

Figure 3-2 Flowchart for Creation of GE 7x7 Lattice

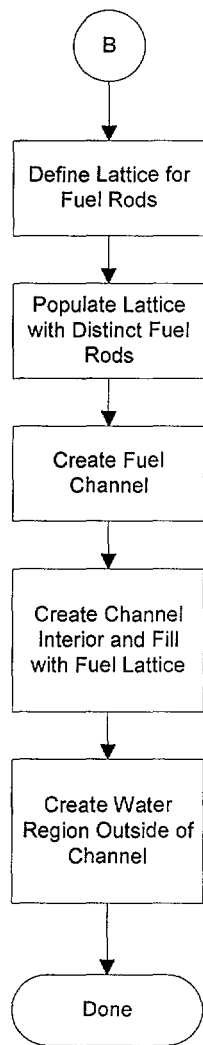


Figure 3-2 (cont'd)

Title: Methodology for GE 8x8 Fuel Lattice with Small Water Rods Model

Document Identifier B000000000-01717-0210-00010 REV 01 Attachment XI Page 1 of 13

CONTENTS

	Page
1. Introduction	4
2. Specifications	5
2.1. Definition of Fuel Rods	5
2.2. Definition of Fuel Lattice	5
2.3. Definition of Fuel Channel	5
2.4. Integration of Components	5
3. Encoding of Process	11
3.1. Driver Function	11
3.2. Lattice-preparation Function	11

Title: Methodology for GE 8x8 Fuel Lattice with Small Water Rods Model

Document Identifier B000000000-01717-0210-00010 REV 01

Attachment XI Page 2 of 13

FIGURES

	Page
2-1 Fuel Rod	6
2-2 Lattice Population	7
2-3 Integration of Components into Lattice	8
3-1 Flowchart for Creation of GE 8x8 Lattice with Small Water Rods	12

Title: Methodology for GE 8x8 Fuel Lattice with Small Water Rods Model

Document Identifier B000000000-01717-0210-00010 REV 01 Attachment XI Page 3 of 13

WORKSHEETS

		Page
2-1	Computation of Surface Coordinates	9
2-2	Cell Definitions	10

Title: Methodology for GE 8x8 Fuel Lattice with Small Water Rods Model**Document Identifier** B000000000-01717-0210-00010 REV 01 Attachment XI Page 4 of 13

1. Introduction

This attachment describes the methodology used to create GE 8x8 (small water rods) fuel lattice models for use in MCNP (References 7.1, 7.2, 7.5, and 7.6) representations of Boiling Water Reactor (BWR) cores. The methodology assumes the existence of a dataset describing the blade geometry that has been generated according to the requirements of Attachment V.

The driver coding within the linkage automation is intended to be sufficiently robust to readily model all the varieties of BWR fuel and is described in Attachment X. This attachment describes the modeling of a GE 8x8 fuel lattice with small water rods. Here "small" indicates that the water rod dimensions are identical to the fuel cladding dimensions.

Title: Methodology for GE 8x8 Fuel Lattice with Small Water Rods Model

Document Identifier B000000000-01717-0210-00010 REV 01 Attachment XI Page 5 of 13

2. Specifications

This process is a subset of the larger process described and illustrated in Attachment X and this attachment describes the particular processing for a GE 8x8 fuel lattice with small water rods (and for fuel assemblies designed and manufactured by other vendors which are consistent with this geometric representation). The basis values used in this attachment are shown in Attachment IV. Note that the values shown herein do not necessarily correspond to a particular lattice used in the analyses, but are shown to illustrate the process whereby such an MCNP input deck is constructed.

2.1. Definition of Fuel Rods

The fuel rods are comprised of zircaloy tubes filled with UO₂ ceramic pellets (Reference 7.8) as shown in Figure 2-1. Computation of the values for the absorber tubes defining surfaces is shown in Worksheet 2-1, while the cell and universe definitions are shown in Worksheet 2-2.

2.2. Definition of Fuel Lattice

The fuel rods fill a regular lattice. In the case of the GE 8x8 fuel design with small water rods, one or more of these fuel rods are displaced with cladding hulls through which non-voided water flows (see Figure 2-2).

2.3. Definition of Fuel Channel

The channel is assumed to be of uniform thickness with rounded corners and is as described in Attachment X for a "D" lattice geometry.

2.4. Integration of Components

The combination of the lattice components into a complete model is illustrated in Figure 2-3.

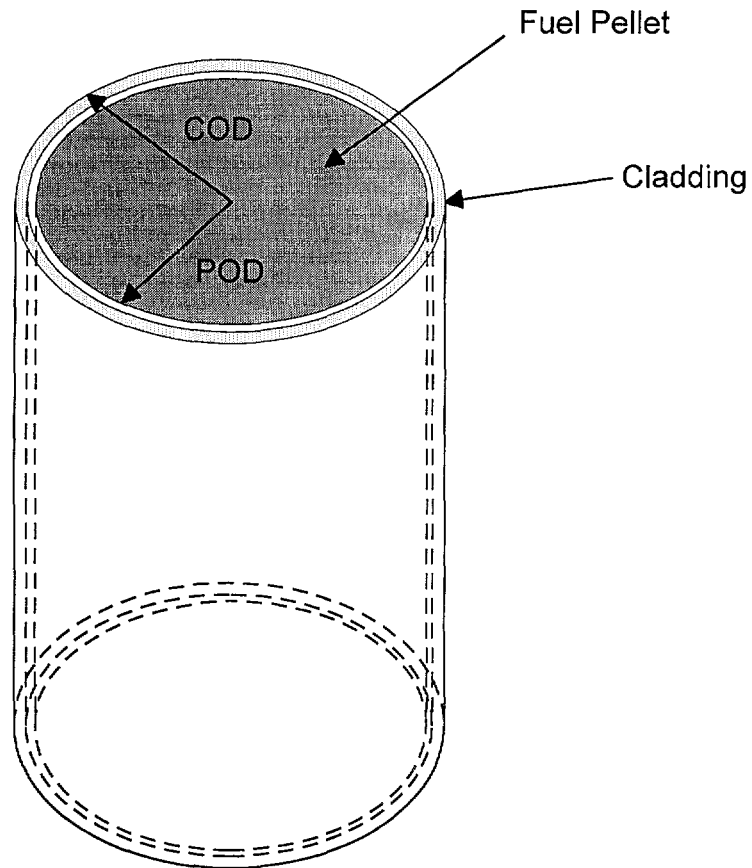


Figure 2-1 Fuel Rod

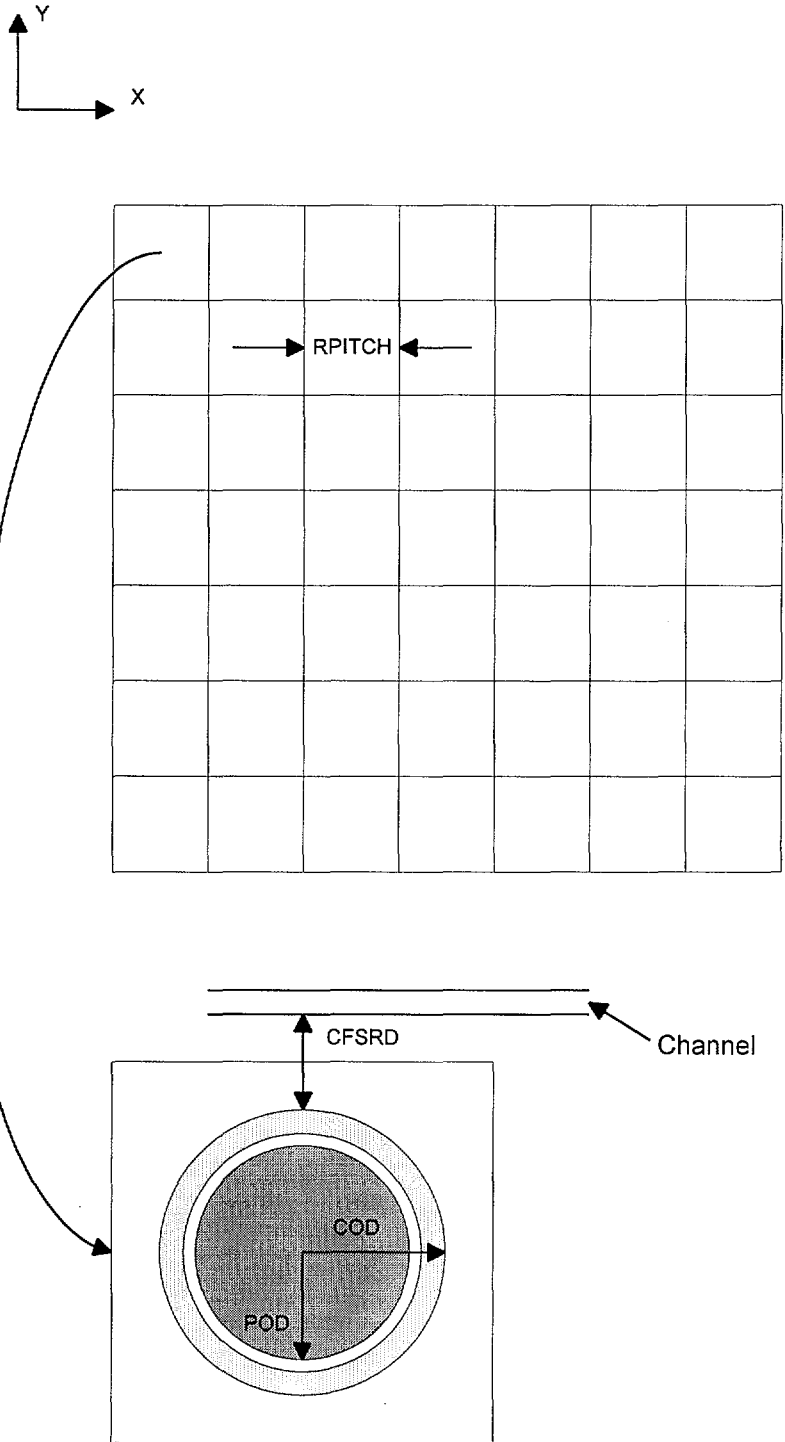


Figure 2-2 Lattice Population

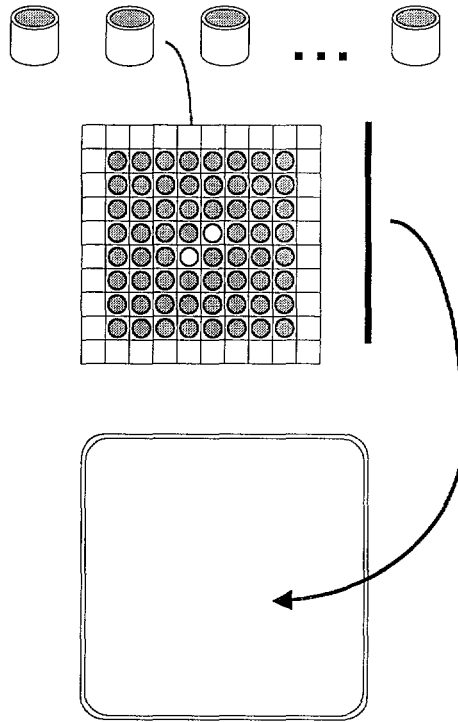


Figure 2-3 Integration of Components into Lattice

Title: Methodology for GE 8x8 Fuel Lattice with Small Water Rods Model

Document Identifier B000000000-01717-0210-00010 REV 01

Attachment XI Page 9 of 13

Worksheet 2-1 Computation of Surface Coordinates

Index	Symbol	Definition	Mnemonic	Parameters	Computation
1	SPOR	Reference Fuel Pellet Outer Surface	c/z	2.16916	= WGAP+CTHICK+CFSRD+(COD/2)
				-2.16916	= -(WGAP+CTHICK+CFSRD+(COD/2))
				0.5283	= POD/2
2	SCIR	Reference Cladding Inner Surface	c/z	2.16916	= WGAP+CTHICK+CFSRD+(COD/2)
				-2.16916	= -(WGAP+CTHICK+CFSRD+(COD/2))
				0.53975	= (COD/2)-CLD
3	SCOR	Reference Cladding Outer Surface	c/z	2.16916	= WGAP+CTHICK+CFSRD+(COD/2)
				-2.16916	= -(WGAP+CTHICK+CFSRD+(COD/2))
				0.62611	= COD/2
4	XMINFRW	XMIN Surface for Fuel Rod Window	px	-0.8128	= -(RPITCH/2)
5	XMAXFRW	XMAX Surface for Fuel Rod Window	px	0.8128	= RPITCH/2
6	YMINFRW	XMIN Surface for Fuel Rod Window	py	-0.8128	= -(RPITCH/2)
7	YMAXFRW	XMAX Surface for Fuel Rod Window	py	0.8128	= RPITCH/2
8	WGCOWX	Wide Gap, Channel Outside Wall, pX Surface	px	0.9525	= WGAP
9	WGCIWX	Wide Gap, Channel Inside Wall, pX Surface	px	1.1557	= WGAP+CTHICK
10	NGCIWX	Narrow Gap, Channel Inside Wall, pX Surface	px	14.56182	= WGAP+CTHICK+ASIN
11	NGCOWX	Narrow Gap, Channel Outside Wall, pX Surface	px	14.76502	= WGAP+(2*CTHICK)+ASIN
12	WGCOWY	Wide Gap, Channel Outside Wall, pY Surface	py	-0.9525	= -WGAP
13	WGCIWY	Wide Gap, Channel Inside Wall, pY Surface	py	-1.1557	= -(WGAP+CTHICK)
14	NGCIWY	Narrow Gap, Channel Inside Wall, pY Surface	py	-14.56182	= -(WGAP+CTHICK+ASIN)
15	NGCOWY	Narrow Gap, Channel Outside Wall, pY Surface	py	-14.76502	= -(WGAP+(2*CTHICK)+ASIN)
16	XAMBIG1	Ambiguity Surface for Channel Corners (Wide Gap)	px	2.1717	= WGAP+CTHICK+CRADIUS
17	XAMBIG2	Ambiguity Surface for Channel Corners (Narrow Gap)	px	13.54582	= WGAP+CTHICK+ASIN-CRADIUS
18	YAMBIG1	Ambiguity Surface for Channel Corners (Wide Gap)	py	-2.1717	= -(WGAP+CTHICK+CRADIUS)
19	YAMBIG2	Ambiguity Surface for Channel Corners (Narrow Gap)	py	-13.54582	= -(WGAP+CTHICK+ASIN-CRADIUS)
20	CC1RO	Outer Radius for Corner 1	c/z	2.1717	= WGAP+CTHICK+CRADIUS
				-2.1717	= -(WGAP+CTHICK+CRADIUS)
				1.2192	= CRADIUS+CTHICK
21	CC1RI	Inner Radius for Corner 1	c/z	2.1717	= WGAP+CTHICK+CRADIUS
				-2.1717	= -(WGAP+CTHICK+CRADIUS)
				1.016	= CRADIUS
22	CC2RO	Outer Radius for Corner 2	c/z	13.54582	= WGAP+CTHICK+ASIN-CRADIUS

Title: Methodology for GE 8x8 Fuel Lattice with Small Water Rods Model

Document Identifier B000000000-01717-0210-00010 REV 01 Attachment XI Page 10 of 13

Index	Symbol	Definition	Mnemonic	Parameters	Computation
				-2.1717	= -(WGAP+CTHICK+CRADIUS)
				1.2192	= CRADIUS+CTHICK
23	CC2RI	Inner Radius for Corner 2	c/z	13.54582	= WGAP+CTHICK+ASIN-CRADIUS
				-2.1717	= -(WGAP+CTHICK+CRADIUS)
				1.016	= CRADIUS
24	CC3RO	Outer Radius for Corner 3	c/z	13.54582	= WGAP+CTHICK+ASIN-CRADIUS
				-13.54582	= -(WGAP+CTHICK+ASIN-CRADIUS)
				1.2192	= CRADIUS+CTHICK
25	CC3RI	Inner Radius for Corner 3	c/z	13.54582	= WGAP+CTHICK+ASIN-CRADIUS
				-13.54582	= -(WGAP+CTHICK+ASIN-CRADIUS)
				1.016	= CRADIUS
26	CC4RO	Outer Radius for Corner 4	c/z	2.1717	= WGAP+CTHICK+CRADIUS
				-13.54582	= -(WGAP+CTHICK+ASIN-CRADIUS)
				1.2192	= CRADIUS+CTHICK
27	CC4RI	Inner Radius for Corner 4	c/z	2.1717	= WGAP+CTHICK+CRADIUS
				-13.54582	= -(WGAP+CTHICK+ASIN-CRADIUS)
				1.016	= CRADIUS

Worksheet 2-2 Cell Definitions

Index	Symbol	Universe	Symbol	Definition	Cell Definition
1	CFP	1	UFR	Fuel Pellet	-1 u= 1
2	CFCG	1		Pellet-Cladding Gap	-2 1 u= 1
3	CFRC	1		Cladding	-3 2 u= 1
4	CMOFR	1		Moderator Outside Fuel Rod	3 u= 1
5	CIWR	2	UWR	Water Inside Water Rod	-1 u= 2
6	CWR	2		Water Rod	-3 2 u= 2
7	CMOWR	2		Moderator Outside Water Rod	3 u= 2
8	CWFRL	3	UFRL	Window for Fuel Rod (Lattice)	4 -5 6 -7 u= 3 fill= 1 or fill= 2
9	CCHAN	4	UCHAN	Channel	(8 -9 -18 19);(-12 13 16 -17); (-14 15 16 -17);(10 -11 -18 19); (-20 21 18 -16);(-22 23 18 17); (-24 25 17 -19);(-26 27 -19 -16) u= 4
10	CWIC	4		Water Inside Channel	(16 -17 -13 14);(9 -16 -18 19); (17 -10 -18 19);(-21 -16 18); (-23 17 19);(-25 17 -19);(27 -16 -19) u= 4 fill= 3
11	CWOC	4		Water Outside Channel	#10 #9 u= 4

Title: Methodology for GE 8x8 Fuel Lattice with Small Water Rods Model**Document Identifier** B000000000-01717-0210-00010 REV 01 Attachment XI Page 11 of 13

3. Encoding of Process

There are two distinct parts of the process for creating lattice models. The first is a driver function that manages the selection of the appropriate data for the lattice and second is a lattice-geometry-specific function that creates the detailed lattice model.

3.1. Driver Function

A description of this function is provided in Attachment X.

3.2. Lattice-preparation Function

A global description of this function is provided in Attachment X. The flowchart of this process for GE 8x8 fuel assemblies with small water rods is shown in Figure 3-1.

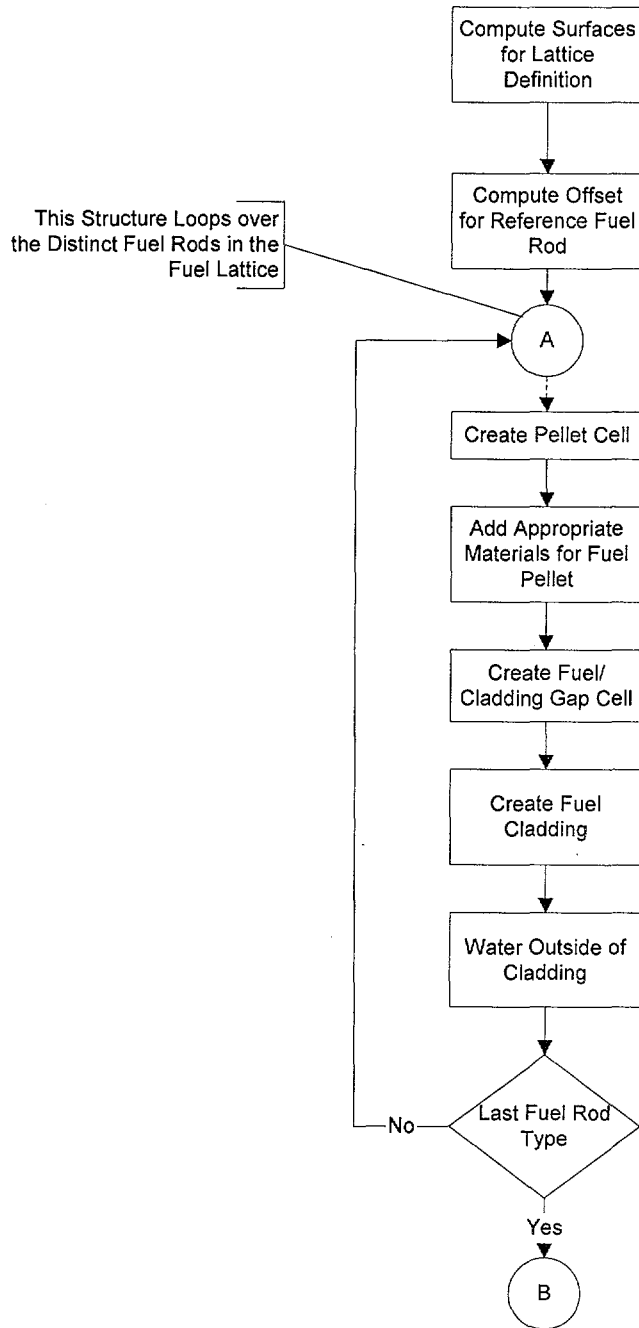


Figure 3-1 Flowchart for Creation of GE 8x8 Lattice with Small Water Rods

Title: Methodology for GE 8x8 Fuel Lattice with Small Water Rods Model

Document Identifier B000000000-01717-0210-00010 REV 01 Attachment XI Page 13 of 13

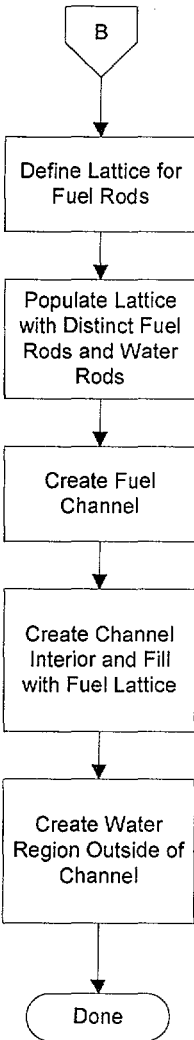


Figure 3-1 (cont'd)

Title: Creation of MCNP Model for Quad Cities Unit 1 Beginning-of-Life Core

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XII **Page 1 of 8**

CONTENTS

	Page
1. Introduction	4
2. Construction of MCNP Input Streams	4
2.1. BLINK Input Specification	4

Title: Creation of MNCP Model for Quad Cities Unit 1 Beginning-of-Life Core

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XII **Page 2 of 8**

FIGURES

	Page
2-1 BLINK Fuel Assembly Identifiers for QC1 Initial Core	5
2-2 Quarter-core Control Blade Position for QC1 BOL	8

Title: Creation of MNCP Model for Quad Cities Unit 1 Beginning-of-Life Core

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XII **Page 3 of 8**

TABLES

	Page
2-1 BLINK NAMELIST Input for BOL CRC Reactivity Analysis	6
2-2 Spacer Locations	7

Title: Creation of MNCP Model for Quad Cities Unit 1 Beginning-of-Life Core

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XII Page 4 of 8

1. Introduction

This attachment describes the creation of the MCNP input streams to model the initial core critical for the Quad Cities Unit 1 core. Thus it documents the processing of Fuel Material Intermediate Datasets (FMID's) from lattice nuclear and geometric specifications and the preparation of input to BLINK, the linkage software routine used to prepare the MCNP input streams.

2. Construction of MCNP Input Streams

The documentation of this portion of the work consists of three parts:

1. identifying the relationship between BLINK fuel assemblies indices and the identifiers used for the unexposed fuel inventories;
2. specifying the various datasets used to construct the MCNP input streams and geometrical and thermodynamic parameters; and
3. defining the control blade positioning.

2.1. BLINK Input Specification

This startup test was performed at the beginning of the first cycle for Quad Cities Unit 1. This evaluation was performed to validate the linkage methodology described in Attachment VI. As such the results must be essentially the same as those obtained from a previous evaluation (Reference 7.7, hereafter cited as the "previous evaluation") where a "hand-crafted" MCNP model was used.

2.1.1. Correspondence of Fuel Assembly Identifiers

The BLINK fuel assembly indices are shown in Figure 2-1. These indices are the same as those used in the reference document for Quad Cities Unit 1 (Reference 7.8, hereafter cited to as the "EPRI Report").

The Fuel Material Intermediate Datasets (FMID's) are created with the IDSGEN software routine, which is documented in Attachment VIII. The SAS2H "cut" files processed in this manner have the following file name nomenclature: AaNn.dat -- where "a" is the BLINK fuel assembly index and "n" is the axial node index.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1											3	3	3	3	3
2										3	3	1	3	1	3
3							3	3	3	3	1	4	1	3	1
4						3	3	1	3	1	4	1	4	2	4
5					3	3	1	3	2	4	2	4	2	4	2
6				3	3	1	3	2	4	2	4	1	4	2	4
7			3	3	1	3	1	4	2	4	2	4	2	4	2
8			3	1	3	2	4	2	4	2	4	1	3	1	3
9			3	3	2	4	2	4	2	4	1	3	1	3	1
10		3	3	1	4	2	4	2	4	1	3	1	3	1	3
11	3	3	1	4	2	4	2	4	1	3	1	3	1	3	1
12	3	1	3	2	4	2	4	1	3	1	3	1	3	1	3
13	3	3	1	4	2	4	2	3	1	3	1	3	1	3	1
14	3	1	3	2	4	2	4	1	3	1	3	1	3	1	3
15	3	3	1	4	2	4	2	3	1	3	1	3	1	3	1

Figure 2-1 BLINK Fuel Assembly Identifiers for QC1 Initial Core

Title: Creation of MNCP Model for Quad Cities Unit 1 Beginning-of-Life Core

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XII Page 6 of 8

2.1.2. Dataset, Geometrical and Thermodynamic Input Values

These values are those documented in Attachment VI, §3.2. The specific values selected for the FORTRAN NAMELIST portion of input for this CRC are shown in Table 2-1. The values for the assignment of fuel assembly geometrical and material types and the underlying lattice geometric and material types are consistent with the fuel assembly indices. The axial node length was a constant value of 15.24 cm (6 inches). This value was explicitly used for the BLINK Version 1 testing, but was implicit for the Version 0 testing. The spacer centers are shown in Table 2-2. The material definition for the smear spacer composition is "SG7D1" and is defined in Attachment III.

Table 2-1 BLINK NAMELIST Input for BOL CRC Reactivity Analysis

Variable	Value	Justification/Reference
CORE_DB	bwr3_724bundle.dat [a]	See Attachment II for complete description of dataset
CORE_MTLS	core_materials.dat [a]	See Attachment III for complete description of dataset
BLADE_DB	ge_d_lattice.dat [a]	See Attachment V for complete description of dataset
LPREFIX	[b]	
FPREFIX	[b]	
NAXIAL	24	This is the number of axial nodes used in the SAS2H depletion calculations.
AFL	365.760	Fuel length is fixed to be 145.24 inches
NCOLP	15	Maximum dimensionality of the quarter core in units of fuel assemblies.
NROWP	15	Maximum dimensionality of the quarter core in units of fuel assemblies.
RHO	0.981 [d]	For this value, see Table 7.3.3-3 of the previous evaluation.
RHOBYP	0.981 [d]	The moderator density throughout the problem is assumed uniform, which is consistent with the startup conditions when the criticality test is performed.
TEMPK	337.04	This value corresponds to 147°F (see §7.1 of the previous evaluation).
MUTP	7GUTP1 [c]	A homogenized mixture of materials with an assumed moderator density of 0.981 g/cm ³ is used.
MLTP	7GLTP1 [c]	A homogenized mixture of materials with an assumed moderator density of 0.981 g/cm ³ is used.

- [a]. The prefix is consistent with the location where the software routine is executed.
- [b]. This input is consistent with location of FMID's processed for this analysis.
- [c]. See Attachment III for a complete description of this composition.
- [d]. The input to BLINK implies greater precision; however, the effect is negligible.

Title: Creation of MNCP Model for Quad Cities Unit 1 Beginning-of-Life Core

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XII Page 7 of 8

Table 2-2 Spacer Locations

Index	Spacer Center Location	
	(inches) [a]	(cm)
1	18.50	46.99
2	38.00	96.52
3	57.50	146.05
4	77.00	195.58
5	96.50	245.11
6	116.00	294.64
7	135.50	344.17

[a]. These values are from the EPRI Report, page A-8.

Title: Creation of MNCP Model for Quad Cities Unit 1 Beginning-of-Life Core

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XII Page 8 of 8

2.1.3. Control Blade Positions

The control blade positions at which the reactor attained criticality are shown in Figure 2-2. The core is modeled in quarter-core symmetry, but since the critical control blade pattern is quarter-core symmetric, no approximations were involved in the control blade pattern used.

	1	2	3	4	5	6	7	8
1						48	0	48
2				0	0	0	0	0
3			0	48	0	48	0	48
4		0	0	0	0	0	0	0
5		48	0	48	0	48	0	48
6	0	0	0	0	0	0	0	0
7	0	48	0	48	0	48	0	48
8	0	0	0	0	0	0	0	0

Figure 2-2 Quarter-core Control Blade Position for QC1 BOL

2.1.4. Input Deck Generated by BLINK

A copy of the input deck generated by BLINK, Version 0, is given in Attachment XIII.

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 1 of 76

This attachment provides a Listing of the MCNP input deck generated by BLINK, version 0, for QC1, BOL.

```
Quad Cities-1, Beginning of Life
c
c   Input Deck Generated by BLINK, Version 0:-
c   Generated on 03/23/98 at 16:26:43 by Process 20234
c
c   Cell Cards
c
c   Cells Defining Problem Domain
c
c   Pressure Vessel
2001      1 -7.9000E+00 -1 2 -3 4 -5 6
      imp:n= 1.0
c   Outside World
2002      0 ( 3: 1: -2: -4: 5)
      imp:n= 0.0
c   Jet Pump Region
2003      2 -9.8114E-01 -6 7 4 -5 -1 2
      imp:n=1.0
c   Core Shroud
2004      1 -7.9000E+00 -7 8 4 -5 -1 2
      imp:n=1.0
c   Upper Tie Plate Region
2005      3 -1.3970E+00 -8 4 -5 9 -10
      imp:n=1.0
c   Core Grid Region
2006      4 -1.0450E+00 -8 4 -5 10 -11
      imp:n=1.0
c   Upper Plenum Region
2007      2 -9.8114E-01 -8 4 -5 11 -1
      imp:n=1.0
c   Lower Tie Plate Region
2008      5 -1.3970E+00 -8 4 -5 -12 13
      imp:n=1.0
c   Fuel Support/Core Plate
2009      6 -2.5430E+00 -8 4 -5 -13 14
      imp:n=1.0
c   Lower Plenum Region
2010      2 -9.8114E-01 -8 4 -5 -14 2
      imp:n=1.0
c   Boron Carbide in Reference Absorber Tube
2011      7 -1.7600E+00 -15
      imp:n=1.0 u= 1
c   Reference Absorber Tube
2012      1 -7.9000E+00 -16 15
      imp:n=1.0 u= 1
c   Water Outside Reference Absorber Tube
2013      2 -9.8114E-01 16
      imp:n=1.0 u= 1
c   Window for Reference Absorber Tube Cell
2014      2 -9.8114E-01 -17 18 -19 20
      imp:n=1.0 u= 2 lat= 1 fill= 0:21 0:0 0:0 1 20r 2
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 2 of 76

c Tie Rod
2015 1 -7.9000E+00 (-21 22 -18 23 -24 25):(26 -27 28 -29 -24 25)
imp:n=1.0 u= 3

c Control Blade Sheath
2016 1 -7.9000E+00 (18 -30 19 -21 -24 25):(18 -30 -20 22 -24 25):
(30 31 -32 -24 25)
imp:n=1.0 u= 4

c Inside of Blade Wing
2017 2 -9.8114E-01 (33 34 -35 -30 -24 25):(-31 -24 25)
u= 4 fill= 2 imp:n= 1.0

c Region Outside of Blade Wing
2018 2 -9.8114E-01 #2016 #2017
imp:n=1.0 u= 4

c Window for Wing of Blade
1 2 -9.8114E-01 36 -37 38 -39 fill= 4
imp:n=1.0 u= 3

c Wing Cloned to North Position
2 like 1 but *trcl=(0 0 0 90 0 90 180 90 90 90 0)
u= 3

c Wing Cloned to West Position
3 like 1 but *trcl=(0 0 0 180 90 90 -90 180 90 90 0)
u= 3

c Wing Cloned to South Position
4 like 1 but *trcl=(0 0 0 -90 180 90 0 -90 90 90 0)
u= 3

c Region Outside of Blade
2019 2 -9.8114E-01 #2015 #1 #2 #3 #4
imp:n=1.0 u= 3

c Fuel Pellet, #1, L1
2020 9 -1.0340E+01 -40 u= 5 imp:n= 1.0

c Fuel/Cladding Gap, #1, L1
2021 0 40 -41 u= 5 imp:n= 1.0

c Cladding, #1, L1
2022 10 -6.5600E+00 41 -42 u= 5 imp:n= 1.0

c Water Outside Cladding #1, L1
2023 8 -9.8114E-01 42
u= 5 imp:n= 1.0

c Fuel Pellet, #2, L1
2024 like 2020 but
mat= 11 rho= -9.9400E+00
u= 6 imp:n= 1.0

c Fuel/Cladding Gap, #2, L1
2025 like 2021 but
u= 6 imp:n= 1.0

c Cladding, #2, L1
2026 like 2022 but
u= 6 imp:n= 1.0

c Water Outside Cladding #2, L1
2027 like 2023 but
u= 6 imp:n= 1.0

c Fuel Pellet, #3, L1
2028 like 2020 but
mat= 12 rho= -9.4800E+00
u= 7 imp:n= 1.0

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 3 of 76

c Fuel/Cladding Gap, #3, L1
2029 like 2021 but
u= 7 imp:n= 1.0

c Cladding, #3, L1
2030 like 2022 but
u= 7 imp:n= 1.0

c Water Outside Cladding #3, L1
2031 like 2023 but
u= 7 imp:n= 1.0

c Fuel Pellet, #4, L1
2032 like 2020 but
mat= 13 rho= -1.0340E+01
u= 8 imp:n= 1.0

c Fuel/Cladding Gap, #4, L1
2033 like 2021 but
u= 8 imp:n= 1.0

c Cladding, #4, L1
2034 like 2022 but
u= 8 imp:n= 1.0

c Water Outside Cladding #4, L1
2035 like 2023 but
u= 8 imp:n= 1.0

c Fuel Pellet, #5, L1
2036 like 2020 but
mat= 14 rho= -9.9400E+00
u= 9 imp:n= 1.0

c Fuel/Cladding Gap, #5, L1
2037 like 2021 but
u= 9 imp:n= 1.0

c Cladding, #5, L1
2038 like 2022 but
u= 9 imp:n= 1.0

c Water Outside Cladding #5, L1
2039 like 2023 but
u= 9 imp:n= 1.0

c Fuel Pellet, #6, L1
2040 like 2020 but
mat= 15 rho= -9.9400E+00
u= 10 imp:n= 1.0

c Fuel/Cladding Gap, #6, L1
2041 like 2021 but
u= 10 imp:n= 1.0

c Cladding, #6, L1
2042 like 2022 but
u= 10 imp:n= 1.0

c Water Outside Cladding #6, L1
2043 like 2023 but
u= 10 imp:n= 1.0

c Fuel Pellet, #7, L1
2044 like 2020 but
mat= 16 rho= -1.0300E+01
u= 11 imp:n= 1.0

c Fuel/Cladding Gap, #7, L1
2045 like 2021 but

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 4 of 76

```

      u= 11 imp:n= 1.0
c    Cladding, #7, L1
2046 like 2022 but
      u= 11 imp:n= 1.0
c    Water Outside Cladding #7, L1
2047 like 2023 but
      u= 11 imp:n= 1.0
c    Fuel Pellet, #8, L1
2048 like 2020 but
      mat= 17 rho= -1.0340E+01
      u= 12 imp:n= 1.0
c    Fuel/Cladding Gap, #8, L1
2049 like 2021 but
      u= 12 imp:n= 1.0
c    Cladding, #8, L1
2050 like 2022 but
      u= 12 imp:n= 1.0
c    Water Outside Cladding #8, L1
2051 like 2023 but
      u= 12 imp:n= 1.0
c    Reference Fuel Rod Cell, L1
2052      8 -9.8114E-01 -43 44 45 -46 lat= 1 u = 13 imp:n= 1.0
      fill= -1:7 -1:7 0:0
          13 13 13 13 13 13 13 13 13
          13 10 10 9 9 9 9 10 13
          13 10 8 8 5 5 6 9 13
          13 9 8 12 5 5 5 6 13
          13 9 5 5 7 5 11 6 13
          13 9 5 5 5 5 5 6 13
          13 9 6 5 11 5 5 6 13
          13 10 9 6 6 6 6 9 13
          13 13 13 13 13 13 13 13 13
c    Channel, L1
2053      18 -6.5600E+00 ( 47 -48 -49 50):( -51 52 53 -54):
      ( -55 56 53 -54):( 57 -58 50 -49):
      ( -59 60 49 -53):( -61 62 49 54):
      ( -63 64 54 -50):( -65 66 -50 -53)
      u= 14 imp:n= 1.0
c    Active Fuel Area, L1
2054      8 -9.8114E-01 ( 53 -54 -52 55):( 48 -53 -49 50 ):
      ( 54 -57 -49 50):( -60 -53 49):
      ( -62 54 49):( -64 54 -50):
      ( -66 -53 -50)
      fill= 13 u= 14 imp:n= 1.0
c    Water Outside of Channel, L1
2055      2 -9.8114E-01 ( -47 : 51 : 58 : -56 ):
      ( 59 -53 49 ):
      ( 61 54 49 ):
      ( 63 54 -50 ):
      ( 65 -53 -50 )
      u= 14 imp:n= 1.0
c    Fuel Pellet, #1, L2
2056      19 -1.0340E+01 -40      u= 15 imp:n= 1.0
c    Fuel/Cladding Gap, #1, L2

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 5 of 76

2057 0 40 -41 u= 15 imp:n= 1.0
c Cladding, #1, L2
2058 10 -6.5600E+00 41 -42 u= 15 imp:n= 1.0
c Water Outside Cladding #1, L2
2059 8 -9.8114E-01 42
u= 15 imp:n= 1.0
c Fuel Pellet, #2, L2
2060 like 2056 but
mat= 20 rho= -9.9400E+00
u= 16 imp:n= 1.0
c Fuel/Cladding Gap, #2, L2
2061 like 2057 but
u= 16 imp:n= 1.0
c Cladding, #2, L2
2062 like 2058 but
u= 16 imp:n= 1.0
c Water Outside Cladding #2, L2
2063 like 2059 but
u= 16 imp:n= 1.0
c Fuel Pellet, #3, L2
2064 like 2056 but
mat= 21 rho= -9.4800E+00
u= 17 imp:n= 1.0
c Fuel/Cladding Gap, #3, L2
2065 like 2057 but
u= 17 imp:n= 1.0
c Cladding, #3, L2
2066 like 2058 but
u= 17 imp:n= 1.0
c Water Outside Cladding #3, L2
2067 like 2059 but
u= 17 imp:n= 1.0
c Fuel Pellet, #4, L2
2068 like 2056 but
mat= 22 rho= -1.0340E+01
u= 18 imp:n= 1.0
c Fuel/Cladding Gap, #4, L2
2069 like 2057 but
u= 18 imp:n= 1.0
c Cladding, #4, L2
2070 like 2058 but
u= 18 imp:n= 1.0
c Water Outside Cladding #4, L2
2071 like 2059 but
u= 18 imp:n= 1.0
c Fuel Pellet, #5, L2
2072 like 2056 but
mat= 23 rho= -9.9400E+00
u= 19 imp:n= 1.0
c Fuel/Cladding Gap, #5, L2
2073 like 2057 but
u= 19 imp:n= 1.0
c Cladding, #5, L2
2074 like 2058 but

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 6 of 76

```

      u= 19 imp:n= 1.0
c      Water Outside Cladding #5, L2
2075 like 2059 but
      u= 19 imp:n= 1.0
c      Fuel Pellet, #6, L2
2076 like 2056 but
      mat= 24 rho= -9.9400E+00
      u= 20 imp:n= 1.0
c      Fuel/Cladding Gap, #6, L2
2077 like 2057 but
      u= 20 imp:n= 1.0
c      Cladding, #6, L2
2078 like 2058 but
      u= 20 imp:n= 1.0
c      Water Outside Cladding #6, L2
2079 like 2059 but
      u= 20 imp:n= 1.0
c      Fuel Pellet, #7, L2
2080 like 2056 but
      mat= 25 rho= -1.0300E+01
      u= 21 imp:n= 1.0
c      Fuel/Cladding Gap, #7, L2
2081 like 2057 but
      u= 21 imp:n= 1.0
c      Cladding, #7, L2
2082 like 2058 but
      u= 21 imp:n= 1.0
c      Water Outside Cladding #7, L2
2083 like 2059 but
      u= 21 imp:n= 1.0
c      Fuel Pellet, #8, L2
2084 like 2056 but
      mat= 26 rho= -1.0340E+01
      u= 22 imp:n= 1.0
c      Fuel/Cladding Gap, #8, L2
2085 like 2057 but
      u= 22 imp:n= 1.0
c      Cladding, #8, L2
2086 like 2058 but
      u= 22 imp:n= 1.0
c      Water Outside Cladding #8, L2
2087 like 2059 but
      u= 22 imp:n= 1.0
c      Reference Fuel Rod Cell, L2
2088      8 -9.8114E-01 -43 44 45 -46 lat= 1 u = 23 imp:n= 1.0
      fill= -1:7 -1:7 0:0
      23 23 23 23 23 23 23 23 23
      23 20 20 19 19 19 19 20 23
      23 20 18 18 15 15 16 19 23
      23 19 18 22 15 15 15 16 23
      23 19 15 15 17 15 21 16 23
      23 19 15 15 15 15 15 16 23
      23 19 16 15 21 15 15 16 23
      23 20 19 16 16 16 16 19 23

```

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 7 of 76

```

      23 23 23 23 23 23 23 23 23
c   Channel, L2
2089   18 -6.5600E+00 ( 47 -48 -49 50):( -51 52 53 -54):
      ( -55 56 53 -54):( 57 -58 50 -49):
      ( -59 60 49 -53):( -61 62 49 54):
      ( -63 64 54 -50):( -65 66 -50 -53)
      u= 24 imp:n= 1.0
c   Active Fuel Area, L2
2090   8 -9.8114E-01 ( 53 -54 -52 55):( 48 -53 -49 50 ):
      ( 54 -57 -49 50):( -60 -53 49):
      ( -62 54 49):( -64 54 -50):
      ( -66 -53 -50)
      fill= 23 u= 24 imp:n= 1.0
c   Water Outside of Channel, L2
2091   2 -9.8114E-01 ( -47 : 51 : 58 : -56 ):
      ( 59 -53 49 ):
      ( 61 54 49 ):
      ( 63 54 -50 ):
      ( 65 -53 -50 )
      u= 24 imp:n= 1.0
c   Fuel Pellet, #1, L3
2092   27 -1.0340E+01 -40      u= 25 imp:n= 1.0
c   Fuel/Cladding Gap, #1, L3
2093   0 40 -41      u= 25 imp:n= 1.0
c   Cladding, #1, L3
2094   10 -6.5600E+00 41 -42      u= 25 imp:n= 1.0
c   Water Outside Cladding #1, L3
2095   8 -9.8114E-01 42
      u= 25 imp:n= 1.0
c   Fuel Pellet, #2, L3
2096 like 2092 but
      mat= 28 rho= -9.9400E+00
      u= 26 imp:n= 1.0
c   Fuel/Cladding Gap, #2, L3
2097 like 2093 but
      u= 26 imp:n= 1.0
c   Cladding, #2, L3
2098 like 2094 but
      u= 26 imp:n= 1.0
c   Water Outside Cladding #2, L3
2099 like 2095 but
      u= 26 imp:n= 1.0
c   Fuel Pellet, #3, L3
2100 like 2092 but
      mat= 29 rho= -9.4800E+00
      u= 27 imp:n= 1.0
c   Fuel/Cladding Gap, #3, L3
2101 like 2093 but
      u= 27 imp:n= 1.0
c   Cladding, #3, L3
2102 like 2094 but
      u= 27 imp:n= 1.0
c   Water Outside Cladding #3, L3
2103 like 2095 but

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 8 of 76

u= 27 imp:n= 1.0
c Fuel Pellet, #4, L3
2104 like 2092 but
mat= 30 rho= -1.0340E+01
u= 28 imp:n= 1.0
c Fuel/Cladding Gap, #4, L3
2105 like 2093 but
u= 28 imp:n= 1.0
c Cladding, #4, L3
2106 like 2094 but
u= 28 imp:n= 1.0
c Water Outside Cladding #4, L3
2107 like 2095 but
u= 28 imp:n= 1.0
c Fuel Pellet, #5, L3
2108 like 2092 but
mat= 31 rho= -9.9400E+00
u= 29 imp:n= 1.0
c Fuel/Cladding Gap, #5, L3
2109 like 2093 but
u= 29 imp:n= 1.0
c Cladding, #5, L3
2110 like 2094 but
u= 29 imp:n= 1.0
c Water Outside Cladding #5, L3
2111 like 2095 but
u= 29 imp:n= 1.0
c Fuel Pellet, #6, L3
2112 like 2092 but
mat= 32 rho= -9.9400E+00
u= 30 imp:n= 1.0
c Fuel/Cladding Gap, #6, L3
2113 like 2093 but
u= 30 imp:n= 1.0
c Cladding, #6, L3
2114 like 2094 but
u= 30 imp:n= 1.0
c Water Outside Cladding #6, L3
2115 like 2095 but
u= 30 imp:n= 1.0
c Fuel Pellet, #7, L3
2116 like 2092 but
mat= 33 rho= -1.0300E+01
u= 31 imp:n= 1.0
c Fuel/Cladding Gap, #7, L3
2117 like 2093 but
u= 31 imp:n= 1.0
c Cladding, #7, L3
2118 like 2094 but
u= 31 imp:n= 1.0
c Water Outside Cladding #7, L3
2119 like 2095 but
u= 31 imp:n= 1.0
c Fuel Pellet, #8, L3

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 9 of 76

```

2120 like 2092 but
      mat= 34 rho= -1.0340E+01
      u= 32 imp:n= 1.0
c     Fuel/Cladding Gap, #8, L3
2121 like 2093 but
      u= 32 imp:n= 1.0
c     Cladding, #8, L3
2122 like 2094 but
      u= 32 imp:n= 1.0
c     Water Outside Cladding #8, L3
2123 like 2095 but
      u= 32 imp:n= 1.0
c     Reference Fuel Rod Cell, L3
2124      8 -9.8114E-01 -43 44 45 -46 lat= 1 u = 33 imp:n= 1.0
      fill= -1:7 -1:7 0:0
           33 33 33 33 33 33 33 33 33
           33 30 30 29 29 29 29 30 33
           33 30 28 28 25 25 26 29 33
           33 29 28 32 25 25 25 26 33
           33 29 25 25 27 25 31 26 33
           33 29 25 25 25 25 25 26 33
           33 29 26 25 31 25 25 26 33
           33 30 29 26 26 26 26 29 33
           33 33 33 33 33 33 33 33 33
c     Channel, L3
2125      18 -6.5600E+00 ( 47 -48 -49 50):( -51 52 53 -54):
      ( -55 56 53 -54):( 57 -58 50 -49):
      ( -59 60 49 -53):( -61 62 49 54):
      ( -63 64 54 -50):( -65 66 -50 -53)
      u= 34 imp:n= 1.0
c     Active Fuel Area, L3
2126      8 -9.8114E-01 ( 53 -54 -52 55):( 48 -53 -49 50 ):
      ( 54 -57 -49 50):( -60 -53 49):
      ( -62 54 49):( -64 54 -50):
      ( -66 -53 -50)
      fill= 33 u= 34 imp:n= 1.0
c     Water Outside of Channel, L3
2127      2 -9.8114E-01 ( -47 : 51 : 58 : -56 ):
      ( 59 -53 49 ):
      ( 61 54 49 ):
      ( 63 54 -50 ):
      ( 65 -53 -50 )
      u= 34 imp:n= 1.0
c     Fuel Pellet, #1, L4
2128      35 -1.0340E+01 -40          u= 35 imp:n= 1.0
c     Fuel/Cladding Gap, #1, L4
2129      0 40 -41          u= 35 imp:n= 1.0
c     Cladding, #1, L4
2130      10 -6.5600E+00 41 -42          u= 35 imp:n= 1.0
c     Water Outside Cladding #1, L4
2131      8 -9.8114E-01 42
      u= 35 imp:n= 1.0
c     Fuel Pellet, #2, L4
2132 like 2128 but

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 10 of 76

```
      mat= 36 rho= -9.4800E+00
      u= 36 imp:n= 1.0
c      Fuel/Cladding Gap, #2, L4
2133 like 2129 but
      u= 36 imp:n= 1.0
c      Cladding, #2, L4
2134 like 2130 but
      u= 36 imp:n= 1.0
c      Water Outside Cladding #2, L4
2135 like 2131 but
      u= 36 imp:n= 1.0
c      Fuel Pellet, #3, L4
2136 like 2128 but
      mat= 37 rho= -1.0340E+01
      u= 37 imp:n= 1.0
c      Fuel/Cladding Gap, #3, L4
2137 like 2129 but
      u= 37 imp:n= 1.0
c      Cladding, #3, L4
2138 like 2130 but
      u= 37 imp:n= 1.0
c      Water Outside Cladding #3, L4
2139 like 2131 but
      u= 37 imp:n= 1.0
c      Fuel Pellet, #4, L4
2140 like 2128 but
      mat= 38 rho= -1.0340E+01
      u= 38 imp:n= 1.0
c      Fuel/Cladding Gap, #4, L4
2141 like 2129 but
      u= 38 imp:n= 1.0
c      Cladding, #4, L4
2142 like 2130 but
      u= 38 imp:n= 1.0
c      Water Outside Cladding #4, L4
2143 like 2131 but
      u= 38 imp:n= 1.0
c      Fuel Pellet, #5, L4
2144 like 2128 but
      mat= 39 rho= -1.0300E+01
      u= 39 imp:n= 1.0
c      Fuel/Cladding Gap, #5, L4
2145 like 2129 but
      u= 39 imp:n= 1.0
c      Cladding, #5, L4
2146 like 2130 but
      u= 39 imp:n= 1.0
c      Water Outside Cladding #5, L4
2147 like 2131 but
      u= 39 imp:n= 1.0
c      Fuel Pellet, #6, L4
2148 like 2128 but
      mat= 40 rho= -1.0340E+01
      u= 40 imp:n= 1.0
```

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 11 of 76

```

c    Fuel/Cladding Gap, #6, L4
2149 like 2129 but
     u= 40 imp:n= 1.0
c    Cladding, #6, L4
2150 like 2130 but
     u= 40 imp:n= 1.0
c    Water Outside Cladding #6, L4
2151 like 2131 but
     u= 40 imp:n= 1.0
c    Reference Fuel Rod Cell, L4
2152      8 -9.8114E-01 -43 44 45 -46 lat= 1 u = 41 imp:n= 1.0
     fill= -1:7 -1:7 0:0
           41 41 41 41 41 41 41 41 41
           41 38 38 37 37 37 37 37 38 41
           41 38 37 37 35 35 35 37 41
           41 37 37 40 35 35 35 35 41
           41 37 35 35 36 35 39 35 41
           41 37 35 35 35 35 35 35 41
           41 37 35 35 39 35 35 35 41
           41 38 37 35 35 35 35 37 41
           41 41 41 41 41 41 41 41 41
c    Channel, L4
2153      18 -6.5600E+00 ( 47 -48 -49 50):( -51 52 53 -54):
     ( -55 56 53 -54):( 57 -58 50 -49):
     ( -59 60 49 -53):( -61 62 49 54):
     ( -63 64 54 -50):( -65 66 -50 -53)
     u= 42 imp:n= 1.0
c    Active Fuel Area, L4
2154      8 -9.8114E-01 ( 53 -54 -52 55):( 48 -53 -49 50 ):
     ( 54 -57 -49 50):( -60 -53 49):
     ( -62 54 49):( -64 54 -50):
     ( -66 -53 -50)
     fill= 41 u= 42 imp:n= 1.0
c    Water Outside of Channel, L4
2155      2 -9.8114E-01 ( -47 : 51 : 58 : -56 ):
     ( 59 -53 49 ):
     ( 61 54 49 ):
     ( 63 54 -50 ):
     ( 65 -53 -50 )
     u= 42 imp:n= 1.0
c    Fuel Pellet, #1, L5
2156      41 -1.0340E+01 -40          u= 43 imp:n= 1.0
c    Fuel/Cladding Gap, #1, L5
2157      0 40 -41          u= 43 imp:n= 1.0
c    Cladding, #1, L5
2158      10 -6.5600E+00 41 -42          u= 43 imp:n= 1.0
c    Water Outside Cladding #1, L5
2159      8 -9.8114E-01 42
     u= 43 imp:n= 1.0
c    Fuel Pellet, #2, L5
2160 like 2156 but
     mat= 42 rho= -9.4800E+00
     u= 44 imp:n= 1.0
c    Fuel/Cladding Gap, #2, L5

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 12 of 76

2161 like 2157 but
u= 44 imp:n= 1.0
c Cladding, #2, L5
2162 like 2158 but
u= 44 imp:n= 1.0
c Water Outside Cladding #2, L5
2163 like 2159 but
u= 44 imp:n= 1.0
c Fuel Pellet, #3, L5
2164 like 2156 but
mat= 43 rho= -1.0340E+01
u= 45 imp:n= 1.0
c Fuel/Cladding Gap, #3, L5
2165 like 2157 but
u= 45 imp:n= 1.0
c Cladding, #3, L5
2166 like 2158 but
u= 45 imp:n= 1.0
c Water Outside Cladding #3, L5
2167 like 2159 but
u= 45 imp:n= 1.0
c Fuel Pellet, #4, L5
2168 like 2156 but
mat= 44 rho= -1.0340E+01
u= 46 imp:n= 1.0
c Fuel/Cladding Gap, #4, L5
2169 like 2157 but
u= 46 imp:n= 1.0
c Cladding, #4, L5
2170 like 2158 but
u= 46 imp:n= 1.0
c Water Outside Cladding #4, L5
2171 like 2159 but
u= 46 imp:n= 1.0
c Fuel Pellet, #5, L5
2172 like 2156 but
mat= 45 rho= -1.0260E+01
u= 47 imp:n= 1.0
c Fuel/Cladding Gap, #5, L5
2173 like 2157 but
u= 47 imp:n= 1.0
c Cladding, #5, L5
2174 like 2158 but
u= 47 imp:n= 1.0
c Water Outside Cladding #5, L5
2175 like 2159 but
u= 47 imp:n= 1.0
c Fuel Pellet, #6, L5
2176 like 2156 but
mat= 46 rho= -1.0340E+01
u= 48 imp:n= 1.0
c Fuel/Cladding Gap, #6, L5
2177 like 2157 but
u= 48 imp:n= 1.0

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 13 of 76

```

c      Cladding, #6, L5
2178  like 2158 but
      u= 48 imp:n= 1.0
c      Water Outside Cladding #6, L5
2179  like 2159 but
      u= 48 imp:n= 1.0
c      Reference Fuel Rod Cell, L5
2180      8 -9.8114E-01 -43 44 45 -46 lat= 1 u = 49 imp:n= 1.0
      fill= -1:7 -1:7 0:0
          49 49 49 49 49 49 49 49 49
          49 46 46 45 45 45 45 46 49
          49 46 45 45 43 43 43 45 49
          49 45 45 48 43 43 43 43 49
          49 45 43 43 44 43 47 43 49
          49 45 43 43 43 43 43 43 49
          49 45 43 43 47 43 43 43 49
          49 46 45 43 43 43 43 45 49
          49 49 49 49 49 49 49 49 49
c      Channel, L5
2181      18 -6.5600E+00 ( 47 -48 -49 50):( -51 52 53 -54):
      ( -55 56 53 -54):( 57 -58 50 -49):
      ( -59 60 49 -53):( -61 62 49 54):
      ( -63 64 54 -50):( -65 66 -50 -53)
      u= 50 imp:n= 1.0
c      Active Fuel Area, L5
2182      8 -9.8114E-01 ( 53 -54 -52 55):( 48 -53 -49 50 ):
      ( 54 -57 -49 50):( -60 -53 49):
      ( -62 54 49):( -64 54 -50):
      ( -66 -53 -50)
      fill= 49 u= 50 imp:n= 1.0
c      Water Outside of Channel, L5
2183      2 -9.8114E-01 ( -47 : 51 : 58 : -56 ):
      ( 59 -53 49 ):
      ( 61 54 49 ):
      ( 63 54 -50 ):
      ( 65 -53 -50 )
      u= 50 imp:n= 1.0
c      Fuel Pellet, #1, L6
2184      47 -1.0340E+01 -40      u= 51 imp:n= 1.0
c      Fuel/Cladding Gap, #1, L6
2185      0 40 -41      u= 51 imp:n= 1.0
c      Cladding, #1, L6
2186      10 -6.5600E+00 41 -42      u= 51 imp:n= 1.0
c      Water Outside Cladding #1, L6
2187      8 -9.8114E-01 42
      u= 51 imp:n= 1.0
c      Fuel Pellet, #2, L6
2188  like 2184 but
      mat= 48 rho= -9.4800E+00
      u= 52 imp:n= 1.0
c      Fuel/Cladding Gap, #2, L6
2189  like 2185 but
      u= 52 imp:n= 1.0
c      Cladding, #2, L6

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 14 of 76

2190 like 2186 but
u= 52 imp:n= 1.0
c Water Outside Cladding #2, L6
2191 like 2187 but
u= 52 imp:n= 1.0
c Fuel Pellet, #3, L6
2192 like 2184 but
mat= 49 rho= -1.0340E+01
u= 53 imp:n= 1.0
c Fuel/Cladding Gap, #3, L6
2193 like 2185 but
u= 53 imp:n= 1.0
c Cladding, #3, L6
2194 like 2186 but
u= 53 imp:n= 1.0
c Water Outside Cladding #3, L6
2195 like 2187 but
u= 53 imp:n= 1.0
c Fuel Pellet, #4, L6
2196 like 2184 but
mat= 50 rho= -1.0340E+01
u= 54 imp:n= 1.0
c Fuel/Cladding Gap, #4, L6
2197 like 2185 but
u= 54 imp:n= 1.0
c Cladding, #4, L6
2198 like 2186 but
u= 54 imp:n= 1.0
c Water Outside Cladding #4, L6
2199 like 2187 but
u= 54 imp:n= 1.0
c Fuel Pellet, #5, L6
2200 like 2184 but
mat= 51 rho= -1.0260E+01
u= 55 imp:n= 1.0
c Fuel/Cladding Gap, #5, L6
2201 like 2185 but
u= 55 imp:n= 1.0
c Cladding, #5, L6
2202 like 2186 but
u= 55 imp:n= 1.0
c Water Outside Cladding #5, L6
2203 like 2187 but
u= 55 imp:n= 1.0
c Fuel Pellet, #6, L6
2204 like 2184 but
mat= 52 rho= -1.0340E+01
u= 56 imp:n= 1.0
c Fuel/Cladding Gap, #6, L6
2205 like 2185 but
u= 56 imp:n= 1.0
c Cladding, #6, L6
2206 like 2186 but
u= 56 imp:n= 1.0

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 15 of 76

```

c      Water Outside Cladding #6, L6
2207 like 2187 but
      u= 56 imp:n= 1.0
c      Reference Fuel Rod Cell, L6
2208      8 -9.8114E-01 -43 44 45 -46 lat= 1 u = 57 imp:n= 1.0
      fill= -1:7 -1:7 0:0
          57 57 57 57 57 57 57 57 57
          57 54 54 53 53 53 53 54 57
          57 54 53 53 51 51 51 53 57
          57 53 53 56 51 51 51 51 57
          57 53 51 51 52 51 55 51 57
          57 53 51 51 51 51 51 51 57
          57 53 51 51 55 51 51 51 57
          57 54 53 51 51 51 51 53 57
          57 57 57 57 57 57 57 57 57
c      Channel, L6
2209      18 -6.5600E+00 ( 47 -48 -49 50):( -51 52 53 -54):
      ( -55 56 53 -54):( 57 -58 50 -49):
      ( -59 60 49 -53):( -61 62 49 54):
      ( -63 64 54 -50):( -65 66 -50 -53)
      u= 58 imp:n= 1.0
c      Active Fuel Area, L6
2210      8 -9.8114E-01 ( 53 -54 -52 55):( 48 -53 -49 50 ):
      ( 54 -57 -49 50):( -60 -53 49):
      ( -62 54 49):( -64 54 -50):
      ( -66 -53 -50)
      fill= 57 u= 58 imp:n= 1.0
c      Water Outside of Channel, L6
2211      2 -9.8114E-01 ( -47 : 51 : 58 : -56 ):
      ( 59 -53 49 ):
      ( 61 54 49 ):
      ( 63 54 -50 ):
      ( 65 -53 -50 )
      u= 58 imp:n= 1.0
c      Fuel Pellet, #1, L7
2212      53 -1.0340E+01 -40      u= 59 imp:n= 1.0
c      Fuel/Cladding Gap, #1, L7
2213      0 40 -41      u= 59 imp:n= 1.0
c      Cladding, #1, L7
2214      10 -6.5600E+00 41 -42      u= 59 imp:n= 1.0
c      Water Outside Cladding #1, L7
2215      8 -9.8114E-01 42
      u= 59 imp:n= 1.0
c      Fuel Pellet, #2, L7
2216 like 2212 but
      mat= 54 rho= -9.9400E+00
      u= 60 imp:n= 1.0
c      Fuel/Cladding Gap, #2, L7
2217 like 2213 but
      u= 60 imp:n= 1.0
c      Cladding, #2, L7
2218 like 2214 but
      u= 60 imp:n= 1.0
c      Water Outside Cladding #2, L7

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 16 of 76

2219 like 2215 but
u= 60 imp:n= 1.0
c Fuel Pellet, #3, L7
2220 like 2212 but
mat= 55 rho= -9.4800E+00
u= 61 imp:n= 1.0
c Fuel/Cladding Gap, #3, L7
2221 like 2213 but
u= 61 imp:n= 1.0
c Cladding, #3, L7
2222 like 2214 but
u= 61 imp:n= 1.0
c Water Outside Cladding #3, L7
2223 like 2215 but
u= 61 imp:n= 1.0
c Fuel Pellet, #4, L7
2224 like 2212 but
mat= 56 rho= -1.0340E+01
u= 62 imp:n= 1.0
c Fuel/Cladding Gap, #4, L7
2225 like 2213 but
u= 62 imp:n= 1.0
c Cladding, #4, L7
2226 like 2214 but
u= 62 imp:n= 1.0
c Water Outside Cladding #4, L7
2227 like 2215 but
u= 62 imp:n= 1.0
c Fuel Pellet, #5, L7
2228 like 2212 but
mat= 57 rho= -9.9400E+00
u= 63 imp:n= 1.0
c Fuel/Cladding Gap, #5, L7
2229 like 2213 but
u= 63 imp:n= 1.0
c Cladding, #5, L7
2230 like 2214 but
u= 63 imp:n= 1.0
c Water Outside Cladding #5, L7
2231 like 2215 but
u= 63 imp:n= 1.0
c Fuel Pellet, #6, L7
2232 like 2212 but
mat= 58 rho= -9.9400E+00
u= 64 imp:n= 1.0
c Fuel/Cladding Gap, #6, L7
2233 like 2213 but
u= 64 imp:n= 1.0
c Cladding, #6, L7
2234 like 2214 but
u= 64 imp:n= 1.0
c Water Outside Cladding #6, L7
2235 like 2215 but
u= 64 imp:n= 1.0

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 17 of 76

```

c      Fuel Pellet, #7, L7
2236  like 2212 but
      mat= 59 rho= -1.0300E+01
      u= 65 imp:n= 1.0
c      Fuel/Cladding Gap, #7, L7
2237  like 2213 but
      u= 65 imp:n= 1.0
c      Cladding, #7, L7
2238  like 2214 but
      u= 65 imp:n= 1.0
c      Water Outside Cladding #7, L7
2239  like 2215 but
      u= 65 imp:n= 1.0
c      Reference Fuel Rod Cell, L7
2240      8 -9.8114E-01 -43 44 45 -46 lat= 1 u = 66 imp:n= 1.0
      fill= -1:7 -1:7 0:0
          66 66 66 66 66 66 66 66 66
          66 64 64 63 63 63 63 64 66
          66 64 62 62 59 59 60 63 66
          66 63 62 59 59 59 59 60 66
          66 63 59 59 61 59 65 60 66
          66 63 59 59 59 59 59 60 66
          66 63 60 59 65 59 59 60 66
          66 64 63 60 60 60 60 63 66
          66 66 66 66 66 66 66 66 66
c      Channel, L7
2241      18 -6.5600E+00 ( 47 -48 -49 50):( -51 52 53 -54):
      ( -55 56 53 -54):( 57 -58 50 -49):
      ( -59 60 49 -53):( -61 62 49 54):
      ( -63 64 54 -50):( -65 66 -50 -53)
      u= 67 imp:n= 1.0
c      Active Fuel Area, L7
2242      8 -9.8114E-01 ( 53 -54 -52 55):( 48 -53 -49 50 ):
      ( 54 -57 -49 50):( -60 -53 49):
      ( -62 54 49):( -64 54 -50):
      ( -66 -53 -50)
      fill= 66 u= 67 imp:n= 1.0
c      Water Outside of Channel, L7
2243      2 -9.8114E-01 ( -47 : 51 : 58 : -56 ):
      ( 59 -53 49 ):
      ( 61 54 49 ):
      ( 63 54 -50 ):
      ( 65 -53 -50 )
      u= 67 imp:n= 1.0
c      Fuel Pellet, #1, L8
2244      60 -1.0340E+01 -40          u= 68 imp:n= 1.0
c      Fuel/Cladding Gap, #1, L8
2245      0 40 -41          u= 68 imp:n= 1.0
c      Cladding, #1, L8
2246      10 -6.5600E+00 41 -42          u= 68 imp:n= 1.0
c      Water Outside Cladding #1, L8
2247      8 -9.8114E-01 42
      u= 68 imp:n= 1.0
c      Fuel Pellet, #2, L8

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 18 of 76

2248 like 2244 but
mat= 61 rho= -9.9400E+00
u= 69 imp:n= 1.0
c Fuel/Cladding Gap, #2, L8
2249 like 2245 but
u= 69 imp:n= 1.0
c Cladding, #2, L8
2250 like 2246 but
u= 69 imp:n= 1.0
c Water Outside Cladding #2, L8
2251 like 2247 but
u= 69 imp:n= 1.0
c Fuel Pellet, #3, L8
2252 like 2244 but
mat= 62 rho= -9.4800E+00
u= 70 imp:n= 1.0
c Fuel/Cladding Gap, #3, L8
2253 like 2245 but
u= 70 imp:n= 1.0
c Cladding, #3, L8
2254 like 2246 but
u= 70 imp:n= 1.0
c Water Outside Cladding #3, L8
2255 like 2247 but
u= 70 imp:n= 1.0
c Fuel Pellet, #4, L8
2256 like 2244 but
mat= 63 rho= -1.0340E+01
u= 71 imp:n= 1.0
c Fuel/Cladding Gap, #4, L8
2257 like 2245 but
u= 71 imp:n= 1.0
c Cladding, #4, L8
2258 like 2246 but
u= 71 imp:n= 1.0
c Water Outside Cladding #4, L8
2259 like 2247 but
u= 71 imp:n= 1.0
c Fuel Pellet, #5, L8
2260 like 2244 but
mat= 64 rho= -9.9400E+00
u= 72 imp:n= 1.0
c Fuel/Cladding Gap, #5, L8
2261 like 2245 but
u= 72 imp:n= 1.0
c Cladding, #5, L8
2262 like 2246 but
u= 72 imp:n= 1.0
c Water Outside Cladding #5, L8
2263 like 2247 but
u= 72 imp:n= 1.0
c Fuel Pellet, #6, L8
2264 like 2244 but
mat= 65 rho= -9.9400E+00

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 19 of 76

```

      u= 73 imp:n= 1.0
c    Fuel/Cladding Gap, #6, L8
2265 like 2245 but
      u= 73 imp:n= 1.0
c    Cladding, #6, L8
2266 like 2246 but
      u= 73 imp:n= 1.0
c    Water Outside Cladding #6, L8
2267 like 2247 but
      u= 73 imp:n= 1.0
c    Fuel Pellet, #7, L8
2268 like 2244 but
      mat= 66 rho= -1.0290E+01
      u= 74 imp:n= 1.0
c    Fuel/Cladding Gap, #7, L8
2269 like 2245 but
      u= 74 imp:n= 1.0
c    Cladding, #7, L8
2270 like 2246 but
      u= 74 imp:n= 1.0
c    Water Outside Cladding #7, L8
2271 like 2247 but
      u= 74 imp:n= 1.0
c    Reference Fuel Rod Cell, L8
2272      8 -9.8114E-01 -43 44 45 -46 lat= 1 u = 75 imp:n= 1.0
      fill= -1:7 -1:7 0:0
          75 75 75 75 75 75 75 75 75
          75 73 73 72 72 72 72 73 75
          75 73 71 71 68 68 69 72 75
          75 72 71 68 68 68 68 69 75
          75 72 68 68 70 68 74 69 75
          75 72 68 68 68 68 68 69 75
          75 72 69 68 74 68 68 69 75
          75 73 72 69 69 69 69 72 75
          75 75 75 75 75 75 75 75 75
c    Channel, L8
2273      18 -6.5600E+00 ( 47 -48 -49 50):( -51 52 53 -54):
      ( -55 56 53 -54):( 57 -58 50 -49):
      ( -59 60 49 -53):( -61 62 49 54):
      ( -63 64 54 -50):( -65 66 -50 -53)
      u= 76 imp:n= 1.0
c    Active Fuel Area, L8
2274      8 -9.8114E-01 ( 53 -54 -52 55):( 48 -53 -49 50 ):
      ( 54 -57 -49 50):( -60 -53 49):
      ( -62 54 49):( -64 54 -50):
      ( -66 -53 -50)
      fill= 75 u= 76 imp:n= 1.0
c    Water Outside of Channel, L8
2275      2 -9.8114E-01 ( -47 : 51 : 58 : -56 ):
      ( 59 -53 49 ):
      ( 61 54 49 ):
      ( 63 54 -50 ):
      ( 65 -53 -50 )
      u= 76 imp:n= 1.0

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 20 of 76

c Fuel Pellet, #1, L9
2276 67 -1.0340E+01 -40 u= 77 imp:n= 1.0

c Fuel/Cladding Gap, #1, L9
2277 0 40 -41 u= 77 imp:n= 1.0

c Cladding, #1, L9
2278 10 -6.5600E+00 41 -42 u= 77 imp:n= 1.0

c Water Outside Cladding #1, L9
2279 8 -9.8114E-01 42
u= 77 imp:n= 1.0

c Fuel Pellet, #2, L9
2280 like 2276 but
mat= 68 rho= -9.4800E+00
u= 78 imp:n= 1.0

c Fuel/Cladding Gap, #2, L9
2281 like 2277 but
u= 78 imp:n= 1.0

c Cladding, #2, L9
2282 like 2278 but
u= 78 imp:n= 1.0

c Water Outside Cladding #2, L9
2283 like 2279 but
u= 78 imp:n= 1.0

c Fuel Pellet, #3, L9
2284 like 2276 but
mat= 69 rho= -1.0340E+01
u= 79 imp:n= 1.0

c Fuel/Cladding Gap, #3, L9
2285 like 2277 but
u= 79 imp:n= 1.0

c Cladding, #3, L9
2286 like 2278 but
u= 79 imp:n= 1.0

c Water Outside Cladding #3, L9
2287 like 2279 but
u= 79 imp:n= 1.0

c Fuel Pellet, #4, L9
2288 like 2276 but
mat= 70 rho= -1.0340E+01
u= 80 imp:n= 1.0

c Fuel/Cladding Gap, #4, L9
2289 like 2277 but
u= 80 imp:n= 1.0

c Cladding, #4, L9
2290 like 2278 but
u= 80 imp:n= 1.0

c Water Outside Cladding #4, L9
2291 like 2279 but
u= 80 imp:n= 1.0

c Fuel Pellet, #5, L9
2292 like 2276 but
mat= 71 rho= -1.0300E+01
u= 81 imp:n= 1.0

c Fuel/Cladding Gap, #5, L9
2293 like 2277 but

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 21 of 76

```

      u= 81 imp:n= 1.0
c    Cladding, #5, L9
2294 like 2278 but
      u= 81 imp:n= 1.0
c    Water Outside Cladding #5, L9
2295 like 2279 but
      u= 81 imp:n= 1.0
c    Reference Fuel Rod Cell, L9
2296      8 -9.8114E-01 -43 44 45 -46 lat= 1 u = 82 imp:n= 1.0
      fill= -1:7 -1:7 0:0
          82 82 82 82 82 82 82 82 82
          82 80 80 79 79 79 79 80 82
          82 80 79 79 77 77 77 79 82
          82 79 79 77 77 77 77 77 82
          82 79 77 77 78 77 81 77 82
          82 79 77 77 77 77 77 77 82
          82 79 77 77 81 77 77 77 82
          82 80 79 77 77 77 77 79 82
          82 82 82 82 82 82 82 82 82
c    Channel, L9
2297      18 -6.5600E+00 ( 47 -48 -49 50):( -51 52 53 -54):
      ( -55 56 53 -54):( 57 -58 50 -49):
      ( -59 60 49 -53):( -61 62 49 54):
      ( -63 64 54 -50):( -65 66 -50 -53)
      u= 83 imp:n= 1.0
c    Active Fuel Area, L9
2298      8 -9.8114E-01 ( 53 -54 -52 55):( 48 -53 -49 50 ):
      ( 54 -57 -49 50):( -60 -53 49):
      ( -62 54 49):( -64 54 -50):
      ( -66 -53 -50)
      fill= 82 u= 83 imp:n= 1.0
c    Water Outside of Channel, L9
2299      2 -9.8114E-01 ( -47 : 51 : 58 : -56 ):
      ( 59 -53 49 ):
      ( 61 54 49 ):
      ( 63 54 -50 ):
      ( 65 -53 -50 )
      u= 83 imp:n= 1.0
c    Fuel Pellet, #1, L10
2300      72 -1.0340E+01 -40          u= 84 imp:n= 1.0
c    Fuel/Cladding Gap, #1, L10
2301      0 40 -41          u= 84 imp:n= 1.0
c    Cladding, #1, L10
2302      10 -6.5600E+00 41 -42          u= 84 imp:n= 1.0
c    Water Outside Cladding #1, L10
2303      8 -9.8114E-01 42
      u= 84 imp:n= 1.0
c    Fuel Pellet, #2, L10
2304 like 2300 but
      mat= 73 rho= -9.9400E+00
      u= 85 imp:n= 1.0
c    Fuel/Cladding Gap, #2, L10
2305 like 2301 but
      u= 85 imp:n= 1.0

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 22 of 76

c Cladding, #2, L10
2306 like 2302 but
u= 85 imp:n= 1.0

c Water Outside Cladding #2, L10
2307 like 2303 but
u= 85 imp:n= 1.0

c Fuel Pellet, #3, L10
2308 like 2300 but
mat= 74 rho= -9.4800E+00
u= 86 imp:n= 1.0

c Fuel/Cladding Gap, #3, L10
2309 like 2301 but
u= 86 imp:n= 1.0

c Cladding, #3, L10
2310 like 2302 but
u= 86 imp:n= 1.0

c Water Outside Cladding #3, L10
2311 like 2303 but
u= 86 imp:n= 1.0

c Fuel Pellet, #4, L10
2312 like 2300 but
mat= 75 rho= -1.0340E+01
u= 87 imp:n= 1.0

c Fuel/Cladding Gap, #4, L10
2313 like 2301 but
u= 87 imp:n= 1.0

c Cladding, #4, L10
2314 like 2302 but
u= 87 imp:n= 1.0

c Water Outside Cladding #4, L10
2315 like 2303 but
u= 87 imp:n= 1.0

c Fuel Pellet, #5, L10
2316 like 2300 but
mat= 76 rho= -9.9400E+00
u= 88 imp:n= 1.0

c Fuel/Cladding Gap, #5, L10
2317 like 2301 but
u= 88 imp:n= 1.0

c Cladding, #5, L10
2318 like 2302 but
u= 88 imp:n= 1.0

c Water Outside Cladding #5, L10
2319 like 2303 but
u= 88 imp:n= 1.0

c Fuel Pellet, #6, L10
2320 like 2300 but
mat= 77 rho= -9.9400E+00
u= 89 imp:n= 1.0

c Fuel/Cladding Gap, #6, L10
2321 like 2301 but
u= 89 imp:n= 1.0

c Cladding, #6, L10
2322 like 2302 but

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 23 of 76

```

      u= 89 imp:n= 1.0
c    Water Outside Cladding #6, L10
2323 like 2303 but
      u= 89 imp:n= 1.0
c    Fuel Pellet, #7, L10
2324 like 2300 but
      mat= 78 rho= -1.0290E+01
      u= 90 imp:n= 1.0
c    Fuel/Cladding Gap, #7, L10
2325 like 2301 but
      u= 90 imp:n= 1.0
c    Cladding, #7, L10
2326 like 2302 but
      u= 90 imp:n= 1.0
c    Water Outside Cladding #7, L10
2327 like 2303 but
      u= 90 imp:n= 1.0
c    Reference Fuel Rod Cell, L10
2328      8 -9.8114E-01 -43 44 45 -46 lat= 1 u = 91 imp:n= 1.0
      fill= -1:7 -1:7 0:0
          91 91 91 91 91 91 91 91 91
          91 89 89 88 88 88 88 89 91
          91 89 87 87 84 84 85 88 91
          91 88 87 84 84 84 84 85 91
          91 88 84 84 86 84 90 85 91
          91 88 84 84 84 84 84 85 91
          91 88 85 84 90 84 84 85 91
          91 89 88 85 85 85 85 88 91
          91 91 91 91 91 91 91 91 91
c    Channel, L10
2329      18 -6.5600E+00 ( 47 -48 -49 50):( -51 52 53 -54):
      ( -55 56 53 -54):( 57 -58 50 -49):
      ( -59 60 49 -53):( -61 62 49 54):
      ( -63 64 54 -50):( -65 66 -50 -53)
      u= 92 imp:n= 1.0
c    Active Fuel Area, L10
2330      8 -9.8114E-01 ( 53 -54 -52 55):( 48 -53 -49 50 ):
      ( 54 -57 -49 50):( -60 -53 49):
      ( -62 54 49):( -64 54 -50):
      ( -66 -53 -50)
      fill= 91 u= 92 imp:n= 1.0
c    Water Outside of Channel, L10
2331      2 -9.8114E-01 ( -47 : 51 : 58 : -56 ):
      ( 59 -53 49 ):
      ( 61 54 49 ):
      ( 63 54 -50 ):
      ( 65 -53 -50 )
      u= 92 imp:n= 1.0
c    Fuel Pellet, #1, L2
2332      19 -1.0340E+01 -40      u= 93 imp:n= 1.0
c    Fuel/Cladding Gap, #1, L2
2333      0 40 -41      u= 93 imp:n= 1.0
c    Cladding, #1, L2
2334      10 -6.5600E+00 41 -42      u= 93 imp:n= 1.0

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 24 of 76

c Water Outside Cladding #1, L2
2335 79 -1.1790E+00 42
u= 93 imp:n= 1.0

c Fuel Pellet, #2, L2
2336 like 2332 but
mat= 20 rho= -9.9400E+00
u= 94 imp:n= 1.0

c Fuel/Cladding Gap, #2, L2
2337 like 2333 but
u= 94 imp:n= 1.0

c Cladding, #2, L2
2338 like 2334 but
u= 94 imp:n= 1.0

c Water Outside Cladding #2, L2
2339 like 2335 but
u= 94 imp:n= 1.0

c Fuel Pellet, #3, L2
2340 like 2332 but
mat= 21 rho= -9.4800E+00
u= 95 imp:n= 1.0

c Fuel/Cladding Gap, #3, L2
2341 like 2333 but
u= 95 imp:n= 1.0

c Cladding, #3, L2
2342 like 2334 but
u= 95 imp:n= 1.0

c Water Outside Cladding #3, L2
2343 like 2335 but
u= 95 imp:n= 1.0

c Fuel Pellet, #4, L2
2344 like 2332 but
mat= 22 rho= -1.0340E+01
u= 96 imp:n= 1.0

c Fuel/Cladding Gap, #4, L2
2345 like 2333 but
u= 96 imp:n= 1.0

c Cladding, #4, L2
2346 like 2334 but
u= 96 imp:n= 1.0

c Water Outside Cladding #4, L2
2347 like 2335 but
u= 96 imp:n= 1.0

c Fuel Pellet, #5, L2
2348 like 2332 but
mat= 23 rho= -9.9400E+00
u= 97 imp:n= 1.0

c Fuel/Cladding Gap, #5, L2
2349 like 2333 but
u= 97 imp:n= 1.0

c Cladding, #5, L2
2350 like 2334 but
u= 97 imp:n= 1.0

c Water Outside Cladding #5, L2
2351 like 2335 but

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 25 of 76

```

      u= 97 imp:n= 1.0
c    Fuel Pellet, #6, L2
2352 like 2332 but
      mat= 24 rho= -9.9400E+00
      u= 98 imp:n= 1.0
c    Fuel/Cladding Gap, #6, L2
2353 like 2333 but
      u= 98 imp:n= 1.0
c    Cladding, #6, L2
2354 like 2334 but
      u= 98 imp:n= 1.0
c    Water Outside Cladding #6, L2
2355 like 2335 but
      u= 98 imp:n= 1.0
c    Fuel Pellet, #7, L2
2356 like 2332 but
      mat= 25 rho= -1.0300E+01
      u= 99 imp:n= 1.0
c    Fuel/Cladding Gap, #7, L2
2357 like 2333 but
      u= 99 imp:n= 1.0
c    Cladding, #7, L2
2358 like 2334 but
      u= 99 imp:n= 1.0
c    Water Outside Cladding #7, L2
2359 like 2335 but
      u= 99 imp:n= 1.0
c    Fuel Pellet, #8, L2
2360 like 2332 but
      mat= 26 rho= -1.0340E+01
      u= 100 imp:n= 1.0
c    Fuel/Cladding Gap, #8, L2
2361 like 2333 but
      u= 100 imp:n= 1.0
c    Cladding, #8, L2
2362 like 2334 but
      u= 100 imp:n= 1.0
c    Water Outside Cladding #8, L2
2363 like 2335 but
      u= 100 imp:n= 1.0
c    Reference Fuel Rod Cell, L2
2364 79 -1.1790E+00 -43 44 45 -46 lat= 1 u = 101 imp:n= 1.0
      fill= -1:7 -1:7 0:0
      101 101 101 101 101 101 101 101 101
      101 98 98 97 97 97 97 98 101
      101 98 96 96 93 93 94 97 101
      101 97 96 100 93 93 93 94 101
      101 97 93 93 95 93 99 94 101
      101 97 93 93 93 93 93 94 101
      101 97 94 93 99 93 93 94 101
      101 98 97 94 94 94 94 97 101
      101 101 101 101 101 101 101 101 101
c    Channel, L2
2365 18 -6.5600E+00 ( 47 -48 -49 50):( -51 52 53 -54):

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 26 of 76

```
( -55 56 53 -54):( 57 -58 50 -49):  
( -59 60 49 -53):( -61 62 49 54):  
( -63 64 54 -50):( -65 66 -50 -53)  
u= 102 imp:n= 1.0  
c Active Fuel Area, L2  
2366 79 -1.1790E+00 ( 53 -54 -52 55):( 48 -53 -49 50 ):  
( 54 -57 -49 50):( -60 -53 49):  
( -62 54 49):( -64 54 -50):  
( -66 -53 -50)  
fill= 101 u= 102 imp:n= 1.0  
c Water Outside of Channel, L2  
2367 2 -9.8114E-01 ( -47 : 51 : 58 : -56 ):  
( 59 -53 49 ):  
( 61 54 49 ):  
( 63 54 -50 ):  
( 65 -53 -50 )  
u= 102 imp:n= 1.0  
c Fuel Pellet, #1, L3  
2368 27 -1.0340E+01 -40 u= 103 imp:n= 1.0  
c Fuel/Cladding Gap, #1, L3  
2369 0 40 -41 u= 103 imp:n= 1.0  
c Cladding, #1, L3  
2370 10 -6.5600E+00 41 -42 u= 103 imp:n= 1.0  
c Water Outside Cladding #1, L3  
2371 79 -1.1790E+00 42  
u= 103 imp:n= 1.0  
c Fuel Pellet, #2, L3  
2372 like 2368 but  
mat= 28 rho= -9.9400E+00  
u= 104 imp:n= 1.0  
c Fuel/Cladding Gap, #2, L3  
2373 like 2369 but  
u= 104 imp:n= 1.0  
c Cladding, #2, L3  
2374 like 2370 but  
u= 104 imp:n= 1.0  
c Water Outside Cladding #2, L3  
2375 like 2371 but  
u= 104 imp:n= 1.0  
c Fuel Pellet, #3, L3  
2376 like 2368 but  
mat= 29 rho= -9.4800E+00  
u= 105 imp:n= 1.0  
c Fuel/Cladding Gap, #3, L3  
2377 like 2369 but  
u= 105 imp:n= 1.0  
c Cladding, #3, L3  
2378 like 2370 but  
u= 105 imp:n= 1.0  
c Water Outside Cladding #3, L3  
2379 like 2371 but  
u= 105 imp:n= 1.0  
c Fuel Pellet, #4, L3  
2380 like 2368 but
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 27 of 76

```
mat= 30 rho= -1.0340E+01
u= 106 imp:n= 1.0
c Fuel/Cladding Gap, #4, L3
2381 like 2369 but
u= 106 imp:n= 1.0
c Cladding, #4, L3
2382 like 2370 but
u= 106 imp:n= 1.0
c Water Outside Cladding #4, L3
2383 like 2371 but
u= 106 imp:n= 1.0
c Fuel Pellet, #5, L3
2384 like 2368 but
mat= 31 rho= -9.9400E+00
u= 107 imp:n= 1.0
c Fuel/Cladding Gap, #5, L3
2385 like 2369 but
u= 107 imp:n= 1.0
c Cladding, #5, L3
2386 like 2370 but
u= 107 imp:n= 1.0
c Water Outside Cladding #5, L3
2387 like 2371 but
u= 107 imp:n= 1.0
c Fuel Pellet, #6, L3
2388 like 2368 but
mat= 32 rho= -9.9400E+00
u= 108 imp:n= 1.0
c Fuel/Cladding Gap, #6, L3
2389 like 2369 but
u= 108 imp:n= 1.0
c Cladding, #6, L3
2390 like 2370 but
u= 108 imp:n= 1.0
c Water Outside Cladding #6, L3
2391 like 2371 but
u= 108 imp:n= 1.0
c Fuel Pellet, #7, L3
2392 like 2368 but
mat= 33 rho= -1.0300E+01
u= 109 imp:n= 1.0
c Fuel/Cladding Gap, #7, L3
2393 like 2369 but
u= 109 imp:n= 1.0
c Cladding, #7, L3
2394 like 2370 but
u= 109 imp:n= 1.0
c Water Outside Cladding #7, L3
2395 like 2371 but
u= 109 imp:n= 1.0
c Fuel Pellet, #8, L3
2396 like 2368 but
mat= 34 rho= -1.0340E+01
u= 110 imp:n= 1.0
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 28 of 76

```
c    Fuel/Cladding Gap, #8, L3
2397 like 2369 but
      u= 110 imp:n= 1.0
c    Cladding, #8, L3
2398 like 2370 but
      u= 110 imp:n= 1.0
c    Water Outside Cladding #8, L3
2399 like 2371 but
      u= 110 imp:n= 1.0
c    Reference Fuel Rod Cell, L3
2400   79 -1.1790E+00 -43 44 45 -46 lat= 1 u = 111 imp:n= 1.0
      fill= -1:7 -1:7 0:0
          111 111 111 111 111 111 111 111 111
          111 108 108 107 107 107 107 108 111
          111 108 106 106 103 103 104 107 111
          111 107 106 110 103 103 103 104 111
          111 107 103 103 105 103 109 104 111
          111 107 103 103 103 103 103 104 111
          111 107 104 103 109 103 103 104 111
          111 108 107 104 104 104 104 107 111
          111 111 111 111 111 111 111 111 111
c    Channel, L3
2401   18 -6.5600E+00 ( 47 -48 -49 50):( -51 52 53 -54):
      ( -55 56 53 -54):( 57 -58 50 -49):
      ( -59 60 49 -53):( -61 62 49 54):
      ( -63 64 54 -50):( -65 66 -50 -53)
      u= 112 imp:n= 1.0
c    Active Fuel Area, L3
2402   79 -1.1790E+00 ( 53 -54 -52 55):( 48 -53 -49 50 ):
      ( 54 -57 -49 50):( -60 -53 49):
      ( -62 54 49):( -64 54 -50):
      ( -66 -53 -50)
      fill= 111 u= 112 imp:n= 1.0
c    Water Outside of Channel, L3
2403   2 -9.8114E-01 ( -47 : 51 : 58 : -56 ):
      ( 59 -53 49 ):
      ( 61 54 49 ):
      ( 63 54 -50 ):
      ( 65 -53 -50 )
      u= 112 imp:n= 1.0
c    Fuel Pellet, #1, L5
2404   41 -1.0340E+01 -40      u= 113 imp:n= 1.0
c    Fuel/Cladding Gap, #1, L5
2405   0 40 -41      u= 113 imp:n= 1.0
c    Cladding, #1, L5
2406   10 -6.5600E+00 41 -42      u= 113 imp:n= 1.0
c    Water Outside Cladding #1, L5
2407   79 -1.1790E+00 42
      u= 113 imp:n= 1.0
c    Fuel Pellet, #2, L5
2408 like 2404 but
      mat= 42 rho= -9.4800E+00
      u= 114 imp:n= 1.0
c    Fuel/Cladding Gap, #2, L5
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 29 of 76

2409 like 2405 but
u= 114 imp:n= 1.0
c Cladding, #2, L5
2410 like 2406 but
u= 114 imp:n= 1.0
c Water Outside Cladding #2, L5
2411 like 2407 but
u= 114 imp:n= 1.0
c Fuel Pellet, #3, L5
2412 like 2404 but
mat= 43 rho= -1.0340E+01
u= 115 imp:n= 1.0
c Fuel/Cladding Gap, #3, L5
2413 like 2405 but
u= 115 imp:n= 1.0
c Cladding, #3, L5
2414 like 2406 but
u= 115 imp:n= 1.0
c Water Outside Cladding #3, L5
2415 like 2407 but
u= 115 imp:n= 1.0
c Fuel Pellet, #4, L5
2416 like 2404 but
mat= 44 rho= -1.0340E+01
u= 116 imp:n= 1.0
c Fuel/Cladding Gap, #4, L5
2417 like 2405 but
u= 116 imp:n= 1.0
c Cladding, #4, L5
2418 like 2406 but
u= 116 imp:n= 1.0
c Water Outside Cladding #4, L5
2419 like 2407 but
u= 116 imp:n= 1.0
c Fuel Pellet, #5, L5
2420 like 2404 but
mat= 45 rho= -1.0260E+01
u= 117 imp:n= 1.0
c Fuel/Cladding Gap, #5, L5
2421 like 2405 but
u= 117 imp:n= 1.0
c Cladding, #5, L5
2422 like 2406 but
u= 117 imp:n= 1.0
c Water Outside Cladding #5, L5
2423 like 2407 but
u= 117 imp:n= 1.0
c Fuel Pellet, #6, L5
2424 like 2404 but
mat= 46 rho= -1.0340E+01
u= 118 imp:n= 1.0
c Fuel/Cladding Gap, #6, L5
2425 like 2405 but
u= 118 imp:n= 1.0

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 30 of 76

```
c   Cladding, #6, L5
2426 like 2406 but
    u= 118 imp:n= 1.0
c   Water Outside Cladding #6, L5
2427 like 2407 but
    u= 118 imp:n= 1.0
c   Reference Fuel Rod Cell, L5
2428   79 -1.1790E+00 -43 44 45 -46 lat= 1 u = 119 imp:n= 1.0
    fill= -1:7 -1:7 0:0
        119 119 119 119 119 119 119 119 119
        119 116 116 115 115 115 115 116 119
        119 116 115 115 113 113 113 115 119
        119 115 115 118 113 113 113 113 119
        119 115 113 113 114 113 117 113 119
        119 115 113 113 113 113 113 113 119
        119 115 113 113 117 113 113 113 119
        119 116 115 113 113 113 113 115 119
        119 119 119 119 119 119 119 119 119
c   Channel, L5
2429   18 -6.5600E+00 ( 47 -48 -49 50):( -51 52 53 -54):
    ( -55 56 53 -54):( 57 -58 50 -49):
    ( -59 60 49 -53):( -61 62 49 54):
    ( -63 64 54 -50):( -65 66 -50 -53)
    u= 120 imp:n= 1.0
c   Active Fuel Area, L5
2430   79 -1.1790E+00 ( 53 -54 -52 55):( 48 -53 -49 50 ):
    ( 54 -57 -49 50):( -60 -53 49):
    ( -62 54 49):( -64 54 -50):
    ( -66 -53 -50)
    fill= 119 u= 120 imp:n= 1.0
c   Water Outside of Channel, L5
2431   2 -9.8114E-01 ( -47 : 51 : 58 : -56 ):
    ( 59 -53 49 ):
    ( 61 54 49 ):
    ( 63 54 -50 ):
    ( 65 -53 -50 )
    u= 120 imp:n= 1.0
c   Fuel Pellet, #1, L6
2432   47 -1.0340E+01 -40      u= 121 imp:n= 1.0
c   Fuel/Cladding Gap, #1, L6
2433   0 40 -41      u= 121 imp:n= 1.0
c   Cladding, #1, L6
2434   10 -6.5600E+00 41 -42      u= 121 imp:n= 1.0
c   Water Outside Cladding #1, L6
2435   79 -1.1790E+00 42
    u= 121 imp:n= 1.0
c   Fuel Pellet, #2, L6
2436 like 2432 but
    mat= 48 rho= -9.4800E+00
    u= 122 imp:n= 1.0
c   Fuel/Cladding Gap, #2, L6
2437 like 2433 but
    u= 122 imp:n= 1.0
c   Cladding, #2, L6
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 31 of 76

2438 like 2434 but
u= 122 imp:n= 1.0
c Water Outside Cladding #2, L6
2439 like 2435 but
u= 122 imp:n= 1.0
c Fuel Pellet, #3, L6
2440 like 2432 but
mat= 49 rho= -1.0340E+01
u= 123 imp:n= 1.0
c Fuel/Cladding Gap, #3, L6
2441 like 2433 but
u= 123 imp:n= 1.0
c Cladding, #3, L6
2442 like 2434 but
u= 123 imp:n= 1.0
c Water Outside Cladding #3, L6
2443 like 2435 but
u= 123 imp:n= 1.0
c Fuel Pellet, #4, L6
2444 like 2432 but
mat= 50 rho= -1.0340E+01
u= 124 imp:n= 1.0
c Fuel/Cladding Gap, #4, L6
2445 like 2433 but
u= 124 imp:n= 1.0
c Cladding, #4, L6
2446 like 2434 but
u= 124 imp:n= 1.0
c Water Outside Cladding #4, L6
2447 like 2435 but
u= 124 imp:n= 1.0
c Fuel Pellet, #5, L6
2448 like 2432 but
mat= 51 rho= -1.0260E+01
u= 125 imp:n= 1.0
c Fuel/Cladding Gap, #5, L6
2449 like 2433 but
u= 125 imp:n= 1.0
c Cladding, #5, L6
2450 like 2434 but
u= 125 imp:n= 1.0
c Water Outside Cladding #5, L6
2451 like 2435 but
u= 125 imp:n= 1.0
c Fuel Pellet, #6, L6
2452 like 2432 but
mat= 52 rho= -1.0340E+01
u= 126 imp:n= 1.0
c Fuel/Cladding Gap, #6, L6
2453 like 2433 but
u= 126 imp:n= 1.0
c Cladding, #6, L6
2454 like 2434 but
u= 126 imp:n= 1.0

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 32 of 76

```
c   Water Outside Cladding #6, L6
2455 like 2435 but
    u= 126 imp:n= 1.0
c   Reference Fuel Rod Cell, L6
2456   79 -1.1790E+00 -43 44 45 -46 lat= 1 u = 127 imp:n= 1.0
    fill= -1:7 -1:7 0:0
        127 127 127 127 127 127 127 127 127
        127 124 124 123 123 123 123 124 127
        127 124 123 123 121 121 121 123 127
        127 123 123 126 121 121 121 121 127
        127 123 121 121 122 121 125 121 127
        127 123 121 121 121 121 121 121 127
        127 123 121 121 125 121 121 121 127
        127 124 123 121 121 121 121 123 127
        127 127 127 127 127 127 127 127 127
c   Channel, L6
2457   18 -6.5600E+00 ( 47 -48 -49 50):( -51 52 53 -54):
    ( -55 56 53 -54):( 57 -58 50 -49):
    ( -59 60 49 -53):( -61 62 49 54):
    ( -63 64 54 -50):( -65 66 -50 -53)
    u= 128 imp:n= 1.0
c   Active Fuel Area, L6
2458   79 -1.1790E+00 ( 53 -54 -52 55):( 48 -53 -49 50 ):
    ( 54 -57 -49 50):( -60 -53 49):
    ( -62 54 49):( -64 54 -50):
    ( -66 -53 -50)
    fill= 127 u= 128 imp:n= 1.0
c   Water Outside of Channel, L6
2459   2 -9.8114E-01 ( -47 : 51 : 58 : -56 ):
    ( 59 -53 49 ):
    ( 61 54 49 ):
    ( 63 54 -50 ):
    ( 65 -53 -50 )
    u= 128 imp:n= 1.0
c   Fuel Pellet, #1, L8
2460   60 -1.0340E+01 -40      u= 129 imp:n= 1.0
c   Fuel/Cladding Gap, #1, L8
2461   0 40 -41      u= 129 imp:n= 1.0
c   Cladding, #1, L8
2462   10 -6.5600E+00 41 -42      u= 129 imp:n= 1.0
c   Water Outside Cladding #1, L8
2463   79 -1.1790E+00 42
    u= 129 imp:n= 1.0
c   Fuel Pellet, #2, L8
2464 like 2460 but
    mat= 61 rho= -9.9400E+00
    u= 130 imp:n= 1.0
c   Fuel/Cladding Gap, #2, L8
2465 like 2461 but
    u= 130 imp:n= 1.0
c   Cladding, #2, L8
2466 like 2462 but
    u= 130 imp:n= 1.0
c   Water Outside Cladding #2, L8
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 33 of 76

2467 like 2463 but
u= 130 imp:n= 1.0
c Fuel Pellet, #3, L8
2468 like 2460 but
mat= 62 rho= -9.4800E+00
u= 131 imp:n= 1.0
c Fuel/Cladding Gap, #3, L8
2469 like 2461 but
u= 131 imp:n= 1.0
c Cladding, #3, L8
2470 like 2462 but
u= 131 imp:n= 1.0
c Water Outside Cladding #3, L8
2471 like 2463 but
u= 131 imp:n= 1.0
c Fuel Pellet, #4, L8
2472 like 2460 but
mat= 63 rho= -1.0340E+01
u= 132 imp:n= 1.0
c Fuel/Cladding Gap, #4, L8
2473 like 2461 but
u= 132 imp:n= 1.0
c Cladding, #4, L8
2474 like 2462 but
u= 132 imp:n= 1.0
c Water Outside Cladding #4, L8
2475 like 2463 but
u= 132 imp:n= 1.0
c Fuel Pellet, #5, L8
2476 like 2460 but
mat= 64 rho= -9.9400E+00
u= 133 imp:n= 1.0
c Fuel/Cladding Gap, #5, L8
2477 like 2461 but
u= 133 imp:n= 1.0
c Cladding, #5, L8
2478 like 2462 but
u= 133 imp:n= 1.0
c Water Outside Cladding #5, L8
2479 like 2463 but
u= 133 imp:n= 1.0
c Fuel Pellet, #6, L8
2480 like 2460 but
mat= 65 rho= -9.9400E+00
u= 134 imp:n= 1.0
c Fuel/Cladding Gap, #6, L8
2481 like 2461 but
u= 134 imp:n= 1.0
c Cladding, #6, L8
2482 like 2462 but
u= 134 imp:n= 1.0
c Water Outside Cladding #6, L8
2483 like 2463 but
u= 134 imp:n= 1.0

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 34 of 76

```

c    Fuel Pellet, #7, L8
2484 like 2460 but
      mat= 66 rho= -1.0290E+01
      u= 135 imp:n= 1.0
c    Fuel/Cladding Gap, #7, L8
2485 like 2461 but
      u= 135 imp:n= 1.0
c    Cladding, #7, L8
2486 like 2462 but
      u= 135 imp:n= 1.0
c    Water Outside Cladding #7, L8
2487 like 2463 but
      u= 135 imp:n= 1.0
c    Reference Fuel Rod Cell, L8
2488   79 -1.1790E+00 -43 44 45 -46 lat= 1 u = 136 imp:n= 1.0
      fill= -1:7 -1:7 0:0
          136 136 136 136 136 136 136 136 136
          136 134 134 133 133 133 133 134 136
          136 134 132 132 129 129 130 133 136
          136 133 132 129 129 129 129 130 136
          136 133 129 129 131 129 135 130 136
          136 133 129 129 129 129 129 130 136
          136 133 130 129 135 129 129 130 136
          136 134 133 130 130 130 130 133 136
          136 136 136 136 136 136 136 136 136
c    Channel, L8
2489   18 -6.5600E+00 ( 47 -48 -49 50):( -51 52 53 -54):
      ( -55 56 53 -54):( 57 -58 50 -49):
      ( -59 60 49 -53):( -61 62 49 54):
      ( -63 64 54 -50):( -65 66 -50 -53)
      u= 137 imp:n= 1.0
c    Active Fuel Area, L8
2490   79 -1.1790E+00 ( 53 -54 -52 55):( 48 -53 -49 50 ):
      ( 54 -57 -49 50):( -60 -53 49):
      ( -62 54 49):( -64 54 -50):
      ( -66 -53 -50)
      fill= 136 u= 137 imp:n= 1.0
c    Water Outside of Channel, L8
2491   2 -9.8114E-01 ( -47 : 51 : 58 : -56 ):
      ( 59 -53 49 ):
      ( 61 54 49 ):
      ( 63 54 -50 ):
      ( 65 -53 -50 )
      u= 137 imp:n= 1.0
c    Fuel Pellet, #1, L10
2492   72 -1.0340E+01 -40      u= 138 imp:n= 1.0
c    Fuel/Cladding Gap, #1, L10
2493   0 40 -41      u= 138 imp:n= 1.0
c    Cladding, #1, L10
2494   10 -6.5600E+00 41 -42      u= 138 imp:n= 1.0
c    Water Outside Cladding #1, L10
2495   79 -1.1790E+00 42
      u= 138 imp:n= 1.0
c    Fuel Pellet, #2, L10

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 35 of 76

2496 like 2492 but
mat= 73 rho= -9.9400E+00
u= 139 imp:n= 1.0
c Fuel/Cladding Gap, #2, L10
2497 like 2493 but
u= 139 imp:n= 1.0
c Cladding, #2, L10
2498 like 2494 but
u= 139 imp:n= 1.0
c Water Outside Cladding #2, L10
2499 like 2495 but
u= 139 imp:n= 1.0
c Fuel Pellet, #3, L10
2500 like 2492 but
mat= 74 rho= -9.4800E+00
u= 140 imp:n= 1.0
c Fuel/Cladding Gap, #3, L10
2501 like 2493 but
u= 140 imp:n= 1.0
c Cladding, #3, L10
2502 like 2494 but
u= 140 imp:n= 1.0
c Water Outside Cladding #3, L10
2503 like 2495 but
u= 140 imp:n= 1.0
c Fuel Pellet, #4, L10
2504 like 2492 but
mat= 75 rho= -1.0340E+01
u= 141 imp:n= 1.0
c Fuel/Cladding Gap, #4, L10
2505 like 2493 but
u= 141 imp:n= 1.0
c Cladding, #4, L10
2506 like 2494 but
u= 141 imp:n= 1.0
c Water Outside Cladding #4, L10
2507 like 2495 but
u= 141 imp:n= 1.0
c Fuel Pellet, #5, L10
2508 like 2492 but
mat= 76 rho= -9.9400E+00
u= 142 imp:n= 1.0
c Fuel/Cladding Gap, #5, L10
2509 like 2493 but
u= 142 imp:n= 1.0
c Cladding, #5, L10
2510 like 2494 but
u= 142 imp:n= 1.0
c Water Outside Cladding #5, L10
2511 like 2495 but
u= 142 imp:n= 1.0
c Fuel Pellet, #6, L10
2512 like 2492 but
mat= 77 rho= -9.9400E+00

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 36 of 76

```

      u= 143 imp:n= 1.0
c    Fuel/Cladding Gap, #6, L10
2513 like 2493 but
      u= 143 imp:n= 1.0
c    Cladding, #6, L10
2514 like 2494 but
      u= 143 imp:n= 1.0
c    Water Outside Cladding #6, L10
2515 like 2495 but
      u= 143 imp:n= 1.0
c    Fuel Pellet, #7, L10
2516 like 2492 but
      mat= 78 rho= -1.0290E+01
      u= 144 imp:n= 1.0
c    Fuel/Cladding Gap, #7, L10
2517 like 2493 but
      u= 144 imp:n= 1.0
c    Cladding, #7, L10
2518 like 2494 but
      u= 144 imp:n= 1.0
c    Water Outside Cladding #7, L10
2519 like 2495 but
      u= 144 imp:n= 1.0
c    Reference Fuel Rod Cell, L10
2520 79 -1.1790E+00 -43 44 45 -46 lat= 1 u = 145 imp:n= 1.0
      fill= -1:7 -1:7 0:0
      145 145 145 145 145 145 145 145 145
      145 143 143 142 142 142 142 143 145
      145 143 141 141 138 138 139 142 145
      145 142 141 138 138 138 138 139 145
      145 142 138 138 140 138 144 139 145
      145 142 138 138 138 138 138 139 145
      145 142 139 138 144 138 138 139 145
      145 143 142 139 139 139 139 142 145
      145 145 145 145 145 145 145 145 145
c    Channel, L10
2521 18 -6.5600E+00 ( 47 -48 -49 50):( -51 52 53 -54):
      ( -55 56 53 -54):( 57 -58 50 -49):
      ( -59 60 49 -53):( -61 62 49 54):
      ( -63 64 54 -50):( -65 66 -50 -53)
      u= 146 imp:n= 1.0
c    Active Fuel Area, L10
2522 79 -1.1790E+00 ( 53 -54 -52 55):( 48 -53 -49 50 ):
      ( 54 -57 -49 50):( -60 -53 49):
      ( -62 54 49):( -64 54 -50):
      ( -66 -53 -50)
      fill= 145 u= 146 imp:n= 1.0
c    Water Outside of Channel, L10
2523 2 -9.8114E-01 ( -47 : 51 : 58 : -56 ):
      ( 59 -53 49 ):
      ( 61 54 49 ):
      ( 63 54 -50 ):
      ( 65 -53 -50 )
      u= 146 imp:n= 1.0

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 37 of 76

```
c      Axial Node 1 for Fuel Assembly 1
2524      0 -67
          fill= 14
          u= 147 imp:n= 1.0
c      Axial Node 2 for Fuel Assembly 1
2525      0 67 -68
          fill= 24
          u= 147 imp:n= 1.0
c      Axial Node 3 for Fuel Assembly 1
2526      0 68 -69
          fill= 24
          u= 147 imp:n= 1.0
c      Axial Node 4 for Fuel Assembly 1
2527      0 69 -70
          fill= 102
          u= 147 imp:n= 1.0
c      Axial Node 5 for Fuel Assembly 1
2528      0 70 -71
          fill= 24
          u= 147 imp:n= 1.0
c      Axial Node 6 for Fuel Assembly 1
2529      0 71 -72
          fill= 24
          u= 147 imp:n= 1.0
c      Axial Node 7 for Fuel Assembly 1
2530      0 72 -73
          fill= 102
          u= 147 imp:n= 1.0
c      Axial Node 8 for Fuel Assembly 1
2531      0 73 -74
          fill= 24
          u= 147 imp:n= 1.0
c      Axial Node 9 for Fuel Assembly 1
2532      0 74 -75
          fill= 34
          u= 147 imp:n= 1.0
c      Axial Node 10 for Fuel Assembly 1
2533      0 75 -76
          fill= 112
          u= 147 imp:n= 1.0
c      Axial Node 11 for Fuel Assembly 1
2534      0 76 -77
          fill= 34
          u= 147 imp:n= 1.0
c      Axial Node 12 for Fuel Assembly 1
2535      0 77 -78
          fill= 34
          u= 147 imp:n= 1.0
c      Axial Node 13 for Fuel Assembly 1
2536      0 78 -79
          fill= 112
          u= 147 imp:n= 1.0
c      Axial Node 14 for Fuel Assembly 1
2537      0 79 -80
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 38 of 76

```
fill= 34
u= 147 imp:n= 1.0
c Axial Node 15 for Fuel Assembly 1
2538 0 80 -81
fill= 34
u= 147 imp:n= 1.0
c Axial Node 16 for Fuel Assembly 1
2539 0 81 -82
fill= 34
u= 147 imp:n= 1.0
c Axial Node 17 for Fuel Assembly 1
2540 0 82 -83
fill= 112
u= 147 imp:n= 1.0
c Axial Node 18 for Fuel Assembly 1
2541 0 83 -84
fill= 34
u= 147 imp:n= 1.0
c Axial Node 19 for Fuel Assembly 1
2542 0 84 -85
fill= 24
u= 147 imp:n= 1.0
c Axial Node 20 for Fuel Assembly 1
2543 0 85 -86
fill= 102
u= 147 imp:n= 1.0
c Axial Node 21 for Fuel Assembly 1
2544 0 86 -87
fill= 24
u= 147 imp:n= 1.0
c Axial Node 22 for Fuel Assembly 1
2545 0 87 -88
fill= 24
u= 147 imp:n= 1.0
c Axial Node 23 for Fuel Assembly 1
2546 0 88 -89
fill= 102
u= 147 imp:n= 1.0
c Axial Node 24 for Fuel Assembly 1
2547 0 89
fill= 14
u= 147 imp:n= 1.0
c Axial Node 1 for Fuel Assembly 2
2548 0 -67
fill= 42
u= 148 imp:n= 1.0
c Axial Node 2 for Fuel Assembly 2
2549 0 67 -68
fill= 50
u= 148 imp:n= 1.0
c Axial Node 3 for Fuel Assembly 2
2550 0 68 -69
fill= 50
u= 148 imp:n= 1.0
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 39 of 76

```
c      Axial Node 4 for Fuel Assembly 2
2551      0 69 -70
          fill= 120
          u= 148 imp:n= 1.0
c      Axial Node 5 for Fuel Assembly 2
2552      0 70 -71
          fill= 50
          u= 148 imp:n= 1.0
c      Axial Node 6 for Fuel Assembly 2
2553      0 71 -72
          fill= 50
          u= 148 imp:n= 1.0
c      Axial Node 7 for Fuel Assembly 2
2554      0 72 -73
          fill= 120
          u= 148 imp:n= 1.0
c      Axial Node 8 for Fuel Assembly 2
2555      0 73 -74
          fill= 50
          u= 148 imp:n= 1.0
c      Axial Node 9 for Fuel Assembly 2
2556      0 74 -75
          fill= 58
          u= 148 imp:n= 1.0
c      Axial Node 10 for Fuel Assembly 2
2557      0 75 -76
          fill= 128
          u= 148 imp:n= 1.0
c      Axial Node 11 for Fuel Assembly 2
2558      0 76 -77
          fill= 58
          u= 148 imp:n= 1.0
c      Axial Node 12 for Fuel Assembly 2
2559      0 77 -78
          fill= 58
          u= 148 imp:n= 1.0
c      Axial Node 13 for Fuel Assembly 2
2560      0 78 -79
          fill= 128
          u= 148 imp:n= 1.0
c      Axial Node 14 for Fuel Assembly 2
2561      0 79 -80
          fill= 58
          u= 148 imp:n= 1.0
c      Axial Node 15 for Fuel Assembly 2
2562      0 80 -81
          fill= 58
          u= 148 imp:n= 1.0
c      Axial Node 16 for Fuel Assembly 2
2563      0 81 -82
          fill= 58
          u= 148 imp:n= 1.0
c      Axial Node 17 for Fuel Assembly 2
2564      0 82 -83
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 40 of 76

```
fill= 128
u= 148 imp:n= 1.0
c Axial Node 18 for Fuel Assembly 2
2565 0 83 -84
fill= 58
u= 148 imp:n= 1.0
c Axial Node 19 for Fuel Assembly 2
2566 0 84 -85
fill= 50
u= 148 imp:n= 1.0
c Axial Node 20 for Fuel Assembly 2
2567 0 85 -86
fill= 120
u= 148 imp:n= 1.0
c Axial Node 21 for Fuel Assembly 2
2568 0 86 -87
fill= 50
u= 148 imp:n= 1.0
c Axial Node 22 for Fuel Assembly 2
2569 0 87 -88
fill= 50
u= 148 imp:n= 1.0
c Axial Node 23 for Fuel Assembly 2
2570 0 88 -89
fill= 120
u= 148 imp:n= 1.0
c Axial Node 24 for Fuel Assembly 2
2571 0 89
fill= 42
u= 148 imp:n= 1.0
c Axial Node 1 for Fuel Assembly 3
2572 0 -67
fill= 67
u= 149 imp:n= 1.0
c Axial Node 2 for Fuel Assembly 3
2573 0 67 -68
fill= 76
u= 149 imp:n= 1.0
c Axial Node 3 for Fuel Assembly 3
2574 0 68 -69
fill= 76
u= 149 imp:n= 1.0
c Axial Node 4 for Fuel Assembly 3
2575 0 69 -70
fill= 137
u= 149 imp:n= 1.0
c Axial Node 5 for Fuel Assembly 3
2576 0 70 -71
fill= 76
u= 149 imp:n= 1.0
c Axial Node 6 for Fuel Assembly 3
2577 0 71 -72
fill= 76
u= 149 imp:n= 1.0
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 41 of 76

c Axial Node 7 for Fuel Assembly 3
2578 0 72 -73
fill= 137
u= 149 imp:n= 1.0

c Axial Node 8 for Fuel Assembly 3
2579 0 73 -74
fill= 76
u= 149 imp:n= 1.0

c Axial Node 9 for Fuel Assembly 3
2580 0 74 -75
fill= 76
u= 149 imp:n= 1.0

c Axial Node 10 for Fuel Assembly 3
2581 0 75 -76
fill= 137
u= 149 imp:n= 1.0

c Axial Node 11 for Fuel Assembly 3
2582 0 76 -77
fill= 76
u= 149 imp:n= 1.0

c Axial Node 12 for Fuel Assembly 3
2583 0 77 -78
fill= 76
u= 149 imp:n= 1.0

c Axial Node 13 for Fuel Assembly 3
2584 0 78 -79
fill= 137
u= 149 imp:n= 1.0

c Axial Node 14 for Fuel Assembly 3
2585 0 79 -80
fill= 76
u= 149 imp:n= 1.0

c Axial Node 15 for Fuel Assembly 3
2586 0 80 -81
fill= 76
u= 149 imp:n= 1.0

c Axial Node 16 for Fuel Assembly 3
2587 0 81 -82
fill= 76
u= 149 imp:n= 1.0

c Axial Node 17 for Fuel Assembly 3
2588 0 82 -83
fill= 137
u= 149 imp:n= 1.0

c Axial Node 18 for Fuel Assembly 3
2589 0 83 -84
fill= 76
u= 149 imp:n= 1.0

c Axial Node 19 for Fuel Assembly 3
2590 0 84 -85
fill= 76
u= 149 imp:n= 1.0

c Axial Node 20 for Fuel Assembly 3
2591 0 85 -86

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 42 of 76

```
fill= 137
u= 149 imp:n= 1.0
c Axial Node 21 for Fuel Assembly 3
2592 0 86 -87
fill= 76
u= 149 imp:n= 1.0
c Axial Node 22 for Fuel Assembly 3
2593 0 87 -88
fill= 76
u= 149 imp:n= 1.0
c Axial Node 23 for Fuel Assembly 3
2594 0 88 -89
fill= 137
u= 149 imp:n= 1.0
c Axial Node 24 for Fuel Assembly 3
2595 0 89
fill= 67
u= 149 imp:n= 1.0
c Axial Node 1 for Fuel Assembly 4
2596 0 -67
fill= 83
u= 150 imp:n= 1.0
c Axial Node 2 for Fuel Assembly 4
2597 0 67 -68
fill= 92
u= 150 imp:n= 1.0
c Axial Node 3 for Fuel Assembly 4
2598 0 68 -69
fill= 92
u= 150 imp:n= 1.0
c Axial Node 4 for Fuel Assembly 4
2599 0 69 -70
fill= 146
u= 150 imp:n= 1.0
c Axial Node 5 for Fuel Assembly 4
2600 0 70 -71
fill= 92
u= 150 imp:n= 1.0
c Axial Node 6 for Fuel Assembly 4
2601 0 71 -72
fill= 92
u= 150 imp:n= 1.0
c Axial Node 7 for Fuel Assembly 4
2602 0 72 -73
fill= 146
u= 150 imp:n= 1.0
c Axial Node 8 for Fuel Assembly 4
2603 0 73 -74
fill= 92
u= 150 imp:n= 1.0
c Axial Node 9 for Fuel Assembly 4
2604 0 74 -75
fill= 92
u= 150 imp:n= 1.0
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 43 of 76

```
c      Axial Node 10 for Fuel Assembly 4
2605      0 75 -76
          fill= 146
          u= 150 imp:n= 1.0
c      Axial Node 11 for Fuel Assembly 4
2606      0 76 -77
          fill= 92
          u= 150 imp:n= 1.0
c      Axial Node 12 for Fuel Assembly 4
2607      0 77 -78
          fill= 92
          u= 150 imp:n= 1.0
c      Axial Node 13 for Fuel Assembly 4
2608      0 78 -79
          fill= 146
          u= 150 imp:n= 1.0
c      Axial Node 14 for Fuel Assembly 4
2609      0 79 -80
          fill= 92
          u= 150 imp:n= 1.0
c      Axial Node 15 for Fuel Assembly 4
2610      0 80 -81
          fill= 92
          u= 150 imp:n= 1.0
c      Axial Node 16 for Fuel Assembly 4
2611      0 81 -82
          fill= 92
          u= 150 imp:n= 1.0
c      Axial Node 17 for Fuel Assembly 4
2612      0 82 -83
          fill= 146
          u= 150 imp:n= 1.0
c      Axial Node 18 for Fuel Assembly 4
2613      0 83 -84
          fill= 92
          u= 150 imp:n= 1.0
c      Axial Node 19 for Fuel Assembly 4
2614      0 84 -85
          fill= 92
          u= 150 imp:n= 1.0
c      Axial Node 20 for Fuel Assembly 4
2615      0 85 -86
          fill= 146
          u= 150 imp:n= 1.0
c      Axial Node 21 for Fuel Assembly 4
2616      0 86 -87
          fill= 92
          u= 150 imp:n= 1.0
c      Axial Node 22 for Fuel Assembly 4
2617      0 87 -88
          fill= 92
          u= 150 imp:n= 1.0
c      Axial Node 23 for Fuel Assembly 4
2618      0 88 -89
```

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 44 of 76

```

    fill= 146
    u= 150 imp:n= 1.0
c   Axial Node 24 for Fuel Assembly 4
2619   0 89
    fill= 83
    u= 150 imp:n= 1.0
c   Fuel Assembly in Northwest Quadrant, CC151
  5     2 -9.8114E-01 (( -90 91 -92 93 )
    (#( -90 93 94 -95 96 )))
    u= 151 imp:n= 1.0
    *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
    fill= 0
c   Fuel Assembly in Northeast Quadrant, CC151
  6 like 5 but fill= 0
    *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
    u= 151 imp:n= 1.0
c   Fuel Assembly in Southeast Quadrant, CC151
  7 like 6 but fill= 149
    trcl=( 0 0 0 ) u= 151 imp:n= 1.0
c   Fuel Assembly in Southwest Quadrant, CC151
  8 like 5 but fill= 0
    *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 151 imp:n= 1.0
c   Blade Window, CC151
  9     2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
    u= 151 imp:n= 1.0
c   Balance of Control Cell, CC151
10 2 -9.8114E-01 #5 #6 #7 #8 #9
    u= 151 imp:n= 1.0
c   Fuel Assembly in Northwest Quadrant, CC152
11     2 -9.8114E-01 (( -90 91 -92 93 )
    (#( -90 93 94 -95 96 )))
    u= 152 imp:n= 1.0
    *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
    fill= 149
c   Fuel Assembly in Northeast Quadrant, CC152
12 like 11 but fill= 149
    *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
    u= 152 imp:n= 1.0
c   Fuel Assembly in Southeast Quadrant, CC152
13 like 12 but fill= 147
    trcl=( 0 0 0 ) u= 152 imp:n= 1.0
c   Fuel Assembly in Southwest Quadrant, CC152
14 like 11 but fill= 149
    *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 152 imp:n= 1.0
c   Blade Window, CC152
15     2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
    u= 152 imp:n= 1.0
    fill= 3 trcl=( 0.0 0.0 -3.6576E+02 )
c   Guide Tube Segment, CC152
16     1 -7.9000E+00 -105 106 u= 152 imp:n= 1.0
c   Inside Guide Tube Segment, CC152
17     0 -106 u= 152 imp:n= 1.0
c   Balance of Control Cell, CC152
18 2 -9.8114E-01 #11 #12 #13 #14 #15 #16 #17

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 45 of 76

```
u= 152 imp:n= 1.0
c Fuel Assembly in Northwest Quadrant, CC153
19 2 -9.8114E-01 (( -90 91 -92 93 )
  (#( -90 93 94 -95 96 )))
u= 153 imp:n= 1.0
*trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 0 )
fill= 149
c Fuel Assembly in Northeast Quadrant, CC153
20 like 19 but fill= 149
*trcl=( 0 0 0 90 0 90 180 90 90 90 0 )
u= 153 imp:n= 1.0
c Fuel Assembly in Southeast Quadrant, CC153
21 like 20 but fill= 147
trcl=( 0 0 0 ) u= 153 imp:n= 1.0
c Fuel Assembly in Southwest Quadrant, CC153
22 like 19 but fill= 149
*trcl=( 0 0 0 -90 180 90 0 -90 90 90 0 ) u= 153 imp:n= 1.0
c Blade Window, CC153
23 2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104 )
u= 153 imp:n= 1.0
fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c Guide Tube Segment, CC153
24 1 -7.9000E+00 -105 106
trcl=( -3.0480E+01 0.0 0.0 ) u= 153 imp:n= 1.0
c Inside Guide Tube Segment, CC153
25 0 -106 trcl=( -3.0480E+01 0.0 0.0 ) u= 153 imp:n= 1.0
c Balance of Control Cell, CC153
26 2 -9.8114E-01 #19 #20 #21 #22 #23 #24 #25
u= 153 imp:n= 1.0
c Fuel Assembly in Northwest Quadrant, CC154
27 2 -9.8114E-01 (( -90 91 -92 93 )
  (#( -90 93 94 -95 96 )))
u= 154 imp:n= 1.0
*trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 0 )
fill= 149
c Fuel Assembly in Northeast Quadrant, CC154
28 like 27 but fill= 0
*trcl=( 0 0 0 90 0 90 180 90 90 90 0 )
u= 154 imp:n= 1.0
c Fuel Assembly in Southeast Quadrant, CC154
29 like 28 but fill= 0
trcl=( 0 0 0 ) u= 154 imp:n= 1.0
c Fuel Assembly in Southwest Quadrant, CC154
30 like 27 but fill= 149
*trcl=( 0 0 0 -90 180 90 0 -90 90 90 0 ) u= 154 imp:n= 1.0
c Blade Window, CC154
31 2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104 )
u= 154 imp:n= 1.0
fill= 3 trcl=( 0.0 0.0 -3.6576E+02 )
c Balance of Control Cell, CC154
32 2 -9.8114E-01 #27 #28 #29 #30 #31
u= 154 imp:n= 1.0
c Fuel Assembly in Northwest Quadrant, CC155
33 2 -9.8114E-01 (( -90 91 -92 93 )
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 46 of 76

```
(#( -90 93 94 -95 96 )))
u= 155 imp:n= 1.0
*trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 0 )
fill= 149
c Fuel Assembly in Northeast Quadrant, CC155
34 like 33 but fill= 149
*trcl=( 0 0 0 90 0 90 180 90 90 90 0 )
u= 155 imp:n= 1.0
c Fuel Assembly in Southeast Quadrant, CC155
35 like 34 but fill= 147
trcl=( 0 0 0 ) u= 155 imp:n= 1.0
c Fuel Assembly in Southwest Quadrant, CC155
36 like 33 but fill= 149
*trcl=( 0 0 0 -90 180 90 0 -90 90 90 0 ) u= 155 imp:n= 1.0
c Blade Window, CC155
37 2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
u= 155 imp:n= 1.0
fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c Guide Tube Segment, CC155
38 1 -7.9000E+00 -105 106 u= 155 imp:n= 1.0
c Inside Guide Tube Segment, CC155
39 0 -106 u= 155 imp:n= 1.0
c Balance of Control Cell, CC155
40 2 -9.8114E-01 #33 #34 #35 #36 #37 #38 #39
u= 155 imp:n= 1.0
c Fuel Assembly in Northwest Quadrant, CC156
41 2 -9.8114E-01 (( -90 91 -92 93 )
(#( -90 93 94 -95 96 )))
u= 156 imp:n= 1.0
*trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 0 )
fill= 147
c Fuel Assembly in Northeast Quadrant, CC156
42 like 41 but fill= 150
*trcl=( 0 0 0 90 0 90 180 90 90 90 0 )
u= 156 imp:n= 1.0
c Fuel Assembly in Southeast Quadrant, CC156
43 like 42 but fill= 147
trcl=( 0 0 0 ) u= 156 imp:n= 1.0
c Fuel Assembly in Southwest Quadrant, CC156
44 like 41 but fill= 150
*trcl=( 0 0 0 -90 180 90 0 -90 90 90 0 ) u= 156 imp:n= 1.0
c Blade Window, CC156
45 2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
u= 156 imp:n= 1.0
fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c Guide Tube Segment, CC156
46 1 -7.9000E+00 -105 106
trcl=( 0.0 3.0480E+01 0.0 ) u= 156 imp:n= 1.0
c Inside Guide Tube Segment, CC156
47 0 -106 trcl=( 0.0 3.0480E+01 0.0 ) u= 156 imp:n= 1.0
c Balance of Control Cell, CC156
48 2 -9.8114E-01 #41 #42 #43 #44 #45 #46 #47
u= 156 imp:n= 1.0
c Fuel Assembly in Northwest Quadrant, CC157
```

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 47 of 76

```

49      2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 157 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
      fill= 147
c      Fuel Assembly in Northeast Quadrant, CC157
50 like 49 but fill= 149
      *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
      u= 157 imp:n= 1.0
c      Fuel Assembly in Southeast Quadrant, CC157
51 like 50 but fill= 148
      trcl=( 0 0 0 ) u= 157 imp:n= 1.0
c      Fuel Assembly in Southwest Quadrant, CC157
52 like 49 but fill= 150
      *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 157 imp:n= 1.0
c      Blade Window, CC157
53      2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
      u= 157 imp:n= 1.0
      fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c      Guide Tube Segment, CC157
54      1 -7.9000E+00 -105 106
      trcl=( -3.0480E+01 3.0480E+01 0.0) u= 157 imp:n= 1.0
c      Inside Guide Tube Segment, CC157
55      0 -106      trcl=( -3.0480E+01 3.0480E+01 0.0) u= 157 imp:n= 1.0
c      Balance of Control Cell, CC157
56 2 -9.8114E-01 #49 #50 #51 #52 #53 #54 #55
      u= 157 imp:n= 1.0
c      Fuel Assembly in Northwest Quadrant, CC158
57      2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 158 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
      fill= 147
c      Fuel Assembly in Northeast Quadrant, CC158
58 like 57 but fill= 0
      *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
      u= 158 imp:n= 1.0
c      Fuel Assembly in Southeast Quadrant, CC158
59 like 58 but fill= 0
      trcl=( 0 0 0 ) u= 158 imp:n= 1.0
c      Fuel Assembly in Southwest Quadrant, CC158
60 like 57 but fill= 150
      *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 158 imp:n= 1.0
c      Blade Window, CC158
61      2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
      u= 158 imp:n= 1.0
      fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c      Balance of Control Cell, CC158
62 2 -9.8114E-01 #57 #58 #59 #60 #61
      u= 158 imp:n= 1.0
c      Fuel Assembly in Northwest Quadrant, CC159
63      2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 159 imp:n= 1.0

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 48 of 76

```
*trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
fill= 0
c Fuel Assembly in Northeast Quadrant, CC159
64 like 63 but fill= 0
*trcl=( 0 0 0 90 0 90 180 90 90 90 0 )
u= 159 imp:n= 1.0
c Fuel Assembly in Southeast Quadrant, CC159
65 like 64 but fill= 149
trcl=( 0 0 0 ) u= 159 imp:n= 1.0
c Fuel Assembly in Southwest Quadrant, CC159
66 like 63 but fill= 0
*trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0 ) u= 159 imp:n= 1.0
c Blade Window, CC159
67 2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
u= 159 imp:n= 1.0
c Guide Tube Segment, CC159
68 1 -7.9000E+00 -105 106 u= 159 imp:n= 1.0
c Inside Guide Tube Segment, CC159
69 0 -106 u= 159 imp:n= 1.0
c Balance of Control Cell, CC159
70 2 -9.8114E-01 #63 #64 #65 #66 #67 #68 #69
u= 159 imp:n= 1.0
c Fuel Assembly in Northwest Quadrant, CC160
71 2 -9.8114E-01 (( -90 91 -92 93 )
(#( -90 93 94 -95 96 )))
u= 160 imp:n= 1.0
*trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
fill= 147
c Fuel Assembly in Northeast Quadrant, CC160
72 like 71 but fill= 149
*trcl=( 0 0 0 90 0 90 180 90 90 90 0 )
u= 160 imp:n= 1.0
c Fuel Assembly in Southeast Quadrant, CC160
73 like 72 but fill= 148
trcl=( 0 0 0 ) u= 160 imp:n= 1.0
c Fuel Assembly in Southwest Quadrant, CC160
74 like 71 but fill= 149
*trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0 ) u= 160 imp:n= 1.0
c Blade Window, CC160
75 2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
u= 160 imp:n= 1.0
fill= 3 trcl=( 0.0 0.0 -3.6576E+02 )
c Guide Tube Segment, CC160
76 1 -7.9000E+00 -105 106 u= 160 imp:n= 1.0
c Inside Guide Tube Segment, CC160
77 0 -106 u= 160 imp:n= 1.0
c Guide Tube Segment (2), CC160
78 1 -7.9000E+00 -105 106
trcl=( 0.0 3.0480E+01 0.0 ) u= 160 imp:n= 1.0
c Inside Guide Tube Segment (2), CC160
79 0 -106 trcl=( 0.0 3.0480E+01 0.0 ) u= 160 imp:n= 1.0
c Balance of Control Cell, CC160
80 2 -9.8114E-01 #71 #72 #73 #74 #75 #76 #77 #78 #79
u= 160 imp:n= 1.0
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 49 of 76

```
c      Fuel Assembly in Northwest Quadrant, CC161
81      2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 161 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
      fill= 148
c      Fuel Assembly in Northeast Quadrant, CC161
82 like 81 but fill= 150
      *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
      u= 161 imp:n= 1.0
c      Fuel Assembly in Southeast Quadrant, CC161
83 like 82 but fill= 148
      trcl=( 0 0 0 ) u= 161 imp:n= 1.0
c      Fuel Assembly in Southwest Quadrant, CC161
84 like 81 but fill= 150
      *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 161 imp:n= 1.0
c      Blade Window, CC161
85      2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
      u= 161 imp:n= 1.0
      fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c      Guide Tube Segment, CC161
86      1 -7.9000E+00 -105 106
      trcl=( -3.0480E+01 0.0 0.0) u= 161 imp:n= 1.0
c      Inside Guide Tube Segment, CC161
87      0 -106      trcl=( -3.0480E+01 0.0 0.0) u= 161 imp:n= 1.0
c      Guide Tube Segment (2), CC161
88      1 -7.9000E+00 -105 106
      trcl=( -3.0480E+01 3.0480E+01 0.0) u= 161 imp:n= 1.0
c      Inside Guide Tube Segment (2), CC161
89      0 -106      trcl=( -3.0480E+01 3.0480E+01 0.0) u= 161 imp:n= 1.0
c      Balance of Control Cell, CC161
90      2 -9.8114E-01 #81 #82 #83 #84 #85 #86 #87 #88 #89
      u= 161 imp:n= 1.0
c      Fuel Assembly in Northwest Quadrant, CC162
91      2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 162 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
      fill= 148
c      Fuel Assembly in Northeast Quadrant, CC162
92 like 91 but fill= 150
      *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
      u= 162 imp:n= 1.0
c      Fuel Assembly in Southeast Quadrant, CC162
93 like 92 but fill= 147
      trcl=( 0 0 0 ) u= 162 imp:n= 1.0
c      Fuel Assembly in Southwest Quadrant, CC162
94 like 91 but fill= 150
      *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 162 imp:n= 1.0
c      Blade Window, CC162
95      2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
      u= 162 imp:n= 1.0
      fill= 3 trcl=( 0.0 0.0 -3.6576E+02 )
c      Guide Tube Segment, CC162
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 50 of 76

```
96      1 -7.9000E+00 -105 106 u= 162 imp:n= 1.0
c      Inside Guide Tube Segment, CC162
97      0 -106 u= 162 imp:n= 1.0
c      Balance of Control Cell, CC162
98      2 -9.8114E-01 #91 #92 #93 #94 #95 #96 #97
      u= 162 imp:n= 1.0
c      Fuel Assembly in Northwest Quadrant, CC163
99      2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 163 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
      fill= 148
c      Fuel Assembly in Northeast Quadrant, CC163
100     like 99 but fill= 150
      *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0 )
      u= 163 imp:n= 1.0
c      Fuel Assembly in Southeast Quadrant, CC163
101     like 100 but fill= 148
      trcl=( 0 0 0 ) u= 163 imp:n= 1.0
c      Fuel Assembly in Southwest Quadrant, CC163
102     like 99 but fill= 150
      *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0 ) u= 163 imp:n= 1.0
c      Blade Window, CC163
103      2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
      u= 163 imp:n= 1.0
      fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c      Guide Tube Segment, CC163
104      1 -7.9000E+00 -105 106
      trcl=( -3.0480E+01 0.0 0.0 ) u= 163 imp:n= 1.0
c      Inside Guide Tube Segment, CC163
105      0 -106      trcl=( -3.0480E+01 0.0 0.0 ) u= 163 imp:n= 1.0
c      Balance of Control Cell, CC163
106      2 -9.8114E-01 #99 #100 #101 #102 #103 #104 #105
      u= 163 imp:n= 1.0
c      Fuel Assembly in Northwest Quadrant, CC164
107      2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 164 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
      fill= 148
c      Fuel Assembly in Northeast Quadrant, CC164
108     like 107 but fill= 0
      *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0 )
      u= 164 imp:n= 1.0
c      Fuel Assembly in Southeast Quadrant, CC164
109     like 108 but fill= 0
      trcl=( 0 0 0 ) u= 164 imp:n= 1.0
c      Fuel Assembly in Southwest Quadrant, CC164
110     like 107 but fill= 150
      *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0 ) u= 164 imp:n= 1.0
c      Blade Window, CC164
111      2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
      u= 164 imp:n= 1.0
      fill= 3 trcl=( 0.0 0.0 -3.6576E+02 )
```

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 51 of 76

```

c   Balance of Control Cell, CC164
112 2 -9.8114E-01 #107 #108 #109 #110 #111
    u= 164 imp:n= 1.0
c   Fuel Assembly in Northwest Quadrant, CC165
113  2 -9.8114E-01 (( -90 91 -92 93 )
    (#( -90 93 94 -95 96 )))
    u= 165 imp:n= 1.0
    *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
    fill= 149
c   Fuel Assembly in Northeast Quadrant, CC165
114 like 113 but fill= 149
    *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
    u= 165 imp:n= 1.0
c   Fuel Assembly in Southeast Quadrant, CC165
115 like 114 but fill= 147
    trcl=( 0 0 0 ) u= 165 imp:n= 1.0
c   Fuel Assembly in Southwest Quadrant, CC165
116 like 113 but fill= 149
    *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 165 imp:n= 1.0
c   Blade Window, CC165
117  2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
    u= 165 imp:n= 1.0
    fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c   Guide Tube Segment, CC165
118  1 -7.9000E+00 -105 106
    trcl=( 0.0 3.0480E+01 0.0) u= 165 imp:n= 1.0
c   Inside Guide Tube Segment, CC165
119  0 -106          trcl=( 0.0 3.0480E+01 0.0) u= 165 imp:n= 1.0
c   Balance of Control Cell, CC165
120 2 -9.8114E-01 #113 #114 #115 #116 #117 #118 #119
    u= 165 imp:n= 1.0
c   Fuel Assembly in Northwest Quadrant, CC166
121  2 -9.8114E-01 (( -90 91 -92 93 )
    (#( -90 93 94 -95 96 )))
    u= 166 imp:n= 1.0
    *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
    fill= 147
c   Fuel Assembly in Northeast Quadrant, CC166
122 like 121 but fill= 149
    *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
    u= 166 imp:n= 1.0
c   Fuel Assembly in Southeast Quadrant, CC166
123 like 122 but fill= 148
    trcl=( 0 0 0 ) u= 166 imp:n= 1.0
c   Fuel Assembly in Southwest Quadrant, CC166
124 like 121 but fill= 149
    *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 166 imp:n= 1.0
c   Blade Window, CC166
125  2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
    u= 166 imp:n= 1.0
    fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c   Guide Tube Segment, CC166
126  1 -7.9000E+00 -105 106
    trcl=( -3.0480E+01 3.0480E+01 0.0) u= 166 imp:n= 1.0

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 52 of 76

```
c      Inside Guide Tube Segment, CC166
127      0 -106      trcl=( -3.0480E+01 3.0480E+01 0.0) u= 166 imp:n= 1.0
c      Balance of Control Cell, CC166
128      2 -9.8114E-01 #121 #122 #123 #124 #125 #126 #127
      u= 166 imp:n= 1.0
c      Fuel Assembly in Northwest Quadrant, CC167
129      2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 167 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
      fill= 147
c      Fuel Assembly in Northeast Quadrant, CC167
130      like 129 but fill= 150
      *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
      u= 167 imp:n= 1.0
c      Fuel Assembly in Southeast Quadrant, CC167
131      like 130 but fill= 148
      trcl=( 0 0 0 ) u= 167 imp:n= 1.0
c      Fuel Assembly in Southwest Quadrant, CC167
132      like 129 but fill= 150
      *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 167 imp:n= 1.0
c      Blade Window, CC167
133      2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
      u= 167 imp:n= 1.0
      fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c      Guide Tube Segment, CC167
134      1 -7.9000E+00 -105 106 u= 167 imp:n= 1.0
c      Inside Guide Tube Segment, CC167
135      0 -106 u= 167 imp:n= 1.0
c      Guide Tube Segment (2), CC167
136      1 -7.9000E+00 -105 106
      trcl=( 0.0 3.0480E+01 0.0) u= 167 imp:n= 1.0
c      Inside Guide Tube Segment (2), CC167
137      0 -106      trcl=( 0.0 3.0480E+01 0.0) u= 167 imp:n= 1.0
c      Balance of Control Cell, CC167
138      2 -9.8114E-01 #129 #130 #131 #132 #133 #134 #135 #136 #137
      u= 167 imp:n= 1.0
c      Fuel Assembly in Northwest Quadrant, CC168
139      2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 168 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
      fill= 148
c      Fuel Assembly in Northeast Quadrant, CC168
140      like 139 but fill= 150
      *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
      u= 168 imp:n= 1.0
c      Fuel Assembly in Southeast Quadrant, CC168
141      like 140 but fill= 147
      trcl=( 0 0 0 ) u= 168 imp:n= 1.0
c      Fuel Assembly in Southwest Quadrant, CC168
142      like 139 but fill= 150
      *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 168 imp:n= 1.0
c      Blade Window, CC168
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 53 of 76

```
143      2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
      u= 168 imp:n= 1.0
      fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c      Guide Tube Segment, CC168
144      1 -7.9000E+00 -105 106
      trcl=( 0.0 3.0480E+01 0.0) u= 168 imp:n= 1.0
c      Inside Guide Tube Segment, CC168
145      0 -106      trcl=( 0.0 3.0480E+01 0.0) u= 168 imp:n= 1.0
c      Balance of Control Cell, CC168
146      2 -9.8114E-01 #139 #140 #141 #142 #143 #144 #145
      u= 168 imp:n= 1.0
c      Fuel Assembly in Northwest Quadrant, CC169
147      2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 169 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
      fill= 148
c      Fuel Assembly in Northeast Quadrant, CC169
148      like 147 but fill= 150
      *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
      u= 169 imp:n= 1.0
c      Fuel Assembly in Southeast Quadrant, CC169
149      like 148 but fill= 147
      trcl=( 0 0 0 ) u= 169 imp:n= 1.0
c      Fuel Assembly in Southwest Quadrant, CC169
150      like 147 but fill= 149
      *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 169 imp:n= 1.0
c      Blade Window, CC169
151      2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
      u= 169 imp:n= 1.0
      fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c      Guide Tube Segment, CC169
152      1 -7.9000E+00 -105 106
      trcl=( -3.0480E+01 3.0480E+01 0.0) u= 169 imp:n= 1.0
c      Inside Guide Tube Segment, CC169
153      0 -106      trcl=( -3.0480E+01 3.0480E+01 0.0) u= 169 imp:n= 1.0
c      Balance of Control Cell, CC169
154      2 -9.8114E-01 #147 #148 #149 #150 #151 #152 #153
      u= 169 imp:n= 1.0
c      Fuel Assembly in Northwest Quadrant, CC170
155      2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 170 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
      fill= 148
c      Fuel Assembly in Northeast Quadrant, CC170
156      like 155 but fill= 0
      *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
      u= 170 imp:n= 1.0
c      Fuel Assembly in Southeast Quadrant, CC170
157      like 156 but fill= 0
      trcl=( 0 0 0 ) u= 170 imp:n= 1.0
c      Fuel Assembly in Southwest Quadrant, CC170
158      like 155 but fill= 149
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 54 of 76

```
*trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0 ) u= 170 imp:n= 1.0
c Blade Window, CC170
159 2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
u= 170 imp:n= 1.0
fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c Balance of Control Cell, CC170
160 2 -9.8114E-01 #155 #156 #157 #158 #159
u= 170 imp:n= 1.0
c Fuel Assembly in Northwest Quadrant, CC171
161 2 -9.8114E-01 (( -90 91 -92 93 )
(#( -90 93 94 -95 96 )))
u= 171 imp:n= 1.0
*trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
fill= 148
c Fuel Assembly in Northeast Quadrant, CC171
162 like 161 but fill= 150
*trcl=( 0 0 0 90 0 90 180 90 90 90 90 0 )
u= 171 imp:n= 1.0
c Fuel Assembly in Southeast Quadrant, CC171
163 like 162 but fill= 148
trcl=( 0 0 0 ) u= 171 imp:n= 1.0
c Fuel Assembly in Southwest Quadrant, CC171
164 like 161 but fill= 150
*trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0 ) u= 171 imp:n= 1.0
c Blade Window, CC171
165 2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
u= 171 imp:n= 1.0
fill= 3 trcl=( 0.0 0.0 -3.6576E+02 )
c Guide Tube Segment, CC171
166 1 -7.9000E+00 -105 106 u= 171 imp:n= 1.0
c Inside Guide Tube Segment, CC171
167 0 -106 u= 171 imp:n= 1.0
c Guide Tube Segment (2), CC171
168 1 -7.9000E+00 -105 106
trcl=( 0.0 3.0480E+01 0.0 ) u= 171 imp:n= 1.0
c Inside Guide Tube Segment (2), CC171
169 0 -106 trcl=( 0.0 3.0480E+01 0.0 ) u= 171 imp:n= 1.0
c Balance of Control Cell, CC171
170 2 -9.8114E-01 #161 #162 #163 #164 #165 #166 #167 #168 #169
u= 171 imp:n= 1.0
c Fuel Assembly in Northwest Quadrant, CC172
171 2 -9.8114E-01 (( -90 91 -92 93 )
(#( -90 93 94 -95 96 )))
u= 172 imp:n= 1.0
*trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
fill= 148
c Fuel Assembly in Northeast Quadrant, CC172
172 like 171 but fill= 150
*trcl=( 0 0 0 90 0 90 180 90 90 90 90 0 )
u= 172 imp:n= 1.0
c Fuel Assembly in Southeast Quadrant, CC172
173 like 172 but fill= 147
trcl=( 0 0 0 ) u= 172 imp:n= 1.0
c Fuel Assembly in Southwest Quadrant, CC172
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 55 of 76

```
174 like 171 but fill= 150
    *trcl=( 0 0 0 -90 180 90 0 -90 90 90 0) u= 172 imp:n= 1.0
c   Blade Window, CC172
175     2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
    u= 172 imp:n= 1.0
    fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c   Guide Tube Segment, CC172
176     1 -7.9000E+00 -105 106
    trcl=( -3.0480E+01 0.0 0.0) u= 172 imp:n= 1.0
c   Inside Guide Tube Segment, CC172
177     0 -106          trcl=( -3.0480E+01 0.0 0.0) u= 172 imp:n= 1.0
c   Guide Tube Segment (2), CC172
178     1 -7.9000E+00 -105 106
    trcl=( -3.0480E+01 3.0480E+01 0.0) u= 172 imp:n= 1.0
c   Inside Guide Tube Segment (2), CC172
179     0 -106          trcl=( -3.0480E+01 3.0480E+01 0.0) u= 172 imp:n= 1.0
c   Balance of Control Cell, CC172
180 2 -9.8114E-01 #171 #172 #173 #174 #175 #176 #177 #178 #179
    u= 172 imp:n= 1.0
c   Fuel Assembly in Northwest Quadrant, CC173
181     2 -9.8114E-01 (( -90 91 -92 93 )
    (#( -90 93 94 -95 96 )))
    u= 173 imp:n= 1.0
    *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 0 )
    fill= 147
c   Fuel Assembly in Northeast Quadrant, CC173
182 like 181 but fill= 149
    *trcl=( 0 0 0 90 0 90 180 90 90 90 0)
    u= 173 imp:n= 1.0
c   Fuel Assembly in Southeast Quadrant, CC173
183 like 182 but fill= 147
    trcl=( 0 0 0 ) u= 173 imp:n= 1.0
c   Fuel Assembly in Southwest Quadrant, CC173
184 like 181 but fill= 149
    *trcl=( 0 0 0 -90 180 90 0 -90 90 90 0) u= 173 imp:n= 1.0
c   Blade Window, CC173
185     2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
    u= 173 imp:n= 1.0
    fill= 3 trcl=( 0.0 0.0 -3.6576E+02 )
c   Guide Tube Segment, CC173
186     1 -7.9000E+00 -105 106 u= 173 imp:n= 1.0
c   Inside Guide Tube Segment, CC173
187     0 -106 u= 173 imp:n= 1.0
c   Balance of Control Cell, CC173
188 2 -9.8114E-01 #181 #182 #183 #184 #185 #186 #187
    u= 173 imp:n= 1.0
c   Fuel Assembly in Northwest Quadrant, CC174
189     2 -9.8114E-01 (( -90 91 -92 93 )
    (#( -90 93 94 -95 96 )))
    u= 174 imp:n= 1.0
    *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 0 )
    fill= 147
c   Fuel Assembly in Northeast Quadrant, CC174
190 like 189 but fill= 149
```

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 56 of 76

```

      *trcl=( 0 0 0 90 0 90 180 90 90 90 0 )
      u= 174 imp:n= 1.0
c      Fuel Assembly in Southeast Quadrant, CC174
191 like 190 but fill= 147
      trcl=( 0 0 0 ) u= 174 imp:n= 1.0
c      Fuel Assembly in Southwest Quadrant, CC174
192 like 189 but fill= 149
      *trcl=( 0 0 0 -90 180 90 0 -90 90 90 0 ) u= 174 imp:n= 1.0
c      Blade Window, CC174
193      2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
      u= 174 imp:n= 1.0
      fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c      Guide Tube Segment, CC174
194      1 -7.9000E+00 -105 106
      trcl=( -3.0480E+01 0.0 0.0 ) u= 174 imp:n= 1.0
c      Inside Guide Tube Segment, CC174
195      0 -106      trcl=( -3.0480E+01 0.0 0.0 ) u= 174 imp:n= 1.0
c      Balance of Control Cell, CC174
196 2 -9.8114E-01 #189 #190 #191 #192 #193 #194 #195
      u= 174 imp:n= 1.0
c      Fuel Assembly in Northwest Quadrant, CC175
197      2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 175 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 0 )
      fill= 147
c      Fuel Assembly in Northeast Quadrant, CC175
198 like 197 but fill= 0
      *trcl=( 0 0 0 90 0 90 180 90 90 90 0 )
      u= 175 imp:n= 1.0
c      Fuel Assembly in Southeast Quadrant, CC175
199 like 198 but fill= 0
      trcl=( 0 0 0 ) u= 175 imp:n= 1.0
c      Fuel Assembly in Southwest Quadrant, CC175
200 like 197 but fill= 149
      *trcl=( 0 0 0 -90 180 90 0 -90 90 90 0 ) u= 175 imp:n= 1.0
c      Blade Window, CC175
201      2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
      u= 175 imp:n= 1.0
      fill= 3 trcl=( 0.0 0.0 -3.6576E+02 )
c      Balance of Control Cell, CC175
202 2 -9.8114E-01 #197 #198 #199 #200 #201
      u= 175 imp:n= 1.0
c      Fuel Assembly in Northwest Quadrant, CC176
203      2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 176 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 0 )
      fill= 149
c      Fuel Assembly in Northeast Quadrant, CC176
204 like 203 but fill= 149
      *trcl=( 0 0 0 90 0 90 180 90 90 90 0 )
      u= 176 imp:n= 1.0
c      Fuel Assembly in Southeast Quadrant, CC176

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 57 of 76

205 like 204 but fill= 147
trcl=(0 0 0) u= 176 imp:n= 1.0
c Fuel Assembly in Southwest Quadrant, CC176

206 like 203 but fill= 149
*trcl=(0 0 0 -90 180 90 0 -90 90 90 90 0) u= 176 imp:n= 1.0
c Blade Window, CC176

207 2 -9.8114E-01 (97 -98 99 -100):(101 -102 103 -104)
u= 176 imp:n= 1.0
fill= 3 trcl=(0.0 0.0 0.0000E+00)
c Balance of Control Cell, CC176

208 2 -9.8114E-01 #203 #204 #205 #206 #207
u= 176 imp:n= 1.0
c Fuel Assembly in Northwest Quadrant, CC177

209 2 -9.8114E-01 ((-90 91 -92 93)
(#(-90 93 94 -95 96)))
u= 177 imp:n= 1.0
*trcl=(0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0)
fill= 147
c Fuel Assembly in Northeast Quadrant, CC177

210 like 209 but fill= 150
*trcl=(0 0 0 90 0 90 180 90 90 90 90 0)
u= 177 imp:n= 1.0
c Fuel Assembly in Southeast Quadrant, CC177

211 like 210 but fill= 148
trcl=(0 0 0) u= 177 imp:n= 1.0
c Fuel Assembly in Southwest Quadrant, CC177

212 like 209 but fill= 149
*trcl=(0 0 0 -90 180 90 0 -90 90 90 90 0) u= 177 imp:n= 1.0
c Blade Window, CC177

213 2 -9.8114E-01 (97 -98 99 -100):(101 -102 103 -104)
u= 177 imp:n= 1.0
fill= 3 trcl=(0.0 0.0 0.0000E+00)
c Guide Tube Segment, CC177

214 1 -7.9000E+00 -105 106
trcl=(0.0 3.0480E+01 0.0) u= 177 imp:n= 1.0
c Inside Guide Tube Segment, CC177

215 0 -106 trcl=(0.0 3.0480E+01 0.0) u= 177 imp:n= 1.0
c Balance of Control Cell, CC177

216 2 -9.8114E-01 #209 #210 #211 #212 #213 #214 #215
u= 177 imp:n= 1.0
c Fuel Assembly in Northwest Quadrant, CC178

217 2 -9.8114E-01 ((-90 91 -92 93)
(#(-90 93 94 -95 96)))
u= 178 imp:n= 1.0
*trcl=(0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0)
fill= 148
c Fuel Assembly in Northeast Quadrant, CC178

218 like 217 but fill= 150
*trcl=(0 0 0 90 0 90 180 90 90 90 90 0)
u= 178 imp:n= 1.0
c Fuel Assembly in Southeast Quadrant, CC178

219 like 218 but fill= 148
trcl=(0 0 0) u= 178 imp:n= 1.0
c Fuel Assembly in Southwest Quadrant, CC178

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 58 of 76

```
220 like 217 but fill= 150
    *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 178 imp:n= 1.0
c   Blade Window, CC178
221     2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
    u= 178 imp:n= 1.0
    fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c   Guide Tube Segment, CC178
222     1 -7.9000E+00 -105 106
    trcl=( -3.0480E+01 3.0480E+01 0.0) u= 178 imp:n= 1.0
c   Inside Guide Tube Segment, CC178
223     0 -106          trcl=( -3.0480E+01 3.0480E+01 0.0) u= 178 imp:n= 1.0
c   Balance of Control Cell, CC178
224 2 -9.8114E-01 #217 #218 #219 #220 #221 #222 #223
    u= 178 imp:n= 1.0
c   Fuel Assembly in Northwest Quadrant, CC179
225     2 -9.8114E-01 (( -90 91 -92 93 )
    (#( -90 93 94 -95 96 )))
    u= 179 imp:n= 1.0
    *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
    fill= 147
c   Fuel Assembly in Northeast Quadrant, CC179
226 like 225 but fill= 149
    *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
    u= 179 imp:n= 1.0
c   Fuel Assembly in Southeast Quadrant, CC179
227 like 226 but fill= 147
    trcl=( 0 0 0 ) u= 179 imp:n= 1.0
c   Fuel Assembly in Southwest Quadrant, CC179
228 like 225 but fill= 149
    *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 179 imp:n= 1.0
c   Blade Window, CC179
229     2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
    u= 179 imp:n= 1.0
    fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c   Guide Tube Segment, CC179
230     1 -7.9000E+00 -105 106
    trcl=( -3.0480E+01 3.0480E+01 0.0) u= 179 imp:n= 1.0
c   Inside Guide Tube Segment, CC179
231     0 -106          trcl=( -3.0480E+01 3.0480E+01 0.0) u= 179 imp:n= 1.0
c   Balance of Control Cell, CC179
232 2 -9.8114E-01 #225 #226 #227 #228 #229 #230 #231
    u= 179 imp:n= 1.0
c   Fuel Assembly in Northwest Quadrant, CC180
233     2 -9.8114E-01 (( -90 91 -92 93 )
    (#( -90 93 94 -95 96 )))
    u= 180 imp:n= 1.0
    *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
    fill= 147
c   Fuel Assembly in Northeast Quadrant, CC180
234 like 233 but fill= 149
    *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
    u= 180 imp:n= 1.0
c   Fuel Assembly in Southeast Quadrant, CC180
235 like 234 but fill= 147
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 59 of 76

```
      trcl=( 0 0 0 ) u= 180 imp:n= 1.0
c      Fuel Assembly in Southwest Quadrant, CC180
236 like 233 but fill= 149
      *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0 ) u= 180 imp:n= 1.0
c      Blade Window, CC180
237      2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
      u= 180 imp:n= 1.0
      fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c      Guide Tube Segment, CC180
238      1 -7.9000E+00 -105 106 u= 180 imp:n= 1.0
c      Inside Guide Tube Segment, CC180
239      0 -106 u= 180 imp:n= 1.0
c      Guide Tube Segment (2), CC180
240      1 -7.9000E+00 -105 106
      trcl=( 0.0 3.0480E+01 0.0 ) u= 180 imp:n= 1.0
c      Inside Guide Tube Segment (2), CC180
241      0 -106      trcl=( 0.0 3.0480E+01 0.0 ) u= 180 imp:n= 1.0
c      Balance of Control Cell, CC180
242 2 -9.8114E-01 #233 #234 #235 #236 #237 #238 #239 #240 #241
      u= 180 imp:n= 1.0
c      Fuel Assembly in Northwest Quadrant, CC181
243      2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 181 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
      fill= 147
c      Fuel Assembly in Northeast Quadrant, CC181
244 like 243 but fill= 149
      *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0 )
      u= 181 imp:n= 1.0
c      Fuel Assembly in Southeast Quadrant, CC181
245 like 244 but fill= 147
      trcl=( 0 0 0 ) u= 181 imp:n= 1.0
c      Fuel Assembly in Southwest Quadrant, CC181
246 like 243 but fill= 149
      *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0 ) u= 181 imp:n= 1.0
c      Blade Window, CC181
247      2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
      u= 181 imp:n= 1.0
      fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c      Guide Tube Segment, CC181
248      1 -7.9000E+00 -105 106
      trcl=( -3.0480E+01 0.0 0.0 ) u= 181 imp:n= 1.0
c      Inside Guide Tube Segment, CC181
249      0 -106      trcl=( -3.0480E+01 0.0 0.0 ) u= 181 imp:n= 1.0
c      Guide Tube Segment (2), CC181
250      1 -7.9000E+00 -105 106
      trcl=( -3.0480E+01 3.0480E+01 0.0 ) u= 181 imp:n= 1.0
c      Inside Guide Tube Segment (2), CC181
251      0 -106      trcl=( -3.0480E+01 3.0480E+01 0.0 ) u= 181 imp:n= 1.0
c      Balance of Control Cell, CC181
252 2 -9.8114E-01 #243 #244 #245 #246 #247 #248 #249 #250 #251
      u= 181 imp:n= 1.0
c      Fuel Assembly in Northwest Quadrant, CC182
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 60 of 76

```
253      2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 182 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
      fill= 147
c      Fuel Assembly in Northeast Quadrant, CC182
254 like 253 but fill= 0
      *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
      u= 182 imp:n= 1.0
c      Fuel Assembly in Southeast Quadrant, CC182
255 like 254 but fill= 0
      trcl=( 0 0 0 ) u= 182 imp:n= 1.0
c      Fuel Assembly in Southwest Quadrant, CC182
256 like 253 but fill= 149
      *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 182 imp:n= 1.0
c      Blade Window, CC182
257      2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
      u= 182 imp:n= 1.0
      fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c      Balance of Control Cell, CC182
258 2 -9.8114E-01 #253 #254 #255 #256 #257
      u= 182 imp:n= 1.0
c      Fuel Assembly in Northwest Quadrant, CC183
259      2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 183 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
      fill= 147
c      Fuel Assembly in Northeast Quadrant, CC183
260 like 259 but fill= 150
      *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
      u= 183 imp:n= 1.0
c      Fuel Assembly in Southeast Quadrant, CC183
261 like 260 but fill= 148
      trcl=( 0 0 0 ) u= 183 imp:n= 1.0
c      Fuel Assembly in Southwest Quadrant, CC183
262 like 259 but fill= 149
      *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 183 imp:n= 1.0
c      Blade Window, CC183
263      2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
      u= 183 imp:n= 1.0
      fill= 3 trcl=( 0.0 0.0 -3.6576E+02 )
c      Guide Tube Segment, CC183
264      1 -7.9000E+00 -105 106 u= 183 imp:n= 1.0
c      Inside Guide Tube Segment, CC183
265      0 -106 u= 183 imp:n= 1.0
c      Balance of Control Cell, CC183
266 2 -9.8114E-01 #259 #260 #261 #262 #263 #264 #265
      u= 183 imp:n= 1.0
c      Fuel Assembly in Northwest Quadrant, CC184
267      2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 184 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 61 of 76

```
fill= 148
c Fuel Assembly in Northeast Quadrant, CC184
268 like 267 but fill= 149
*trcl=( 0 0 0 90 0 90 180 90 90 90 0)
u= 184 imp:n= 1.0
c Fuel Assembly in Southeast Quadrant, CC184
269 like 268 but fill= 147
trcl=( 0 0 0 ) u= 184 imp:n= 1.0
c Fuel Assembly in Southwest Quadrant, CC184
270 like 267 but fill= 150
*trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 184 imp:n= 1.0
c Blade Window, CC184
271 2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
u= 184 imp:n= 1.0
fill= 3 trcl=( 0.0 0.0 -3.6576E+02 )
c Guide Tube Segment, CC184
272 1 -7.9000E+00 -105 106 u= 184 imp:n= 1.0
c Inside Guide Tube Segment, CC184
273 0 -106 u= 184 imp:n= 1.0
c Balance of Control Cell, CC184
274 2 -9.8114E-01 #267 #268 #269 #270 #271 #272 #273
u= 184 imp:n= 1.0
c Fuel Assembly in Northwest Quadrant, CC185
275 2 -9.8114E-01 (( -90 91 -92 93 )
(#( -90 93 94 -95 96 )))
u= 185 imp:n= 1.0
*trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
fill= 147
c Fuel Assembly in Northeast Quadrant, CC185
276 like 275 but fill= 149
*trcl=( 0 0 0 90 0 90 180 90 90 90 0)
u= 185 imp:n= 1.0
c Fuel Assembly in Southeast Quadrant, CC185
277 like 276 but fill= 147
trcl=( 0 0 0 ) u= 185 imp:n= 1.0
c Fuel Assembly in Southwest Quadrant, CC185
278 like 275 but fill= 149
*trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 185 imp:n= 1.0
c Blade Window, CC185
279 2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
u= 185 imp:n= 1.0
fill= 3 trcl=( 0.0 0.0 -3.6576E+02 )
c Guide Tube Segment, CC185
280 1 -7.9000E+00 -105 106 u= 185 imp:n= 1.0
c Inside Guide Tube Segment, CC185
281 0 -106 u= 185 imp:n= 1.0
c Guide Tube Segment (2), CC185
282 1 -7.9000E+00 -105 106
trcl=( 0.0 3.0480E+01 0.0) u= 185 imp:n= 1.0
c Inside Guide Tube Segment (2), CC185
283 0 -106 trcl=( 0.0 3.0480E+01 0.0) u= 185 imp:n= 1.0
c Balance of Control Cell, CC185
284 2 -9.8114E-01 #275 #276 #277 #278 #279 #280 #281 #282 #283
u= 185 imp:n= 1.0
```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 62 of 76

```
c   Fuel Assembly in Northwest Quadrant, CC186
285   2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 186 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
      fill= 149
c   Fuel Assembly in Northeast Quadrant, CC186
286 like 285 but fill= 149
      *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
      u= 186 imp:n= 1.0
c   Fuel Assembly in Southeast Quadrant, CC186
287 like 286 but fill= 0
      trcl=( 0 0 0 ) u= 186 imp:n= 1.0
c   Fuel Assembly in Southwest Quadrant, CC186
288 like 285 but fill= 0
      *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 186 imp:n= 1.0
c   Blade Window, CC186
289   2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
      u= 186 imp:n= 1.0
      fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c   Balance of Control Cell, CC186
290 2 -9.8114E-01 #285 #286 #287 #288 #289
      u= 186 imp:n= 1.0
c   Fuel Assembly in Northwest Quadrant, CC187
291   2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 187 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
      fill= 147
c   Fuel Assembly in Northeast Quadrant, CC187
292 like 291 but fill= 150
      *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
      u= 187 imp:n= 1.0
c   Fuel Assembly in Southeast Quadrant, CC187
293 like 292 but fill= 0
      trcl=( 0 0 0 ) u= 187 imp:n= 1.0
c   Fuel Assembly in Southwest Quadrant, CC187
294 like 291 but fill= 0
      *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 187 imp:n= 1.0
c   Blade Window, CC187
295   2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
      u= 187 imp:n= 1.0
      fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c   Guide Tube Segment, CC187
296   1 -7.9000E+00 -105 106
      trcl=( 0.0 3.0480E+01 0.0) u= 187 imp:n= 1.0
c   Inside Guide Tube Segment, CC187
297   0 -106      trcl=( 0.0 3.0480E+01 0.0) u= 187 imp:n= 1.0
c   Balance of Control Cell, CC187
298 2 -9.8114E-01 #291 #292 #293 #294 #295 #296 #297
      u= 187 imp:n= 1.0
c   Fuel Assembly in Northwest Quadrant, CC188
299   2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
```

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 63 of 76

```

u= 188 imp:n= 1.0
*trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
fill= 148
c Fuel Assembly in Northeast Quadrant, CC188
300 like 299 but fill= 150
*trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
u= 188 imp:n= 1.0
c Fuel Assembly in Southeast Quadrant, CC188
301 like 300 but fill= 0
trcl=( 0 0 0 ) u= 188 imp:n= 1.0
c Fuel Assembly in Southwest Quadrant, CC188
302 like 299 but fill= 0
*trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 188 imp:n= 1.0
c Blade Window, CC188
303 2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
u= 188 imp:n= 1.0
fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c Guide Tube Segment, CC188
304 1 -7.9000E+00 -105 106
trcl=( -3.0480E+01 3.0480E+01 0.0) u= 188 imp:n= 1.0
c Inside Guide Tube Segment, CC188
305 0 -106 trcl=( -3.0480E+01 3.0480E+01 0.0) u= 188 imp:n= 1.0
c Balance of Control Cell, CC188
306 2 -9.8114E-01 #299 #300 #301 #302 #303 #304 #305
u= 188 imp:n= 1.0
c Fuel Assembly in Northwest Quadrant, CC189
307 2 -9.8114E-01 (( -90 91 -92 93 )
(#( -90 93 94 -95 96 )))
u= 189 imp:n= 1.0
*trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
fill= 148
c Fuel Assembly in Northeast Quadrant, CC189
308 like 307 but fill= 149
*trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
u= 189 imp:n= 1.0
c Fuel Assembly in Southeast Quadrant, CC189
309 like 308 but fill= 0
trcl=( 0 0 0 ) u= 189 imp:n= 1.0
c Fuel Assembly in Southwest Quadrant, CC189
310 like 307 but fill= 0
*trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 189 imp:n= 1.0
c Blade Window, CC189
311 2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
u= 189 imp:n= 1.0
fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c Guide Tube Segment, CC189
312 1 -7.9000E+00 -105 106
trcl=( 0.0 3.0480E+01 0.0) u= 189 imp:n= 1.0
c Inside Guide Tube Segment, CC189
313 0 -106 trcl=( 0.0 3.0480E+01 0.0) u= 189 imp:n= 1.0
c Balance of Control Cell, CC189
314 2 -9.8114E-01 #307 #308 #309 #310 #311 #312 #313
u= 189 imp:n= 1.0
c Fuel Assembly in Northwest Quadrant, CC190

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 64 of 76

```
315      2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 190 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 0 )
      fill= 147
c      Fuel Assembly in Northeast Quadrant, CC190
316 like 315 but fill= 149
      *trcl=( 0 0 0 90 0 90 180 90 90 90 0 )
      u= 190 imp:n= 1.0
c      Fuel Assembly in Southeast Quadrant, CC190
317 like 316 but fill= 0
      trcl=( 0 0 0 ) u= 190 imp:n= 1.0
c      Fuel Assembly in Southwest Quadrant, CC190
318 like 315 but fill= 0
      *trcl=( 0 0 0 -90 180 90 0 -90 90 90 0 ) u= 190 imp:n= 1.0
c      Blade Window, CC190
319      2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104 )
      u= 190 imp:n= 1.0
      fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c      Guide Tube Segment, CC190
320      1 -7.9000E+00 -105 106
      trcl=( -3.0480E+01 3.0480E+01 0.0 ) u= 190 imp:n= 1.0
c      Inside Guide Tube Segment, CC190
321      0 -106      trcl=( -3.0480E+01 3.0480E+01 0.0 ) u= 190 imp:n= 1.0
c      Balance of Control Cell, CC190
322 2 -9.8114E-01 #315 #316 #317 #318 #319 #320 #321
      u= 190 imp:n= 1.0
c      Fuel Assembly in Northwest Quadrant, CC191
323      2 -9.8114E-01 (( -90 91 -92 93 )
      (#( -90 93 94 -95 96 )))
      u= 191 imp:n= 1.0
      *trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 0 )
      fill= 147
c      Fuel Assembly in Northeast Quadrant, CC191
324 like 323 but fill= 149
      *trcl=( 0 0 0 90 0 90 180 90 90 90 0 )
      u= 191 imp:n= 1.0
c      Fuel Assembly in Southeast Quadrant, CC191
325 like 324 but fill= 0
      trcl=( 0 0 0 ) u= 191 imp:n= 1.0
c      Fuel Assembly in Southwest Quadrant, CC191
326 like 323 but fill= 0
      *trcl=( 0 0 0 -90 180 90 0 -90 90 90 0 ) u= 191 imp:n= 1.0
c      Blade Window, CC191
327      2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104 )
      u= 191 imp:n= 1.0
      fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c      Guide Tube Segment, CC191
328      1 -7.9000E+00 -105 106
      trcl=( 0.0 3.0480E+01 0.0 ) u= 191 imp:n= 1.0
c      Inside Guide Tube Segment, CC191
329      0 -106      trcl=( 0.0 3.0480E+01 0.0 ) u= 191 imp:n= 1.0
c      Balance of Control Cell, CC191
330 2 -9.8114E-01 #323 #324 #325 #326 #327 #328 #329
```

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 65 of 76

```

u= 191 imp:n= 1.0
c Fuel Assembly in Northwest Quadrant, CC192
331 2 -9.8114E-01 (( -90 91 -92 93 )
  (#( -90 93 94 -95 96 )))
u= 192 imp:n= 1.0
*trcl=( 0.0 0.0 0.0 180 90 90 -90 180 90 90 90 0 )
fill= 147
c Fuel Assembly in Northeast Quadrant, CC192
332 like 331 but fill= 0
*trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)
u= 192 imp:n= 1.0
c Fuel Assembly in Southeast Quadrant, CC192
333 like 332 but fill= 0
trcl=( 0 0 0 ) u= 192 imp:n= 1.0
c Fuel Assembly in Southwest Quadrant, CC192
334 like 331 but fill= 0
*trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0) u= 192 imp:n= 1.0
c Blade Window, CC192
335 2 -9.8114E-01 ( 97 -98 99 -100 ):( 101 -102 103 -104)
u= 192 imp:n= 1.0
fill= 3 trcl=( 0.0 0.0 0.0000E+00 )
c Balance of Control Cell, CC192
336 2 -9.8114E-01 #331 #332 #333 #334 #335
u= 192 imp:n= 1.0
c Active Core
2620 2 -9.8114E-01 -8 4 -5 -9 12
fill=235
imp:n= 1.0
c Core Lattice
2621 2 -9.8114E-01 -107 108 109 -110
lat=1 u=235 imp:n= 1.0
fill= -8:0 -8:0 0:0
235 235 235 235 235 235 235 235 235
235 235 235 235 235 151 152 153 154
235 235 235 151 155 153 156 157 158
235 235 159 153 160 161 162 163 164
235 235 165 166 167 161 168 169 170
235 151 152 163 171 172 173 174 175
235 176 177 178 168 179 180 181 182
235 176 183 163 184 174 185 181 175
235 186 187 188 189 190 191 190 192

c Surface Cards
c
c Surfaces for Problem Domain
c Top of Problem
1 pz 4.3300E+02
c Bottom of Problem
2 pz -4.8570E+01
c Maximum Radial Extent of Problem
3 cz 3.3433E+02
c X-Z Plane
*4 py 0.0
c Y-Z Plane

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 66 of 76

*5 px 0.0
c Inner Radial Surface of Vessel
6 cz 3.1877E+02
c Outer Radial Surface of Core Shroud
7 cz 2.6305E+02
c Inner Radial Surface of Core Shroud
8 cz 2.5797E+02
c Top of Active Fuel
9 pz 3.6576E+02
c Top of Upper Tie Plate Region
10 pz 3.9900E+02
c Top of Core Grid Region
11 pz 4.0300E+02
c Bottom of Active Fuel
12 pz 0.0
c Bottom of Lower Tie Plate Region
13 pz -1.3490E+01
c Bottom of Fuel Support/Core Plate Region
14 pz -1.8570E+01
c Outer Radius of Absorber
15 c/z 2.2292E+00 0.0000E+00 1.7550E-01
c Outer Radius of Absorber Tube
16 c/z 2.2292E+00 0.0000E+00 2.3900E-01
c XMAX for Reference Absorber Tube Cell
17 px 2.4733E+00
c XMIN for Reference Absorber Tube Cell
18 px 1.9850E+00
c YMAX for Reference Absorber Tube Cell
19 py 2.5400E-01
c YMIN for Reference Absorber Tube Cell
20 py -2.5400E-01
c Tie Rod Surface #6
21 py 3.2400E-01
c Tie Rod Surface #7
22 py -3.2400E-01
c Tie Rod Surface #4
23 px -1.9850E+00
c Top of Absorber Column
24 pz 3.6512E+02
c Bottom of Absorber Column
25 pz 1.9250E+00
c Tie Rod Surface #3
26 px -3.2400E-01
c Tie Rod Surface #2
27 px 3.2400E-01
c Tie Rod Surface #5
28 py -1.9850E+00
c Tie Rod Surface #8
29 py 1.9850E+00
c Ambiguity Surface for Sheath Corner
30 px 1.1996E+01
c Inner Surface for Sheath Corner
31 c/z 1.1996E+01 0.0000E+00 2.5400E-01
c Outer Surface for Sheath Corner

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 67 of 76

32 c/z 1.1996E+01 0.0000E+00 3.2400E-01
c XMIN for Sheath Interior
33 px 1.9860
c YMIN for Sheath Interior
34 py -0.2530
c YMAX for Sheath Interior
35 py -0.2530
c XMIN for Blade Window
36 px 1.9950E+00
c XMAX for Blade Window
37 px 1.2390E+01
c YMIN for Blade Window
38 py -6.5800E-01
c YMAX for Blade Window
39 py 6.5800E-01
c Fuel Pellet Outer Surface, #1, L1
40 c/z 2.2325E+00 -2.2325E+00 6.2000E-01
c Fuel Cladding Inner Surface, #1, L1
41 c/z 2.2325E+00 -2.2325E+00 6.3400E-01
c Fuel Cladding Outer Surface, #1, L1
42 c/z 2.2325E+00 -2.2325E+00 7.1500E-01
c XMAX Surface for Fuel Rod Window, L1
43 px 3.1700E+00
c XMIN Surface for Fuel Rod Window, L1
44 px 1.2950E+00
c YMIN Surface for Fuel Rod Window, L1
45 py -3.1700E+00
c YMAX Surface for Fuel Rod Window, L1
46 py -1.2950E+00
c Wide Gap, Channel Outside Wall, pX Surface, L1
47 px 9.5300E-01
c Wide Gap, Channel Inside Wall, pX Surface, L1
48 px 1.1530E+00
c Ambiguity Surface for Channel Corners (Wide Gap), pY, L1
49 py -2.1730E+00
c Ambiguity Surface for Channel Corners (Narrow Gap), pY, L1
50 py -1.3539E+01
c Wide Gap, Channel Outside Wall, pY Surface, L1
51 py -9.5300E-01
c Wide Gap, Channel Inside Wall, pY Surface, L1
52 py -1.1530E+00
c Ambiguity Surface for Channel Corners (Wide Gap), pX, L1
53 px 2.1730E+00
c Ambiguity Surface for Channel Corners (Narrow Gap), pX, L1
54 px 1.3539E+01
c Narrow Gap, Channel Inside Wall, pY Surface, L1
55 py -1.4559E+01
c Narrow Gap, Channel Outside Wall, pY Surface, L1
56 py -1.4759E+01
c Narrow Gap, Channel Inside Wall, pX Surface, L1
57 px 1.4559E+01
c Narrow Gap, Channel Outside Wall, pX Surface, L1
58 px 1.4759E+01
c Channel Corner Outer Radius (NW), L1

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 68 of 76

59 c/z 2.1730E+00 -2.1730E+00 1.2200E+00
c Channel Corner Inner Radius (NW), L1
60 c/z 2.1730E+00 -2.1730E+00 1.0200E+00
c Channel Corner Outer Radius (NE), L1
61 c/z 1.3539E+01 -2.1730E+00 1.2200E+00
c Channel Corner Inner Radius (NE), L1
62 c/z 1.3539E+01 -2.1730E+00 1.0200E+00
c Channel Corner Outer Radius (SE), L1
63 c/z 1.3539E+01 -1.3539E+01 1.2200E+00
c Channel Corner Inner Radius (SE), L1
64 c/z 1.3539E+01 -1.3539E+01 1.0200E+00
c Channel Corner Outer Radius (SW), L1
65 c/z 2.1730E+00 -1.3539E+01 1.2200E+00
c Channel Corner Inner Radius (SW), L1
66 c/z 2.1730E+00 -1.3539E+01 1.0200E+00
c Top of Node 1
67 pz 1.5240E+01
c Top of Node 2
68 pz 3.0480E+01
c Top of Node 3
69 pz 4.5720E+01
c Top of Node 4
70 pz 6.0960E+01
c Top of Node 5
71 pz 7.6200E+01
c Top of Node 6
72 pz 9.1440E+01
c Top of Node 7
73 pz 1.0668E+02
c Top of Node 8
74 pz 1.2192E+02
c Top of Node 9
75 pz 1.3716E+02
c Top of Node 10
76 pz 1.5240E+02
c Top of Node 11
77 pz 1.6764E+02
c Top of Node 12
78 pz 1.8288E+02
c Top of Node 13
79 pz 1.9812E+02
c Top of Node 14
80 pz 2.1336E+02
c Top of Node 15
81 pz 2.2860E+02
c Top of Node 16
82 pz 2.4384E+02
c Top of Node 17
83 pz 2.5908E+02
c Top of Node 18
84 pz 2.7432E+02
c Top of Node 19
85 pz 2.8956E+02
c Top of Node 20

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 69 of 76

86 pz 3.0480E+02
c Top of Node 21
87 pz 3.2004E+02
c Top of Node 22
88 pz 3.3528E+02
c Top of Node 23
89 pz 3.5052E+02
c Window for Fuel Assembly (max X), CC151
90 px 1.5140E+01
c Window for Fuel Assembly (min X), CC151
91 px 5.0000E-01
c Window for Fuel Assembly (max Y), CC151
92 py -5.0000E-01
c Window for Fuel Assembly (min Y), CC151
93 py -1.5140E+01
c Curved Corner in Window for Fuel Assembly, CC151
94 c/z 1.3091E+01 -1.3091E+01 2.0486E+00
c Y Ambiguity Surface for Fuel Assembly Window, CC151
95 py -1.3091E+01
c X Ambiguity Surface for Fuel Assembly Window, CC151
96 px 1.3091E+01
c Surface 1 for Control Blade Window
97 py -4.0000E-01
c Surface 2 for Control Blade Window
98 py 4.0000E-01
c Surface 3 for Control Blade Window
99 px -1.2480E+01
c Surface 4 for Control Blade Window
100 px 1.2480E+01
c Surface 5 for Control Blade Window
101 px -4.0000E-01
c Surface 6 for Control Blade Window
102 px 4.0000E-01
c Surface 7 for Control Blade Window
103 py -1.2480E+01
c Surface 8 for Control Blade Window
104 py 1.2480E+01
c Outer Surface of Guide Tube
105 c/z 1.5240E+01 -1.5240E+01 8.9000E-01
c Inner Surface of Guide Tube
106 c/z 1.5240E+01 -1.5240E+01 8.1500E-01
c Edge of Control Cell (max X)
107 px 1.5240E+01
c Edge of Control Cell (min X)
108 px -1.5240E+01
c Edge of Control Cell (min Y)
109 py -1.5240E+01
c Edge of Control Cell (max Y)
110 py 1.5240E+01

c Material Cards
c SS304
m1 006000.50c -8.0000E-02 025055.50c -2.0000E+00 015031.50c -4.5000E-02
016032.50c -3.0000E-02 014000.50c -7.5000E-01 024050.60c -7.9401E-01

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 70 of 76

```

024052.60c -1.5903E+01 024053.60c -1.8379E+00 024054.60c -4.6512E-01
028058.60c -6.2161E+00 028060.60c -2.4765E+00 028061.60c -1.0943E-01
028062.60c -3.5427E-01 028064.60c -9.3702E-02 007014.50c -1.0000E-01
026054.60c -3.8841E+00 026056.60c -6.3175E+01 026057.60c -1.4856E+00
026058.60c -1.9936E-01
c   Bypass Water
m2 001001.50c 2.0
    008016.50c 1.0
mt2 lwtr.01
c   7GUTP1
m3 001001.50c -7.0530E+00 006000.50c -4.0000E-03 007014.50c -6.0000E-03
    008016.50c -5.6015E+01 014000.50c -4.1000E-02 015031.50c -2.0000E-03
    016032.50c -2.0000E-03 024050.60c -4.5175E-02 024052.60c -9.0481E-01
    024053.60c -1.0457E-01 024054.60c -2.6463E-02 025055.50c -1.1000E-01
    026054.60c -2.1781E-01 026056.60c -3.5427E+00 026057.60c -8.3307E-02
    026058.60c -1.1179E-02 028058.60c -3.4474E-01 028060.60c -1.3735E-01
    028061.60c -6.0688E-03 028062.60c -1.9648E-02 028064.60c -5.1967E-03
    040000.60c -3.0876E+01 050000.35c -4.4000E-01
c   3TG1
m4 001001.50c -1.0415E+01 006000.50c -6.0000E-03 007014.50c -7.0000E-03
    008016.50c -8.2657E+01 014000.50c -5.2000E-02 015031.50c -3.0000E-03
    016032.50c -2.0000E-03 024050.60c -5.4996E-02 024052.60c -1.1015E+00
    024053.60c -1.2730E-01 024054.60c -3.2216E-02 025055.50c -1.3900E-01
    026054.60c -2.6911E-01 026056.60c -4.3771E+00 026057.60c -1.0293E-01
    026058.60c -1.3813E-02 028058.60c -4.3076E-01 028060.60c -1.7161E-01
    028061.60c -7.5830E-03 028062.60c -2.4550E-02 028064.60c -6.4933E-03
c   7GLTP1
m5 001001.50c -3.0160E+00 006000.50c -4.4000E-02 007014.50c -5.5000E-02
    008016.50c -2.3957E+01 014000.50c -4.1200E-01 015031.50c -2.5000E-02
    016032.50c -1.6000E-02 024050.60c -4.3725E-01 024052.60c -8.7576E+00
    024053.60c -1.0121E+00 024054.60c -2.5613E-01 025055.50c -1.0990E+00
    026054.60c -2.1366E+00 026056.60c -3.4752E+01 026057.60c -8.1720E-01
    026058.60c -1.0967E-01 028058.60c -3.4205E+00 028060.60c -1.3627E+00
    028061.60c -6.0215E-02 028062.60c -1.9495E-01 028064.60c -5.1562E-02
    040000.60c -1.7754E+01 050000.35c -2.5300E-01
c   3CPI
m6 001001.50c -3.3430E+00 006000.50c -5.6000E-02 007014.50c -7.0000E-02
    008016.50c -2.6533E+01 014000.50c -5.2600E-01 015031.50c -3.2000E-02
    016032.50c -2.1000E-02 024050.60c -5.5681E-01 024052.60c -1.1152E+01
    024053.60c -1.2888E+00 024054.60c -3.2617E-01 025055.50c -1.4020E+00
    026054.60c -2.7237E+00 026056.60c -4.4301E+01 026057.60c -1.0418E+00
    026058.60c -1.3980E-01 028058.60c -4.3587E+00 028060.60c -1.7365E+00
    028061.60c -7.6729E-02 028062.60c -2.4841E-01 028064.60c -6.5703E-02
c   B4C
m7 005010.50c -1.4420E+01 005011.56c -6.3840E+01 006000.50c -2.0000E+01
c   Inchannel Water
m8 001001.50c 2.0
    008016.50c 1.0
mt8 lwtr.01
c   G7212G003DL1.dat (1)
m9 092234.50c -1.8000E-04 092235.50c -2.1770E-02 092236.50c -1.0000E-04
    092238.50c -8.5942E-01 008016.50c -1.1853E-01
c   ZIRC2
m10 008016.50c -1.2000E-01 024050.60c -4.1790E-03 024052.60c -8.3701E-02

```

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 71 of 76

	024053.60c	-9.6730E-03	024054.60c	-2.4480E-03	026054.60c	-5.6500E-03
	026056.60c	-9.1898E-02	026057.60c	-2.1610E-03	026058.60c	-2.9000E-04
	028058.60c	-3.3600E-02	028060.60c	-1.3387E-02	028061.60c	-5.9150E-04
	028062.60c	-1.9150E-03	028064.60c	-5.0650E-04	050000.35c	-1.4000E+00
	040000.60c	-9.8230E+01				
c	G7212G003DL1.dat (2)					
m11	092234.50c	-1.8000E-04	092235.50c	-2.1770E-02	092236.50c	-1.0000E-04
	092238.50c	-8.5942E-01	008016.50c	-1.1853E-01		
c	G7212G003DL1.dat (3)					
m12	092234.50c	-1.8000E-04	092235.50c	-2.1770E-02	092236.50c	-1.0000E-04
	092238.50c	-8.5942E-01	008016.50c	-1.1853E-01		
c	G7212G003DL1.dat (4)					
m13	092234.50c	-1.2000E-04	092235.50c	-1.4990E-02	092236.50c	-7.0000E-05
	092238.50c	-8.6631E-01	008016.50c	-1.1852E-01		
c	G7212G003DL1.dat (5)					
m14	092234.50c	-1.2000E-04	092235.50c	-1.4990E-02	092236.50c	-7.0000E-05
	092238.50c	-8.6631E-01	008016.50c	-1.1852E-01		
c	G7212G003DL1.dat (6)					
m15	092234.50c	-8.0000E-05	092235.50c	-1.0580E-02	092236.50c	-5.0000E-05
	092238.50c	-8.7078E-01	008016.50c	-1.1851E-01		
c	G7212G003DL1.dat (7)					
m16	092234.50c	-1.8000E-04	092235.50c	-2.1450E-02	092236.50c	-1.0000E-04
	092238.50c	-8.4653E-01	008016.50c	-1.1873E-01	064152.50c	-3.0000E-05
	064154.50c	-2.8000E-04	064155.50c	-1.9000E-03	064156.50c	-2.6400E-03
	064157.50c	-2.0300E-03	064158.50c	-3.2500E-03	064160.50c	-2.8900E-03
c	G7212G003DL1.dat (8)					
m17	092234.50c	-1.8000E-04	092235.50c	-2.1770E-02	092236.50c	-1.0000E-04
	092238.50c	-8.5942E-01	008016.50c	-1.1853E-01		
c	ZIRC4					
m18	008016.50c	-1.2000E-01	024050.60c	-4.1790E-03	024052.60c	-8.3701E-02
	024053.60c	-9.6730E-03	024054.60c	-2.4480E-03	026054.60c	-1.1300E-02
	026056.60c	-1.8380E-01	026057.60c	-4.3220E-03	026058.60c	-5.8000E-04
	050000.35c	-1.4000E+00	040000.60c	-9.8180E+01		
c	G7212G006DL2.dat (1)					
m19	092234.50c	-1.8000E-04	092235.50c	-2.1770E-02	092236.50c	-1.0000E-04
	092238.50c	-8.5942E-01	008016.50c	-1.1853E-01		
c	G7212G006DL2.dat (2)					
m20	092234.50c	-1.8000E-04	092235.50c	-2.1770E-02	092236.50c	-1.0000E-04
	092238.50c	-8.5942E-01	008016.50c	-1.1853E-01		
c	G7212G006DL2.dat (3)					
m21	092234.50c	-1.8000E-04	092235.50c	-2.1770E-02	092236.50c	-1.0000E-04
	092238.50c	-8.5942E-01	008016.50c	-1.1853E-01		
c	G7212G006DL2.dat (4)					
m22	092234.50c	-1.2000E-04	092235.50c	-1.4990E-02	092236.50c	-7.0000E-05
	092238.50c	-8.6631E-01	008016.50c	-1.1852E-01		
c	G7212G006DL2.dat (5)					
m23	092234.50c	-1.2000E-04	092235.50c	-1.4990E-02	092236.50c	-7.0000E-05
	092238.50c	-8.6631E-01	008016.50c	-1.1852E-01		
c	G7212G006DL2.dat (6)					
m24	092234.50c	-8.0000E-05	092235.50c	-1.0580E-02	092236.50c	-5.0000E-05
	092238.50c	-8.7078E-01	008016.50c	-1.1851E-01		
c	G7212G006DL2.dat (7)					
m25	092234.50c	-1.8000E-04	092235.50c	-2.1120E-02	092236.50c	-1.0000E-04
	092238.50c	-8.3364E-01	008016.50c	-1.1894E-01	064152.50c	-5.0000E-05

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 72 of 76

	064154.50c	-5.6000E-04	064155.50c	-3.8000E-03	064156.50c	-5.2800E-03
	064157.50c	-4.0600E-03	064158.50c	-6.4900E-03	064160.50c	-5.7900E-03
c	G7212G006DL2.dat (8)					
m26	092234.50c	-1.8000E-04	092235.50c	-2.1770E-02	092236.50c	-1.0000E-04
	092238.50c	-8.5942E-01	008016.50c	-1.1853E-01		
c	G7212G007DL3.dat (1)					
m27	092234.50c	-1.8000E-04	092235.50c	-2.1770E-02	092236.50c	-1.0000E-04
	092238.50c	-8.5942E-01	008016.50c	-1.1853E-01		
c	G7212G007DL3.dat (2)					
m28	092234.50c	-1.8000E-04	092235.50c	-2.1770E-02	092236.50c	-1.0000E-04
	092238.50c	-8.5942E-01	008016.50c	-1.1853E-01		
c	G7212G007DL3.dat (3)					
m29	092234.50c	-1.8000E-04	092235.50c	-2.1770E-02	092236.50c	-1.0000E-04
	092238.50c	-8.5942E-01	008016.50c	-1.1853E-01		
c	G7212G007DL3.dat (4)					
m30	092234.50c	-1.2000E-04	092235.50c	-1.4990E-02	092236.50c	-7.0000E-05
	092238.50c	-8.6631E-01	008016.50c	-1.1852E-01		
c	G7212G007DL3.dat (5)					
m31	092234.50c	-1.2000E-04	092235.50c	-1.4990E-02	092236.50c	-7.0000E-05
	092238.50c	-8.6631E-01	008016.50c	-1.1852E-01		
c	G7212G007DL3.dat (6)					
m32	092234.50c	-8.0000E-05	092235.50c	-1.0580E-02	092236.50c	-5.0000E-05
	092238.50c	-8.7078E-01	008016.50c	-1.1851E-01		
c	G7212G007DL3.dat (7)					
m33	092234.50c	-1.8000E-04	092235.50c	-2.1120E-02	092236.50c	-1.0000E-04
	092238.50c	-8.3364E-01	008016.50c	-1.1894E-01	064152.50c	-5.0000E-05
	064154.50c	-5.6000E-04	064155.50c	-3.8000E-03	064156.50c	-5.2800E-03
	064157.50c	-4.0600E-03	064158.50c	-6.4900E-03	064160.50c	-5.7900E-03
c	G7212G007DL3.dat (8)					
m34	092234.50c	-1.8000E-04	092235.50c	-2.1660E-02	092236.50c	-1.0000E-04
	092238.50c	-8.5512E-01	008016.50c	-1.1859E-01	064152.50c	-1.0000E-05
	064154.50c	-9.0000E-05	064155.50c	-6.3000E-04	064156.50c	-8.8000E-04
	064157.50c	-6.8000E-04	064158.50c	-1.0800E-03	064160.50c	-9.6000E-04
c	G7211G003DL4.dat (1)					
m35	092234.50c	-1.8000E-04	092235.50c	-2.1770E-02	092236.50c	-1.0000E-04
	092238.50c	-8.5942E-01	008016.50c	-1.1853E-01		
c	G7211G003DL4.dat (2)					
m36	092234.50c	-1.8000E-04	092235.50c	-2.1770E-02	092236.50c	-1.0000E-04
	092238.50c	-8.5942E-01	008016.50c	-1.1853E-01		
c	G7211G003DL4.dat (3)					
m37	092234.50c	-1.2000E-04	092235.50c	-1.4990E-02	092236.50c	-7.0000E-05
	092238.50c	-8.6631E-01	008016.50c	-1.1852E-01		
c	G7211G003DL4.dat (4)					
m38	092234.50c	-8.0000E-05	092235.50c	-1.0580E-02	092236.50c	-5.0000E-05
	092238.50c	-8.7078E-01	008016.50c	-1.1851E-01		
c	G7211G003DL4.dat (5)					
m39	092234.50c	-1.8000E-04	092235.50c	-2.1450E-02	092236.50c	-1.0000E-04
	092238.50c	-8.4653E-01	008016.50c	-1.1873E-01	064152.50c	-3.0000E-05
	064154.50c	-2.8000E-04	064155.50c	-1.9000E-03	064156.50c	-2.6400E-03
	064157.50c	-2.0300E-03	064158.50c	-3.2500E-03	064160.50c	-2.8900E-03
c	G7211G003DL4.dat (6)					
m40	092234.50c	-1.8000E-04	092235.50c	-2.1770E-02	092236.50c	-1.0000E-04
	092238.50c	-8.5942E-01	008016.50c	-1.1853E-01		
c	G7211G006DL5.dat (1)					

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 73 of 76

m41 092234.50c -1.8000E-04 092235.50c -2.1770E-02 092236.50c -1.0000E-04
092238.50c -8.5942E-01 008016.50c -1.1853E-01
c G7211G006DL5.dat (2)

m42 092234.50c -1.8000E-04 092235.50c -2.1770E-02 092236.50c -1.0000E-04
092238.50c -8.5942E-01 008016.50c -1.1853E-01
c G7211G006DL5.dat (3)

m43 092234.50c -1.2000E-04 092235.50c -1.4990E-02 092236.50c -7.0000E-05
092238.50c -8.6631E-01 008016.50c -1.1852E-01
c G7211G006DL5.dat (4)

m44 092234.50c -8.0000E-05 092235.50c -1.0580E-02 092236.50c -5.0000E-05
092238.50c -8.7078E-01 008016.50c -1.1851E-01
c G7211G006DL5.dat (5)

m45 092234.50c -1.8000E-04 092235.50c -2.1120E-02 092236.50c -1.0000E-04
092238.50c -8.3364E-01 008016.50c -1.1894E-01 064152.50c -5.0000E-05
064154.50c -5.6000E-04 064155.50c -3.8000E-03 064156.50c -5.2800E-03
064157.50c -4.0600E-03 064158.50c -6.4900E-03 064160.50c -5.7900E-03
c G7211G006DL5.dat (6)

m46 092234.50c -1.8000E-04 092235.50c -2.1770E-02 092236.50c -1.0000E-04
092238.50c -8.5942E-01 008016.50c -1.1853E-01
c G7211G007DL6.dat (1)

m47 092234.50c -1.8000E-04 092235.50c -2.1770E-02 092236.50c -1.0000E-04
092238.50c -8.5942E-01 008016.50c -1.1853E-01
c G7211G007DL6.dat (2)

m48 092234.50c -1.8000E-04 092235.50c -2.1770E-02 092236.50c -1.0000E-04
092238.50c -8.5942E-01 008016.50c -1.1853E-01
c G7211G007DL6.dat (3)

m49 092234.50c -1.2000E-04 092235.50c -1.4990E-02 092236.50c -7.0000E-05
092238.50c -8.6631E-01 008016.50c -1.1852E-01
c G7211G007DL6.dat (4)

m50 092234.50c -8.0000E-05 092235.50c -1.0580E-02 092236.50c -5.0000E-05
092238.50c -8.7078E-01 008016.50c -1.1851E-01
c G7211G007DL6.dat (5)

m51 092234.50c -1.8000E-04 092235.50c -2.1120E-02 092236.50c -1.0000E-04
092238.50c -8.3364E-01 008016.50c -1.1894E-01 064152.50c -5.0000E-05
064154.50c -5.6000E-04 064155.50c -3.8000E-03 064156.50c -5.2800E-03
064157.50c -4.0600E-03 064158.50c -6.4900E-03 064160.50c -5.7900E-03
c G7211G007DL6.dat (6)

m52 092234.50c -1.8000E-04 092235.50c -2.1660E-02 092236.50c -1.0000E-04
092238.50c -8.5512E-01 008016.50c -1.1859E-01 064152.50c -1.0000E-05
064154.50c -9.0000E-05 064155.50c -6.3000E-04 064156.50c -8.8000E-04
064157.50c -6.8000E-04 064158.50c -1.0800E-03 064160.50c -9.6000E-04
c G7212G003DL7.dat (1)

m53 092234.50c -1.8000E-04 092235.50c -2.1770E-02 092236.50c -1.0000E-04
092238.50c -8.5942E-01 008016.50c -1.1853E-01
c G7212G003DL7.dat (2)

m54 092234.50c -1.8000E-04 092235.50c -2.1770E-02 092236.50c -1.0000E-04
092238.50c -8.5942E-01 008016.50c -1.1853E-01
c G7212G003DL7.dat (3)

m55 092234.50c -1.8000E-04 092235.50c -2.1770E-02 092236.50c -1.0000E-04
092238.50c -8.5942E-01 008016.50c -1.1853E-01
c G7212G003DL7.dat (4)

m56 092234.50c -1.2000E-04 092235.50c -1.4990E-02 092236.50c -7.0000E-05
092238.50c -8.6631E-01 008016.50c -1.1852E-01
c G7212G003DL7.dat (5)

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 74 of 76

m57 092234.50c -1.2000E-04 092235.50c -1.4990E-02 092236.50c -7.0000E-05
092238.50c -8.6631E-01 008016.50c -1.1852E-01
c G7212G003DL7.dat (6)

m58 092234.50c -8.0000E-05 092235.50c -1.0580E-02 092236.50c -5.0000E-05
092238.50c -8.7078E-01 008016.50c -1.1851E-01
c G7212G003DL7.dat (7)

m59 092234.50c -1.8000E-04 092235.50c -2.1450E-02 092236.50c -1.0000E-04
092238.50c -8.4653E-01 008016.50c -1.1873E-01 064152.50c -3.0000E-05
064154.50c -2.8000E-04 064155.50c -1.9000E-03 064156.50c -2.6400E-03
064157.50c -2.0300E-03 064158.50c -3.2500E-03 064160.50c -2.8900E-03
c G7212G006DL8.dat (1)

m60 092234.50c -1.8000E-04 092235.50c -2.1770E-02 092236.50c -1.0000E-04
092238.50c -8.5942E-01 008016.50c -1.1853E-01
c G7212G006DL8.dat (2)

m61 092234.50c -1.8000E-04 092235.50c -2.1770E-02 092236.50c -1.0000E-04
092238.50c -8.5942E-01 008016.50c -1.1853E-01
c G7212G006DL8.dat (3)

m62 092234.50c -1.8000E-04 092235.50c -2.1770E-02 092236.50c -1.0000E-04
092238.50c -8.5942E-01 008016.50c -1.1853E-01
c G7212G006DL8.dat (4)

m63 092234.50c -1.2000E-04 092235.50c -1.4990E-02 092236.50c -7.0000E-05
092238.50c -8.6631E-01 008016.50c -1.1852E-01
c G7212G006DL8.dat (5)

m64 092234.50c -1.2000E-04 092235.50c -1.4990E-02 092236.50c -7.0000E-05
092238.50c -8.6631E-01 008016.50c -1.1852E-01
c G7212G006DL8.dat (6)

m65 092234.50c -8.0000E-05 092235.50c -1.0580E-02 092236.50c -5.0000E-05
092238.50c -8.7078E-01 008016.50c -1.1851E-01
c G7212G006DL8.dat (7)

m66 092234.50c -1.8000E-04 092235.50c -2.1120E-02 092236.50c -1.0000E-04
092238.50c -8.3364E-01 008016.50c -1.1894E-01 064152.50c -5.0000E-05
064154.50c -5.6000E-04 064155.50c -3.8000E-03 064156.50c -5.2800E-03
064157.50c -4.0600E-03 064158.50c -6.4900E-03 064160.50c -5.7900E-03
c G7211G003DL9.dat (1)

m67 092234.50c -1.8000E-04 092235.50c -2.1770E-02 092236.50c -1.0000E-04
092238.50c -8.5942E-01 008016.50c -1.1853E-01
c G7211G003DL9.dat (2)

m68 092234.50c -1.8000E-04 092235.50c -2.1770E-02 092236.50c -1.0000E-04
092238.50c -8.5942E-01 008016.50c -1.1853E-01
c G7211G003DL9.dat (3)

m69 092234.50c -1.2000E-04 092235.50c -1.4990E-02 092236.50c -7.0000E-05
092238.50c -8.6631E-01 008016.50c -1.1852E-01
c G7211G003DL9.dat (4)

m70 092234.50c -8.0000E-05 092235.50c -1.0580E-02 092236.50c -5.0000E-05
092238.50c -8.7078E-01 008016.50c -1.1851E-01
c G7211G003DL9.dat (5)

m71 092234.50c -1.8000E-04 092235.50c -2.1450E-02 092236.50c -1.0000E-04
092238.50c -8.4653E-01 008016.50c -1.1873E-01 064152.50c -3.0000E-05
064154.50c -2.8000E-04 064155.50c -1.9000E-03 064156.50c -2.6400E-03
064157.50c -2.0300E-03 064158.50c -3.2500E-03 064160.50c -2.8900E-03
c G7212G006DL10.dat (1)

m72 092234.50c -1.8000E-04 092235.50c -2.1770E-02 092236.50c -1.0000E-04
092238.50c -8.5942E-01 008016.50c -1.1853E-01
c G7212G006DL10.dat (2)

Title: Index for Computer Files Supporting this Analysis**Document Identifier:** B00000000-01717-0210-00010 REV 01 Attachment XIII Page 75 of 76

```

m73 092234.50c -1.8000E-04 092235.50c -2.1770E-02 092236.50c -1.0000E-04
    092238.50c -8.5942E-01 008016.50c -1.1853E-01
c   G7212G006DL10.dat (3)
m74 092234.50c -1.8000E-04 092235.50c -2.1770E-02 092236.50c -1.0000E-04
    092238.50c -8.5942E-01 008016.50c -1.1853E-01
c   G7212G006DL10.dat (4)
m75 092234.50c -1.2000E-04 092235.50c -1.4990E-02 092236.50c -7.0000E-05
    092238.50c -8.6631E-01 008016.50c -1.1852E-01
c   G7212G006DL10.dat (5)
m76 092234.50c -1.2000E-04 092235.50c -1.4990E-02 092236.50c -7.0000E-05
    092238.50c -8.6631E-01 008016.50c -1.1852E-01
c   G7212G006DL10.dat (6)
m77 092234.50c -8.0000E-05 092235.50c -1.0580E-02 092236.50c -5.0000E-05
    092238.50c -8.7078E-01 008016.50c -1.1851E-01
c   G7212G006DL10.dat (7)
m78 092234.50c -1.8000E-04 092235.50c -2.1120E-02 092236.50c -1.0000E-04
    092238.50c -8.3364E-01 008016.50c -1.1894E-01 064152.50c -5.0000E-05
    064154.50c -5.6000E-04 064155.50c -3.8000E-03 064156.50c -5.2800E-03
    064157.50c -4.0600E-03 064158.50c -6.4900E-03 064160.50c -5.7900E-03
c   SG7D1
m79 001001.50c -9.3090E+00 006000.50c -1.0000E-03 008016.50c -7.3901E+01
    013027.50c -2.4000E-02 014000.50c -8.0000E-03 022000.50c -6.7000E-02
    024050.60c -1.9474E-02 024052.60c -3.9005E-01 024053.60c -4.5076E-02
    024054.60c -1.1408E-02 025055.50c -1.3000E-02 026054.60c -1.5312E-02
    026056.60c -2.4904E-01 026057.60c -5.8563E-03 026058.60c -7.8590E-04
    028058.60c -1.2667E+00 028060.60c -5.0467E-01 028061.60c -2.2300E-02
    028062.60c -7.2196E-02 028064.60c -1.9095E-02 040000.60c -1.3857E+01
    050000.35c -1.9800E-01
mt79 lwtr.01
c   Control Cards
c
tmp1 2.9043E-08 956r
kcode 10000 1.000 10 310
print
prdmp 310 5 1 2
sdef  erg=d1 pos=d2 axs= 0 0 1 rad=d3 ext=d4
spl   -3
si2   L -1.5240E+02 3.0480E+01 1.8288E+02
      -1.2192E+02 3.0480E+01 1.8288E+02
      -9.1440E+01 3.0480E+01 1.8288E+02
      -6.0960E+01 3.0480E+01 1.8288E+02
      -3.0480E+01 3.0480E+01 1.8288E+02
      -1.8288E+02 6.0960E+01 1.8288E+02
      -1.5240E+02 6.0960E+01 1.8288E+02
      -1.2192E+02 6.0960E+01 1.8288E+02
      -9.1440E+01 6.0960E+01 1.8288E+02
      -6.0960E+01 6.0960E+01 1.8288E+02
      -3.0480E+01 6.0960E+01 1.8288E+02
      -1.8288E+02 9.1440E+01 1.8288E+02
      -1.5240E+02 9.1440E+01 1.8288E+02
      -1.2192E+02 9.1440E+01 1.8288E+02
      -9.1440E+01 9.1440E+01 1.8288E+02
      -6.0960E+01 9.1440E+01 1.8288E+02
      -3.0480E+01 9.1440E+01 1.8288E+02

```

Title: Index for Computer Files Supporting this Analysis

Document Identifier: B00000000-01717-0210-00010 REV 01 Attachment XIII Page 76 of 76

```
-2.1336E+02 1.2192E+02 1.8288E+02
-1.8288E+02 1.2192E+02 1.8288E+02
-1.5240E+02 1.2192E+02 1.8288E+02
-1.2192E+02 1.2192E+02 1.8288E+02
-9.1440E+01 1.2192E+02 1.8288E+02
-6.0960E+01 1.2192E+02 1.8288E+02
-3.0480E+01 1.2192E+02 1.8288E+02
-2.1336E+02 1.5240E+02 1.8288E+02
-1.8288E+02 1.5240E+02 1.8288E+02
-1.5240E+02 1.5240E+02 1.8288E+02
-1.2192E+02 1.5240E+02 1.8288E+02
-9.1440E+01 1.5240E+02 1.8288E+02
-6.0960E+01 1.5240E+02 1.8288E+02
-3.0480E+01 1.5240E+02 1.8288E+02
-2.1336E+02 1.8288E+02 1.8288E+02
-1.8288E+02 1.8288E+02 1.8288E+02
-1.5240E+02 1.8288E+02 1.8288E+02
-1.2192E+02 1.8288E+02 1.8288E+02
-9.1440E+01 1.8288E+02 1.8288E+02
-6.0960E+01 1.8288E+02 1.8288E+02
-3.0480E+01 1.8288E+02 1.8288E+02
-2.1336E+02 2.1336E+02 1.8288E+02
-1.8288E+02 2.1336E+02 1.8288E+02
-1.5240E+02 2.1336E+02 1.8288E+02
-1.2192E+02 2.1336E+02 1.8288E+02
-9.1440E+01 2.1336E+02 1.8288E+02
-6.0960E+01 2.1336E+02 1.8288E+02
-3.0480E+01 2.1336E+02 1.8288E+02
sp2 1 1 1 1 1 1 1 1 1 1
    1 1 1 1 1 1 1 1 1 1
    1 1 1 1 1 1 1 1 1 1
    1 1 1 1 1 1 1 1 1 1
    1 1 1 1 1
si3 3.0470E+01
si4 1.8288E+02
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 1 of 241

CONTENTS

	Page
1. Introduction	3
2. Routines for BLINK	4
2.1. Main Routine	4
2.2. Service Routines	21
2.3. Input Routines	57
2.4. Input Editing Routines	72
2.5. Deck Generation Routines	90

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV **Page 2 of 241**

TABLES

	Page
1-1 Location of Listings for Each Functional Block	3

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 3 of 241

1. Introduction

This attachment contains listings of the FORTRAN routines and C functions that comprise version 0 of the BLINK software routine. The listings are divided by functional block as in the description of the software routine given in Attachment VI. Section 2 of this attachment provides the listing of the code consistent with the functional blocks of Attachment VI and Table 1-1 below.

Table 1-1 Location of Listings for Each Functional Block

Section	Contents
2.1	Main Function
2.2	Service Routines
2.3	Input Routines
2.4	Input Editing Routines
2.5	Deck Generation Routines

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 4 of 241

2. Routines for BLINK

2.1. Main Routine

```
/* -----  
- - Load Library Header Files - - - - -  
----- */  
#include <stdio.h>  
#include <string.h>  
#include <time.h>  
#include <malloc.h>  
#include <errno.h>  
  
/* -----  
- - Global Type Definitions - - - - -  
----- */  
typedef char ascii_string[133];  
typedef struct ascii_record(  
    struct ascii_record *last;  
    ascii_string line;  
    struct ascii_record *next;  
    ) a_record;  
typedef struct s_material(  
    struct s_material *last;  
    int atomic_number;  
    int mass_number;  
    float weight_percentage;  
    char library_suffix[5];  
    struct s_material *next;  
    ) ll_material;  
  
typedef struct u_list(  
    struct u_list *last;  
    int index;  
    ascii_string label;  
    struct u_list *next;  
    ) usage_list;  
  
typedef struct su_list(  
    struct su_list *last;  
    int index;  
    ascii_string label;  
    ascii_string value;  
    char mnemonic[4];  
    ascii_string equivalent_label;  
    struct su_list *next;  
    ) surface_usage_list;  
  
typedef struct fuel_geometry(  
    ascii_string gds_name;  
    int latdim;  
    int nwr;
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 5 of 241

```

        float cthick;
        float asin;
        float wgap;
        float ngap;
        float cradius;
        float fsrd;
        float cfsrd;
        float rpitch;
        float cod;
        float cld;
        float pod;
        char frcmat[6];
        char fcmat[6];
    } fg_list;

typedef struct all{
    struct all *last;
    int basis_lattice_material_index;
    int lattice_material_index;
    struct all *next;
} augmented_lattice_list;

/* - - - - -
- - Function Prototypes - - - - -
- - - - -
*/
void abort();
void header();
void lines();
void echo(char [],char []);
int *memory_integer(int,int,int *);
float *memory_float(int,int,float *);
ascii_string *memory_ascii_string(int,int,ascii_string *);
void readin(char[],int *,char[],int *,int *,int *,float *,int *
,int *,int *,int *,float *,float *,float *,float *,float *,float *
,float *,float *,float *,char *,char *,char [],char *,char *
,char *,char *,char *,char *,int *,char[],char[],float *
,float *,float *,float *,float *);
void redmaps(int *,int *,int *,int *,int *,int *,int *,int *,int *
,int *,int *,int *,int *,int *,int *,int *,int *,int *, int *
,int *);
void redlats(int *, int *, int*, int *);
void ldlv(int *,int *,int *,int *);
void ldlvr(int *,int *,int *,float *);
void nladd(int, ascii_string *);
void ftclose(int *);
void rblade(int *,char[],int *,int *,float *,float *,float *,float *
,float *,float *,float *,float *,int *,float *,float *,char[],char[]
,char[],char[]);
void rlattice(int *,char[],int *,float *,float *,float *,float *
,float *,float *,float *,float *,float *, float *,float*,char[]
,char[],int *,int *);
fg_list *memory_fg_list(int,int,fg_list *);
void lodct(int *,int *,int *,int *,int *,int *,int *,int *,int *

```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 6 of 241

```
,int *);
void editin(char[],char [],char [],int,int,float,int,int,int,int,int
,int,int *,int *,int *, int *,int *,int,int,ascii_string *
,ascii_string *,char [],int,char[],float,float,float,char[],char[]);
void coredb_edt(int,int,int,int,float,float,float,float,float,float
,float,float,float,int *,char [],char [],char [],char []
,char [],float,float);
void bladedb_edt(int,float,float,float,float,float,float,float,float
,int,float,float,char[],char[],char[],char[]);
void fgds_edt(int,char[],ascii_string *,fg_list *);
void edit_ct(int,int *p);
int clattice(int *, int *,int *,int *,int *);
void strngr(int *,int *,int *,ascii_string *);
a_record *load_core_mtls(char[],int);
a_record *memory_ascii_record(int,a_record *);
usage_list *memory_usage_list(int,usage_list *);
surface_usage_list *memory_surface_usage_list(int
,surface_usage_list *);
void vessel_generation(float,float,float,float,float,float,float,float
,float,FILE *,FILE *,FILE *,int *,int *,int *,a_record *,char[]
,char[],char[],char[],char[],int,surface_usage_list *
,usage_list **,float,float);
void ccmgen(int *,int *,int *,int *,int *,int *,int *,int *
,int *,int *,int *,int *,int *,int *,int *);
void build_control_blade(int,float,float,float,float,float,float
,float,float,int,float,float,char[],char[],char[],char[],usage_list *
,surface_usage_list *,a_record *,FILE *,FILE *,FILE *,int *,int *
,float,int *,int *,int *,int *);
void generate_deck(FILE *,FILE *,FILE *,FILE *);
void copy_ascii_file(FILE *,FILE *);
void echo_MCNP_deck(FILE *);
void discard_scratch_file(char []);
void memsum();
int mchar(int *,char []);
ll_material *load_fuel_material(int *,char[],char[],int,int **,float **);
void generate_lattice_model(int *,int *,FILE *,FILE *,FILE *
,surface_usage_list *,usage_list *,int *,fg_list *,ascii_string *
,int,char[],float,float,int *,a_record *,int *,int *,char[]);
void build_assemblies(int *,int *,FILE *,FILE *,surface_usage_list *
,float *,int **,int,int *,int,int *,int *);
usage_list *load_usage_list(char[],int,usage_list *);
void build_control_cells(int *,int *,int,int,int *,int *,int,FILE *,FILE *
,float,surface_usage_list *,float,int,float,float,char[],usage_list *
,a_record *,int *,int *,FILE *,float,float,float,int,int *,int *);
void core_lattice_generation(int *,int *,float,float,surface_usage_list *
,usage_list *,int *,int *,int,int,FILE *,FILE *,int,int *,float);
void source_specification(int,float,int,int,FILE *,int,int,int *,int *
,float,float,int,int,int,int,float);
void edit_universes(int,int,int *,int *,int,int,int,int,int,int *,int *
,int *);
void edit_surfaces(surface_usage_list *);
void editmaterials(usage_list *);
void edit_spacer(int,int *,float *,ascii_string *);
void spacer_location(int,int,float,int *,float *,int *);
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 7 of 241

```

augmented_lattice_list *memory_lattice_list(int
, augmented_lattice_list *);
void augment_lattice_list(int *, int *, int *, int, int, int, int, int, int *
, int *, augmented_lattice_list *, int *);
ll_material *material_match(a_record *, char[], float *, int *);
void rollup_llm(ll_material *);

/* - - - - -
- - Global Variables - - - - -
- - - - -
- -
- - Integer Variables
- -     nline - number of lines on current output page
- -     version - version number of code
- -     storage_i - current integer storage requested (bytes)
- -     storage_r - current real storage requested (bytes)
- -     storage_c - current character storage requested (bytes)
- -     storage_it - maximum integer storage requested (bytes)
- -     storage_rt - maximum real storage requested (bytes)
- -     storage_ct - maximum character storage requested (bytes)
- -     cdate - date of code execution
- -     crtime - time of code execution
- -     pid - process identifier for code execution
*/
short nline = 0, version = 0;
int storage_i = 0, storage_r = 0, storage_it = 0, storage_rt = 0,
    storage_c = 0, storage_ct = 0;
int long pid;
/* - Character Variables */
char modification_level = '-';
char codenm[8] = "BLINK";
char cdate[9] = {NULL};
char crtime[9] = {NULL};
ascii_string case_title;
/* - FILE Pointers */
FILE *nout;

void main(int argc, char *argv[]) {
/* - - - - -
- - * B W R L I N K * Creates MCNP Input Decks for Modeling of
- -     Commercial Reactor Critical "Experiments"
- -     (CRC) with Fuel Constituent Number Densities
- -     from SAS2H Analyses
- - - - -
- - Command Line Argument(s):
- -     prefix - Prefix for Names of Input and Output Files
- - - - -
- - Variable Declarations - - - - -
- - Integer Variable(s)
- -     nin - FORTRAN logical unit number for input file
- -     lucgeom - FORTRAN logical unit number for core geometry dataset
- -     file
- -     length_fin - length of input file name

```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 8 of 241

```
-- naxial - number of axial nodes in fuel assemblies
-- nrow - number of rows in core map for the whole core
-- ncol - number of columns in core map for the whole core
-- nrowp - number of rows in core map in problem
-- ncolp - number of columns in the core map in the problem
-- nrowb - number of control blade location rows for the whole
-- core
-- ncolb - number of control blade location columns for the
-- whole core
-- nrowbp - number of control blade rows in the problem
-- ncolbp - number of control blade columns in the problem
-- nbundlg - number of unique fuel assemblies geometrical indices
-- nbundlm - number of unique fuel assemblies material indices
-- nlatticg - number of unique fuel lattice geometrical indices
-- nlatticm - number of unique fuel lattice material indices
-- naxpl - naxial+1
-- valid - map of valid fuel locations in core (dynamic)
-- gmap - map of fuel assembly geometrical indices (dynamic)
-- mmap - map of fuel assembly geometrical indices (dynamic)
-- lgvect - vector for lattice geometrical assignment to fuel
-- lattices
-- lmvect - vector for lattice material assignment to fuel
-- lattices
-- validb - map of valid control blade locations in core (dynamic)
-- bladep - map of control blade axial positions (dynamic)
-- incore_loc - map of locations of incore instrumentation locations
-- (dynamic)
-- lt - scratch integer array used for input processing
-- (dynamic)
-- nsrck - nominal source size per cycle in MCNP
-- ikz - number of cycles to be skipped before beginning tally
-- accumulation in MCNP
-- kct - number of cycles to be performed in MCNP
-- ncell - counter for number of cells generated for non-trans-
-- lated cells (this begins @ 2001 to give headroom
-- for indexing of cells that must be translated)
-- nsurface - counter for number of surfaces generated
-- nmaterial - counter for number of materials generated
-- core_f - fraction of core in problem
-- (1 - full, 2 - half, 4 - quarter)
-- nuniverse - number of universes in problem
-- nrowcc - number of rows in control cell map
-- ncolcc - number of columns in control cell map
-- ccmmap - pointer to map of unique control cells
-- ntube - number of absorber tubes in control blade
-- ncs - number of central stiffeners in control blade wing
-- ptr_correspondence_table
-- - pointer to table containing array giving correspondence
-- between fuel geometry indices and fuel material indices
-- ptr_ufl - pointer to vector of universe indices corresponding to
-- the lattice models created for each unique lattice type
-- ptr_ufa - pointer to vector of universe indices corresponding to
-- the fuel assembly models created for each unique
-- fuel assembly type
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 9 of 241

```

- -      ucb - universe of control blade
- -      fau_fill - pointer to array containing fuel assembly assignments
- -                to control cells
- -      ndm - periodicity for dumps to TPE file
- -      mct - flag for creating MCTAL file
- -                (0 - no file, <> 0 - create file)
- -      ndmp - maximum number of dumps
- -      ptr_n_spacer
- -                - pointer to vector containing number of spacers for
- -                each fuel assembly geometrical type
- -      ptr_spacer_node
- -                - pointer to array containing the axial node locations
- -                for each fuel assembly geometrical type
- -      total_spacer_locations
- -                - total number of spacer locations in geometrically
- -                unique fuel assemblies
- -      nlatticm_ref
- -                - number of lattice material indices before addition of
- -                indices for spacer grid treatment
- -      ncell_tr - cell indices for translated cells
*/
int nin = 5, lucgeom = 1, nsrck = 10000, ikz = 10, kct = 310;
int ndm = 5, mct = 1, ndmp = 2;
int ncell = 2000, nsurface = 0, nmaterial = 0, core_f = 4;
int nuniverse = 0, ncell_tr = 0;
int ntube = 0, ncs = 0, ucb, nlatticm_ref;
int nrowcc, ncolcc, total_spacer_locations;
int length_fin = 132;
int naxial, nrow, ncol, nrowp, ncolp, nrowb, ncolb, nrowbp, ncolbp
, nbundlg, nbundlm, nbundlg, nlatticg, nlatticm, naxpl;
int *valid, *gmap, *mmap, *lgvect, *lmvect, *validb, *bladepl, *incore_loc
, *lt;
int *ccmap, *ptr_n_spacer, *ptr_spacer_node;
int *ptr_correspondence_table, *ptr_ufl, *ptr_ufa, *fau_fill;
/* - Float Variable(s)
- -      afl - active fuel length
- -      apitch - fuel assembly pitch
- -      sod - shroud outer radius
- -      sthick - shroud thickness
- -      vod - pressure vessel outer radius
- -      vthick - pressure vessle thickness
- -      tutpr - Top of Upper Tie Plate Region
- -      tcgr - Top of Core Grid Region
- -      bltpr - Bottom of Lower Tie Plate Region
- -      bcpr - Bottom of Core Plate Region
- -      rkk - initial guess for eigenvalue in MCNP
- -      bypass_density - bypass water density
- -      inchannel_density - in-channel water density
- -      blade_window_offset
- -                - sizing of windows for control blades
- -                and fuel assemblies in control cell
- -                template
- -      cbspan - control blade span
- -      atid - blade absorber tube inner diameter (cm)

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 10 of 241

```

- -         atod - blade absorber tube outer diameter (cm)
- -         cbthick - blade wing thickness (cm)
- -         trspan - blade tie rod span (cm)
- -         trthick - blade tie rod thickness (cm)
- -         cblength - blade active absorber length (cm)
- -         wsthick - blade wing thickness (cm)
- -         csoff - blade central stiffener offset (cm)
- -         cswidth - blade central stiffener width (cm)
- -         dtod - incore guide tube outer diameter (cm)
- -         dtid - incore guide tube inner diameter (cm)
- -         ptr_nab - axial nodal boundary coordinates
- -         cb_stroke - control blade stroke (cm)
- -         rho - input in-channel density value (g/cc)
- -         rhobyp - input bypass density value (g/cc)
- -         tempk - input problem temperature (Kelvins)
- - ptr_spacer_location
- -             - pointer to array containing spacer
- -             axial location (w/r/t Bottom of Active
- -             Fuel)
*/
float afl, apitch, sod, sthick, vod, vthick, tutpr, tcgr, bltpr
, bcpr, dtod, dtid, rho = 0.0, rhobyp = 0.0, tempk = 293.15;
float cbspan = 0.0, atid = 0.0, atod = 0.0, cbthick = 0.0
, trspan = 0.0, trthick = 0.0, cblength = 0.0, wsthick = 0.0
, csoff = 0.0, cswidth = 0.0, cb_stroke = 365.76;
float rkk = 1.0, bypass_density = 1.0, inchannel_density = 1.0;
float blade_window_offset = 0.40;
float *ptr_nab, *ptr_spacer_location;
/* - Character Variables
- -         fin - name for input file
- -         fout - name for output file
- -         inname - FORTRAN NAMELIST on input file
- -         prefix - prefix for names of input and output files
- -         core_db - name of core geometry database file
- -         lprefix - prefix for lattice geometry database
- -         fprefix - prefix for fuel material intermediate database
- -         lgdsnam - pointer to names of datasets for unique geometrical
- -                 lattice types
- -         lmdsnam - pointer to names of datasets for unique geometrical
- -                 lattice types
- -         cell_file - name for scratch file for cell definitions
- -         surface_file - name for scratch file for surface definitions
- -         material_file - name for scratch file for material definitions
- -         MCNP_file - name for MCNP input file
- -         core_mtls - name of core materials database file
- -         mvessel - vessel material identifier
- -         mshroud - core shroud material identifier
- -         mtg - core top guide region material identifier
- -         mcp - core plate region material identifier
- -         mutp - upper tie plate region material identifier
- -         mltp - lower tie plate region material identifier
- -         migt - incore guide tube material identifier
- -         blade_db - dataset containing blade geometry for problem
- -         cbpmat - identifier for blade poison material

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 11 of 241

```

- -          atmat - identifier for absorber tube poison material
- -          cbsmat - identifier for blade sheath material
- -          cbtrmat - identifier for blade tie rod material
- - ptr_spacer_material
- -          - pointer to vector containing alphanumeric
- -          identifiers for each fuel assembly spacer
- -          material mnemonic
*/
char fin[133], fout[133];
char inname[8] = "LINKIN";
char prefix[133], core_db[133], lprefix[133], fprefix[133]
, cell_file[133], surface_file[133], material_file[133]
, MCNP_file[133], core_mtls[133], blade_db[133];
char mshroud[6], mvessel[6], mtg[6], mcp[6], mutp[6], mltp[6]
, migt[6];
char cbpmat[6], atmat[6], cbsmat[6], cbtrmat[6];
ascii_string *lgdsnam, *lmdsnam, *ptr_spacer_material;
/* - file pointers
- -   lmcnp - file for MCNP Input Deck
- -   lu8 - scratch file for MCNP card images for cell definitions
- -   lu9 - scratch file for MCNP card images for surface definitions
- -   lul0 - scratch file for MCNP card images for material defini-
- -         tions
*/
FILE *lmcnp, *lu8, *lu9, *lul0;
/* - Structured Variable(s)
- -   ptr_core_mtls - pointer to first record of linked list for
contents
- -                   of core materials dataset
- -   ptr_surface_usage - pointer to first record of linked list for
- -                   surface labels and indices
- -   ptr_material_usage - pointer to first record of linked list for
- -                   material labels and indices
- -   ptr_lg_ds - pointer to lattice geometry datasets
- -   additional_lattices - material lattice indices for lattices added for
- -                   spacer grid treatment
*/
a_record *ptr_core_mtls;
usage_list *ptr_material_usage;
surface_usage_list *ptr_surface_usage;
fg_list *ptr_lg_ds;
augmented_lattice_list *additional_lattices;
/* - initialize global variables not explicitly initialized elsewhere */
/* - verify presence of prefix for input and output file names, if found
- - construct file names - - - - - */
if(argc < 2) {
    puts("No file prefix available from command line -- abort\n");
    return;}
else {
    strcpy(prefix,argv[1]);
    strcpy(fin,prefix);
    strcat(fin, ".inp");
    strcpy(fout,prefix);
    strcat(fout, ".out");

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 12 of 241

```

    }
/* - open input and output files for processing */
{ FILE *inp;
  inp = fopen(fin,"r");
  if(!inp) {
    puts("Input file cannot be opened\n");
    return;}
  fclose(inp);
}
nout = fopen(fout,"w");
header();
/* - echo input stream to output stream */
echo(inname,fin);
/* - Process Input Directives - - - - - */
/* - Read Namelists */
readin(fin,&length_fin,core_db,&nin,&lucgeom,&naxial,&afl,&nrow
, &ncol,&nrowp,&ncolp,&apitch,&sod,&sthick,&vod,&vthick
, &tutpr,&tcgr,&bltpr,&bcpr,lprefix,fprefix,core_mtls,mvessel
, mshroud,mtg,mcp,mutp,mltp,&core_f,blade_db,migt,&dtod,&dtid
, &rho,&rhobyp,&tempk);
/* - Set interal variables for in-channel and bypass water densities */
if(rho != 0.0) inchannel_density = rho;
if(rhobyp != 0)
  bypass_density = rhobyp;
else
  if(rho != 0)
    bypass_density = rho;
/* - Allocate Memory for Arrays */
nrow%2?(nrowb = nrow/2+1):(nrowb = nrow/2);
ncol%2?(ncolb = ncol/2+1):(ncolb = ncol/2);
nrowp%2?(nrowbp = nrowp/2+1):(nrowbp = nrowp/2);
ncolp%2?(ncolbp = ncolp/2+1):(ncolbp = ncolp/2);
valid = memory_integer(1, (nrow*ncol), valid);
gmap = memory_integer(1, (nrowp*ncolp), gmap);
mmap = memory_integer(1, (nrowp*ncolp), mmap);
validb = memory_integer(1, (nrowb*ncolb), validb);
bladeb = memory_integer(1, (nrowbp*ncolbp), bladeb);
incore_loc = memory_integer(1, (nrowb*ncolb), incore_loc);
lt = memory_integer(1, (nrowp*ncolp), lt);
/* - Read Arrays from Input and Core Geometry File */
{ int maxb;
  maxb = nrowp*ncolp;
  redmaps(&nin,&lucgeom,&nrow,&ncol,&nrowp,&ncolp,valid,gmap,validb
, bladeb,&nbundlg,&maxb,lt,&nrowb,&ncolb,&nrowbp,&ncolbp
, incore_loc,mmap,&nbundlm);
}
/* - Return Scratch Memory and Arrays that are No Longer Needed - - */
valid = memory_integer(-1, (nrow*ncol), valid);
validb = memory_integer(-1, (nrowb*ncolb), validb);
lt = memory_integer(-1, (nrowp*ncolp), lt);
/* - Read Lattice Assignments to Fuel Types - - - - - */
naxpl = naxial+1;
lgvect = memory_integer(1, (naxpl*nbundlg), lgvect);
ldlv(&nbundlg,&naxpl,&nin,lgvect);

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 13 of 241

```

    lmvect = memory_integer(1, (naxpl*nbundlm), lmvect);
    ldlv(&nbundlm, &naxpl, &nin, lmvect);
/* - Determine the Number of Unique Lattices */
{ int max;
  max = naxpl*nbundlg;
  lt = memory_integer(1, max, lt);
  nlatticg = (int) clattice(&naxpl, &nbundlg, &max, lgvect, lt);
  lt = memory_integer(-1, max, lt);
  max = naxpl*nbundlm;
  lt = memory_integer(1, max, lt);
  nlatticm = (int) clattice(&naxpl, &nbundlm, &max, lmvect, lt);
  lt = memory_integer(-1, max, lt);
}
/* - Read Names of Datasets for Unique Lattice Geometrical and - - -
   - - Material Indices */
/* - Dataset Names are Limited to 132 Characters */
  lgdsnam = memory_ascii_string(1, (nlatticg), lgdsnam);
  lmdsnam = memory_ascii_string(1, (nlatticm), lmdsnam);
/* - Dataset Names for Fuel Geometries */
{ int len = 133;
  strngr(&nin, &len, &nlatticg, lgdsnam);
/* - Dataset Names for Fuel Material Intermediate Datasets */
  strngr(&nin, &len, &nlatticm, lmdsnam);
}
/* - Spacer Information - - - - - */
{ short int i;
  int one = 1;
/* - Read Number of Spacers for Each Fuel Assembly Geometrical
   - - Type */
  ptr_n_spacer = memory_integer(1, nbundlg, ptr_n_spacer);
  ldlv(&one, &nbundlg, &nin, ptr_n_spacer);
/* - Compute Total Number of Spacers in Problem */
  { short int i;
    int *ptr = ptr_n_spacer;
    for(i = 1; i <= nbundlg; i++){
      total_spacer_locations += *ptr;
      ptr++;}
  }
/* - Read in Spacer Locations and Material Mnemonic for each Fuel
   - - Assembly Geometrical Type */
  ptr_spacer_location = memory_float(1, total_spacer_locations
    , ptr_spacer_location);
  ptr_spacer_material = memory_ascii_string(1, nbundlg
    , ptr_spacer_material);
  ldldr(&one, &total_spacer_locations, &nin, ptr_spacer_location);
  { short int i;
    int len = 133;
    strngr(&nin, &len, &nbundlg, ptr_spacer_material);
  }
}
/* - Load Contents of Fuel Geometry Datasets - - - - - */
/* - Allocate Memory */
ptr_lg_ds = memory_fg_list(1, nlatticg, ptr_lg_ds);
{ short int i;

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 14 of 241

```

int latdim, nwr;
float cthick, asin, wgap, ngap, cradius, fsrd, cfsrd, rpitch
, cod, cld, pod;
char frpmat[7], fpmat[7];
ascii_string *as = lgdsnam, fn;
fg_list *p;
p = ptr_lg_ds;
{ int len=132, length;
  length = mchar(&len, lprefix);
  lprefix[length] = '\0';
  length = mchar(&len, fprefix);
  fprefix[length] = '\0';
}
for( i = 1; i <= nlatticg; i++, p++, as++){
  { int len=132, length;
    ascii_string holder;
    strncpy(holder, *as, len);
    length = mchar(&len, holder);
    holder[length] = '\0';
    strncpy(*as, holder, (length+1));
  }
  strcpy(fn, lprefix);
  strcat(fn, *as);
  rlattice(&length_fin, fn, &lucgeom, &cthick, &asin, &wgap, &ngap
, &cradius, &fsrd, &cfsrd, &rpitch, &cod, &cld, &pod, frpmat, fpmat
, &latdim, &nwr);
  strcpy(p->gds_name, *as);
  p->latdim = latdim; p->nwr = nwr; p->cthick = cthick;
  p->asin = asin; p->wgap = wgap; p->ngap = ngap;
  p->radius = cradius; p->fsrd = fsrd; p->cfsrd = cfsrd;
  p->rpitch = rpitch; p->cod = cod; p->cld = cld;
  p->pod = pod;
  { int len=6, length;
    length = mchar(&len, frpmat);
    frpmat[length] = '\0';
    length = mchar(&len, fpmat);
    fpmat[length] = '\0';
  }
  strncpy(p->frpmat, frpmat, 6);
  strncpy(p->fpmat, fpmat, 6);}
}
/* - Close Input File */
ftclose(&nin);
/* - Compute Nodal Locations of Spacers */
ptr_spacer_node = memory_integer(1, total_spacer_locations
, ptr_spacer_node);
spacer_location(naxial, nbundlg, afl, ptr_n_spacer
, ptr_spacer_location, ptr_spacer_node);
/* - Read Contents of Blade Geometry Dataset */
rblade(&length_fin, blade_db, &lucgeom, &ntube, &cbspan, &atid, &atod
, &cbthick, &trspan, &trthick, &wsthick, &cblength, &nCS, &CSoff, &CSwidth
, cbpmat, atmat, cbpmat, cbtrmat);
/* - Generate Correspondence Table of Fuel Geometry Indices and Fuel
- - Material Indices */

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 15 of 241

```

ptr_correspondence_table =
  memory_integer(1, (2*nlattcm), ptr_correspondence_table);
lodct(&nrowp, &ncolp, &naxpl, &nlattcg, &nlattcm, gmap, mmap
  , lgvect, lmvect, ptr_correspondence_table);
/* - Edit Input Instructions - - - - - */
editin(core_db, lprefix, fprefix, naxial, naxpl, afl, nrowp, ncolp, nrowbp
  , ncolbp, nbundlg, nbundlm, gmap, mmap, lgvect, lmvect, bladep, nlattcg
  , nlattcm, lgdsnam, lmdsnam, core_mtls, core_f, blade_db, rho, rhobyp
  , tempk, mutp, mltp);
/* - Edit Contents of Core Geometry Dataset - - - - - */
coredb_edt(nrow, ncol, nrowb, ncolb, apitch, vod, vthick, sod, sthick, tutpr
  , tcgr, bltpr, bcpr, incore_loc, mvessel, mshroud, mtg, mcp, migt, dtod, dtid);
/* - Edit Contents of Blade Geometry Dataset - - - - - */
bladedb_edt(ntube, cbspan, atid, atod, cbthick, trspan, trthick, wsthick
  , cblength, ncs, csoff, cswidth, cbpmat, atmat, cbsmat, cbtrmat);
/* - Edit Contents of Fuel Geometry Datasets - - - - - */
fgds_edt(nlattcg, lprefix, lgdsnam, ptr_lg_ds);
/* - Edit Spacer Input Information - - - - - */
edit_spacer(nbundlg, ptr_n_spacer, ptr_spacer_location
  , ptr_spacer_material);
/* - Edit Fuel Material/Geometry Correspondence Table - - - - - */
edit_ct(nlattcm, ptr_correspondence_table);
/* - - - - -
- - Add Lattice Definitions for Spacer Grid Treatment - - - - -
- - - - - */
nlattcm_ref = nlattcm;
additional_lattices = memory_lattice_list(1, additional_lattices);
augment_lattice_list(&nlattcm, lmvect, lgvect, nlattcm_ref, nlattcg
  , nbundlm, nbundlg, naxial, ptr_n_spacer, ptr_spacer_node
  , additional_lattices, ptr_correspondence_table);
/* - Regenerate Correspondence Table */
ptr_correspondence_table = memory_integer(-1, (2*nlattcm_ref)
  , ptr_correspondence_table);
ptr_correspondence_table = memory_integer(1, (2*nlattcm)
  , ptr_correspondence_table);
lodct(&nrowp, &ncolp, &naxpl, &nlattcg, &nlattcm, gmap, mmap, lgvect
  , lmvect, ptr_correspondence_table);
/* - Edit New Lattice Material Index Vector and Correspondence Table */
lines(7);
fprintf(nout, "0Lattice Material Index Vector has been Regenerated");
fprintf(nout, " to Accomodate Spacer Modeling\n");
fprintf(nout, "0Fuel Assembly Index/Lattice Index\n");
{ short int i, j;
  int *p = lmvect;
  fprintf(nout, "bundle");
  for(i = 1; i <= naxial; i++){
    fprintf(nout, "%5i", i);
    fprintf(nout, "\n\n");
  }
  for(j = 0; j < nbundlm; j++){
    fprintf(nout, "%5i ", *p);
    p++;
    for(i = 1; i < naxpl; i++){
      fprintf(nout, "%5i", *p);
      p++;
    }
  }
}

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 16 of 241

```

    }
    if(i > 1) lines(1);
    fprintf(nout, "\n");
}
}
lines(2);
fprintf(nout, "0New Correspondence Table\n\n");
edit_ct(nlatticm, ptr_correspondence_table);
/* - - - - -
- - Generate Input Deck for MCNP - - - - -
- - - - - */

strcpy(cell_file, prefix);
strcat(cell_file, ".cel");
strcpy(surface_file, prefix);
strcat(surface_file, ".sur");
strcpy(material_file, prefix);
strcat(material_file, ".mat");
/* - Open Scratch Files for Segments of MCNP Input Deck */
lu8 = fopen(cell_file, "w");
lu9 = fopen(surface_file, "w");
lu10 = fopen(material_file, "w");
/* - Load Core Materials Database into Memory */
ptr_core_mtls = load_core_mtls(core_mtls, length_fin);
/* - Set up first access to surface usage pointer */
ptr_surface_usage =
memory_surface_usage_list(1, ptr_surface_usage);
ptr_surface_usage->last = NULL;
ptr_surface_usage->index = 0;
strcpy("", ptr_surface_usage->label);
strcpy("", ptr_surface_usage->value);
strcpy("", ptr_surface_usage->mnemonic);
strcpy("", ptr_surface_usage->equivalent_label);
ptr_surface_usage->next = NULL;
/* - Write Representations for Core Structural Components */
vessel_generation(apitch, vod, vthick, sod, sthick, tutpr, tcgr, bltpr, bcpr
, lu8, lu9, lu10, &ncell, &nsurface, &nmaterial, ptr_core_mtls, mvessel
, mshroud, mtg, mcp, mutp, mltp, core_f, ptr_surface_usage
, &ptr_material_usage, bypass_density, afl);
/* - Generate Control Blade Model */
build_control_blade(ntube, cbspan, atid, atod, cbthick, trspan, trthick
, wsthick, cblength, ncs, csoff, cswidth, cbpmat, atmat, cbsmat, cbtrmat
, ptr_material_usage, ptr_surface_usage, ptr_core_mtls, lu8, lu9, lu10
, &nuniverse, &nmaterial, bypass_density, &ncell, &uch, &nsurface
, &ncell_tr);
/* - Create Fuel Assembly Models - - - - - */
/* - Create In-channel Moderator Material */
{ short int i, nloc;
char *cp, zaid[11];
ascii_string label;
usage_list *ptr_ml = ptr_material_usage;
do{
ptr_ml = ptr_ml->next;}
while(ptr_ml->next != NULL);
strcpy(label, "Inchannel Water");

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 17 of 241

```

    fprintf(lu10,"c      %s\n",label);
    nmaterial++;
    ptr_ml = load_usage_list(label,nmaterial,ptr_ml);
    strcpy(zaid,"001001.50c");
    nloc = 0;
    if(nmaterial < 1000) nloc++;
    if(nmaterial < 100) nloc++;
    if(nmaterial < 10) nloc++;
    strcpy(label,"2.0");
    { short int ip;
      for( ip = 1; ip <= nloc; ip++) fprintf(lu10," ");
      fprintf(lu10,"m%i %s %s\n",nmaterial,zaid,label);
    }
    strcpy(zaid,"008016.50c");
    strcpy(label,"1.0");
    fprintf(lu10,"      %s %s\n",zaid,label);
    { short int ip;
      for( ip = 1; ip <= (nloc-1); ip++) fprintf(lu10," ");
      fprintf(lu10,"mt%i lwtr.01\n",nmaterial);
    }
  }
ptr_ufl = memory_integer(1,nlatticm,ptr_ufl);
{ int n, ufl, nlat, index, n_entries;
  float density;
  int *p_ufl = ptr_ufl;
  ascii_string *p_sm = ptr_spacer_material;
  ll_material *ptr_mtl;
  usage_list *ptr_ml;
  augmented_lattice_list *p_addlat = additional_lattices;
  for(n = 1; n <= nlatticm_ref; n++,p_ufl++){
    generate_lattice_model(&ncell,&nmaterial,lu8,lu9,lu10
      ,ptr_surface_usage,ptr_material_usage
      ,ptr_correspondence_table,ptr_lg_ds,lmdsnam,n,fprefix
      ,inchannel_density,bypass_density,&nuniverse
      ,ptr_core_mtls,&ufl,&nsurface,"Inchannel Water");
    *p_ufl = ufl;}
/* - Add Lattices Incorporating Spacer Grids */
  for(n = (nlatticm_ref+1);n <= nlatticm;n++,p_ufl++){
/* - Determine Material Density and Add Composition to Input
- - Deck */
    { short int i;
      int len = 132, length, *lmv = lmvect, nlatm, nlatg;
      int *p = ptr_correspondence_table;
      ascii_string buffer;
/* - Find Lattice Geometry Index Corresponding to Lattice
- - Material Index */
      for(i = 1;i < n;i++) lmv++;
      nlatm = *lmv;
      for(i = 1;i < nlatm;i++) {p++; p++;}
      p++;
      nlatg = *p;
      for(i = 1;i < nlatg;i++,p_sm++);
      strncpy(buffer,*p_sm,len);
      length = mchar(&len,*p_sm);

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 18 of 241

```

        buffer[length] = '\0';
        strncpy(*p_sm,buffer,len);
    }
    ptr_mtl = material_match(ptr_core_mtls,*p_sm,&density
        ,&n_entries);
    search_usage_list(1,*p_sm,&index,ptr_material_usage);
    if(index == 0){
        nmaterial++;
        ptr_ml = ptr_material_usage;
        while(ptr_ml->next != NULL) ptr_ml = ptr_ml->next;
        index = (ptr_ml->index)+1;
        ptr_ml = load_usage_list(*p_sm,index,ptr_ml);
        add_material(lu10,&nmaterial,*p_sm,n_entries,ptr_mtl);
        { short int nb,nblank = 0;
            if(nmaterial < 100) nblank = 1;
            if(nmaterial < 10) nblank = 2;
            for(nb = 1;nb <= nblank;nb++) fprintf(lu10," ");
            fprintf(lu10,"mt%i lwtr.01\n",nmaterial);
        }
        rollup_llm(ptr_mtl);}
/* - Generate New Lattices */
    nlat = p_addlat->basis_lattice_material_index;
    generate_lattice_model(&ncell,&nmaterial,lu8,lu9,lu10
        ,ptr_surface_usage,ptr_material_usage
        ,ptr_correspondence_table,ptr_lg_ds,lmdsnam,nlat
        ,fprefix,density,bypass_density,&nuniverse
        ,ptr_core_mtls,&ufl,&nsurface,*p_sm);
    *p_ufl = ufl;
    p_addlat = p_addlat->next;
}
}
/* - Compute Nodal Axial Boundaries */
ptr_nab = memory_float(1,(naxial-1),ptr_nab);
{ short int i;
    float delta, last, *p=ptr_nab;
    delta = afl/((float) naxial);
    last = 0.0;
    for(i = 1;i <= (naxial-1);i++,p++){
        *p = last+delta;
        last = *p;
    }
}
/* - Generate Unique Lattice Definitions - - - - - */
ptr_ufa = memory_integer(1,nbundlm,ptr_ufa);
build_assemblies(&ncell,&nuniverse,lu8,lu9,ptr_surface_usage
    ,ptr_nab,&ptr_ufa,nbundlm,lmvect,naxial,ptr_ufl,&nsurface);
ptr_nab = memory_float(-1,(naxial-1),ptr_nab);
/* - Determine the Location and Number of Unique Control Cells */
{ int *ccmapw, nu_cc;
    nu_cc = nuniverse+1;
    if(core_f == 4){
        ncolcc = (ncolp/2)+1;
        nrowcc = (nrowp/2)+1;}
    if(core_f == 1){

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 19 of 241

```

        nrowp%4?(nrowcc = nrowp/2):(nrowcc = (nrowp/2)+1);
        ncolp%4?(ncolcc = ncolp/2):(ncolcc = (ncolp/2)+1);}
if(core_f == 2){
    ncolcc = (ncolp/2)+1;
    nrowp%4?(nrowcc = nrowp/2):(nrowcc = (nrowp/2)+1);}
ccmap = memory_integer(1, (ncolcc*nrowcc), ccmmap);
ccmapw = memory_integer(1, (6*ncolcc*nrowcc), ccmmapw);
ccngen(&nrowp, &ncolp, mmap, &nrowcc, &ncolcc, ccmmap, ccmmapw, &core_f
    , &ncolbp, &nrowbp, bladep, &nuniverse, incore_loc, &nrowb, &ncolb);
/* - Generate Control Cells */
fau_fill = memory_integer(1, 4*(nuniverse-nu_cc+1), fau_fill);
build_control_cells(&ncell_tr, &nuniverse, ncolcc, nrowcc, ccmmap, ccmmapw
    , (nuniverse-nu_cc+1), lu8, lu9, blade_window_offset, ptr_surface_usage
    , apitch, ucb, cspan, cb_stroke, migt, ptr_material_usage, ptr_core_mtls
    , &nmaterial, ptr_ufa, lu10, bypass_density, dtid, dtod, nu_cc, fau_fill
    , &nsurface);
/* - Genrate Control Cell Lattice */
core_lattice_generation(&ncell, &nsurface, apitch, bypass_density
    , ptr_surface_usage, ptr_material_usage, ccmmap, ncolcc, nrowcc
    , lu8, lu9, core_f, &nuniverse, afl);
edit_universes(ncolcc, nrowcc, ccmmap, fau_fill, core_f, ncolp, nrowp
    , nu_cc, nlatticm, nbundlm, ptr_ufl, ptr_ufa);
ptr_ufl = memory_integer(-1, nlatticm, ptr_ufl);
source_specification(nsrck, rkk, ikz, kct, lu10, ncolcc, nrowcc, ccmmap
    , fau_fill, apitch, afl, core_f, ndm, mct, ndmp, (ncell-2000+ncell_tr)
    , tempk);
fau_fill = memory_integer(-1, 4*(nuniverse-nu_cc+1), fau_fill);
ccmapw = memory_integer(-1, (6*ncolcc*nrowcc), ccmmapw);
}
ptr_ufa = memory_integer(-1, nbundlm, ptr_ufa);
/* - Edit Surfaces Generated */
edit_surfaces(ptr_surface_usage);
edit_materials(ptr_material_usage);
/* - Close Scratch File for Deck Segments */
fclose(lu8);
fclose(lu9);
fclose(lu10);
/* - Build MCNP Deck from Segment Files */
strcpy(MCNP_file, prefix);
strcat(MCNP_file, "_m.inp");
lmcnp = fopen(MCNP_file, "w");
/* - Open Scratch Files for Segments of MCNP Input Deck */
lu8 = fopen(cell_file, "r");
lu9 = fopen(surface_file, "r");
lu10 = fopen(material_file, "r");
generate_deck(lmcnp, lu8, lu9, lu10);
fclose(lu8); fclose(lu9); fclose(lu10);
fclose(lmcnp);
/* - Discard Scratch Files */
discard_scratch_file(cell_file);
discard_scratch_file(surface_file);
discard_scratch_file(material_file);
/* - Echo MCNP Input Deck to Output File */
lmcnp = fopen(MCNP_file, "r");

```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 20 of 241

```
    echo_MCNP_deck(lmcnp);
/* - Return Memory */
    incore_loc = memory_integer(-1, (nrowb*ncolb), incore_loc);
    bladep = memory_integer(-1, (nrowbp*ncolbp), bladep);
    gmap = memory_integer(-1, (nrowp*ncolp), gmap);
    mmap = memory_integer(-1, (nrowp*ncolp), mmap);
    lgvect = memory_integer(-1, (naxpl*nbundlg), lgvect);
    lmvect = memory_integer(-1, (naxpl*nbundlm), lmvect);
    lgdsnam = memory_ascii_string(-1, (nlatticg), lgdsnam);
    lmdsnam = memory_ascii_string(-1, (nlatticm_ref), lmdsnam);
    ptr_lg_ds = memory_fg_list(-1, nlatticg, ptr_lg_ds);
    ptr_correspondence_table =
        memory_integer(-1, (2*nlatticm), ptr_correspondence_table);
/* - end of processing - - - - - */
    memsum();
    fclose(nout);
    return;
}
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 21 of 241

2.2. Service Routines

Function abort

```
#include <stdio.h>

void abort(){
/* -----
- - * a b o r t * Handles abnormal terminations detected by the
- -               calling routine
-----
- -
- - Variable Definition(s) -----
- - FILE Variable(s)
- - nout - pointer to code output file
*/
extern FILE *nout;
/* - Write Termination Processing Message ----- */
fprintf(nout,"0*** Processing Stops ***\n");
exit(0);
return;
}
```

Function bufferpad

```
void bufferpad(char buffer[], int start, int last){
/* -----
- - * b u f f e r p a d * Adds blanks to character string
-----
- - Argument(s):
- -     buffer - string variable to which add blanks           (input)
- -     start  - starting location for adding blanks           (input)
- -     last   - ending location for adding blanks             (input)
-----
- -
- - Variable Definition(s) -----
- - Integer Variable(s)
- - i - counter
*/
short int i, begin;
/* - Character Variable(s)
- - blank - blank character */
char blank = ' ';
if(start == 0){
    strncpy(buffer,"",1);
    begin = 1;}
else begin = start;
for( i = begin; i <= last; i++)
    strncat(buffer,&blank,1);
}
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 22 of 241

Function copy_ascii_file

```

#include<stdio.h>
#include<string.h>

void copy_ascii_file(FILE *lu1,FILE *lu2){
/* - - - - -
- - * c o p y _ a s c i i _ f i l e * Copies Contents of One ASCII
- -                               File to Another
- - - - -
- - Argument(s):
- -     lu1 - pointer to target file (input)
- -     lu2 - pointer to source file (input)
- - - - -
- - Variable Definition(s) - - - - -
- - Character Variable(s)
- -     c - single character used to read from source file
- -     p - pointer to character variable
- -     buffer - string variable used to load entire lines for print
*/
    unsigned char c;
    char *p;
    char buffer[81];
/* - Copy Line by Line - - - - - */
    do
    { strcpy(buffer,"");
      p = buffer;
      do
      { c = fgetc(lu2);
        *p = c;
        p++;}
      while((c != 10) && (c != 255));
      if(c != 255){
        p = buffer;
        c = *p;
        while(c != 10){
          fputc(*p,lu1);
          p++;
          c = *p;}
        fprintf(lu1,"\n");}
    } while(c != 255);
    return;
}

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 23 of 241

Function discard_scratch_file

```

#include<stdio.h>
#include<stdlib.h>
#include<string.h>

void discard_scratch_file(char filename[]){
/* -----
- - * d i s c a r d _ s c r a t c h _ f i l e * Creates sub-pro-
- -                                     to Delete Files
- -                                     used for Scratch
- -                                     Processing
-----
- - Argument(s):
- -   filename - specification for file to be discarded      (input)
-----
- - Variable Definition(s) -----
- -   Integer Variable(s)
- -   return_code - return code from system call
*/
int return_code = 0;
/* - Character Variable(s)
- - buffer - string variable used to load command for sub-process
*/
char buffer[133];
/* - FILE Variable(s)
- - nout - output file string pointer
*/
extern FILE *nout;
/* - Create Command String for Sub-process ----- */
sprintf(buffer,"rm -f %s\n",filename);
return_code = system(buffer);
if(return_code != 0){
    lines(5);
    fprintf(nout,"0*** W a r n i n g *** Function");
    fprintf(nout," discard_scratch_file -- Error on rm Call\n");
    fprintf(nout,"0Error Number %i Detected\n");
    fprintf(nout,"\n");
}
return;
}

```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 24 of 241

Function fortran_message

```
#include <stdio.h>

void fortran_message(char message[132]){
/* -----
- - *   f o r t r a n _ m e s s a g e   * writes messages from FORTRAN
- -                                     routines to output files
- - -----
- -
- - Variable Definition(s) -----
- - FILE Variable(s)
- -   nout - output file
*/
  extern FILE *nout;
/* - Write Message to Output File ----- */
  fprintf(nout,message);
  fprintf(nout,"\n");
  return;
}
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 25 of 241

Subroutine fclose

```

      SUBROUTINE FTCLOSE(UNIT)
C ---
C -- * F T C L O S E * Closes Sequential Files
C ---
C -- Argument(s):
C --     UNIT - Logical Unit Number Associated with the      (input)
C --           File to be Closed
C ---
C --
C -- Type Statement(s) ---
C -- MESSAGE - String for Messages
      INTEGER UNIT
      CHARACTER*132 MESSAGE
C ---
      CLOSE(UNIT=UNIT,STATUS='KEEP',IOSTAT=IOS,ERR=500)
C -- Normal End of Processing ---
      RETURN
C -- Error Processing ---
500 CALL lines(2)
      MESSAGE =
      2 '0*** F A T A L E R R O R *** SUBR. FCLOSE -- Error Number'
      LOC = MCHAR(132,MESSAGE)+1
      WRITE(MESSAGE(LOC:(LOC+3)), '(I4)') IOS
      WRITE(MESSAGE((MCHAR(132,MESSAGE)+1):)
      2 , '(' Encountered Closing Logical Unit: ''')
      LOC = MCHAR(132,MESSAGE)+1
      WRITE(MESSAGE(LOC:(LOC+1)), '(I2)') UNIT
      CALL fortran_message(MESSAGE)
      CALL lines(2)
      CALL abort()
      RETURN
      END

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 26 of 241

Subroutine ftopen

```

      SUBROUTINE FTOPEN(UNIT, FNAME)
C ---
C -- * F T O P E N * Opens Sequential Files for Processing
C ---
C -- Argument(s):
C --     UNIT - Logical Unit Number to Associate with the (input)
C --           File to be Opened
C --     FNAME - Name of File to Open (input)
C ---
C --
C -- Type Statement(s) ---
C -- MESSAGE - message to pass to output file
      INTEGER UNIT
      CHARACTER*132 MESSAGE
      CHARACTER*(*) FNAME
C ---
      OPEN(UNIT=UNIT, NAME=FNAME, IOSTAT=IOS, ERR=500)
C -- Normal End of Processing ---
      RETURN
C -- Error Processing ---
500 CALL lines(3)
      MESSAGE =
      2 '0*** F A T A L E R R O R *** SUBR. FTOPEN -- '
      WRITE(MESSAGE((MCHAR(132,MESSAGE)+1):)
      2 ,('Error Number '))
      LOC = MCHAR(132,MESSAGE)+1
      WRITE(MESSAGE(LOC:(LOC+3)), '(I4)') IOS
      WRITE(MESSAGE((MCHAR(132,MESSAGE)+1):)
      2 ,('Encounterd Opening File:'))
      CALL fortran_message(MESSAGE)
      MESSAGE = ' '
      WRITE(MESSAGE, '(A)') FNAME
      CALL fortran_message(MESSAGE)
      CALL lines(2)
      CALL abort()
      END

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 27 of 241

Function header

```

#include <stdio.h>
#include <string.h>
#include <time.h>
typedef char ascii_string[133];

void header(){
/* - - - - -
- - * h e a d e r * Prints the header at the top of each page
- - - - -
- -
- - Variable Declaration(s) - - - - - */
/* - Integer Variables
- - npage - page number
- - pid - process id for linkage code execution
- - nline - current line number on output page
- - version - version number for code
*/
static short npage = 0;
extern long pid;
extern short nline;
extern short version;
/* - Character Variables
- - cdate - date of execution
- - crtime - time of execution
- - codenm - name of code
- - modification_level - modification level for code
- - case_title - title for this analysis
*/
extern char cdate[9];
extern char crtime[9];
extern char modification_level;
extern char codenm[5];
extern ascii_string case_title;
/* - FILE Variable(s)
- - nin - input file
- - nout - output file
*/
extern FILE *nin, *nout;
/* - Structures
*/
struct tm *timeptr;
/* - Initialization for first call - - - - - */
if(strlen(cdate) == NULL)
{ size_t maxsize = 9;
long tloc = {NULL};
time(&tloc);
timeptr = localtime(&tloc);
strftime(cdate,maxsize,"%x",timeptr);
strftime(crtime,maxsize,"%X",timeptr);
pid = getpid();
}
/* - Print FORTRAN-type page eject and header record - - - - - */

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 28 of 241

```

npage++;
fprintf(nout, "1");
fprintf(nout, "**** ");
{ register unsigned int i;
  char *p;
  p = codenm;
  for (i=0; i < strlen(codenm); i++,p++)
    fprintf(nout, "%c ", *p);
}
fprintf(nout, "**** VERS. ");
fprintf(nout, "%2i", version);
fprintf(nout, ":");
fprintf(nout, "%c", modification_level);
fprintf(nout, " ** CRWMS,M&O ** DATE: ");
fprintf(nout, cdate);
fprintf(nout, ", TIME: %s", crtime);
fprintf(nout, " PID: %5i", pid);
{ register unsigned int i;
  for(i = 1; i <= 37; i++) fprintf(nout, " ");
}
fprintf(nout, " PAGE: %u \n", npage);
if(npage != 1) {
  fprintf(nout, " ");
  fprintf(nout, case_title);
  fprintf(nout, "\n");
  fprintf(nout, "\n");
  nline = 4;}
else
  nline = 3;
return;
}

```

Subroutine invali

```

      SUBROUTINE INVALI(NVAL, VECTOR, VALUE)
C - - - - -
C - - * I N V A L I *   Initializes a integer vector to a single value
C - - - - -
C - - Argument(s):
C - -     NVAL - Number of entries in vector           (input)
C - -     VECTOR - Vector to be initialized           (output)
C - -     VALUE - Value to initialize vector with     (input)
C - - - - -
C - -
C - - Type Statement(s) - - - - -
      INTEGER VALUE
      INTEGER VECTOR(NVAL)
C - - Sweep through vector initializing entries - - - - -
      DO N = 1, NVAL
        VECTOR(N) = VALUE
      ENDDO
C - - End of Normal Processing - - - - -
      RETURN
      END

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 29 of 241

Subroutine invalr

```

      SUBROUTINE INVALR(NVAL,VECTOR,VALUE)
C ---
C -- * I N V A L R *  Initializes a real vector to a single value
C ---
C -- Argument(s):
C --     NVAL - Number of entries in vector                (input)
C --     VECTOR - Vector to be initialized                (output)
C --     VALUE - Value to initialize vector with          (input)
C ---
C --
C -- Dimension Statement(s) - - - - -
      DIMENSION VECTOR(NVAL)
C -- Sweep through vector initializing entries - - - - -
      DO N = 1,NVAL
         VECTOR(N) = VALUE
      ENDDO
C -- End of Normal Processing - - - - -
      RETURN
      END

```

Function lines

```

void lines(int nl){
/* ---
-- * l i n e s *  Computes the number of cumulative lines printed
--                and determines if a page eject is necessary
-- ---
-- Argument(s):
--     bnl - number of new lines to be printed            (input)
-- ---
-- Variable Declaration(s) - - - - -
-- Integer Variables
--     maxlin - maximum number of lines per page
--     nline - number of lines on current page
*/
  static short maxlin = 63;
  extern short nline;
/* - Add line print request to current page location and determine
   whether page eject is necessary */
  nline += nl;
  if(nline >= maxlin)
  { header();
    nline += nl;}
  return;
}

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 30 of 241

Function load_core_mtls

```

#include<stdio.h>
#include<string.h>

typedef char ascii_string[133];
typedef struct ascii_record{
        struct ascii_record *last;
        ascii_string line;
        struct ascii_record *next;
} a_record;

a_record *memory_ascii_record(int,int,a_record *);

a_record *load_core_mtls(char core_mtls[],int length_fin){
/* -----
- - * l o a d _ c o r e _ m t l s *   Loads Core Materials Dataset
-----
- - Argument(s):
- -   core_mtls - specification for file containing core mat- (input)
- -               materials definitions
- -   length_fin - length of file name (input)
- -   first - pointer to first element in linked list (output)
-----
- -
- - Variable Definition(s) -----
- - Integer Variable(s)
- -   c - single character used to process successive characters
- -       from the input file
*/
    unsigned char c;
/* - Character Variable(s)
- -   record - string used to hold single lines from the input file
- -   p - pointer to location in record
*/
    ascii_string record;
    char *p;
/* - Structured Variable(s)
- -   first - pointer to first member of linked list
- -   ptr_current - pointer to current member of linked list
- -   ptr_next - pointer to next member of linked list
*/
    a_record *first, *ptr_current, *ptr_next;
/* - FILE Variable(s)
- -   lu - pointer to core materials dataset
*/
    FILE *lu;
/* - Open Core Materials Dataset ----- */
    { short int i;
      for(i = 0;i < length_fin;i++){
        if(core_mtls[i] == ' '){
          core_mtls[i] = '\0';
          break;}

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 31 of 241

```

    }
}
lu = fopen(core_mtls,"r");
/* - Allocate First Structure for Processing File */
ptr_current = memory_ascii_record(1,1,ptr_current);
first = ptr_current;
ptr_current->last = NULL;
p = record;
while((c = fgetc(lu)) != 10){
    *p = c;
    p++;}
sprintf(p,"\n");
strcpy(ptr_current->line,record);
ptr_current->next = NULL;
/* - read balance of records in dataset */
p = record;
while((c = fgetc(lu)) != 255){
    *p = c;
    p++;
    if(c == 10){
        sprintf(p,"\n");
        ptr_next = memory_ascii_record(1,1,ptr_next);
        ptr_next->next = NULL;
        ptr_current->next = ptr_next;
        ptr_next->last = ptr_current;
        ptr_current = ptr_next;
        strcpy(ptr_current->line,record);
        p = record;}
}
return first;
}

```

Function load_fuel_material

```

#include <stdio.h>
#include <string.h>
typedef char ascii_string[133];
typedef struct s_material{
    struct s_material *last;
    int atomic_number;
    int mass_number;
    float weight_percentage;
    char library_suffix[5];
    struct s_material *next;
} ll_material;

ll_material *memory_s_material(int,int,ll_material *);
float *memory_float(int,int,float *);

ll_material *load_fuel_material(int *nft,char fprefix[],char dataset[]
,int nlat,int **lattice,float **fp_density){
/* -----
- - * l o a d _ f u e l _ m a t e r i a l * Loads Contents of Fuel
- - Material Dataset into

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 32 of 241

```

- -                                     Memory for Subsequent
- -                                     Processing
- - -----
- - Argument(s):
- -     nft - Number of Fuel Rod Types in Lattice           (output)
- -     fprefix - Prefix for Lattice Composition Database     (input)
- -     dataset - Dataset Name for Fuel Intermediate Dataset of (input)
- -               Interest
- -     nlat - lattice dimensionality                         (input)
- -     lattice - Fuel Rod Type Map                           (output)
- -     p - Vector of Pointers to Start of Linked List       (output)
- -           Containing Material Inventories for each
- -           Distinct Fuel Rod Type
- -     fp_density - densities for each fuel type             (output)
- - -----
- - Variable Declarations - - - - -
- - Integer Variable(s)
- -     n - loop index
*/
short int n;
/* - Character Variable(s)
- -     c - single character for reading input file
- -     buffer - string variable used to manage dataset processing
- -     pb - pointer to buffer
- -     fn - complete dataset file descriptor
- -     header - header from dataset
*/
unsigned char c;
char *pb;
ascii_string fn, buffer, header;
/* - File Variable(s)
- -     ds - dataset stream pointer
- -     nout - output file
*/
FILE *ds;
extern FILE *nout;
/* - Structured Variable(s)
*/
ll_material *p, *p_return;
/* - Put Dataset Name in C String Format - - - - - */
{ int len = 132, length;
  length = mchar(&len,dataset);
  dataset[length] = '\0';
}
strcpy(fn,fprefix);
strcat(fn,dataset);
/* - Open File for Processing - - - - - */
ds = fopen(fn,"r");
if(!ds){
  lines(5);
  fprintf(nout,"0*** F a t a l E r r o r *** -- Function");
  fprintf(nout
    , " Load Fuel Material, Fuel Material Dataset not Found\n");
}

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 33 of 241

```

    fprintf(nout,"0Dataset Name:\n");
    fprintf(nout," %s\n",fn);
    abort();}
/* - Read Header Record */
strcpy(buffer,"");
pb = buffer;
do{
    c = fgetc(ds);
    *pb = c;
    pb++;}
while (c != 10);
*pb = '\0';
strcpy(header,buffer);
/* - Skip QA Record */
do
    c = fgetc(ds);
while (c != 10);
/* - Skip Fuel Rod Type Map Title */
pb = buffer;
do{
    c = fgetc(ds);
    *pb = c;
    pb++;}
while (c != 10);
/* - Read Fuel Rod Type Map Line by Line */
{ short int j;
  int *plattice;
  ascii_string holder;
  plattice = *lattice;
  for( j = 1; j <= nlat; j++){
    pb = buffer;
    do{
        c = fgetc(ds);
        *pb = c;
        pb++;}
    while(c != 10);
    *pb = '\0';
    pb = buffer;
    for(n = 0; n < nlat; n++,plattice++){
        while(*pb == ' ') pb++;
        strcpy(holder,"");
        do{
            strncat(holder,pb,1);
            pb++;}
        while(*pb != ' ');
        pb++;
        sscanf(holder,"%i",plattice);}
    }
/* - Compute Number of Lattice Types */
*nft = 0;
plattice = *lattice;
for( j = 1; j <= (nlat*nlat); j++,plattice++)
    if(*plattice > *nft) *nft = *plattice;
p = memory_s_material(1,*nft,p);

```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 34 of 241

```
    }
/* - Skip Title for Density Vector */
pb = buffer;
do{
    c = fgetc(ds);
    *pb = c;
    pb++;}
while (c != 10);
/* - Allocate Memory for Density Vector */
*fp_density = memory_float(1,*nft,*fp_density);
/* - Read Density Values */
{ short int nmax,i;
  ascii_string holder;
  float *p = *fp_density;
  n = 1;
  do{
    pb = buffer;
    do{
      c = fgetc(ds);
      *pb = c;
      pb++;}
    while(c != 10);
    *pb = '\0';
    pb = buffer;
    if((n+5) > *nft) nmax = *nft;
    else nmax = n+4;
    for( i = n; i <= nmax; i++,p++){
      while(*pb == ',') pb++;
      strcpy(holder,"");
      c = *pb;
      while((*pb != ',') && (c != 10)){
        strncat(holder,pb,1);
        pb++;
        c = *pb;}
      pb++;
      sscanf(holder,"%f",p);}
    n = nmax+1;
  }
  while(n <= *nft);
}
/* - Skip Title for Fuel Material Compositions */
pb = buffer;
do{
  c = fgetc(ds);
  *pb = c;
  pb++;}
while (c != 10);
/* - Read Each Materials for Each Fuel Rod Type */
/* - Allocate Memory for Vector of Linked Lists */
p = memory_s_material(1,*nft,p);
p_return = p;
{ unsigned char c;
  int index, n_entries, i, j;
  char label[6];
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 36 of 241

```

        ptr_mtl->mass_number = value;
        pb++;
    }
    break;
case 4:
    { float value;
      sscanf(holder,"%f",&value);
      ptr_mtl->weight_percentage = value;
      pb++;
    }
}
ptr_mtl->next = NULL;
ptr_mtl->last = ptr_last;
ptr_last = ptr_mtl;
if( i < (n_entries-1)){
    ptr_next = memory_s_material(1,1,ptr_next);
    ptr_mtl->next = ptr_next;
    ptr_mtl = ptr_next;
    ptr_mtl->atomic_number = 0;
    strcpy(ptr_mtl->library_suffix,"");
    ptr_mtl->mass_number = 0;
    ptr_mtl->weight_percentage = 0.0;
}
}
p++;
}
}
return p_return;
}

```

Function load_surface_usage_list

```

#include <stdio.h>
#include <string.h>

typedef char ascii_string[133];

typedef struct su_list{
    struct su_list *last;
    int index;
    ascii_string label;
    ascii_string value;
    char mnemonic[4];
    ascii_string equivalent_label;
    struct su_list *next;
} surface_usage_list;

surface_usage_list *memory_surface_usage_list(int,surface_usage_list *);

surface_usage_list *load_surface_usage_list(ascii_string label,int index
,ascii_string value,char mnemonic[4],ascii_string equivalent_label
,surface_usage_list *p){
/* -----
- - load_surface_usage_list - loads new values into surface usage list

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 37 of 241

```

- - - - -
- - Argument(s):
- -     label - character string to be added to new structure   (input)
- -     index - index for character string                     (input)
- -     value - definition of surface                          (input)
- -     mnemonic - surface definition mnemonic                 (input)
- -     equivalent_label
- -         - label of
- -     p - pointer to memory locations                        (i&o)
- - - - -
- -
- - Variable Definition(s) - - - - -
- - Structured Variable(s)
- - p_new - pointer to new structure
*/
    surface_usage_list *p_new;
/* - Check for Initial Member of String - - - - - */
    if(p == NULL){
/* - Initial Member */
        p = memory_surface_usage_list(1,p);
        p->last = NULL;
        p->next = NULL;
        strcpy(p->label,label);
        strcpy(p->value,value);
        strcpy(p->mnemonic,mnemonic);
        strcpy(p->equivalent_label,equivalent_label);
        p->index = index;
        return p;}
/* - Addition to Linked List */
    else{
        p_new = memory_surface_usage_list(1,p_new);
        p->next = p_new;
        p_new->last = p;
        p = p_new;
        p->next = NULL;
        strcpy(p->label,label);
        strcpy(p->value,value);
        strcpy(p->mnemonic,mnemonic);
        strcpy(p->equivalent_label,equivalent_label);
        p->index = index;
        return p;}
}

```

Function load_usage_list

```

#include <stdio.h>
#include <string.h>

typedef char ascii_string[133];

typedef struct u_list{
    struct u_list *last;
    int index;
    ascii_string label;
    struct u_list *next;
}

```



```

        } usage_list;

usage_list *memory_usage_list(int,usage_list *);

usage_list *load_usage_list(char label[],int index,usage_list *p){
/* - - - - -
- - load_usage_list - loads new values into usage list
- - - - -
- - Argument(s):
- -     label - character string to be added to new structure (input)
- -     index - index for character string (input)
- -     p - pointer to memory locations (i&o)
- - - - -
- -
- - Variable Definition(s) - - - - -
- - Structured Variable(s)
- - p_new - pointer to new structure
*/
usage_list *p_new;
/* - Check for Initial Member of String - - - - - */
if(p == NULL){
/* - Initial Member */
p = memory_usage_list(1,p);
p->last = NULL;
p->next = NULL;
strcpy(p->label,label);
p->index = index;
return p;}
/* - Addition to Linked List */
else{
p_new = memory_usage_list(1,p_new);
p->next = p_new;
p_new->last = p;
p = p_new;
p->next = NULL;
strcpy(p->label,label);
p->index = index;
return p;}
}

```

Function mchar

```

FUNCTION MCHAR(MAXLEN,BUFFER)
C - - - - -
C - - * M C H A R * Determines the number of non-blank characters in
C - - character variable of total length MAXLEN
C - - - - -
C - - Argument(s):
C - -     MAXLEN - dimension of character variable (input)
C - -     BUFFER - character variable (input)
C - - - - -
C - -
C - - Type statement(s) - - - - -
CHARACTER*1 BLANK
CHARACTER*(*) BUFFER

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 39 of 241

```

C -- Data statement(s) -----
  DATA BLANK/' '/
C -----
  MCHAR = MAXLEN
  DO 10 NC = MAXLEN,1,-1
  IF(BUFFER(NC:NC) .NE. BLANK) THEN
    MCHAR = NC
    GO TO 20
  ENDIF
  10 CONTINUE
  MCHAR = 0
C -- End of processing -----
  20 CONTINUE
  END

```

Function memory_ascii_record

```

#include<stdio.h>
#include<malloc.h>
#include<errno.h>

typedef char ascii_string[133];
typedef struct ascii_record{
    struct ascii_record *last;
    ascii_string line;
    struct ascii_record *next;
} a_record;

a_record *memory_ascii_record(int operation,int number, a_record *p){
/* -----
- - memory_ascii_record - manages memory requests for character storage
- -                       for variables of the ascii_record structure
- -                       type
-----
- - Argument(s):
- - operation - flag for memory operation to perform          (input)
- -             ( 1 - allocate memory,
- -             -1 - return memory)
- -   number - number of structures for which to allocate      (input)
- -             memory
- -   p - pointer to memory locations                          (i&o)
-----
- - Variable Definition(s) -----
- - Integer Variables
- -   n - number of entries in proper size
- -   size - number of bytes requested
- -   storage_c - current bytes of character storage requested
- -   storage_ct - maximum bytes of character storage requested
*/
  size_t size ;
  extern int storage_c, storage_ct;
/* - FILE Pointer Variable(s)
- - nout - output file */
  extern FILE *nout;

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 40 of 241

```

/* -- Process Memory Request - - - - - */
if(operation == 1) {
    size = number*sizeof(a_record);
    p = (a_record *)malloc(size);
    if((errno != NULL) || (p == NULL)) {
        if(p == NULL) {
            lines(2);
            fprintf(nout
, "0*** F A T A L E R R O R *** -- Function memory_ascii_record: ");
            fprintf(nout, "Null Pointer Returned\n");
        }
        if(errno != NULL){
            lines(2);
            fprintf(nout
, "0*** F A T A L E R R O R *** -- Function memory_ascii_record: ");
            fprintf(nout, "Error Number %i Detected\n");
        }
        lines(2);
        abort();
    }
    storage_c += number*sizeof(ascii_string);
    if(storage_c > storage_ct) storage_ct = storage_c;
    return p;}
else {
    size = number*sizeof(ascii_string);
    storage_c -= size;
    free(p);
    return p;}
}

```

Function memory_ascii_string

```

#include<stdio.h>
#include<string.h>
#include<malloc.h>
#include<errno.h>
typedef char ascii_string[133];

ascii_string *memory_ascii_string(int operation, int number
,ascii_string *p){
/* - - - - -
- - memory_ascii_string - manages memory requests for character storage
- - for variables of the ascii_string type
- - - - -
- - Argument(s):
- - operation - flag for memory operation to perform (input)
- - ( 1 - allocate memory,
- - -1 - return memory)
- - number - number of storage locations to request (input)
- - (n.b., not used for return requests)
- - p - pointer to memory locations (i&o)
- - - - -
- - Variable Definition(s) - - - - -
- - Integer Variables

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 41 of 241

```

- -          n - number of entries in proper size
- -          size - number of bytes requested
- -    storage_c - current bytes of character storage requested
- -    storage_ct - maximum bytes of character storage requested
*/
    size_t size ;
    extern int storage_c, storage_ct;
/* - FILE Pointer Variable(s)
- - nout - output file */
    extern FILE *nout;
/* - - Process Memory Request - - - - - - - - - - - - - - - - - - */
    if(operation == 1) {
        size = ((size_t) number)*sizeof(ascii_string);
        p = (ascii_string *)malloc(size);
        if((errno != NULL) || (p == NULL)) {
            if(p == NULL) {
                lines(2);
                fprintf(nout
                    , "0*** F A T A L E R R O R *** -- Function memory_character: ");
                fprintf(nout, "Null Pointer Returned\n");
            }
            if(errno != NULL){
                lines(2);
                fprintf(nout
                    , "0*** F A T A L E R R O R *** -- Function memory_character: ");
                fprintf(nout, "Error Number %i Detected\n");
            }
        }
        lines(2);
        abort();
    }
    storage_c += size;
    if(storage_c > storage_ct) storage_ct = storage_c;
    return p;}
else {
    size = ((size_t) number)*sizeof(ascii_string);
    storage_c -= size;
    free(p);
    return p;}
}

```

Function memory_fg_list

```

#include <stdio.h>
#include <malloc.h>
#include <errno.h>

typedef char ascii_string[133];

typedef struct fuel_geometry{
    ascii_string gds_name;
    int latdim;
    int nwr;
    float cthick;
    float asin;
    float wgap;
}

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 42 of 241

```

        float ngap;
        float cradius;
        float fsrd;
        float cfsrd;
        float rpitch;
        float cod;
        float cld;
        float pod;
        char frm[6];
        char fcm[6];
    } fg_list;

fg_list *memory_fg_list(int operation, int number, fg_list *p){
/* -----
- - memory_fg_list - manages memory requests for storage for
-----
- - Argument(s):
- - operation - flag for memory operation to perform          (input)
- -             ( 1 - allocate memory,
- -             -1 - return memory)
- -   number - number of structures to allocate                (input)
- -     p - pointer to memory locations                        (i&o)
-----
- -
- - Variable Definition(s) -----
- - Integer Variables
- -     n - number of entries in proper size
- -     size - number of bytes requested
- -     storage_i - current bytes of integer storage requested
- -     storage_it - maximum bytes of integer storage requested
- -     storage_r - current bytes of single-precision real storage
- -                 requested
- -     storage_rt - maximum bytes of single-precision real storage
- -                 requested
*/
    size_t size ;
    extern int storage_i, storage_it;
    extern int storage_r, storage_rt;
    extern int storage_c, storage_ct;
/* - FILE Pointer Variable(s)
- - nout - output file */
    extern FILE *nout;
/* - - Process Memory Request ----- */
    if(operation == 1) {
        size = number*sizeof(fg_list);
        p = (fg_list *)malloc(size);
        if((errno != NULL) || (p == NULL)) {
            if(p == NULL) {
                lines(2);
                fprintf(nout
, "0*** F A T A L E R R O R *** -- Function memory_fg_list: ");
                fprintf(nout, "Null Pointer Returned\n");
            }
            if(errno != NULL){

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 43 of 241

```

        lines(2);
        fprintf(nout
, "0*** F A T A L   E R R O R   *** -- Function memory_fg_list: ");
        fprintf(nout, "Error Number %i Detected\n");
    }
    lines(2);
    abort();
}
storage_i += 2*sizeof(int);
if(storage_i > storage_it) storage_it = storage_i;
storage_r += 10*sizeof(float);
if(storage_r > storage_rt) storage_rt = storage_r;
storage_c += 12*sizeof(char)+sizeof(ascii_string);
if(storage_c > storage_ct) storage_ct = storage_c;
return p;}
else {
    free(p);
    storage_i -= 2*sizeof(int);
    storage_r -= 10*sizeof(float);
    storage_c -= 12*sizeof(char)+sizeof(ascii_string);
    return p;}
}

```

Function memory_float

```

#include <stdio.h>
#include <stdlib.h>
#include <malloc.h>
#include <errno.h>

float *memory_float(int operation, int number, float *p){
/* -----
- - memory_integer - manages memory requests for single-precision real
- -                    storage
-----
- - Argument(s):
- - operation - flag for memory operation to perform          (input)
- -              ( 1 - allocate memory,
- -              -1 - return memory)
- -   number - number of storage locations to request          (input)
- -              (n.b., not used for return requests)
- -   p - pointer to memory location                          (i&o)
-----
- -
- - Variable Definition(s) -----
- - Integer Variables
- -   storage_r - current bytes of float storage requested
- -   storage_rt - maximum bytes of float storage requested
*/
extern int storage_r, storage_rt;
/* - FILE Pointer Variable(s)
- - nout - output file */
extern FILE *nout;
/* - - Process Memory Request ----- */
if(operation == 1) {

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 44 of 241

```

p = (float *)calloc((size_t)number, (size_t) sizeof(float));
if((errno != NULL) || (p == NULL)) {
    if(p == NULL) {
        lines(2);
        fprintf(nout
            , "0*** F A T A L E R R O R *** -- Function memory_float: ");
        fprintf(nout, "Null Pointer Returned\n");
    }
    if(errno != NULL){
        lines(2);
        fprintf(nout
            , "0*** F A T A L E R R O R *** -- Function memory_float: ");
        fprintf(nout, "Error Number %i Detected\n");
    }
    lines(2);
    abort();
}
storage_r += number;
if(storage_r > storage_rt) storage_rt = storage_r;
return p;}
else {
    storage_r -= number;
    free(p);
    return p;};
}

```

Function memory_integer

```

#include <stdio.h>
#include <stdlib.h>
#include <malloc.h>
#include <errno.h>

int *memory_integer(int operation, int number, int *p){
/* - - - - -
- - memory_integer - manages memory requests for integer storage
- - - - -
- - Argument(s):
- - operation - flag for memory operation to perform          (input)
- -             ( 1 - allocate memory,
- -             -1 - return memory)
- - number - number of storage locations to request            (input)
- -             (n.b., not used for return requests)
- - p - pointer to memory location                             (i&o)
- - - - -
- - Variable Definition(s) - - - - -
- - Integer Variables
- - storage_i - current bytes of integer storage requested
- - storage_it - maximum bytes of integer storage requested
*/
extern int storage_i, storage_it;
/* - FILE Pointer Variable(s)
- - nout - output file */
extern FILE *nout;

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 45 of 241

```

/* - - Process Memory Request - - - - - */
if(operation == 1) {
    p = (int *)calloc((size_t)number, (size_t) sizeof(int));
    if((errno != NULL) || (p == NULL)) {
        if(p == NULL) {
            lines(2);
            fprintf(nout
, "0*** F A T A L E R R O R *** -- Function memory_integer: ");
            fprintf(nout, "Null Pointer Returned\n");
        }
        if(errno != NULL){
            lines(2);
            fprintf(nout
, "0*** F A T A L E R R O R *** -- Function memory_integer: ");
            fprintf(nout, "Error Number %i Detected\n");
        }
        lines(2);
        abort();
    }
    storage_i += number;
    if(storage_i > storage_it) storage_it = storage_i;
    return p;}
else {
    storage_i -= number;
    free(p);
    return p;};
}

```

Function memory_lattice_list

```

#include <stdio.h>
#include <malloc.h>
#include <errno.h>

typedef struct all{
    struct all *last;
    int basis_lattice_material_index;
    int lattice_material_index;
    struct all *next;
} augmented_lattice_list;

augmented_lattice_list *memory_lattice_list(int operation
, augmented_lattice_list *p){
/* - - - - -
- - memory_usage_list - manages memory requests for storage for vari-
- -                         ables of the augmented_lattice_list structure
- -                         type
- - - - -
- - Argument(s):
- - operation - flag for memory operation to perform           (input)
- -             ( 1 - allocate memory,
- -             -1 - return memory)
- -             p - pointer to memory locations                 (i&o)
- - - - -
- -

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 46 of 241

```

- - Variable Definition(s) - - - - -
- - Integer Variables
- -     n - number of entries in proper size
- -     size - number of bytes requested
- -     storage_i - current bytes of integer storage requested
- -     storage_it - maximum bytes of integer storage requested
*/
size_t size ;
extern int storage_i, storage_it;
/* - FILE Pointer Variable(s)
- - nout - output file */
extern FILE *nout;
/* - - Process Memory Request - - - - - */
if(operation == 1) {
    size = sizeof(augmented_lattice_list);
    p = (augmented_lattice_list *)calloc((size_t) 1
    , (size_t) sizeof(augmented_lattice_list));
    if((errno != NULL) || (p == NULL)) {
        if(p == NULL) {
            lines(2);
            fprintf(nout
, "0*** F A T A L E R R O R *** -- Function memory_lattice_list: ");
            fprintf(nout, "Null Pointer Returned\n");
        }
        if(errno != NULL){
            lines(2);
            fprintf(nout
, "0*** F A T A L E R R O R *** -- Function memory_lattice_list: ");
            fprintf(nout, "Error Number %i Detected\n");
        }
        lines(2);
        abort();
    }
    storage_i += 2*sizeof(int);
    if(storage_i > storage_it) storage_it = storage_i;
    return p;}
else {
    free(p);
    storage_i -= 2*sizeof(int);
    return p;}
}

```

Function memory_s_material

```

#include<stdio.h>
#include<malloc.h>
#include<errno.h>

typedef char ascii_string[133];
typedef struct ascii_record{
    struct ascii_record *last;
    ascii_string line;
    struct ascii_record *next;
} a_record;

typedef struct s_material{

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B000000000-01717-0210-00010 REV 01 Attachment XIV Page 47 of 241

```

        struct s_material *last;
        int atomic_number;
        int mass_number;
        float weight_percentage;
        char library_suffix[5];
        struct s_material *next;
    } ll_material;

ll_material *memory_s_material(int operation,int number,ll_material *p){
/* - - - - -
- - memory_s_material - manages memory requests for storage for vari-
- -                      ables of the s_material structure type
- - - - -
- - Argument(s):
- - operation - flag for memory operation to perform          (input)
- -              ( 1 - allocate memory,
- -              -1 - return memory)
- -   number - number of structures to process                (input)
- -   p - pointer to memory locations                          (i&o)
- - - - -
- - Variable Definition(s) - - - - -
- - Integer Variables
- -   n - number of entries in proper size
- -   size - number of bytes requested
- -   storage_i - current bytes of integer storage requested
- -   storage_it - maximum bytes of integer storage requested
- -   storage_r - current bytes of single-precision real storage
- -               requested
- -   storage_rt - maximum bytes of single-precision real storage
- -               requested
- -   storage_c - current bytes of character storage requested
- -   storage_ct - maximum bytes of character storage requested
*/
    size_t size ;
    extern int storage_i, storage_it;
    extern int storage_r, storage_rt;
    extern int storage_c, storage_ct;
/* - FILE Pointer Variable(s)
- - nout - output file */
    extern FILE *nout;
/* - - Process Memory Request - - - - - */
    if(operation == 1) {
        size = number*sizeof(ll_material);
        p = (ll_material *)malloc(size);
        if((errno != NULL) || (p == NULL)) {
            if(p == NULL) {
                lines(2);
                fprintf(nout
, "0*** F A T A L   E R R O R   *** -- Function memory_s_material: ");
                fprintf(nout,"Null Pointer Returned\n");
            }
            if(errno != NULL){

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 48 of 241

```

        lines(2);
        fprintf(nout
, "0*** F A T A L E R R O R *** -- Function memory_s_material: ");
        fprintf(nout, "Error Number %i Detected\n");
    }
    lines(2);
    abort();
}
storage_c += 3*number*sizeof(char);
if(storage_c > storage_ct) storage_ct = storage_c;
storage_i += 2*number*sizeof(int);
if(storage_i > storage_it) storage_it = storage_i;
storage_r += number*sizeof(float);
if(storage_r > storage_rt) storage_rt = storage_r;
return p;}
else {
    free(p);
    storage_c -= 3*number*sizeof(char);
    storage_i -= 2*number*sizeof(int);
    storage_r -= number*sizeof(float);
    return p;}
}

```

Function memory_surface_usage_list

```

#include <stdio.h>
#include <malloc.h>
#include <errno.h>

typedef char ascii_string[133];

typedef struct su_list{
        struct su_list *last;
        int index;
        ascii_string label;
        ascii_string value;
        char mnemonic[4];
        ascii_string equivalent_label;
        struct su_list *next;
    } surface_usage_list;

surface_usage_list *memory_surface_usage_list(int operation
, surface_usage_list *p){
/* -----
- - memory_surface_usage_list - manages memory requests for storage for
- -                               variables of the surface_usage_list
- -                               structure type
-----
- - Argument(s):
- - operation - flag for memory operation to perform           (input)
- -             ( 1 - allocate memory,
- -             -1 - return memory)
- -           p - pointer to memory locations                   (i&o)
-----
- -

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 49 of 241

```

- - Variable Definition(s) - - - - -
- - Integer Variables
- -     n - number of entries in proper size
- -     size - number of bytes requested
- -     storage_i - current bytes of integer storage requested
- -     storage_it - maximum bytes of integer storage requested
- -     storage_c - current bytes of character storage requested
- -     storage_ct - maximum bytes of character storage requested
*/
size_t size ;
extern int storage_i, storage_it;
extern int storage_c, storage_ct;
/* - FILE Pointer Variable(s)
- - nout - output file */
extern FILE *nout;
/* - - Process Memory Request - - - - - */
if(operation == 1) {
    size = sizeof(surface_usage_list);
    p = (surface_usage_list *)malloc(size);
    if((errno != NULL) || (p == NULL)) {
        if(p == NULL) {
            lines(2);
            fprintf(nout
, "0*** F A T A L   E R R O R   *** -- Function");
            fprintf(nout, " memory_surface_usage_list: ");
            fprintf(nout, "Null Pointer Returned\n");
        }
        if(errno != NULL){
            lines(2);
            fprintf(nout
, "0*** F A T A L   E R R O R   *** -- Function");
            fprintf(nout, " memory_surface_usage_list: ");
            fprintf(nout, "Error Number %i Detected\n");
        }
        lines(2);
        abort();
    }
    storage_c += 3*sizeof(ascii_string)+4*sizeof(char);
    if(storage_c > storage_ct) storage_ct = storage_c;
    storage_i += sizeof(int);
    if(storage_i > storage_it) storage_it = storage_i;
    return p;}
else {
    free(p);
    storage_c -= (3*sizeof(ascii_string)+4*sizeof(char));
    storage_i -= sizeof(int);
    return p;}
}

```

Function memory_usage_list

```

#include <stdio.h>
#include <malloc.h>
#include <errno.h>

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 50 of 241

```

typedef char ascii_string[133];

typedef struct u_list{
    struct u_list *last;
    int index;
    ascii_string label;
    struct u_list *next;
} usage_list;

usage_list *memory_usage_list(int operation, usage_list *p){
/* - - - - -
- - memory_usage_list - manages memory requests for storage for vari-
- -                      ables of the usage_list structure type
- - - - -
- - Argument(s):
- - operation - flag for memory operation to perform          (input)
- -              ( 1 - allocate memory,
- -              -1 - return memory)
- -          p - pointer to memory locations                    (i&o)
- - - - -
- - Variable Definition(s) - - - - -
- - Integer Variables
- -          n - number of entries in proper size
- -          size - number of bytes requested
- -          storage_i - current bytes of integer storage requested
- -          storage_it - maximum bytes of integer storage requested
- -          storage_c - current bytes of character storage requested
- -          storage_ct - maximum bytes of character storage requested
*/
    size_t size ;
    extern int storage_i, storage_it;
    extern int storage_c, storage_ct;
/* - FILE Pointer Variable(s)
- - nout - output file */
    extern FILE *nout;
/* - - Process Memory Request - - - - - */
    if(operation == 1) {
        size = sizeof(usage_list);
        p = (usage_list *)malloc(size);
        if((errno != NULL) || (p == NULL)) {
            if(p == NULL) {
                lines(2);
                fprintf(nout
, "0*** FATAL ERROR *** -- Function memory_usage_list: ");
                fprintf(nout, "Null Pointer Returned\n");
            }
            if(errno != NULL){
                lines(2);
                fprintf(nout
, "0*** FATAL ERROR *** -- Function memory_usage_list: ");
                fprintf(nout, "Error Number %i Detected\n");
            }
        }
    }
}

```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 51 of 241

```
    }
    lines(2);
    abort();
}
storage_c += sizeof(ascii_string);
if(storage_c > storage_ct) storage_ct = storage_c;
storage_i += sizeof(int);
if(storage_i > storage_it) storage_it = storage_i;
return p;}
else {
    free(p);
    storage_c -= sizeof(ascii_string);
    storage_i -= sizeof(int);
    return p;}
}
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 52 of 241

Function memsum

```

#include <stdio.h>

void memsum(){
/* - - - - -
- - * m e m s u m * Edits the Dynamic Storage Usage
- - - - -
- -
- - Variable Definition(s) - - - - -
- - Integer Variable(s)
- - integer_storage - total integer storage expressed in bytes
- - real_storage - total real storage expressed in bytes
- - character_storage - total character storage expressed in bytes
- - total_storage - total storage expressed in bytes
*/
int integer_storage, real_storage, character_storage, total_storage;
extern int storage_it, storage_rt, storage_ct;
/* - FILE Variable(s)
- - nout - pointer to code output file
*/
extern FILE *nout;
/* - Compute Storage Totals in Bytes - - - - - */
integer_storage = storage_it*sizeof(int);
real_storage = storage_rt*sizeof(float);
character_storage = storage_ct*sizeof(char);
total_storage = integer_storage+real_storage+character_storage;
/* - Write the Total to Output - - - - - */
lines(8);
fprintf(nout,"0Total Dynamic Memory Usage in Bytes:\n");
fprintf(nout,"\n");
fprintf(nout," Integer = %7i \n",integer_storage);
fprintf(nout," Real = %7i \n",real_storage);
fprintf(nout," Character = %7i \n",character_storage);
fprintf(nout," ----- \n");
fprintf(nout," Total = %7i \n",total_storage);
return;
}

```

Function rollup_llm

```

#include<stdlib.h>

typedef struct s_material{
    struct s_material *last;
    int atomic_number;
    int mass_number;
    float weight_percentage;
    char library_suffix[5];
    struct s_material *next;
} ll_material;

ll_material *memory_s_material(int,int,ll_material *);

void rollup_llm(ll_material *p){

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 53 of 241

```

/* - - - - -
- - rollup_llm - returns memory associated with a linked list with a
- -             s_material structure type
- - - - -
- - Argument(s):
- -           p - pointer to start of linked list             (input)
- - - - -
- - Variable Definition(s) - - - - -
- - Structured Variable(s)
- - last - pointer to previous structure in linked list
*/
ll_material *last;
/* - Search for End of Linked List - - - - - */
while(p->next != NULL) p = p->next;
/* - Sweep through List Backwards, Returning Memory - - - - - */
while(p->last != NULL){
    last = p->last;
    p = memory_s_material(-1,l,p);
    p = last;
}
return;
}

```

Function search_surface_usage_list

```

#include <stdio.h>
#include <string.h>

typedef char ascii_string[133];

typedef struct su_list{
    struct su_list *last;
    int index;
    ascii_string label;
    ascii_string value;
    char mnemonic[4];
    ascii_string equivalent_label;
    struct su_list *next;
} surface_usage_list;

void search_surface_usage_list(int option,ascii_string label,int *index
,ascii_string value,char mnemonic[4],ascii_string equivalent_label
,surface_usage_list *p){
/* - - - - -
- - search_surface_usage_list - searches a surface usage linked list
- -                             seeking match
- - - - -
- - Argument(s):
- -   option - flag to indicate processing to be performed      (input)
- -           (0 - search for label based on index,
- -           1 - search for index based on label,
- -           2 - serach for index based on surface defini- (input)
- -             tion)
- -   label - character string either to be sought or          (i|o)

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 54 of 241

```

- -          returned
- -      index - index either to be sought or returned          (i|o)
- -      value - definition of surface                          (input)
- -      mnemonic - mnemonic for surface                       (input)
- -      equivalent_label
- -          - label for equivalent surface                    (input)
- -          p - pointer to start of linked list              (input)
- - -----
- -
- - Variable Definition(s) - - - - -
- - FILE Variable(s)
- -      nout - pointer to output stream
*/
extern FILE *nout;
/* - Search based on Index - - - - - */
if(option == 0){
    do{
        if(p->index == *index){
            strcpy(label,p->label);
            return;}
        p = p->next;
    } while(p->next != NULL);
/* - Process the Last Entry */
if(p->index == *index){
    strcpy(label,p->label);
    return;}
else{
    lines(2);
    fprintf(nout
        ,"0*** F A T A L E R R O R *** Function search_usage_list:");
    fprintf(nout," No match found for index = %i\n",*index);
    abort();}
}
if(option == 1){
    do{
        if(strcmp(label,p->label) == 0){
            *index = p->index;
            return;}
        p = p->next;
    } while(p->next != NULL);
/* - Process the Last Entry */
if(strcmp(label,p->label) == 0){
    *index = p->index;
    return;}
else{
    *index = 0;
}
}
if(option == 2){
    do{
        if((strcmp(value,p->value) == 0) &&
            (strcmp(mnemonic,p->mnemonic) == 0)){
            *index = p->index;
            strcpy(equivalent_label,p->label);

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 55 of 241

```

        return;}
    p = p->next;
} while(p->next != NULL);
/* - Process the Last Entry */
if((strcmp(value,p->value) == 0) &&
   (strcmp(mnemonic,p->mnemonic) == 0)){
    *index = p->index;
    strcpy(equivalent_label,p->label);
    return;}
else
    *index = 0;
}
}

```

Function search_usage_list

```

#include <stdio.h>
#include <string.h>

typedef char ascii_string[133];

typedef struct u_list{
    struct u_list *last;
    int index;
    ascii_string label;
    struct u_list *next;
} usage_list;

void search_usage_list(int option,char label[],int *index
,usage_list *p){
/* -----
- - search_usage_list - searches a usage linked list seeking match
-----
- - Argument(s):
- -     option - flag to indicate processing to be performed      (input)
- -             (0 - search for label based on index,
- -             1 - search for index based on label)
- -     label - character string either to be sought or          (i|o)
- -             returned
- -     index - index either to be sought or returned            (i|o)
- -     p - pointer to start of linked list                       (input)
-----
- - Variable Definition(s) -----
- - FILE Variable(s)
- - nout - pointer to output stream
*/
extern FILE *nout;
/* - Search based on Index ----- */
if(option == 0){
    do{
        if(p->index == *index){
            strcpy(label,p->label);
            return;
        }
    }
}

```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 56 of 241

```
    p = p->next;
  } while(p->next != NULL);
/* - Process the Last Entry */
  if(p->index == *index){
    strcpy(label,p->label);
    return;}
  else{
    lines(2);
    fprintf(nout
      , "0*** FATAL ERROR *** -- Function search_usage_list:");
      fprintf(nout, " No match found for index = %i\n", *index);
      abort();}
}
else{
  do{
    if(strcmp(label,p->label) == 0){
      *index = p->index;
      return;
    }
    p = p->next;
  } while(p->next != NULL);
/* - Process the Last Entry */
  if(strcmp(label,p->label) == 0){
    *index = p->index;
    return;}
  else{
    *index = 0;
  }
}
}
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 57 of 241

2.3. Input Routines

Subroutine bpchek

```

      SUBROUTINE BPCHEK(NROWB,NCOLB,NROWBP,NCOLBP,VALIDB,BLADEP)
C - - - - -
C - - * B P C H E K * Determines whether Blade Positions are Valid
C - - - - -
C - - Argument(s):
C - -     NROWB - Number of Control Blade Rows in Full Core (input)
C - -     NCOLB - Number of Control Blade Columns in Full (input)
C - -             Core
C - -     NROWBP - Number of Control Blade Rows in Problem (input)
C - -     NCOLBP - Number of Control Blade Columns in Problem (input)
C - -     VALIDB - Integer Array Containing Valid Locations (input)
C - -             for Control Blades
C - -     BLADEP - Integer Array Containing Blade Insertion (input)
C - -             Values
C - - - - -
C - -
C - - Type Statement(s) - - - - -
C - -     ERROR - Logical Flag Indicating that Fatal Error has been
C - -             Detected
C - -     MESSAGE - String for Messages
      LOGICAL ERROR
      INTEGER VALIDB(NCOLB,NROWB), BLADEP(NCOLBP,NROWBP)
      CHARACTER*132 MESSAGE
C - - Sweep through Map - - - - -
      ERROR = .FALSE.
      DO J = 1,NROWBP
        DO I = 1,NCOLBP
          IF(VALIDB(I,J) .EQ. 0) THEN
            IF(BLADEP(I,J) .EQ. -1) THEN
              CONTINUE
            ELSE
              ERROR = .TRUE.
              CALL lines(3)
              MESSAGE =
2 '0*** F A T A L E R R O R *** SUBR. BPCHEK -- Control'
              WRITE(MESSAGE((MCHAR(132,MESSAGE)+1):))
2 , '(' Blade Assignment to Invalid Location ('))'
              LOC = MCHAR(132,MESSAGE)+1
              WRITE(MESSAGE(LOC:(LOC+1)), '(I2)') I
              WRITE(MESSAGE((MCHAR(132,MESSAGE)+1):), '('','')')
              LOC = MCHAR(132,MESSAGE)+1
              WRITE(MESSAGE(LOC:(LOC+1)), '(I2)') I
              WRITE(MESSAGE((MCHAR(132,MESSAGE)+1):), '('')')
              CALL fortran_message(MESSAGE)
              MESSAGE = ' Blade Position = '
              WRITE(MESSAGE((MCHAR(132,MESSAGE)+1):), '(I2)') BLADEP(I,J)
              CALL fortran_message(MESSAGE)
            ENDIF
          ENDIF
        ENDDO
      ENDDO

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 58 of 241

```

ENDDO
C - - Check for Valid Range
DO J = 1,NROWBP
DO I = 1,NCOLBP
IF (VALIDB(I,J) .NE. 0) THEN
IF ((BLADEP(I,J) .LT. 0) .OR. (BLADEP(I,J) .GT. 48)) THEN
ERROR = .TRUE.
CALL lines(2)
MESSAGE =
2 '0*** F A T A L E R R O R *** SUBR. BPCHEK -- '
WRITE(MESSAGE((MCHAR(132,MESSAGE)+1):)
2 , '(' Invalid Blade Position: Location ('))'
LOC = MCHAR(132,MESSAGE)+1
WRITE(MESSAGE(LOC:(LOC+1)), '(I2)') I
WRITE(MESSAGE((MCHAR(132,MESSAGE)+1):), '('', ''')'
LOC = MCHAR(132,MESSAGE)+1
WRITE(MESSAGE(LOC:(LOC+1)), '(I2)') I
WRITE(MESSAGE((MCHAR(132,MESSAGE)+1):), '('')''')'
CALL fortran_message(MESSAGE)
ENDIF
ENDIF
ENDDO
ENDDO
C - - Abort for Assignment Errors - - - - -
IF(ERROR) THEN
CALL lines(2)
CALL abort()
ENDIF
C - - End of Processing - - - - -
RETURN
END

```

Function cbundle

```

FUNCTION CBUNDLE(NCOL, NROW, MAX, MAP, LIST)
C - - - - -
C - - * C B U N D L E * Determines the Number of Unique Fuel As-
C - - assembly Types in the Core
C - - - - -
C - - Argument(s):
C - - NCOL - Number of "Columns" in Core Map (input)
C - - NROW - Number of "Rows" in Core Map (input)
C - - MAX - Maximum Possible Number of Unique Assem- (input)
C - - blies
C - - MAP - Map of Indices for Core Locations (input)
C - - (integer)
C - - LIST - Scratch Array for Accumulating a List of (scratch)
C - - Unique Fuel Assembly Indices
C - - - - -
C - -
C - - Type Statement(s) - - - - -
C - - MATCH - Logical Variable used to Indicate that a Fuel Assembly
C - - Identifier Already exists in the Unique Index List
LOGICAL MATCH

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 59 of 241

```

      INTEGER CBUNDLE
C - - Dimension Statement(s) - - - - -
      DIMENSION LIST(MAX)
      DIMENSION MAP(NCOL,NROW)
C - - Sweep through Map Looking for Unique Fuel Assembly Identifiers -
      NU = 0
      DO J = 1,NROW
        DO I = 1,NCOL
          IF(MAP(I,J) .NE. 0) THEN
            IF(NU .EQ. 0) THEN
              NU = 1
              LIST(NU) = MAP(I,J)
            ELSE
              MATCH = .FALSE.
              DO N = 1,NU
                IF(LIST(N) .EQ. MAP(I,J)) MATCH = .TRUE.
              ENDDO
              IF(.NOT. MATCH) THEN
                NU = NU+1
                LIST(NU) = MAP(I,J)
              ENDIF
            ENDIF
          ENDIF
        ENDDO
      ENDDO
      CBUNDLE = NU
C - - End of Normal Processing - - - - -
      RETURN
      END
  
```

Function clattice

```

      FUNCTION CLATTICE(NAXP1,NBUNDLE,MAX,LVECT,LIST)
C - - - - -
C - - * C L A T T I C E * Determines the Number of Unique Fuel
C - - Lattices in the Core
C - - - - -
C - - Argument(s):
C - - NAXP1 - Number of Axial Nodes in the Core Plus One (input)
C - - NBUNDLE - Number of Unique Fuel Assemblies in the (input)
C - - Core
C - - MAX - Maximum Possible Number of Unique Lat- (input)
C - - tices
C - - LVECT - Vectors Assigning Lattice Indices to Fuel (input)
C - - Assemblies
C - - LIST - Scratch Array for Accumulating a List of (scratch)
C - - Unique Fuel Lattice Indices
C - - - - -
C - -
C - - Type Statement(s) - - - - -
C - - LOGICAL MATCH - Logical Variable used to Indicate that a Fuel Assembly
C - - Identifier Already exists in the Unique Index List
      LOGICAL MATCH
      INTEGER CLATTICE
C - - Dimension Statement(s) - - - - -
  
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 60 of 241

```

        DIMENSION LIST(MAX)
        DIMENSION LVECT(NAXP1,NBUNDLE)
C - - Sweep through Fuel Assembly Loading Vectors Looking for Unique
C - - Lattice Identifiers
        NU = 0
        DO J = 1,NBUNDLE
C - - Note that the first entry for each Fuel Assembly is the Fuel
C - - Assembly Sequential Index
        DO I = 2,NAXP1
            IF(LVECT(I,J) .NE. 0) THEN
                IF(NU .EQ. 0) THEN
                    NU = 1
                    LIST(NU) = LVECT(I,J)
                ELSE
                    MATCH = .FALSE.
                    DO N = 1,NU
                        IF(LIST(N) .EQ. LVECT(I,J)) MATCH = .TRUE.
                    ENDDO
                    IF(.NOT. MATCH) THEN
                        NU = NU+1
                        LIST(NU) = LVECT(I,J)
                    ENDIF
                ENDIF
            ENDIF
        ENDDO
        CLATTICE = NU
C - - End of Normal Processing - - - - -
        RETURN
    END

```

Function echo

```

#include <stdio.h>
#include <string.h>
typedef char ascii_string[133];

void echo(char inname[8], char fin[133]){
/* - - - - -
- - * e c h o * Copies the input file to the output stream
- - - - -
- - Argument(s):
- -     inname - Name for Input FORTRAN NAMELIST           (input)
- -     fin - Name for Input File                          (input)
- - - - -
- - Variable Declaration(s) - - - - -
- - Integer Variables
- -     infind - Flag to Indicate that the FORTRAN NAMELIST Directive
- -             has been Encountered
*/
    unsigned short infind = 0;
/* - Character Variable(s)
- -     c - single character used to read from input file
- -     p - pointer to character variable

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 61 of 241

```

- -      buffer - string variable used to copy files from input file
- - case_title - title for analysis
*/
  unsigned char c;
  char *p;
  char buffer[133] = "";
  extern codenm[8];
  extern ascii_string case_title;
/* - FILE Variable(s)
- - nin - input file
*/
  FILE *nin;
  extern FILE *nout;
/* - - Echo input stream - - - - - - - - - - - - - - - - - - - - */
  nin = fopen(fin,"r");
  lines(4);
  fprintf(nout,"0Input to %s :\n",codenm);
  fprintf(nout,"\n");
/* - Read Title Line and Copy to Output File*/
  p = buffer;
  do
    { c = fgetc(nin);
      *p = c;
      p++;}
  while(c != 10);
  fprintf(nout," ");
  p = buffer;
  c = *p;
  while(c != 10){
    fputc(*p,nout);
    p++;
    c = *p;
  }
  fprintf(nout,"\n");
  strncpy(case_title,buffer,strlen(buffer));
  lines(1);
/* - Read Second Line and Look for FORTRAN NAMELIST Input Directive */
  do
  { strcpy(buffer,"");
    p = buffer;
    do
    { c = fgetc(nin);
      *p = c;
      p++;}
    while((c != 10) && (c != 255));
    if(strstr(buffer,inname) != NULL) infind = 1;
    if(c != 255) fprintf(nout," ");
    if(c != 255){
      p = buffer;
      c = *p;
      while(c != 10){
        fputc(*p,nout);
        p++;
        c = *p;
      }
    }
  }

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 62 of 241

```

    }
    fprintf(nout, "\n");
    lines(1);
  }
}
while(c != 255);
if(!ifind)
{
  lines(4);
  fprintf(nout, "0*** F A T A L   E R R O R   *** Function echo --");
  fprintf(nout, " No Namelist Directive Encountered.\n");
  abort();
}
rewind(nin);
fclose(nin);
return;
}

```

Subroutine Idlv

```

      SUBROUTINE LDLV(NBUNDLE, NAXP1, LU, LVECT)
C - - - - -
C - - * L D L V * Reads Vectors giving Lattice Assignments to Fuel
C - - Assemblies
C - - - - -
C - - Argument(s):
C - - NBUNDLE - Number of Fuel Assemblies in Core (input)
C - - NAXP1 - Number of Axial Nodes in Core +1 (input)
C - - LU - Logical Unit from which to Read Vectors (input)
C - - LVECT - Lattice Loading Vectors for each Fuel (output)
C - - Assembly
C - - - - -
C - -
C - - Type Statement(s) - - - - -
C - - MESSAGE - Character String for Messages
C - - CHARACTER*132 MESSAGE
C - - Dimension Statement(s) - - - - -
C - - DIMENSION LVECT(NAXP1, NBUNDLE)
C - - Skip Header Record - - - - -
C - - READ(LU, '(1X)')
C - - DO NB=1, NBUNDLE
C - - READ(LU, *, END=500) (LVECT(I, NB), I=1, NAXP1)
C - - ENDDO
C - - End of Normal Processing - - - - -
C - - RETURN
C - - Error Processing - - - - -
500 MESSAGE = '0*** F A T A L   E R R O R   *** SUBR. LDLV --'
WRITE(MESSAGE((MCHAR(132, MESSAGE)+1):)
2, '(' Premature End-of-File Encountered on Logical Unit')')
WRITE(MESSAGE((MCHAR(132, MESSAGE)+1):)
2, '(' ', I2)') LU
CALL lines(5)
CALL fortran_message(MESSAGE)
CALL abort()
END

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 63 of 241

Subroutine ldlvr

```

      SUBROUTINE LDLVR(NBUNDLE,NAXP1,LU,LVECT)
C ---
C -- * L D L V * Reads Vectors giving Lattice Assignments to Fuel
C -- Assemblies (Real Version of LDLV)
C ---
C -- Argument(s):
C -- NBUNDLE - Number of Fuel Assemblies in Core (input)
C -- NAXP1 - Number of Axial Nodes in Core +1 (input)
C -- LU - Logical Unit from which to Read Vectors (input)
C -- LVECT - Lattice Loading Vectors for each Fuel (output)
C -- Assembly
C ---
C --
C -- Type Statement(s) ---
C -- MESSAGE - Character String for Messages
      CHARACTER*132 MESSAGE
      REAL LVECT(NAXP1,NBUNDLE)
C -- Skip Header Record ---
      READ(LU,'(1X)')
      DO NB=1,NBUNDLE
         READ(LU,*,END=500) (LVECT(I,NB),I=1,NAXP1)
      ENDDO
C -- End of Normal Processing ---
      RETURN
C -- Error Processing ---
500 MESSAGE = '0*** F A T A L E R R O R *** SUBR. LDLVR --'
      WRITE(MESSAGE((MCHAR(132,MESSAGE)+1):)
2 , '(' Premature End-of-File Encountered on Logical Unit')' )
      WRITE(MESSAGE((MCHAR(132,MESSAGE)+1):)
2 , '(' ' ',I2)') LU
      CALL lines(5)
      CALL fortran_message(MESSAGE)
      CALL abort()
      END

```

Subroutine lodct

```

      SUBROUTINE LODCT(NCOLP,NROWP,NAXP1,NLATTICG,NLATTICM,GMAP,MMAP
2 ,LGVECT,LMVECT,CTABLE)
C ---
C -- * L O D C T * Load Correspondence Table Matching Lattice
C -- Geometry Types with Lattice Material Types
C ---
C -- Argument(s):
C -- NCOLP - Number of Columns in Core Fuel Assembly (input)
C -- Map
C -- NROWP - Number of Rows in Core Fuel Assembly Map (input)
C -- NAXP1 - Number of Axial Nodes + 1 (input)
C -- NLATTICG - Number of Unique Lattice Geometry Types (input)
C -- NLATTICM - Number of Unique Lattice Material Types (input)
C -- GMAP - Fuel Assembly Geometry Index Map (input)
C -- MMAP - Fuel Assembly Material Index Map (input)
C -- LGVECT - Vectors Provides Lattice Geometry Assign- (input)

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 64 of 241

```

C - -          ment Indices for a Given Fuel Assembly
C - -          Geometry Index
C - -          LMVECT - Vectors Provides Lattice Material Assign- (input)
C - -          ment Indices for a Given Fuel Assembly
C - -          Material Index
C - -          CTABLE - Correspondence Table Providing Lattice (output)
C - -          Geometry Index for a Given Lattice Mater-
C - -          ial Index
C - -          -----
C - -
C - - Type Statement(s) -----
C - - MATCH - Flag to Indicate that a Lattice Material/Lattice
C - -          Geometry Pair has Already been Added to CTABLE
          LOGICAL MATCH
          INTEGER GMAP(NCOLP,NROWP),CTABLE(2,NLATTICM)
C - - Dimension Statement(s) -----
          DIMENSION MMAP(NCOLP,NROWP),LGVECT(NAXP1,NLATTICG)
          2 ,LMVECT(NAXP1,NLATTICM)
C - - Sweep Over Entire Core -----
          CALL INVALI((2*NLATTICM),CTABLE,0)
          IC = 0
          DO J = 1,NROWP
            DO I = 1,NCOLP
              IF(MMAP(I,J) .NE. 0) THEN
C - - Examine Each Node of Fuel Assembly Loading Vector
                DO K = 2,NAXP1
                  LGV = LGVECT(K,GMAP(I,J))
                  LMV = LMVECT(K,MMAP(I,J))
                  IF(IC .EQ. 0) THEN
                    IC = IC+1
                    CTABLE(1,IC) = LMV
                    CTABLE(2,IC) = LGV
                  ELSE
                    MATCH = .FALSE.
                    DO N = 1,IC
                      IF((CTABLE(1,N) .EQ. LMV) .AND.
2                      (CTABLE(2,N) .EQ. LGV)) MATCH = .TRUE.
                    ENDDO
                    IF(.NOT. MATCH) THEN
                      IC = IC+1
                      CTABLE(1,IC) = LMV
                      CTABLE(2,IC) = LGV
                    ENDIF
                  ENDIF
                ENDDO
              ENDDO
            ENDDO
          ENDDO
C - - End of Normal Processing -----
          RETURN
          END

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 65 of 241

Subroutine mapchek

```

      SUBROUTINE MAPCHEK(NROW,NCOL,NROWP,NCOLP,VALID,MAP)
C ---
C - - * M A P C H E K * Determines whether Input Fuel Assembly
C - -                      Locations are Valid and if all Valid
C - -                      Locations have Fuel Assemblies Indices
C ---
C - - Argument(s):
C - -     NROW - Number of Rows in Full Core (input)
C - -     NCOL - Number of Columns in Full Core (input)
C - -     NROWP - Number of Rows in Problem (input)
C - -     NCOLP - Number of Columns in Problem (input)
C - -     MAP - Integer Array Containing Fuel Assembly (input)
C - -           Identifier Map
C - -     VALID - Integer Array Containing Valid Locations (input)
C - -           in Core for Fuel Assemblies
C ---
C - -
C - - Type Statement(s) - - - - -
C - -     ERROR - Logical Flag Indicating that Fatal Error has been
C - -           Detected
C - -     MESSAGE - String for Message
      LOGICAL ERROR
      INTEGER VALID(NCOL,NROW)
      CHARACTER*132 MESSAGE
C - - Dimension Statement(s) - - - - -
      DIMENSION MAP(NCOLP,NROWP)
C - - Sweep through Map - - - - -
      ERROR = .FALSE.
      DO J = 1,NROWP
        DO I = 1,NCOLP
          IF(VALID(I,J) .EQ. 0) THEN
            IF(MAP(I,J) .EQ. 0) THEN
              CONTINUE
            ELSE
              ERROR = .TRUE.
              CALL lines(3)
              MESSAGE =
2 '0*** F A T A L E R R O R *** SUBR. MAPCHEK -- Fuel '
              WRITE(MESSAGE((MCHAR(132,MESSAGE)+1):)
2 ,>('Assembly Assignment to Invalid Location ('))
              LOC = MCHAR(132,MESSAGE)+1
              WRITE(MESSAGE(LOC:(LOC+1)),'(I2)') I
              WRITE(MESSAGE((MCHAR(132,MESSAGE)+1):),('',''))
              LOC = MCHAR(132,MESSAGE)+1
              WRITE(MESSAGE(LOC:(LOC+1)),'(I2)') J
              WRITE(MESSAGE((MCHAR(132,MESSAGE)+1):),('')')
              CALL fortran_message(MESSAGE)
              MESSAGE = 'Fuel Assmebly Index = '
              LOC = MCHAR(132,MESSAGE)+1
              WRITE(MESSAGE(LOC:(LOC+2)),'(I3)') MAP(I,J)
              CALL fortran_message(MESSAGE)
            ENDIF
          ENDIF
        END DO
      END DO
      ENDIF

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 66 of 241

```

ELSE
  IF(MAP(I,J) .NE. 0) THEN
    CONTINUE
  ELSE
    ERROR = .TRUE.
    CALL lines(3)
    MESSAGE =
2 '0*** F A T A L E R R O R *** SUBR. MAPCHEK -- No '
    WRITE(MESSAGE((MCHAR(132,MESSAGE)+1):))
2 , '('Fuel Assembly Assignment to Valid Location ('))'
    LOC = MCHAR(132,MESSAGE)+1
    WRITE(MESSAGE(LOC:(LOC+1)), '(I2)') I
    WRITE(MESSAGE((MCHAR(132,MESSAGE)+1):), '('',''))
    LOC = MCHAR(132,MESSAGE)+1
    WRITE(MESSAGE(LOC:(LOC+1)), '(I2)') J
    WRITE(MESSAGE((MCHAR(132,MESSAGE)+1):), '('')')
    CALL fortran_message(MESSAGE)
  ENDIF
ENDIF
ENDDO
ENDDO
C -- Abort for Assignment Errors - - - - -
IF(ERROR) THEN
  CALL lines(2)
  CALL abort()
ENDIF
C -- End of Processing - - - - -
RETURN
END

```

Subroutine r2dmap

```

SUBROUTINE R2DMAP(NCOL, NROW, MAP, LU)
C - - - - -
C - - * R 2 D M A P * Reads Two-dimensional Map giving Fuel
C - - Assembly Locations
C - - - - -
C - - Argument(s):
C - - NCOL - Number of "Columns" in Core Map (input)
C - - NROW - Number of "Rows" in Core Map (input)
C - - MAP - Map of Indices for Core Locations (output)
C - - (integer)
C - - LU - Logical Unit from which to Read Map (input)
C - - - - -
C - -
C - - Type Statement(s) - - - - -
C - - MESSAGE - message to pass to output file
C - - INTEGER MAP(NCOL, NROW)
C - - CHARACTER*132 MESSAGE
C - - - - -
DO J = 1, NROW
  READ(LU, *, END=500) (MAP(I, J), I=1, NCOL)
ENDDO
C - - End of Normal Processing - - - - -

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 67 of 241

```

      RETURN
C -- Error Processing - - - - -
500 CALL lines(3)
      MESSAGE = '0*** F A T A L E R R O R *** SUBR. R2DMAP -- '
      WRITE(MESSAGE((MCHAR(132,MESSAGE)+1):)
2 , '('Premature End-of-File Encountered on Logical Unit ')')
      LOC = MCHAR(132,MESSAGE)
      WRITE(MESSAGE(LOC:(LOC+1)), '(I2)') LU
      CALL fortran_message(MESSAGE)
      CALL abort()
      END

```

Subroutine rblade

```

      SUBROUTINE RBLADE(LENFN, BLADE_DB, LUGEOM, NTUBE, CBSPAN, ATID, ATOD
2 , CBTHICK, TRSPAN, TRTHICK, WSTHICK, CBLENGTH, NCS, CSOFF, CSWIDTH
3 , CBPMAT, ATMAT, CBSMAT, CBTRMAT)
C -- - - - - -
C -- * R B L A D E * Reads Dataset Containing Control Blade
C -- Geometry Dataset
C -- - - - - -
C -- Argument(s):
C -- LENFN - Length of File Name (input)
C -- BLADE_DB - Name of File Containing Appropriate Core- (input)
C -- layout Database
C -- LUGEOM - Logical Unit Number for Blade Geometry (input)
C -- Dataset
C -- NTUBE - Number of Absorber Tubes in All Four (output)
C -- Wings of Blade
C -- CBSPAN - Control Blade Span (Tie Rod Center to (output)
C -- Tip)
C -- ATID - Absorber Tube Inner Diameter (cm) (output)
C -- ATOD - Absorber Tube Outer Diameter (cm) (output)
C -- CBTHICK - Blade Wing Thickness (cm) (output)
C -- TRSPAN - Tie Rod Span (cm) (output)
C -- TRTHICK - Tie Rod Thickness (cm) (output)
C -- WSTHICK - Wing Thickness (cm) (output)
C -- CBLENGTH - Active Absrober Length (cm) (output)
C -- NCS - Number of Central Stiffeners per Wing (output)
C -- CSOFF - Central Stiffener Offset (output)
C -- CSWIDTH - Central Stiffener Width (output)
C -- CBPMAT - Identifier for Blade Poison Material (output)
C -- ATMAT - Identifier for Blade Absorber Tube Mat- (output)
C -- erial
C -- CBSMAT - Identifier for Blade Sheath Material (output)
C -- CBTRMAT - Identifier for Blade Tie Rod Material (output)
C -- - - - - -
C --
C -- Type Statement(s) - - - - -
      CHARACTER*6 CBPMAT, ATMAT, CBSMAT, CBTRMAT
      CHARACTER*(LENFN) BLADE_DB
C -- Namelist Statements - - - - -
      NAMELIST /BLADE/ NTUBE, CBSPAN, ATID, ATOD, CBTHICK, TRSPAN, TRTHICK
2 , CBLENGTH, WSTHICK, NCS, CSOFF, CSWIDTH, CBPMAT, ATMAT, CBSMAT, CBTRMAT
C -- Read in Namelist - - - - -

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 68 of 241

```

      CALL FTOPEN(LUGEOM, BLADE_DB)
C - - Skip Record Containing Dataset Title
      READ(LUGEOM, '(1X)')
C - - Read NAMELIST Input Block
      READ(LUGEOM, BLADE)
C - - Close Dataset
      CALL FTCLOSE(LUGEOM)
C - - End of Normal Processing - - - - -
10 RETURN
      END

```

Subroutine readin

```

      SUBROUTINE READIN(FIN, LENFN, CORE_DB, NIN, LUCGEOM, NAXIAL, AFL, NROW
2 , NCOL, NROWP, NCOLP, APITCH, SOD, STHICK, VOD, VTHICK, TUTPR, TCGR, BLTPR
3 , BCPR, LPREFIX, FPREFIX, CORE_MTLS, MVESSEL, MSHROUD, MTG, MCP, MUTP
4 , MLTP, CORE_F, BLADE_DB, MIGT, DTOD, DTID, RHO, RHOBYP, TEMPK)
C - - - - -
C - - * R E A D I N * Reads NAMELIST input to program -- including
C - - the core geometry file (defined as function
C - - for compatibility with C invocation)
C - - - - -
C - - Argument(s):
C - - FIN - File Name for Input File (input)
C - - LENFN - Length of File Name (input)
C - - CORE_DB - Name of File Containing Appropriate Core- (output)
C - - layout Database
C - - NIN - Logical Unit Number for Input File (input)
C - - LUCGEOM - Logical Unit Number for Core Geometry (input)
C - - Dataset File
C - - NAXIAL - Number of Axial Nodes in Active Fuel (output)
C - - AFL - Active Fuel Length (cm) (output)
C - - NROW - Number of Rows in Full Core (output)
C - - NCOL - Number of Columns in Full Core (output)
C - - NROWP - Number of Rows in Core Represented in the (output)
C - - Problem
C - - NCOLP - Number of Columns in the Core Repr- (output)
C - - sented in the Problem
C - - APITCH - Assembly Pitch (cm) (output)
C - - SOD - Outer Radius of Core Shroud (cm) (output)
C - - STHICK - Core Shroud Thickness (cm) (output)
C - - VOD - Pressure Vessel Outer Radius (cm) (output)
C - - VTHICK - Pressure Vessel Thickness (cm) (output)
C - - TUTPR - Top of the Upper Tie Plate Region (cm) (output)
C - - TCGR - Top of the Core Grid Region (cm) (output)
C - - BLTPR - Bottom of the Lower Tie Plate Region (cm) (output)
C - - BCPR - Bottom of the Core Plate Region (cm) (output)
C - - LPREFIX - File Description Prefix for Lattice Geo- (output)
C - - metry Files
C - - FPREFIX - File Description Prefix for Fuel Materi (output)
C - - al Intermediate Datasets
C - - CORE_MTLS - File Specification for Core Materials (output)
C - - Dataset
C - - MVESSEL - Material Identifier for Vessel (output)
C - - MSHROUD - Material Identifier for Core Shroud (output)

```


Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 70 of 241

Subroutine rlattice

```

SUBROUTINE RLATTICE(LENFN,FUEL_DB,LUGEOM,CTHICK,ASIN,WGAP,NGAP
2 ,CRADIUS,FSRD,CFSRD,RPITCH,COD,CLD,POD,FRCMAT,FCMAT,LATDIM,NWR)
C - - - - -
C - - * R L A T T I C E * Reads Dataset Containing Lattice
C - - Geometry Dataset
C - - - - -
C - - Argument(s):
C - - LENFN - Length of File Name (input)
C - - FUEL_DB - Name of File Containing Appropriate (input)
C - - Lattice-layout Database
C - - CTHICK - Channel Thickness (cm) (output)
C - - ASIN - Inner Span of Channel (cm) (output)
C - - WGAP - Wide Gap Thickness (cm) (output)
C - - NGAP - Narrow Gap Thickness (cm) (output)
C - - CRADIUS - Inner Radius of Channel Corner (cm) (output)
C - - FSRD - Clad Surface to Clad Surface Separation (output)
C - - (cm)
C - - CFSRD - Clad Surface to Inner Channel Surface (output)
C - - Separation (cm)
C - - RPITCH - Fuel Rod Pitch (cm) (output)
C - - COD - Cladding Outer Diameter (cm) (output)
C - - CLD - Cladding Thickness (cm) (output)
C - - POD - Fuel Pellet Outer Diameter (cm) (output)
C - - FRCMAT - Material Identifier for Fuel Rod Cladding (output)
C - - FCMAT - Material Identifier for Channel (output)
C - - LATDIM - Lattice Dimensionality (output)
C - - NWR - Number of Water Rods (output)
C - - - - -
C - -
C - - Type Statement(s) - - - - -
REAL NGAP
CHARACTER*6 FRCMAT,FCMAT
CHARACTER*(LENFN) FUEL_DB
C - - Namelist Statements - - - - -
NAMELIST /FUEL/ CTHICK,ASIN,WGAP,NGAP,CRADIUS,FSRD,CFSRD,RPITCH
2 ,COD,CLD,POD,LATDIM,NWR,FRCMAT,FCMAT
C - - Read in Namelist - - - - -
CALL FTOPEN(LUGEOM,FUEL_DB)
C - - Skip Record Containing Dataset Title
READ(LUGEOM,'(1X)')
C - - Read NAMELIST Input Block
READ(LUGEOM,FUEL)
C - - Close Dataset
CALL FTCLOSE(LUGEOM)
C - - End of Normal Processing - - - - -
10 RETURN
END

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 71 of 241

Subroutine strngr

```

      SUBROUTINE STRNGR(LU, LEN, NSTR, VECTOR)
C ---
C - - * S T R N G R * Reads Character Strings from FORTRAN Logical
C - - Unit
C ---
C - - Argument(s):
C - - LU - Logical Unit from which to Read Character (input)
C - - Strings
C - - LEN - Length of Character Strings (input)
C - - NSTR - Number of FORTRAN Records to Read (input)
C - - VECTOR - Character String Storage Locations (output)
C ---
C - -
C - - Type Statement(s) - - - - -
C - - MESSAGE - Character String for Message
      CHARACTER*132 MESSAGE
      CHARACTER*(LEN) VECTOR(NSTR)
C - - Skip Header Record - - - - -
      READ(LU, '(1X)')
      DO N=1, NSTR
         READ(LU, '(A)', END=500) VECTOR(N)
      ENDDO
C - - End of Normal Processing - - - - -
      RETURN
C - - Error Processing - - - - -
500 MESSAGE = '0*** F A T A L E R R O R *** SUBR. STRNGR --'
      WRITE(MESSAGE((MCHAR(132, MESSAGE)+1):)
2 , '(' Premature End-of-File Encountered on Logical Unit')')
      WRITE(MESSAGE((MCHAR(132, MESSAGE)+1):)
2 , '(' ', I2)') LU
      CALL lines(5)
      CALL fortran_message(MESSAGE)
      CALL abort()
      RETURN
      END

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 72 of 241

2.4. Input Editing Routines

Function bladedb_edt

```

#include <stdio.h>
#include <string.h>

void bladedb_edt(int ntube,float cbspan,float atid,float atod
, float cbthick,float trspan,float trthick,float wsthick
, float cblength,int ncs,float csoff,float cswidth,char cbpmat[]
, char atmat[],char cbsmat[],char cbtrmat[]){
/* -----
- - * b l a d e d b _ e d t * Edits Contents of Control Blade
- -                               Geometry Dataset
- - -----
- - Argument(s):
- -     ntube - number of absorber tubes in control blade      (input)
- -     cbspan - control blade span                             (input)
- -     atid - blade absorber tube inner diameter (cm)         (input)
- -     atod - blade absorber tube outer diameter (cm)         (input)
- -     cbthick - blade wing thickness                         (input)
- -     trspan - blade tie rod span (cm)                       (input)
- -     trthick - blade tie rod thickness (cm)                 (input)
- -     wsthick - blade wing thickness (cm)                   (input)
- -     cblength - active absorber length (cm)                (input)
- -     ncs - number of central stiffeners in control         (input)
- -           blade wing
- -     csoff - blade central stiffener offset (cm)            (input)
- -     cswidth - blade central stiffener width (cm)           (input)
- -     cbpmat - material identifier for control blade poi-    (input)
- -           sen
- -     atmat - material identifier for absorber tube mater-  (input)
- -           ial
- -     cbsmat - material identifier for blade sheath mater-  (input)
- -           ial
- -     cbtrmat - material identifier for blade tie rod mat-  (input)
- -           erial
- - -----
- - Variable Definition(s) -----
- - Character Variable(s)
- -     buffer - string variable used to load entire lines for print
- -     sbuff - short string variable used to load a segment of a line
*/
char sbuff[5];
char buffer[133];
/* - FILE Variable(s)
- - nout - output file
*/
extern FILE *nout;
/* - Summarize NAMELIST Input Directives ----- */
header();
lines(5);
fprintf(nout

```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 73 of 241

```
, "0Dataset Edit -- Values from Blade Geometry Dataset\n");
fprintf(nout, "0 Variable      Value      Definition\n");
fprintf(nout, "\n");
lines(1);
sprintf(buffer, "      NTUBE      %6i", ntube);
bufferpad(buffer, strlen(buffer), 23);
strcat(buffer, "Number of Absorber Tubes in Blade\n");
fprintf(nout, buffer);
lines(1);
sprintf(buffer, "      CBSPAN      %6.2f", cbspan);
bufferpad(buffer, strlen(buffer), 23);
strcat(buffer, "Control Blade Span (cm)\n");
fprintf(nout, buffer);
lines(1);
sprintf(buffer, "      ATID      %6.2f", atid);
bufferpad(buffer, strlen(buffer), 23);
strcat(buffer, "Absorber Tube Inner Diameter (cm)\n");
fprintf(nout, buffer);
lines(1);
sprintf(buffer, "      ATOD      %6.2f", atod);
bufferpad(buffer, strlen(buffer), 23);
strcat(buffer, "Absorber Tube Outer Diameter (cm)\n");
fprintf(nout, buffer);
lines(1);
sprintf(buffer, "      CBTHICK     %6.2f", cbthick);
bufferpad(buffer, strlen(buffer), 23);
strcat(buffer, "Wing Thickness (cm)\n");
fprintf(nout, buffer);
lines(1);
sprintf(buffer, "      TRSPAN     %6.2f", trspan);
bufferpad(buffer, strlen(buffer), 23);
strcat(buffer, "Tie Rod Span\n");
fprintf(nout, buffer);
lines(1);
sprintf(buffer, "      TRTHICK    %6.2f", trthick);
bufferpad(buffer, strlen(buffer), 23);
strcat(buffer, "Tie Rod Thickness (cm)\n");
fprintf(nout, buffer);
lines(1);
sprintf(buffer, "      WSTHICK    %6.2f", wsthick);
bufferpad(buffer, strlen(buffer), 23);
strcat(buffer, "Wing Thickness (cm)\n");
fprintf(nout, buffer);
lines(1);
sprintf(buffer, "      CBLENGTH   %6.2f", cblength);
bufferpad(buffer, strlen(buffer), 23);
strcat(buffer, "Active Absorber Length (cm)\n");
fprintf(nout, buffer);
lines(1);
sprintf(buffer, "      CBPMAT     %s", cbpmat);
bufferpad(buffer, strlen(buffer), 23);
strcat(buffer, "Blade Poison Material\n");
fprintf(nout, buffer);
lines(1);
```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 74 of 241

```
    sprintf(buffer, "    ATMAT      %s", atmat);
    bufferpad(buffer, strlen(buffer), 23);
    strcat(buffer, "Absorber Tube Material\n");
    fprintf(nout, buffer);
    lines(1);
    sprintf(buffer, "    CBSMAT      %s", cbsmat);
    bufferpad(buffer, strlen(buffer), 23);
    strcat(buffer, "Blade Sheath Material\n");
    fprintf(nout, buffer);
    lines(1);
    sprintf(buffer, "    CBTRMAT      %s", cbtrmat);
    bufferpad(buffer, strlen(buffer), 23);
    strcat(buffer, "Blade Tie Rod Material\n");
    fprintf(nout, buffer);
    switch (ncs){
    case 0:
        lines(1);
        sprintf(buffer, "    CSOFF      %6.2f", csoff);
        bufferpad(buffer, strlen(buffer), 23);
        strcat(buffer, "Central Stiffener Offset (cm)\n");
        fprintf(nout, buffer);
        lines(1);
        sprintf(buffer, "    CSWIDTH      %6.2f", cswidth);
        bufferpad(buffer, strlen(buffer), 23);
        strcat(buffer, "Central Stiffener Width (cm)\n");
        fprintf(nout, buffer);
        break;
    case 1:
        break;
    default:
        lines(3);
        fprintf(nout
            , "0*** F a t a l E r r o r *** -- Function");
        fprintf(nout, " bladedb_edt:\n");
        fprintf(nout
            , " NCS must be either Zero or Unity,");
        fprintf(nout, " Value Encountered = %i\n", ncs);
        abort();}
    lines(1);
    fprintf(nout, "\n");
    return;
}
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 75 of 241

Function coredb_edt

```

#include <stdio.h>
#include <string.h>

void coredb_edt(int nrow,int ncol,int nrowb,int ncolb,float apitch
, float vod,float vthick,float sod,float sthick,float tutpr
, float tcgr,float bltpr,float bcpr,int *incore_loc,char mvessel[]
, char mshroud[],char mtg[],char mcp[],char migt[],float dtod
, float dtid){
/* -----
- - * c o r e d b _ e d t * Edits Contents of Core Geometry Data
- -                               set
- - -----
- - Argument(s):
- -     nrow - number of rows in core map for the whole core (input)
- -     ncol - number of columns in core map for the whole (input)
- -           core
- -     nrowb - number of rows of control cell locations in (input)
- -           the core
- -     ncolb - number of columnes of control cell locations (input)
- -           in the core
- -     apitch - fuel assembly pitch (input)
- -     sod - shroud outer radius (input)
- -     sthick - shroud thickness (input)
- -     vod - pressure vessel outer radius (input)
- -     vthick - pressure vessle thickness (input)
- -     tutpr - Top of Upper Tie Plate Region (input)
- -     tcgr - Top of Core Grid Region (input)
- -     bltpr - Bottom of Lower Tie Plate Region (input)
- -     bcpr - Bottom of Core Plate Region (input)
- -     incore_loc - positions for incore instrumentation (input)
- -     mvessel - vessel material identifier (input)
- -     mshroud - core shroud material identifier (input)
- -     mtg - core top guide region material identifier (input)
- -     mcp - core plate region material identifier (input)
- -     migt - incore guide tube material identifier (input)
- -     dtod - outer diameter of incore guide tube (input)
- -     dtid - inner diameter of incore guide tube (input)
- - -----
- -
- - Variable Definition(s) -----
- - Character Variable(s)
- -     buffer - string variable used to load entire lines for print
- -     sbuff - short string variable used to load a segment of a line
*/
char sbuff[5];
char buffer[133];
/* - FILE Variable(s)
- - nout - output file
*/
extern FILE *nout;
/* - Summarize NAMELIST Input Directives ----- */
header();

```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 76 of 241

```
lines(5);
fprintf(nout,"0Dataset Edit -- Values from Core Geometry Dataset\n");
fprintf(nout,"0Variable      Value      Definition\n");
fprintf(nout,"\n");
lines(2);
sprintf(buffer,"      NCOL      %6i",ncol);
bufferpad(buffer,strlen(buffer),22);
strcat(buffer,"Number of Columns of Fuel Assemblies in the Core\n");
fprintf(nout,buffer);
lines(2);
sprintf(buffer,"      NROW      %6i",nrow);
bufferpad(buffer,strlen(buffer),22);
strcat(buffer,"Number of Rows of Fuel Assemblies in the Core\n");
fprintf(nout,buffer);
lines(3);
sprintf(buffer,"      NCOLB      %6i",ncolb);
bufferpad(buffer,strlen(buffer),22);
strcat(buffer,"Number of Columns of Control Blades in Core\n");
fprintf(nout,buffer);
sprintf(buffer,"      NROWB      %6i",nrowb);
bufferpad(buffer,strlen(buffer),22);
strcat(buffer,"Number of Rows of Control Blades in Core\n");
fprintf(nout,buffer);
sprintf(buffer,"      APITCH      %6.2f",apitch);
bufferpad(buffer,strlen(buffer),22);
strcat(buffer,"Fuel Assembly Pitch\n");
fprintf(nout,buffer);
sprintf(buffer,"      VOD      %6.2f",vod);
bufferpad(buffer,strlen(buffer),22);
strcat(buffer,"Vessel Outer Diameter (cm)\n");
fprintf(nout,buffer);
lines(1);
sprintf(buffer,"      VTHICK      %6.2f",vthick);
bufferpad(buffer,strlen(buffer),22);
strcat(buffer,"Vessel Thickness (cm)\n");
fprintf(nout,buffer);
lines(1);
sprintf(buffer,"      SOD      %6.2f",sod);
bufferpad(buffer,strlen(buffer),22);
strcat(buffer,"Core Shroud Outer Diameter (cm)\n");
fprintf(nout,buffer);
lines(1);
sprintf(buffer,"      STHICK      %6.2f",sthick);
bufferpad(buffer,strlen(buffer),22);
strcat(buffer,"Core Shroud Thickness (cm)\n");
fprintf(nout,buffer);
lines(1);
sprintf(buffer,"      TUTPR      %6.2f",tutpr);
bufferpad(buffer,strlen(buffer),22);
strcat(buffer,"Top of the Upper Tie Plate Region (cm)\n");
fprintf(nout,buffer);
lines(1);
sprintf(buffer,"      TCGR      %6.2f",tcgr);
bufferpad(buffer,strlen(buffer),22);
```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 77 of 241

```
strcat(buffer,"Top of the Core Grid Region (cm)\n");
fprintf(nout,buffer);
lines(1);
sprintf(buffer,"  BLTPR      %6.2f",bltpr);
bufferpad(buffer,strlen(buffer),22);
strcat(buffer,"Bottom of the Lower Tie Plate Region (cm)\n");
fprintf(nout,buffer);
lines(1);
sprintf(buffer,"  BCPR      %6.2f",bcpr);
bufferpad(buffer,strlen(buffer),22);
strcat(buffer
,"Bottom of the Fuel Support/Core Plate Region (cm)\n");
fprintf(nout,buffer);
lines(1);
sprintf(buffer,"  DTOD      %6.2f",dtod);
bufferpad(buffer,strlen(buffer),22);
strcat(buffer
,"Outer Diameter of Incore Guide Tube (cm)\n");
fprintf(nout,buffer);
lines(1);
sprintf(buffer,"  DTID      %6.2f",dtid);
bufferpad(buffer,strlen(buffer),22);
strcat(buffer
,"Inner Diameter of Incore Guide Tube (cm)\n");
fprintf(nout,buffer);
/* - Material Identifiers */
lines(1);
sprintf(buffer," MVESSEL      %s",mvessel);
bufferpad(buffer,strlen(buffer),22);
strcat(buffer
,"Material Identifier for Vessel\n");
fprintf(nout,buffer);
lines(1);
sprintf(buffer," MSHROUD      %s",mshroud);
bufferpad(buffer,strlen(buffer),22);
strcat(buffer
,"Material Identifier for Core Shroud\n");
fprintf(nout,buffer);
lines(1);
sprintf(buffer,"  MTG      %s",mtg);
bufferpad(buffer,strlen(buffer),22);
strcat(buffer
,"Material Identifier for Core Top Guide Region\n");
fprintf(nout,buffer);
lines(1);
sprintf(buffer,"  MCP      %s",mcp);
bufferpad(buffer,strlen(buffer),22);
strcat(buffer
,"Material Identifier for Core Plate Region\n");
fprintf(nout,buffer);
lines(1);
sprintf(buffer,"  MIGT      %s",migt);
bufferpad(buffer,strlen(buffer),22);
strcat(buffer
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 78 of 241

```

    ,"Material Identifier for Incore Guide Tube\n");
    fprintf(nout,buffer);
/* - Edit Incore Instrumentation Location Map */
    lines(5+nrowb);
    strcpy(buffer,"");
    bufferpad(buffer,0,132);
    fprintf(nout,"0Incore Instrumentation Location Map\n");
    fprintf(nout," (1 - indicates instrumentation dry tube present)\n");
    fprintf(nout,"\n");
    { short int i,j;
      int *p = incore_loc;
      fprintf(nout,"  ");
      for(i = 1; i <= ncolb; i++){
        short int ib;
        ib = (2*i)-1;
        fprintf(nout," %2i",ib);
      }
      fprintf(nout,"\n");
      fprintf(nout,"\n");
      for(j = 0; j < nrowb; j++){
        short int jb;
        jb = (2*(j+1))-1;
        sprintf(buffer," %2i",jb);
        for(i = 0; i < ncolb; i++){
          if(*p == 0)
            strncat(buffer," ",4);
          else{
            sprintf(sbuff,"%3i",*p);
            strncat(buffer,sbuff,4);
          }
          p++;
        }
        fprintf(nout,buffer);
        fprintf(nout,"\n");
      }
    }
  }
  return;
}

```

Function edit_ct

```

#include <stdio.h>
#include <string.h>

void edit_ct(int nlatticm,int *p){
/* -----
- - *   e d i t   _   c t   *   Edits Contents of Correspondence Table, which
- -                               Relates Lattice Geometry Indices to Lattice
- -                               Material Indices
-----
- - Argument(s):
- -   nlatticm - Number of Unique Lattice Material Indices      (input)
- -           p - Pointer to Correspondence Table                (input)
-----
- -
- -

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 79 of 241

```

- - Variable Declarations - - - - -
- - Integer Variable(s)
- -   n - index for loop
- - p2 - pointer to second member of correspondence table
*/
  int n, *p2;
/* - File Variable(s)
- -   nout - output stream
*/
  extern FILE *nout;
/* - Edit Correspondence Table - - - - - */
  lines(7);
  fprintf(nout,"0Correspondence Table for Lattice Material and Geometry");
  fprintf(nout," Indices\n");
  fprintf(nout,"0Index Material Geometry\n\n");
  p2 = p+1;
  for( n = 1; n <= nlatticm; n++){
    if(n > 7) lines(1);
    fprintf(nout," %3i   %6i   %6i\n",n,*p,*p2);
    p++; p++;
    p2 = p+1;}
}

```

Function edit_spacer

```

#include <stdio.h>
#include <string.h>

typedef char ascii_string[133];

void edit_spacer(int nbundlg,int *ptr_n_spacer
, float *ptr_spacer_location,ascii_string *ptr_spacer_material){
/* - - - - -
- - * e d i t _ s p a c e r * Edits Input Variables Defining
- -                               Fuel Assembly Spacers
- - - - -
- - Argument(s):
- -   nbundlg - number of fuel assembly geometrical types   (input)
- -   ptr_n_spacer
- -             - number of spacers in each fuel assembly type (input)
- -   ptr_spacer_location
- -             - pointer to array containing location of spa- (input)
- -               cers for each fuel assembly type
- -   ptr_spacer_material
- -             - pointer to array containing spacer material (input)
- -               mnemonic for each fue assembly type
- - - - -
- - Variable Definition(s) - - - - -
- - Integer Variable(s)
- -   i - utility index variable
- -   j - utility index variable
- -   ptr_n - pointer to integer vector
*/
  short int i,j;

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 80 of 241

```

    int *ptr_n = ptr_n_spacer;
/* - Float Variable(s)
- - ptr_f - pointer to floating point array
*/
    float *ptr_f = ptr_spacer_location;
/* - Character Variable(s)
- - ptr_c - pointer to ascii_string vector
*/
    ascii_string *ptr_c = ptr_spacer_material;
/* - FILE Variable(s)
- - nout - output file
*/
    extern FILE *nout;
/* - Summarize NAMELIST Input Directives - - - - - */
    header();
    lines(5+nbundlg);
    fprintf(nout
        ,"0Fuel Assembly Spacer Information\n");
/* - Number of Spacers for Each Unique Fuel Assembly Geometry */
    fprintf(nout,"0Index  Number of Spacers  Material Mnemonic\n\n");
    for(i = 1;i <= nbundlg;i++){
        fprintf(nout," %3i          %2i          %s\n"
            ,i,*ptr_n,*ptr_c);
        ptr_n++;
        ptr_c++;}
/* - Spacer Coordinates for Each Type */
    lines(3+nbundlg);
    fprintf(nout,"0Index Axial Distance from BAF (cm)\n\n");
    ptr_n = ptr_n_spacer;
    for(i = 1;i <= nbundlg;i++){
        fprintf(nout," %3i",i);
        for(j = 1;j <= *ptr_n;j++){
            fprintf(nout," %7.3f",*ptr_f);
            *ptr_f++;
        }
        fprintf(nout,"\n");
        ptr_n++;
    }
}

```

Function editin

```

#include<stdio.h>
#include<string.h>
typedef char ascii_string[133];

void editin(char core_db[],char lprefix[],char fprefix[],int naxial
    ,int naxpl,float afl,int nrowp,int ncolp,int nrowbp,int ncolbp
    ,int nbundlg,int nbundlm,int *gmap,int *mmap,int *lgvect,int *lmvect
    ,int *bladep,int nlatticg,int nlatticm,ascii_string *lgdsnam
    ,ascii_string *lmdsnam,char core_mtls[],int core_f,char blade_db[]
    ,float rho,float rhobyp,float tempk,char mutp[],char mltp[]){
/* - - - - -
- - * e d i t i n * Edits Input Directives
- - - - -

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 81 of 241

```

-- Argument(s):
--   core_db - file name for core geometry dataset          (input)
--   lprefix - file specification prefix for lattice geo-   (input)
--             metry datasets
--   fprefix - file specification prefix for fuel material  (input)
--             intermediate datasets
--   naxial  - number of axial nodes in problem            (input)
--   naxpl  - number of axial nodes in core + 1            (input)
--   afl    - active fuel length                          (input)
--   nrowp  - number of rows of fuel in the problem       (input)
--   ncolp  - number of columns of fuel in the problem    (input)
--   ncolbp - number of control blade rows in the problem (input)
--   nbundlg - number of distinct fuel assembly geometries (input)
--             in the core
--   nbundlm - number of distinct fuel assembly material   (input)
--             types in the core
--   gmap   - map for fuel assembly geometrical indices   (input)
--   mmap   - map for fuel assembly material indices      (input)
--   lgvect - vectors mapping fuel lattice indices to fuel (input)
--             assembly indices for geometries
--   lmvect - vectors mapping fuel lattice material indices (input)
--             to fuel assembly material indices
--   bladep - map of control blade positions              (input)
--   nlatticg - number of unique lattice geometries       (input)
--   nlatticm - number of unique lattice materials        (input)
--   lgsdsnam - pointer to character strings holding names of (input)
--             lattices geometry datasets
--   lmsdsnam - pointer to character strings holding names of (input)
--             fuel material intermediate datasets
--   core_mtls - file description for core materials dataset (input)
--   core_f    - fraction of core modeled                  (input)
--             (1 - full, 2 - half, 4 - quarter)
--   blade_db - control blade dataset for problem         (input)
--   rho      - input density of inchannel moderator (g/cc) (input)
--   rhobyp  - input density of bypass moderator (g/cc)   (input)
--   tempk   - problem temperature (Kelvin)               (input)
--   mutp    - material identifier for Upper Tie Plate    (input)
--             Region
--   mltp    - material identifier for Lower Tie Plate    (input)
--             Region
-----
-- Variable Definition(s) -----
-- Character Variable(s)
--   buffer - string variable used to load entire lines for print
--   sbuff  - short string variable used to load a segment of a line
*/
char sbuff[5];
char buffer[133];
/* - FILE Variable(s)
-- nout - output file
*/
extern FILE *nout;
/* - Summarize NAMELIST Input Directives ----- */

```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 82 of 241

```
header();
lines(5);
fprintf(nout,"0Input Edit -- Values from Input Deck Card Images\n");
fprintf(nout,"0 Variable      Value      Definition\n");
fprintf(nout,"\n");
lines(2);
strcpy(buffer,"  CORE_DB");
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Location of Core Geometry Database\n");
fprintf(nout,buffer);
fprintf(nout,"          %s\n",core_db);
lines(2);
strcpy(buffer,"  CORE_MTLS");
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Location of Core Materials Database\n");
fprintf(nout,buffer);
fprintf(nout,"          %s\n",core_mtls);
lines(2);
strcpy(buffer,"  LPREFIX");
bufferpad(buffer,strlen(buffer),23);
strcat(buffer
,"File Specification for Lattice Geometry Datasets\n");
fprintf(nout,buffer);
fprintf(nout,"          %s\n",lprefix);
lines(2);
strcpy(buffer,"  FPREFIX");
bufferpad(buffer,strlen(buffer),23);
strcat(buffer
,"File Specification for Lattice Material Datasets\n");
fprintf(nout,buffer);
strcpy(buffer,"  BLADE_DB");
bufferpad(buffer,strlen(buffer),23);
strcat(buffer
,"File Specification for Blade Geometry Datasets\n");
fprintf(nout,buffer);
fprintf(nout,"          %s\n",fprefix);
lines(4);
sprintf(buffer,"      NROWP      %6i",nrowp);
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Number of Rows in Problem Core Map\n");
fprintf(nout,buffer);
sprintf(buffer,"      NCOLP      %6i",ncolp);
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Number of Columns in Problem Core Map\n");
fprintf(nout,buffer);
sprintf(buffer,"      NAXIAL      %6i",naxial);
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Number of Axial Nodes\n");
fprintf(nout,buffer);
sprintf(buffer,"      AFL        %7.2f",afl);
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Active Core Height (cm)\n");
fprintf(nout,buffer);
sprintf(buffer,"  CORE_F        %6i",core_f);
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 83 of 241

```

bufferpad(buffer, strlen(buffer), 23);
strcat(buffer, "Fraction of Core Modeled:");
switch (core_f){
  case 1:
    strcat(buffer, " full-core problem\n"); break;
  case 2:
    strcat(buffer, " half-core problem\n"); break;
  case 4:
    strcat(buffer, " quarter-core problem\n"); break;
  default:
    lines(3);
    fprintf(nout, "0*** F a t a l E r r o r -- Function");
    fprintf(nout, " editin: Illegal Input for core_f\n");
    fprintf(nout, " Value Encountered = %i\n");
    abort();
}
fprintf(nout, buffer);
lines(2);
sprintf(buffer, "      RHO      %7.5f", rho);
bufferpad(buffer, strlen(buffer), 23);
strcat(buffer, "Inchannel Density (g/cc)\n");
fprintf(nout, buffer);
sprintf(buffer, "      RHOBYP      %7.5f", rhobyp);
bufferpad(buffer, strlen(buffer), 23);
strcat(buffer, "Bypass Density (g/cc)\n");
fprintf(nout, buffer);
lines(1);
sprintf(buffer, "      TEMPK      %7.2f", tempk);
bufferpad(buffer, strlen(buffer), 23);
strcat(buffer, "Problem Temperature (Kelvin)\n");
fprintf(nout, buffer);
lines(1);
sprintf(buffer, "      MUTP      %s", mutp);
bufferpad(buffer, strlen(buffer), 23);
strcat(buffer
, "Material Identifier for Upper Tie Plate Region\n");
fprintf(nout, buffer);
lines(1);
sprintf(buffer, "      MLTP      %s", mltp);
bufferpad(buffer, strlen(buffer), 23);
strcat(buffer
, "Material Identifier for Lower Tie Plate Region\n");
fprintf(nout, buffer);
/* - Edit Fuel Assembly Geometry Assignment Map */
lines(4+nrowp);
fprintf(nout, "0Fuel Assembly Geometry Index Map\n");
fprintf(nout, "\n");
{ short int i, j;
  int *p = gmap;
  fprintf(nout, "  ");
  for(i = 1; i <= ncolp; i++)
    fprintf(nout, " %2i", i);
  fprintf(nout, "\n");
  fprintf(nout, "\n");
}

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 84 of 241

```

    for(j = 0; j < nrowp; j++){
        sprintf(buffer, " %2i", (j+1));
        for(i = 0; i < ncolp; i++){
            if(*p == 0)
                strcat(buffer, "    ", 4);
            else{
                sprintf(sbuff, " %3i", *p);
                strcat(buffer, sbuff, 4);
            }
            p++;
        }
        fprintf(nout, buffer);
        fprintf(nout, "\n");
    }
}
/* - Edit Lattice Geometry Assignment Vectors */
lines(7);
fprintf(nout, "0Lattice Geometry Loading Vectors\n");
fprintf(nout, "0Fuel Assembly Index/Lattice Index\n");
{ short int i, j;
  int *p = lgvect;
  fprintf(nout, " bundle");
  for(i = 1; i <= naxial; i++)
      fprintf(nout, "%5i", i);
  fprintf(nout, "\n");
  fprintf(nout, "\n");
  for(j = 0; j < nbundlg; j++){
      fprintf(nout, " %5i ", *p);
      p++;
      for(i = 1; i < naxpl; i++){
          fprintf(nout, "%5i", *p);
          p++;
      }
      if(i > 1) lines(1);
      fprintf(nout, "\n");
  }
}
}
/* - Edit Lattice Geometry Dataset File Specifications */
lines(5);
strcpy(buffer, "");
bufferpad(buffer, 0, 132);
fprintf(nout, "0Lattice Geometry Datasets\n");
fprintf(nout, "\n");
{ short int i;
  ascii_string *p = lgdsnam;
  for(i = 0; i < nlatticg; i++, p++){
      sprintf(buffer, " %2i ", i+1);
      strcat(buffer, *p, 132);
      if(i > 4) lines(1);
      fprintf(nout, buffer);
      fprintf(nout, "\n");
  }
}
}
/* - Edit Fuel Assembly Material Assignment Map */
lines(4+nrowp);

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 85 of 241

```

strcpy(buffer, "");
bufferpad(buffer, 0, 132);
fprintf(nout, "0Fuel Assembly Material Index Map\n");
fprintf(nout, "\n");
{ short int i, j;
  int *p = mmap;
  fprintf(nout, "    ");
  for(i = 1; i <= ncolp; i++)
    fprintf(nout, " %2i", i);
  fprintf(nout, "\n");
  fprintf(nout, "\n");
  for(j = 0; j < nrowp; j++){
    sprintf(buffer, " %2i", (j+1));
    for(i = 0; i < ncolp; i++){
      if(*p == 0)
        strncat(buffer, "    ", 4);
      else{
        sprintf(sbuff, " %3i", *p);
        strncat(buffer, sbuff, 4);
      }
      p++;
    }
    fprintf(nout, buffer);
    fprintf(nout, "\n");
  }
}
/* - Edit Lattice Material Assignment Vectors */
lines(7);
fprintf(nout, "0Lattice Material Loading Vectors\n");
fprintf(nout, "0Fuel Assembly Index/Lattice Index\n");
{ short int i, j;
  int *p = lmvect;
  fprintf(nout, " bundle");
  for(i = 1; i <= naxial; i++)
    fprintf(nout, "%5i", i);
  fprintf(nout, "\n");
  fprintf(nout, "\n");
  for(j = 0; j < nbundlm; j++){
    fprintf(nout, " %5i ", *p);
    p++;
    for(i = 1; i < naxpl; i++){
      fprintf(nout, "%5i", *p);
      p++;
    }
    if(i > 1) lines(1);
    fprintf(nout, "\n");
  }
}
/* - Edit Lattice Material Intermediate Dataset File Specifications */
lines(5);
strcpy(buffer, "");
bufferpad(buffer, 0, 132);
fprintf(nout, "0Lattice Material Intermediate Datasets\n");
fprintf(nout, "\n");

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 86 of 241

```

    { short int i;
      ascii_string *p = lmdsnam;
      for(i = 0; i < nlatticm; i++, p++){
        sprintf(buffer, " %2i ", i+1);
        strncat(buffer, *p, 132);
        if(i > 4) lines(1);
        fprintf(nout, buffer);
        fprintf(nout, "\n");
      }
  /* - Edit Blade Positioning Map */
  lines(4+nrowbp);
  strcpy(buffer, "");
  bufferpad(buffer, 0, 132);
  fprintf(nout, "0Control Blade Positioning Map\n");
  fprintf(nout, "\n");
  { short int i, j;
    int *p = bladep;
    fprintf(nout, " ");
    for(i = 1; i <= ncolbp; i++){
      short int ib;
      ib = (2*i)-1;
      fprintf(nout, " %2i", ib);
    }
    fprintf(nout, "\n");
    fprintf(nout, "\n");
    for(j = 0; j < nrowbp; j++){
      short int jb;
      jb = (2*(j+1))-1;
      sprintf(buffer, " %2i", jb);
      for(i = 0; i < ncolbp; i++){
        if((*p == 48) | (*p == -1))
          strncat(buffer, " ", 4);
        else{
          sprintf(sbuff, " %3i", *p);
          strncat(buffer, sbuff, 4);
        }
        p++;
      }
      fprintf(nout, buffer);
      fprintf(nout, "\n");
    }
  }
  return;
}

```

Function fgds_edt

```

#include <stdio.h>
#include <string.h>

typedef char ascii_string[133];
typedef struct fuel_gometry{
    ascii_string gds_name;
    int latdim;
    int nwr;

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 87 of 241

```

        float cthick;
        float asin;
        float wgap;
        float ngap;
        float cradius;
        float fsrd;
        float cfsrd;
        float rpitch;
        float cod;
        float cld;
        float pod;
        char frmat[6];
        char fcmat[6];
    } fg_list;

void fgds_edt(int nlatticg,char lprefix[],ascii_string *lgdsnam
,fg_list *ptr_lg_ds){
/* -----
- - fgds_edt - edits contents of fuel geometry datasets
- - -----
- - Argument(s):
- - operation - flag for memory operation to perform          (input)
- - lprefix - prefix for lattice geometry datasets             (input)
- - lgdsnam - file descriptions for lattice geometry datasets (input)
- - ptr_lg_ds - pointer to first element in list of fuel geo- (input)
- - datasets
- - -----
- -
- - Variable Definition(s) - - - - -
- - Integer Variables
- - i - loop control variable
*/
    short int i;
/* - Character Variable(s)
- - fn - filename for lattice geometry dataset
- - buffer - string used to manage output
*/
    ascii_string fn, buffer;
/* - FILE Variable(s)
- - nout - stream for output file
*/
    extern FILE *nout;
/* - Structured Variable(s)
- - p - pointer to current element in list of fuel geometry datasets
- - l - pointer to current element in list of lattice geometry dataset
- - names
*/
    fg_list *p = ptr_lg_ds;
    ascii_string *l = lgdsnam;
/* - Loop Over all the Lattice Geometry Types - - - - - */
    for( i = 1; i <= nlatticg; i++,p++,l++){
        lines(20);
        strcpy(fn,lprefix);
        strcat(fn,*l);

```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 88 of 241

```
fprintf(nout,"0Values from Lattice Geometry Dataset:\n  %s\n",fn);
fprintf(nout,"0 Variable      Value      Definition\n\n");
sprintf(buffer,"      LATDIM      %6i",p->latdim);
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Lattice Dimensionality\n");
fprintf(nout,buffer);
sprintf(buffer,"      NWR      %6i",p->nwr);
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Number of Water Rods\n");
fprintf(nout,buffer);
sprintf(buffer,"      CTHICK      %5.2f",p->cthick);
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Channel Thickness (cm)\n");
fprintf(nout,buffer);
sprintf(buffer,"      ASIN      %5.2f",p->asin);
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Channel Interior Span (cm)\n");
fprintf(nout,buffer);
sprintf(buffer,"      WGAP      %5.2f",p->wgap);
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Wide Gap Half-thickness (cm)\n");
fprintf(nout,buffer);
sprintf(buffer,"      NGAP      %5.2f",p->ngap);
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Narrow Gap Half-thickness (cm)\n");
fprintf(nout,buffer);
sprintf(buffer,"      CRADIUS      %5.2f",p->cradius);
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Channel Corner Inner Radius (cm)\n");
fprintf(nout,buffer);
sprintf(buffer,"      FSRD      %5.2f",p->fsrd);
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Clad Surface to Clad Surface Separation (cm)\n");
fprintf(nout,buffer);
sprintf(buffer,"      CFSRD      %5.2f",p->cfsrd);
bufferpad(buffer,strlen(buffer),23);
strcat(buffer
, "Clad Surface to Inner Channel Surface Separation (cm)\n");
fprintf(nout,buffer);
sprintf(buffer,"      RPITCH      %5.2f",p->rpitch);
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Fuel Rod Pitch (cm)\n");
fprintf(nout,buffer);
sprintf(buffer,"      COD      %5.2f",p->cod);
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Fuel Rod Cladding Outer Diameter (cm)\n");
fprintf(nout,buffer);
sprintf(buffer,"      CLD      %5.2f",p->cld);
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Fuel Rod Cladding Thickness (cm)\n");
fprintf(nout,buffer);
sprintf(buffer,"      POD      %5.2f",p->pod);
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Fuel Pellet Outer Diameter (cm)\n");
```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 89 of 241

```
fprintf(nout,buffer);
sprintf(buffer,"      FRCMAT      %s",p->frpmat);
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Fuel Rod Cladding Material Identifier\n");
fprintf(nout,buffer);
sprintf(buffer,"      FCMAT      %s",p->fcmat);
bufferpad(buffer,strlen(buffer),23);
strcat(buffer,"Fuel Channel Material Identifier\n");
fprintf(nout,buffer);
}
}
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 90 of 241

2.5. Deck Generation Routines

Function add_cell

```

#include <stdio.h>
#include <string.h>

typedef char ascii_string[133];

void bufferpad(char[],int,int);

void add_cell(char label[],char material[],FILE *lu8,int *ncell
,int index,float density,int *nuniverse){
/* -----
- - *   a d d _ c e l l   *   adds cell definition to MCNP model
-----
- - Argument(s):
- -   label - Label for Cell                               (input)
- -   material - Label for Cell Material                   (input)
- -   lu8 - Pointer to Scratch File Stream for Cell Def-   (input)
- -         initions
- -   ncell - Cell Number                                  (input)
- -   index - Index for Cell Material                      (input)
- -   density - Density for Cell Material                  (input)
- -   nuniverse - Universe for Cell                       (input)
-----
- -
- - Variable Declarations - - - - -
- - Character Variable(s)
- -   buffer - buffer for processing lines of output
- -   ptr_buf - pointer to buffer
*/
  ascii_string buffer;
  char *ptr_buf;
/* - FILE Variable(s) - - - - -
- - nout - pointer to output stream
*/
  extern FILE *nout;
/* - Write Cell Definition to MCNP Input Deck - - - - - */
  (*ncell)++;
  fprintf(lu8,"c      %s\n",label);
  if(index != 0)
    fprintf(lu8,"%5i %5i %10.4E",*ncell,index,density);
  else
    fprintf(lu8,"%5i %5i",*ncell,index);
/* - Write Cell Definition to Output File - - - - - */
  sprintf(buffer," %5i",*ncell);
  bufferpad(buffer,strlen(buffer),(30-strlen(material)));
  ptr_buf = buffer+(30-strlen(material));
  sprintf(ptr_buf,"%s/%5i/%10.4E",material,index,density);
  if(*nuniverse != 0){
    ptr_buf = buffer+49*sizeof(char);
    bufferpad(buffer,strlen(buffer),49);
    sprintf(ptr_buf,"%4i",*nuniverse);}

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 91 of 241

```

ptr_buf = buffer+59*sizeof(char);
bufferpad(buffer,strlen(buffer),59);
sprintf(ptr_buf,"%s\n",label);
lines(1);
fprintf(nout,buffer);
}

```

Function add_like_but

```

#include <stdio.h>
#include <string.h>

typedef char ascii_string[133];

void bufferpad(char[],int,int);

void add_like_but(char label[],char material[],FILE *lu8,int *ncell
, char but_string[],int ncref,int *nuniverse){
/* - - - - -
- - *   a d d _ c e l l *   adds cell definition to MCNP model
- - - - -
- - Argument(s):
- -   label - Label for Cell (input)
- -   material - Label for Cell Material (input)
- -   lu8 - Pointer to Scratch File Stream for Cell Def- (input)
- -         initions
- -   ncell - Cell Number (input)
- -   but_string - String Variable Containing Exceptions to (input)
- -         Cloned Cell
- -   ncref - Index of Cell to Clone (input)
- -   nuniverse - Universe for Cell (input)
- - - - -
- - Variable Declarations - - - - -
- - Character Variable(s)
- -   buffer - buffer for processing lines of output
- -   ptr_buf - pointer to buffer
*/
  ascii_string buffer;
  char *ptr_buf;
/* - FILE Variable(s) - - - - -
- -   nout - pointer to output stream
*/
  extern FILE *nout;
/* - Write Cell Definition to MCNP Input Deck - - - - - */
  (*ncell)++;
  fprintf(lu8,"c      %s\n",label);
  fprintf(lu8,"%5i like %i but \n",*ncell,ncref);
  if(strlen(but_string) != 0)
    fprintf(lu8,"      %s\n",but_string);
/* - Write Cell Definition to Output File - - - - - */
  sprintf(buffer," %5i",*ncell);
  bufferpad(buffer,strlen(buffer),(30-strlen(material)));
  ptr_buf = buffer+(30-strlen(material));
  sprintf(ptr_buf,"%s",material,ncref);

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 92 of 241

```

if(*nuniverse != 0){
    ptr_buf = buffer+49*sizeof(char);
    bufferpad(buffer, strlen(buffer), 49);
    sprintf(ptr_buf, "%4i", *nuniverse);}
ptr_buf = buffer+59*sizeof(char);
bufferpad(buffer, strlen(buffer), 59);
sprintf(ptr_buf, "%s\n", label);
lines(1);
fprintf(nout, buffer);
lines(1);
fprintf(nout, "      %s\n", but_string);
}

```

Function add_material

```

#include <stdio.h>
#include <string.h>

typedef char ascii_string[133];
typedef struct s_material{
    struct s_material *last;
    int atomic_number;
    int mass_number;
    float weight_percentage;
    char library_suffix[5];
    struct s_material *next;
} ll_material;

void add_material(FILE *lu10, int *nmaterial, char label[], int n_entries
, ll_material *ptr_mtl) {
/* -----
- - *   a d d _ m a t e r i a l   *   adds material definition
- - -----
- - Argument(s):
- -     lu10 - pointer to scratch stream for MCNP material      (input)
- -           definitions
- -     nmaterial - number of materials in MCNP problem          (i&o)
- -     label - material identifier                             (input)
- -     n_entries - number of entries in material definition     (input)
- -           linked list
- -     ptr_mtl - pointer to linked list for a specific material (input)
- - -----
- - Variable Declarations - - - - -
- - Integer Variable(s)
- -     i - loop index
- -     nloc - location in string
- -     n_print - number of lines printed
- -     nval - number of entries printed on a single line
*/
int i, nloc, n_print = 1, nval = 0;
/* - Character Variable(s)
- -     zaid - MCNP material identifier
- -     cp - pointer to character string

```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 93 of 241

```
- - suffix - local variable for library suffix
*/
char zaid[11];
char *cp;
char suffix[5];
/* - Structured Variable(s) */
ll_material *pm;
/* - Copy Material Definition from Linked List - - - - - */
fprintf(lu10,"c      %s\n",label);
pm = ptr_mtl;
for(i = 1; i <= n_entries; i++){
  if(pm->weight_percentage != 0.0){
    cp = zaid;
    nloc = 3;
    if(pm->atomic_number < 100){
      sprintf(cp,"0");
      cp++;
      nloc--;}
    if(pm->atomic_number < 10){
      sprintf(cp,"0");
      cp++;
      nloc--;}
    sprintf(cp,"%i",pm->atomic_number);
    { short int ip;
      for(ip = nloc; ip > 0; ip--) cp++;
    }
    nloc = 3;
    if(pm->mass_number < 100){
      sprintf(cp,"0");
      cp++;
      nloc--;}
    if(pm->mass_number < 10){
      sprintf(cp,"0");
      cp++;
      nloc--;}
    sprintf(cp,"%i",pm->mass_number);
    { short int ip;
      for(ip = 1; ip <= nloc; ip++) cp++;
    }
    strncpy(suffix,pm->library_suffix,4);
    suffix[4] = '\0';
    sprintf(cp,"%s",suffix);
    if(n_print == 1){
      n_print++;
      nloc = 0;
      if(*nmaterial < 1000) nloc++;
      if(*nmaterial < 100) nloc++;
      if(*nmaterial < 10) nloc++;
      { short int ip;
        for(ip = 1; ip <= nloc; ip++) fprintf(lu10," ");
        fprintf(lu10,"m%i %s %10.4E",*nmaterial,zaid
          ,-(pm->weight_percentage));
        nval++;
      }
    }
  }
}
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 94 of 241

```

    }
    else{
        if(nval == 0) fprintf(lu10, "      ");
        fprintf(lu10, " %s %10.4E",zaid
            ,-(pm->weight_percentage));
        n_print++;
        nval++;
        if(nval == 3){
            fprintf(lu10, "\n");
            nval = 0;
        }
    }
    pm = pm->next;
}
}
if(nval != 0) fprintf(lu10, "\n");
}

```

Function add_surface

```

#include <stdio.h>
#include <string.h>

typedef char ascii_string[133];

typedef struct su_list{
    struct su_list *last;
    int index;
    ascii_string label;
    ascii_string value;
    char mnemonic[4];
    ascii_string equivalent_label;
    struct su_list *next;
} surface_usage_list;

surface_usage_list *load_surface_usage_list(char[],int
, char[],char[],char[],surface_usage_list *);
void search_surface_usage_list(int,char[],int *,char[],char[]
, char[],surface_usage_list *);

void add_surface(int operation,ascii_string label,char mnemonic[4]
,FILE *lu8,FILE *lu9,int *index,ascii_string value
,surface_usage_list *ptr_surface_usage,int *nsurface){
/* -----
- - * a d d _ s u r f a c e * adds surface to MCNP model
-----
- - Argument(s):
- - operation - operation to perform (input)
- -             (0 - add new surface card,
- -             1 - do not add new surface card)
- - label - Label Surface (input)
- - mnemonic - MCNP mnemonic for surface (first edit only) (input)
- - lu8 - Pointer to Scratch File Stream for Cell Def- (input)
- -         initions
- - lu9 - Pointer to Scratch File Stream for Surface (input)

```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 95 of 241

```
--          Definitions
--          index - Index for Surface                      (i|o)
--          value - String Containing Values for Defining Surface (input)
-- ptr_surface_usage
--          - Points to Surface Usage Linked List          (input)
--          nsurface - total number of surfaces            (i|o)
--          -----
--
-- Variable Declarations -----
-- Integer Variable(s)
--          multiplier - multiplier on index for cell card
--          equivalent_index - index for equivalent surface
*/
int multiplier = 1.0, equivalent_index;
/* - Character Variable(s)
--          buffer - buffer for processing lines of output
--          ptr_buf - pointer to buffer
--          equivalent_label - label for surface with identical definition
*/
ascii_string buffer, equivalent_label = "";
char *ptr_buf;
/* - FILE Variable(s) -----
--          nout - pointer to output stream
*/
extern FILE *nout;
/* - Structured Variable(s) -----
--          ptr_sl - pointer to surface usage list
*/
surface_usage_list *ptr_sl;
/* - For New Surfaces, Write to Surface Accumulation Scratch File */
if(*index < 0) {
    multiplier = -1;
    *index = -(*index);}
if(operation == 0){
/* - Search for Equivalent Surface that has already been Defined */
search_surface_usage_list(2,"",&equivalent_index,value,mnemonic
, equivalent_label,ptr_surface_usage);
ptr_sl = ptr_surface_usage;
while(ptr_sl->next != NULL) ptr_sl = ptr_sl->next;
if(equivalent_index == 0){
    (*nsurface)++;
    *index = *nsurface;}
else
    *index = equivalent_index;
ptr_sl = load_surface_usage_list(label,*index,value,mnemonic
, equivalent_label,ptr_sl);
if(equivalent_index == 0){
    fprintf(lu9,"c      %s\n",label);
    fprintf(lu9,"%5i %s %s\n",*index,mnemonic,value);}
}
/* - Write Surface Definition to Cell Card in MCNP Input Deck ----- */
fprintf(lu8," %i", (multiplier*(*index)));
/* - Write to Edit */
sprintf(buffer,"          %6i %s", (multiplier*(*index)),mnemonic);
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 96 of 241

```

bufferpad(buffer, strlen(buffer), 59);
ptr_buf = buffer+59*sizeof(char);
sprintf(ptr_buf, "%s\n", label);
lines(1);
fprintf(fout, buffer);
}

```

Function add_symmetry_surfaces

```

#include <stdio.h>
#include <string.h>

typedef char ascii_string[133];

typedef struct su_list{
    struct su_list *last;
    int index;
    ascii_string label;
    ascii_string value;
    char mnemonic[4];
    ascii_string equivalent_label;
    struct su_list *next;
} surface_usage_list;

void search_surface_usage_list(int, char[], int *, char[], char[], char[]
, surface_usage_list *);
surface_usage_list *load_surface_usage_list(char[], int, char[], char[]
, char[], surface_usage_list *);

void add_symmetry_surfaces(int core_f, FILE *lu8, FILE *lu9
, surface_usage_list *ptr_surface_usage, int colons){
/* -----
- - *   a d d _ s y m m e t r y _ s u r f a c e s   *   adds symmetry sur-
- -                                           faces to MCNP
- -                                           model
-----
- - Argument(s):
- -     core_f - Flag for Fraction of Core in Model           (input)
- -              (0 - full, 2 - half, 4 - quarter)
- -     lu8 - Pointer to Scratch File Stream for Cell Def-   (input)
- -           initions
- -     lu9 - Pointer to Scratch File Stream for Surface   (input)
- -           Definitions
- - ptr_surface_usage
- -       - Pointer to Surface Usage Linked List           (input)
- -     colons - Flag to Indicate that colons should be added (input)
- -              before Surface Indices in Cell Definition
- -              (0 - no
- -                1 - yes)
-----
- -
- - Variable Declarations -----
- - ^ Integer Variable(s)
- -   index - index to surface

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 97 of 241

```

*/
  int index;
/* - Character Variable(s)
  - -   buffer - buffer for processing lines of output
  - -   label - label for surface
  - -   ptr_buf - pointer to buffer
  - -   mnemonic - MCNP mnemonic for surface
  - -   value - surface definition
*/
  char mnemonic[4];
  ascii_string buffer, label, value;
  char *ptr_buf;
/* - FILE Variable(s) - - - - -
  - -   nout - pointer to output stream
*/
  extern FILE *nout;
/* - Structured Variable(s)
  - -   ptr_sl - pointer to surface usage list
*/
  surface_usage_list *ptr_sl;
/* - Write Surface Definition to Cell Card in MCNP Input Deck - - - - */
  strcpy(label,"X-Z Plane");
  search_surface_usage_list(1,label,&index,"","","ptr_surface_usage);
  strcpy(mnemonic,"");
  if(colons == 1) fprintf(lu8,":");
/* - Add Surface for First Occurrence */
  if(index == 0){
    ptr_sl = ptr_surface_usage;
    while(ptr_sl->next != NULL) ptr_sl = ptr_sl->next;
    index = (ptr_sl->index)+1;
    strcpy(mnemonic,"py");
    strcpy(value,"0.0");
    ptr_sl = load_surface_usage_list(label,index,value,mnemonic
      , "", ptr_sl);
    fprintf(lu9,"c      %s\n",label);
    fprintf(lu9," *%5i 0.0\n",index);}
  fprintf(lu8," %i",index);
  sprintf(buffer,"      %6i %s",index,mnemonic);
  bufferpad(buffer,strlen(buffer),59);
  ptr_buf = buffer+59*sizeof(char);
  sprintf(ptr_buf,"%s\n",label);
  lines(1);
  fprintf(nout,buffer);
  if(core_f == 4){
    if(colons == 1) fprintf(lu8,":");
    strcpy(label,"Y-Z Plane");
    search_surface_usage_list(1,label,&index,"","","
      , ptr_surface_usage);
    strcpy(mnemonic,"");
    if(index == 0){
      ptr_sl = ptr_surface_usage;
      while(ptr_sl->next != NULL) ptr_sl = ptr_sl->next;
      index = (ptr_sl->index)+1;
      strcpy(mnemonic,"px");

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 98 of 241

```

        strcpy(value, "0.0");
        ptr_sl = load_surface_usage_list(label, index, value, mnemonic
        , "", ptr_sl);
        fprintf(lu9, "c      %s\n", label);
        fprintf(lu9, " *%5i 0.0\n", index);}
        fprintf(lu8, " %i", (-index));
        sprintf(buffer, "      %6i %s", (-index), mnemonic);
        bufferpad(buffer, strlen(buffer), 59);
        ptr_buf = buffer+59*sizeof(char);
        sprintf(ptr_buf, "%s\n", label);
        lines(1);
        fprintf(nout, buffer);
    }
}

```

Function augment_lattice_list

```

#include <stdio.h>
#include <string.h>

typedef char ascii_string[133];

typedef struct all{
        struct all *last;
        int basis_lattice_material_index;
        int lattice_material_index;
        struct all *next;
    } augmented_lattice_list;

augmented_lattice_list *memory_lattice_list(int
, augmented_lattice_list *);

void augment_lattice_list(int *nlatticm, int *lmvect, int *lgvect
, int nlatticm_ref, int nlatticg, int nbundlm, int nbundlg, int naxial
, int *ptr_n_spacer, int *ptr_spacer_node
, augmented_lattice_list *additional_lattices
, int *ptr_correspondence_table){
/* -----
- - * a u g m e n t _ l a t t i c e _ l i s t * adds lattices incor-
- -                                           porating spacer grids
- -                                           into material lattice
- -                                           loading vectors
-----
- - Argument(s):
- -   nlatticm - total number of unique material lattice indices   (i&o)
- -   lmvect   - material lattice loading vectors                   (i&o)
- -   lgvect   - geometrical lattice loading vectors                (input)
- -   nlatticm_ref
- -           - total number of unique material lattice indi-   (input)
- -             ces without augmentation for lattices incor-
- -             porating spacers
- -   nlatticg - total number of unique geometrical lattice in- (input)
- -             dices
- -   nbundlm  - number of unique fuel assembly materials         (input)

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 99 of 241

```

- -     nbundlg - number of unique fuel assembly geometries      (input)
- -     naxial  - number of axial nodes in problem              (input)
- -     ptr_n_spacer
- -           - number of spacer grids associated with each      (input)
- -             geometrically unique fuel assembly
- -     ptr_spacer_node
- -           - nodal location of spacer grids for each geo-    (input)
- -             metric fuel assembly type
- -     additional_lattices
- -           - linked list of additional lattices generated    (output)
- -             to accomodate spacer grids
- -     ptr_correspondence_table
- -           - table providing relationship between lattice     (input)
- -             material indices and lattice geometrical in-
- -             dices
- -     - - - - -
- -
- - Variable Declarations - - - - -
- - Integer Variable(s)
- -     k - axial index for looping
- -     n - utility loop variable
- -     nglat - lattice geometry index
- -     ngasm - fuel assembly geometry index
- -     lmv - pointer to lattice material index list
- -     lgv - pointer to lattice geometry index list
- -     spacer_present - flag to indicate whether a spacer grid is present
- -                   at this location
*/
short int k,n;
int ngasm, spacer_present, nglat;
int *lmv = lmvect, *lgv;
/* - Structured Variable(s)
- - ptr_al - pointer to additional lattice linked list
*/
augmented_lattice_list *ptr_al = additional_lattices;
/* - FILE Variable(s) - - - - -
- - nout - pointer to output stream
*/
extern FILE *nout;
/* - Sweep through Material Assembly Types - - - - - */
for(n = 1;n <= nbundlm;n++){
/* - Skip Fuel Assembly Index */
    lmv++;
/* - Sweep from Bottom of Assmby to Top */
    for(k = 1;k <= naxial;k++,lmv++){
/* - Determine Corresponding Lattice Geometrical Index */
        { short int i;
          int *p = ptr_correspondence_table;
          while(*p != *lmv) {p++;p++;}
          nglat = *(p+1);
        }
}
/* - Find Fuel Assembly Geometrical Type of which the Lattice Geometry
- - is a Member */
    lgv = lgvect;

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 100 of 241

```

    { short int i,bk,ns;
      int *ptr = ptr_n_spacer;
      int *ptr_n = ptr_spacer_node;
      ngasm = 0;
      for(i = 1;i <= nbundlg;i++)
        if(ngasm == 0){
/* - Skip Fuel Assembly Index */
          lgv++;
          for(bk = 1;bk <= naxial;bk++,lgv++)
            if(*lgv == nglat){
              ngasm = i;
              break;}
        }
      else break;
/* - Compare Nodes at which Spacer Grids are Located with Current Axial
   - - Node Index */
      spacer_present = 0;
      for(i = 1;i < ngasm;i++,ptr++)
        for(ns = 1;ns <= *ptr;ns++,ptr_n++);
      for(bk = 1;bk <= *ptr;bk++,ptr_n++)
        if(k == *ptr_n){
          spacer_present = 1;
          break;}
    }
/* - Determine whether this Lattice Incorporating a Spacer Grid has
   - - already been defined */
    if(spacer_present == 1){
      { short int found = 0, first = 0;
        augmented_lattice_list *ptr_next;
        ptr_al = additional_lattices;
        do{
          if(first != 0) ptr_al = ptr_al->next;
          first = 1;
          if(ptr_al->basis_lattice_material_index == *lmv){
            found = 1;
            *lmv = ptr_al->lattice_material_index;
            break;}
        }
        while(ptr_al->next != NULL);
/* - Spacer Grid not already Defined, Augment Lattice List */
        if(found == 0){
          ptr_al = additional_lattices;
/* - Check for First Entry in Linked List */
          if(ptr_al->lattice_material_index != NULL){
            while(ptr_al->next != NULL) ptr_al = ptr_al->next;
            ptr_next = memory_lattice_list(1,ptr_next);
            ptr_next->last = ptr_al;
            ptr_al->next = ptr_next;
            ptr_al = ptr_next;}
          (*nlatticm)++;
          ptr_al->basis_lattice_material_index = *lmv;
          ptr_al->lattice_material_index = *nlatticm;
          ptr_al->next = NULL;
          *lmv = *nlatticm;

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 101 of 241

```
    }  
  }  
}  }  
}
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV **Page 102 of 241**

Function build_assemblies

```

#include <stdio.h>
#include <string.h>
typedef char ascii_string[133];

typedef struct su_list{
    struct su_list *last;
    int index;
    ascii_string label;
    ascii_string value;
    char mnemonic[4];
    ascii_string equivalent_label;
    struct su_list *next;
} surface_usage_list;

void add_cell(char[],char[],FILE *,int *,int,float,int *);
void add_surface(int,char[],char[],FILE *,FILE *,int *,char[]
, surface_usage_list *,int *);
void lines(int);
void search_surface_usage_list(int,char[],int *,char[],char[]
, char[],surface_usage_list *);

void build_assemblies(int *ncell,int *nuniverse,FILE *lu8,FILE *lu9
, surface_usage_list *ptr_surface_usage,float *ptr_nab,int **ptr_ufa
,int nbundlm,int *lmvect,int naxial,int *ptr_ufl,int *nsurface){
/* -----
- - * b u i l d _ a s s e m b l i e s * builds fuel assemblies from
- -                               constituent lattices
- - -----
- - Argument(s):
- -     ncell - number of cells in MCNP model           (input)
- -     nuniverse - number of "universes" in MCNP model (input)
- -     lu8 - stream pointer for scratch file for cell   (input)
- -           definition accumulation
- -     lu9 - stream pointer for scratch file for surface (input)
- -           definition accumulation
- -     ptr_surface_usage
- -           - linked list for surfaces already defined in (input)
- -             the problem
- -     ptr_nab - vector with axial corrdinates for the tops of (input)
- -             nodes
- -     ptr_ufa - vector with universe indices for unique fuel (output)
- -             assembly types
- -     nbundlm - number of unique fuel assembly types   (input)
- -     lmvect - vector containing assignment of lattice mat- (input)
- -             erial indices to fuel assembly material indi-
- -             ces
- -     naxial - number of axial nodes                   (input)
- -     ptr_ufl - vector with universe indices for unique lat- (input)
- -             tice types
- -     nsurface - total number of surfaces              (input)
- - -----
- -

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 103 of 241

```

- - Variable Definition(s) - - - - -
- - Integer Variable(s)
- -     n - loop variable
- -     k - loop control for axial location
- -     index - utility index variable
- -     lmv - pointer to lmvect array
- -     p - pointer to **ptr_ufa vector
*/
short int n,k;
int index, *lmv = lmvect;
int *p = *ptr_ufa;
/* - Character Variable(s)
- -     label - label for cell, surface or material
- -     mnemonic - mnemonic for surfaces
- -     material - name of material
- -     value - string for defining surface definition
*/
char mnemonic[4];
ascii_string label, material, value;
/* - FILE Variable(s)
- -     nout - pointer to code output file
*/
extern FILE *nout;
/* - Loop Over Unique Fuel Assemblies in the Core - - - - - */
for(n = 1;n <= nbundlm;n++){
    (*nuniverse)++;
    for(k = 1;k <= naxial;k++){
/* - Cells for Axial Nodes */
        sprintf(label,"Axial Node %i for Fuel Assembly %i",k,n);
        strcpy(material,"void");
        add_cell(label,material,lu8,ncell,0,0.0,nuniverse);
/* - Surfaces for Node Boundaries */
        if(k != 1) {
            sprintf(label,"Top of Node %i", (k-1));
            search_surface_usage_list(1,label,&index,"","",
                ,ptr_surface_usage);
            if(index == 0){
                lines(3);
                fprintf(nout,"0*** F a t a l   E r r o r   *** Function");
                fprintf(nout," populate_control_cells --\n");
                fprintf(nout
                    ," Surface not Found in Linked List, label = %s\n"
                    ,label);}
            add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
                ,nsurface);}
        if(k != naxial){
            sprintf(label,"Top of Node %i",k);
            search_surface_usage_list(1,label,&index,"","",
                ,ptr_surface_usage);
            if(index == 0){
                strcpy(mnemonic,"pz");
                { short int n;
                    float *p = ptr_nab;
                    for(n = 1;n < k;n++) p++;

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B000000000-01717-0210-00010 REV 01 Attachment XIV Page 104 of 241

```

        sprintf(value, "%10.4E", *p);
    }
    index = -1;
    add_surface(0, label, mnemonic, lu8, lu9, &index, value
        , ptr_surface_usage, nsurface);
    else{
        index = -index;
        add_surface(1, label, "", lu8, lu9, &index, "", ptr_surface_usage
            , nsurface);
    }
    fprintf(lu8, "\n");
    { short int nn;
      int *l = lmv;
      int *p = ptr_ufl;
      l++;
      for(nn = 1; nn < k; nn++, l++) continue;
      for(nn = 1; nn < *l; nn++, p++) continue;
      fprintf(lu8, "        fill= %i\n", *p);
      fprintf(lu8, "        u= %i imp:n= 1.0\n", *nuniverse);
    }
}
/*    **ptr_ufa = *nuniverse; */
*p = *nuniverse;
p++;
for(k = 1; k <= (naxial+1); k++) lmv++;
}
}

```

Function build_control_blade

```

#include <stdio.h>
#include <string.h>

typedef char ascii_string[133];
typedef struct ascii_record{
        struct ascii_record *last;
        ascii_string line;
        struct ascii_record *next;
    } a_record;

typedef struct s_material{
        struct s_material *last;
        int atomic_number;
        int mass_number;
        float weight_percentage;
        char library_suffix[5];
        struct s_material *next;
    } ll_material;

typedef struct u_list{
        struct u_list *last;
        int index;
        ascii_string label;
        struct u_list *next;
    } usage_list;

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 105 of 241

```

typedef struct su_list{
    struct su_list *last;
    int index;
    ascii_string label;
    ascii_string value;
    char mnemonic[4];
    ascii_string equivalent_label;
    struct su_list *next;
} surface_usage_list;

void add_material(FILE *,int *,char[],int,ll_material *);
void add_cell(char[],char[],FILE *,int *,int,float,int *);
usage_list *load_usage_list(char[],int,usage_list *);
surface_usage_list *load_surface_usage_list(char[],int,char[]
, char[],char[],surface_usage_list *);
ll_material *material_match(a_record *,char[],float *,int *);
int mchar(int *,char[]);
void lines(int);
void abort();
void add_surface(int,char[],char[],FILE *,FILE *,int *,char[]
,surface_usage_list *,int *);
void search_usage_list(int,char[],int *,usage_list *);
void rollup_llm(ll_material *p);

void build_control_blade(int ntube,float cbspan,float atid,float atod
,float cbthick,float trspan,float trthick,float wsthick,float cblength
,int ncs,float csoff,float cswidth,char cbpmat[],char atmat[]
,char cbsmat[],char cbtrmat[],usage_list *ptr_material_usage
,surface_usage_list *ptr_surface_usage,a_record *ptr_core_mtls
,FILE *lu8,FILE *lu9,FILE *lu10,int *nuniverse,int *nmaterial
,float bypass_density,int *ncell,int *ucb,int *nsurface
,int *ncell_tr){
/* -----
-- *   b u i l d _ c o n t r o l _ b l a d e *   builds control blade
--                                           model
-- -----
-- Argument(s):
--     ntube - number of absorber tubes in control blade      (input)
--     cbspan - control blade span                            (input)
--     atid - blade absorber tube inner diameter (cm)        (input)
--     atod - blade absorber tube outer diameter (cm)        (input)
--     cbthick - blade wing thickness                         (input)
--     trspan - blade tie rod span (cm)                       (input)
--     trthick - blade tie rod thickness (cm)                 (input)
--     wsthick - blade wing thickness (cm)                    (input)
--     cblength - active absorber length (cm)                 (input)
--     ncs - number of central stiffeners in control         (input)
--           blade wing
--     csoff - blade central stiffener offset (cm)            (input)
--     cswidth - blade central stiffener width (cm)           (input)
--     cbpmat - identifier for blade poison material          (input)
--     atmat - identifier for absorber tube material          (input)
--     cbsmat - identifier for blade sheath material          (input)

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 106 of 241

```

- -   cbtrmat - identifier for blade tie rod material           (input)
- -   ptr_material_usage
- -       - pointer to linked list of material labels and   (input)
- -       indices
- -   ptr_surface_usage
- -       - pointer to linked list of surface labels and   (input)
- -       indices
- -   ptr_core_matls
- -       - pointer to linked list of core materials       (input)
- -   lu8 - pointer to stream containing scratch file for   (input)
- -       cell definitions
- -   lu9 - pointer to stream containing scratch file for   (input)
- -       surface definitions
- -   lu10 - pointer to stream containing scratch file for  (input)
- -       material definitions
- -   nuniverse - number of MCNP "universes"               (i&o)
- -   nmaterial - number of MCNP materials                 (i&o)
- -   bypass_density -
- -       density of water in bypass region                (input)
- -   ncell - number of MCNP cells (non-translated)        (i&o)
- -   ucb - universe number for control blade               (output)
- -   nsurface - total number of surfaces                   (i&o)
- -   ncell_tr - number of MCNP cell that are translated   (i&o)

```

```

- - - - -
- -

```

- - Variable Declarations - - - - -

- - Integer Variable(s)

```

- -   sztmi - surface index for top of absorber column
- -   szbmi - surface index for bottom of absorber column
- -   spor - surface index for reference boron carbide outer radius
- -   sator - surface index for reference absorber tube outer radius
- -   sratmax - surface index for XAMX for reference absorber tube cell
- -   sratmin - surface index for XMIN for reference absorber tube cell
- -   sratymax - surface index for YMAX for reference absorber tube cell
- -   sratymin - surface index for YMIN for reference absorber tube cell
- -   trs2 - surface index for tie rod surface #2
- -   trs3 - surface index for tie rod surface #3
- -   trs4 - surface index for tie rod surface #4
- -   trs5 - surface index for tie rod surface #5
- -   trs6 - surface index for tie rod surface #6
- -   trs7 - surface index for tie rod surface #7
- -   trs8 - surface index for tie rod surface #8
- -   sambig - surface index for ambiguity surface for search corner
- -   scor - surface index for outer surface of sheath corner
- -   scir - surface index for inner surface of sheath corner
- -   sxminbw - surface index for XMIN for blade window
- -   sxmaxbw - surface index for XMAX for blade window
- -   syminbw - surface index for YMIN for blade window
- -   symaxbw - surface index for YMAX for blade window
- -   crat - cell index for reference absorber tube
- -   ctr - cell index for tie rod
- -   cs - cell index for control blade sheath
- -   cbw - cell index for wing window
- -   ccbn - cell index for wing window cloned to north location

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 107 of 241

```

- -      ccbw - cell index for wing window cloned to west location
- -      ccbs - cell index for wing window cloned to south location
- -      cis - cell index for inside of sheath
- -      uar - universe index for reference absorber tube
- -      us - universe index for sheath and absorber tubes
- -      ublade - universe index for entire blade
- -      n_entries - number of elements/isotopes of which a material is
- -                  composed
- -      index - index of location in list
- -      zero - the value zero
- -      uwing - universe index for reference blade wing
*/
int sztmi, szbmi, spor, sator, sratmax, sratmin, sratymax, sratymin
    , trs2, trs3, trs4, trs5, trs6, trs7, trs8, sambig, scor, scir
    , sxminbw, sxmaxbw, syminbw, symaxbw;
int crat, ctr, cs, cbw, ccbn, ccbw, ccbs, cis;
int uar, us, ublade, uwing;
int n_entries = 0, index, zero = 0;
/* - Float Variable(s) - - - - -
- -      ztmi - position of tip of blade absorber column when blade fully
- -                  inserted (143.75 inches for all BWR's)
- -      zbmi - position of bottom of blade absorber column when blade
- -                  fully inserted
- -      por - outer radius of boron carbide absorber mass
- -      ator - outer radius of absorber tube
- -      xcorn - Inner Surface of Sheath Corner
- -      tfs - absorber tube field span
- -      acmax - XMAX Surface for Reference Absorber Cell
- -      x0 - X-coordinate for center of reference absorber tube
- -      xambig - ambiguity surface for sheath corner
- -      density - density for material
- -      rzero - the value zero
*/
float ztmi = 365.125, zbmi, por, ator, xcorn, tfs, acmax, x0, xambig;
float density, rzero = 0.0;
/* - Character Variable(s)
- -      buffer - buffer for processing lines of output
- -      ptr_buf - pointer to buffer
- -      label - label for cell or surface
- -      value - character representation of surface definition values
- -      mnemonic - identifier for MCNP surface
*/
char mnemonic[4];
ascii_string buffer, label, value;
char *ptr_buf;
/* - FILE Variable(s) - - - - -
- -      nout - pointer to output stream
*/
extern FILE *nout;
/* - Structured Variable(s) - - - - -
- -      ptr_mtl - pointer to linked list for a specific material
- -      ptr_ml - pointer to material usage list
*/
ll_material *ptr_mtl;

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 108 of 241

```

usage_list *ptr_ml;
/* - Compute Coordinates for Blade Generation - - - - - */
zbmi = ztmi-cblength;
por = atid/2;
ator = atod/2;
xcorn = cbspan-wsthick;
tfs = xcorn-trspan;
acmax = (tfs/((float)(ntube/4)))+trspan;
x0 = (acmax+trspan)/2;
xambig = trspan+(((float)(ntube/4))-0.5)*(acmax-trspan);
/* - Create Blade Model in MCNP - - - - - */
/* - Absorber - - - - - */
/* - Cells */
{ int len = 6 , length;
  length = mchar(&len,cbpmat);
  cbpmat[length] = '\0';
}
search_usage_list(1,cbpmat,&index,ptr_material_usage);
ptr_mtl = material_match(ptr_core_mtls,cbpmat,&density,&n_entries);
if(index == 0){
  (*nmaterial)++;
  ptr_ml = ptr_material_usage;
  while(ptr_ml->next != NULL) ptr_ml = ptr_ml->next;
  index = (ptr_ml->index)+1;
  ptr_ml = load_usage_list(cbpmat,index,ptr_ml);}
else
  n_entries = 0;
strcpy(label,"Boron Carbide in Reference Absorber Tube");
(*nuniverse)++;
uar = *nuniverse;
add_cell(label,cbpmat,lu8,ncell,index,-density,nuniverse);
/* - Surfaces */
strcpy(label,"Outer Radius of Absorber");
strcpy(mnemonic,"c/z");
sprintf(value,"%10.4E %10.4E %10.4E",x0,rzero,por);
index = -1;
add_surface(0,label,mnemonic,lu8,lu9,&index,value,ptr_surface_usage
,nsurface);
fprintf(lu8,"\n      imp:n=1.0 u= %i\n",*nuniverse);
/* - Materials */
if(n_entries != 0)
{ add_material(lu10,nmaterial,cbpmat,n_entries,ptr_mtl);
  rollup_llm(ptr_mtl);
  n_entries = 0;}
/* - Absorber Tube - - - - - */
/* - Cells */
{ int len = 6 , length;
  length = mchar(&len,atmat);
  atmat[length] = '\0';
}
search_usage_list(1,atmat,&index,ptr_material_usage);
ptr_mtl = material_match(ptr_core_mtls,atmat,&density,&n_entries);
if(index == 0){
  (*nmaterial)++;

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 109 of 241

```

    ptr_ml = ptr_material_usage;
    while(ptr_ml->next != NULL) ptr_ml = ptr_ml->next;
    index = (ptr_ml->index)+1;
    ptr_ml = load_usage_list(cbpmat, index, ptr_ml);
else
    n_entries = 0;
    strcpy(label, "Reference Absorber Tube");
    add_cell(label, cbpmat, lu8, ncell, index, -density, nuniverse);
/* - Surfaces */
    strcpy(label, "Outer Radius of Absorber Tube");
    strcpy(mnemonic, "c/z");
    sprintf(value, "%10.4E %10.4E %10.4E", x0, rzero, ator);
    index = -1;
    add_surface(0, label, mnemonic, lu8, lu9, &index, value, ptr_surface_usage
, nsurface);
    strcpy(label, "Outer Radius of Absorber");
    search_surface_usage_list(1, label, &index, "", "", ""
, ptr_surface_usage);
    if(index == 0){
        lines(3);
        fprintf(nout, "0*** F a t a l E r r o r *** Function");
        fprintf(nout, " build_control_blade --\n");
        fprintf(nout, " Surface Not Found in Linked List label =");
        fprintf(nout, " %s\n", label);
        abort();
    }
    add_surface(1, label, "", lu8, lu9, &index, "", ptr_surface_usage
, nsurface);
    fprintf(lu8, "\n      imp:n=1.0 u= %i\n", *nuniverse);
/* - Materials */
    if(n_entries != 0)
    { add_material(lu10, nmaterial, atmat, n_entries, ptr_mtl);
      rollup_llm(ptr_mtl);
      n_entries = 0;
    }
/* - Water Outside of Reference Absorber Tube - - - - - */
/* - Cells */
    strcpy(label, "Bypass Water");
    search_usage_list(1, label, &index, ptr_material_usage);
    if(index == 0){
        lines(3);
        fprintf(nout, "0*** F a t a l E r r o r *** Function");
        fprintf(nout, " build_control_blade --\n");
        fprintf(nout, " Material Not Found in Linked List label =");
        fprintf(nout, " %s\n", label);
        abort();
    }
    strcpy(label, "Water Outside Reference Absorber Tube");
    add_cell(label, "Bypass Water", lu8, ncell, index, -bypass_density
, nuniverse);
/* - Surfaces */
    strcpy(label, "Outer Radius of Absorber Tube");
    search_surface_usage_list(1, label, &index, "", "", ""
, ptr_surface_usage);
    if(index == 0){
        lines(3);
        fprintf(nout, "0*** F a t a l E r r o r *** Function");

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 110 of 241

```

    fprintf(nout," build_control_blade --\n");
    fprintf(nout," Surface Not Found in Linked List label =");
    fprintf(nout," %s\n",label);
    abort();}
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
,nsurface);
fprintf(lu8,"\n      imp:n=1.0 u= %i\n",*nuniverse);
/* - Materials */
/* - Bypass Water should already exist as a material */
/* - Window for Reference Absorber Tube - - - - - */
/* - Cells */
(*nuniverse)++;
us = *nuniverse;
strcpy(label,"Bypass Water");
search_usage_list(1,label,&index,ptr_material_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"0*** F a t a l E r r o r *** Function");
    fprintf(nout," build_control_blade --\n");
    fprintf(nout," Material Not Found in Linked List label =");
    fprintf(nout," %s\n",label);
    abort();}
strcpy(label,"Window for Reference Absorber Tube Cell");
add_cell(label,"Bypass Water",lu8,ncell,index,-bypass_density
,nuniverse);
crat = *ncell;
/* - Surfaces */
strcpy(label,"XMAX for Reference Absorber Tube Cell");
strcpy(mnemonic,"px");
sprintf(value,"%10.4E",acmax);
index = -1;
add_surface(0,label,mnemonic,lu8,lu9,&index,value,ptr_surface_usage
,nsurface);
strcpy(label,"XMIN for Reference Absorber Tube Cell");
strcpy(mnemonic,"px");
sprintf(value,"%10.4E",trspan);
index = 1;
add_surface(0,label,mnemonic,lu8,lu9,&index,value,ptr_surface_usage
,nsurface);
strcpy(label,"YMAX for Reference Absorber Tube Cell");
strcpy(mnemonic,"py");
sprintf(value,"%10.4E", (trthick/2));
index = -1;
add_surface(0,label,mnemonic,lu8,lu9,&index,value,ptr_surface_usage
,nsurface);
strcpy(label,"YMIN for Reference Absorber Tube Cell");
strcpy(mnemonic,"py");
sprintf(value,"%10.4E", (-trthick/2));
index = 1;
add_surface(0,label,mnemonic,lu8,lu9,&index,value,ptr_surface_usage
,nsurface);
fprintf(lu8
," \n      imp:n=1.0 u= %i lat= 1 fill= 0:%i 0:0 0:0 %i %ir %i\n"
,*nuniverse,(ntube/4),uar,((ntube/4)-1),*nuniverse);

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 111 of 241

```

/* - Materials */
/* - Bypass Water should already exist as a material */
/* - Tie Rod - - - - - */
/* - Cells */
(*nuniverse)++;
ublade = *nuniverse;
{ int len = 6 , length;
  length = mchar(&len,cbtrmat);
  cbtrmat[length] = '\0';
}
strcpy(label,cbtrmat);
search_usage_list(1,label,&index,ptr_material_usage);
ptr_mtl = material_match(ptr_core_mtls,cbtrmat,&density,&n_entries);
if(index == 0){
  (*nmaterial)++;
  ptr_ml = ptr_material_usage;
  while(ptr_ml->next != NULL) ptr_ml = ptr_ml->next;
  index = (ptr_ml->index)+1;
  ptr_ml = load_usage_list(cbpmat,index,ptr_ml);}
else
  n_entries = 0;
strcpy(label,"Tie Rod");
add_cell(label,cbtrmat,lu8,ncell,index,-density
  ,nuniverse);
ctr = *ncell;
/* - Surfaces */
fprintf(lu8," (");
strcpy(label,"Tie Rod Surface #6");
strcpy(mnemonic,"py");
sprintf(value,"%10.4E",((trthick+wsthick)/2));
index = -1;
add_surface(0,label,mnemonic,lu8,lu9,&index,value,ptr_surface_usage
  ,nsurface);
strcpy(label,"Tie Rod Surface #7");
strcpy(mnemonic,"py");
sprintf(value,"%10.4E",(-(trthick+wsthick)/2));
index = 1;
add_surface(0,label,mnemonic,lu8,lu9,&index,value,ptr_surface_usage
  ,nsurface);
strcpy(label,"XMIN for Reference Absorber Tube Cell");
search_surface_usage_list(1,label,&index,"","",""
  ,ptr_surface_usage);
index = -index;
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
  ,nsurface);
strcpy(label,"Tie Rod Surface #4");
strcpy(mnemonic,"px");
sprintf(value,"%10.4E",(-trspan));
index = 1;
add_surface(0,label,mnemonic,lu8,lu9,&index,value,ptr_surface_usage
  ,nsurface);
strcpy(label,"Top of Absorber Column");
strcpy(mnemonic,"pz");
sprintf(value,"%10.4E",ztmi);

```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 112 of 241

```
index = -1;
add_surface(0, label, mnemonic, lu8, lu9, &index, value, ptr_surface_usage
, nsurface);
strcpy(label, "Bottom of Absorber Column");
strcpy(mnemonic, "pz");
sprintf(value, "%10.4E", zbmi);
index = 1;
add_surface(0, label, mnemonic, lu8, lu9, &index, value, ptr_surface_usage
, nsurface);
fprintf(lu8, "): (");
strcpy(label, "Tie Rod Surface #3");
strcpy(mnemonic, "px");
sprintf(value, "%10.4E", -(trthick+wsthick)/2));
index = 1;
add_surface(0, label, mnemonic, lu8, lu9, &index, value, ptr_surface_usage
, nsurface);
strcpy(label, "Tie Rod Surface #2");
strcpy(mnemonic, "px");
sprintf(value, "%10.4E", ((trthick+wsthick)/2));
index = -1;
add_surface(0, label, mnemonic, lu8, lu9, &index, value, ptr_surface_usage
, nsurface);
strcpy(label, "Tie Rod Surface #5");
strcpy(mnemonic, "py");
sprintf(value, "%10.4E", (-trspan));
index = 1;
add_surface(0, label, mnemonic, lu8, lu9, &index, value, ptr_surface_usage
, nsurface);
strcpy(label, "Tie Rod Surface #8");
strcpy(mnemonic, "py");
sprintf(value, "%10.4E", trspan);
index = -1;
add_surface(0, label, mnemonic, lu8, lu9, &index, value, ptr_surface_usage
, nsurface);
strcpy(label, "Top of Absorber Column");
search_surface_usage_list(1, label, &index, "", "", ""
, ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout, "0*** F a t a l E r r o r *** Function");
    fprintf(nout, " build_control_blade --\n");
    fprintf(nout, " Surface Not Found in Linked List label =");
    fprintf(nout, " %s\n", label);
    abort();}
index = -index;
add_surface(1, label, "", lu8, lu9, &index, "", ptr_surface_usage
, nsurface);
strcpy(label, "Bottom of Absorber Column");
search_surface_usage_list(1, label, &index, "", "", ""
, ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout, "0*** F a t a l E r r o r *** Function");
    fprintf(nout, " build_control_blade --\n");
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 113 of 241

```

    fprintf(nout," Surface Not Found in Linked List label =");
    fprintf(nout," %s\n",label);
    abort();}
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
,nsurface);
fprintf(lu8,"");
fprintf(lu8,"\n      imp:n=1.0 u= %i\n",*nuniverse);
/* - Materials */
if(n_entries != 0)
{ add_material(lu10,nmaterial,atmat,n_entries,ptr_mtl);
  rollup_llm(ptr_mtl);
  n_entries = 0;}
/* - Control Blade Sheath - - - - - */
/* - Cells */
(*nuniverse)++;
uwing = *nuniverse;
{ int len = 6 , length;
  length = mchar(&len,cbsmat);
  cbsmat[length] = '\0';
}
strcpy(label,cbsmat);
search_usage_list(1,label,&index,ptr_material_usage);
ptr_mtl = material_match(ptr_core_mtls,cbsmat,&density,&n_entries);
if(index == 0){
  (*nmaterial)++;
  ptr_ml = ptr_material_usage;
  while(ptr_ml->next != NULL) ptr_ml = ptr_ml->next;
  index = (ptr_ml->index)+1;
  ptr_ml = load_usage_list(cbpsmat,index,ptr_ml);}
else
  n_entries = 0;
strcpy(label,"Control Blade Sheath");
add_cell(label,cbsmat,lu8,ncell,index,-density
,&us);
cs = *ncell;
/* - Surfaces */
fprintf(lu8," (");
strcpy(label,"XMIN for Reference Absorber Tube Cell");
search_surface_usage_list(1,label,&index,"","",""
,ptr_surface_usage);
if(index == 0){
  lines(3);
  fprintf(nout,"0*** F a t a l E r r o r *** Function");
  fprintf(nout," build_control_blade --\n");
  fprintf(nout," Surface Not Found in Linked List label =");
  fprintf(nout," %s\n",label);
  abort();}
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
,nsurface);
strcpy(label,"Ambiguity Surface for Sheath Corner");
strcpy(mnemonic,"px");
sprintf(value,"%10.4E",xambig);
index = -1;
add_surface(0,label,mnemonic,lu8,lu9,&index,value,ptr_surface_usage

```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 114 of 241

```
,nsurface);
strcpy(label,"YMAX for Reference Absorber Tube Cell");
search_surface_usage_list(1,label,&index,"","",""
,ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"0*** F a t a l E r r o r *** Function");
    fprintf(nout," build_control_blade --\n");
    fprintf(nout," Surface Not Found in Linked List label =");
    fprintf(nout," %s\n",label);
    abort();}
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
,nsurface);
strcpy(label,"Tie Rod Surface #6");
search_surface_usage_list(1,label,&index,"","",""
,ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"0*** F a t a l E r r o r *** Function");
    fprintf(nout," build_control_blade --\n");
    fprintf(nout," Surface Not Found in Linked List label =");
    fprintf(nout," %s\n",label);
    abort();}
index = -index;
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
,nsurface);
strcpy(label,"Top of Absorber Column");
search_surface_usage_list(1,label,&index,"","",""
,ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"0*** F a t a l E r r o r *** Function");
    fprintf(nout," build_control_blade --\n");
    fprintf(nout," Surface Not Found in Linked List label =");
    fprintf(nout," %s\n",label);
    abort();}
index = -index;
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
,nsurface);
strcpy(label,"Bottom of Absorber Column");
search_surface_usage_list(1,label,&index,"","",""
,ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"0*** F a t a l E r r o r *** Function");
    fprintf(nout," build_control_blade --\n");
    fprintf(nout," Surface Not Found in Linked List label =");
    fprintf(nout," %s\n",label);
    abort();}
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
,nsurface);
fprintf(lu8,"):(";
strcpy(label,"XMIN for Reference Absorber Tube Cell");
search_surface_usage_list(1,label,&index,"","",""
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 115 of 241

```

,ptr_surface_usage);
if(index == 0){
  lines(3);
  fprintf(nout,"0*** F a t a l E r r o r *** Function");
  fprintf(nout," build_control_blade --\n");
  fprintf(nout," Surface Not Found in Linked List label =");
  fprintf(nout," %s\n",label);
  abort();}
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
,nsurface);
strcpy(label,"Ambiguity Surface for Sheath Corner");
search_surface_usage_list(1,label,&index,"","",""
,ptr_surface_usage);
if(index == 0){
  lines(3);
  fprintf(nout,"0*** F a t a l E r r o r *** Function");
  fprintf(nout," build_control_blade --\n");
  fprintf(nout," Surface Not Found in Linked List label =");
  fprintf(nout," %s\n",label);
  abort();}
index = -index;
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
,nsurface);
strcpy(label,"YMIN for Reference Absorber Tube Cell");
search_surface_usage_list(1,label,&index,"","",""
,ptr_surface_usage);
if(index == 0){
  lines(3);
  fprintf(nout,"0*** F a t a l E r r o r *** Function");
  fprintf(nout," build_control_blade --\n");
  fprintf(nout," Surface Not Found in Linked List label =");
  fprintf(nout," %s\n",label);
  abort();}
index = -index;
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
,nsurface);
strcpy(label,"Tie Rod Surface #7");
search_surface_usage_list(1,label,&index,"","",""
,ptr_surface_usage);
if(index == 0){
  lines(3);
  fprintf(nout,"0*** F a t a l E r r o r *** Function");
  fprintf(nout," build_control_blade --\n");
  fprintf(nout," Surface Not Found in Linked List label =");
  fprintf(nout," %s\n",label);
  abort();}
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
,nsurface);
strcpy(label,"Top of Absorber Column");
search_surface_usage_list(1,label,&index,"","",""
,ptr_surface_usage);
if(index == 0){
  lines(3);
  fprintf(nout,"0*** F a t a l E r r o r *** Function");

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 116 of 241

```

    fprintf(nout, " build_control_blade --\n");
    fprintf(nout, " Surface Not Found in Linked List label =");
    fprintf(nout, " %s\n", label);
    abort();}
index = -index;
add_surface(1, label, "", lu8, lu9, &index, "", ptr_surface_usage
, nsurface);
strcpy(label, "Bottom of Absorber Column");
search_surface_usage_list(1, label, &index, "", "", ""
, ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout, "0*** F a t a l E r r o r *** Function");
    fprintf(nout, " build_control_blade --\n");
    fprintf(nout, " Surface Not Found in Linked List label =");
    fprintf(nout, " %s\n", label);
    abort();}
add_surface(1, label, "", lu8, lu9, &index, "", ptr_surface_usage
, nsurface);
fprintf(lu8, "):\n      (");
strcpy(label, "Ambiguity Surface for Sheath Corner");
search_surface_usage_list(1, label, &index, "", "", ""
, ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout, "0*** F a t a l E r r o r *** Function");
    fprintf(nout, " build_control_blade --\n");
    fprintf(nout, " Surface Not Found in Linked List label =");
    fprintf(nout, " %s\n", label);
    abort();}
add_surface(1, label, "", lu8, lu9, &index, "", ptr_surface_usage
, nsurface);
strcpy(label, "Inner Surface for Sheath Corner");
strcpy(mnemonic, "c/z");
sprintf(value, "%10.4E %10.4E %10.4E", xambig, rzero, (trthick/2));
index = 1;
add_surface(0, label, mnemonic, lu8, lu9, &index, value, ptr_surface_usage
, nsurface);
strcpy(label, "Outer Surface for Sheath Corner");
strcpy(mnemonic, "c/z");
sprintf(value, "%10.4E %10.4E %10.4E", xambig, rzero, ((trthick+wsthick)/2));
index = -1;
add_surface(0, label, mnemonic, lu8, lu9, &index, value, ptr_surface_usage
, nsurface);
strcpy(label, "Top of Absorber Column");
search_surface_usage_list(1, label, &index, "", "", ""
, ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout, "0*** F a t a l E r r o r *** Function");
    fprintf(nout, " build_control_blade --\n");
    fprintf(nout, " Surface Not Found in Linked List label =");
    fprintf(nout, " %s\n", label);
    abort();}

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 117 of 241

```

index = -index;
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
,nsurface);
strcpy(label,"Bottom of Absorber Column");
search_surface_usage_list(1,label,&index,"","",""
,ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"0*** F a t a l E r r o r *** Function");
    fprintf(nout," build_control_blade --\n");
    fprintf(nout," Surface Not Found in Linked List label =");
    fprintf(nout," %s\n",label);
    abort();}
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
,nsurface);
fprintf(lu8," ");
fprintf(lu8,"\n      imp:n=1.0 u= %i\n",uwing);
/* - Materials */
if(n_entries != 0)
{ add_material(lu10,nmaterial,atmat,n_entries,ptr_mtl);
  rollup_llm(ptr_mtl);
  n_entries = 0;}
/* - Inside of Control Blade Sheath */
/* - Cell */
strcpy(label,"Bypass Water");
search_usage_list(1,label,&index,ptr_material_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"0*** F a t a l E r r o r *** Function");
    fprintf(nout," build_control_blade --\n");
    fprintf(nout," Material Not Found in Linked List label =");
    fprintf(nout," %s\n",label);
    abort();}
strcpy(label,"Inside of Blade Wing");
add_cell(label,"Bypass Water",lu8,ncell,index,-bypass_density
,&uwing);
cis = *ncell;
/* - Surfaces */
fprintf(lu8," (");
strcpy(label,"XMIN for Sheath Interior");
strcpy(mnemonic,"px");
sprintf(value,"%10.4f", (trspan+0.001));
index = 1;
add_surface(0,label,mnemonic,lu8,lu9,&index,value
,ptr_surface_usage,nsurface);
strcpy(label,"YMIN for Sheath Interior");
strcpy(mnemonic,"py");
sprintf(value,"%10.4f", (- (trthick/2)+0.001));
index = 1;
add_surface(0,label,mnemonic,lu8,lu9,&index,value
,ptr_surface_usage,nsurface);
strcpy(label,"YMAX for Sheath Interior");
strcpy(mnemonic,"py");
sprintf(value,"%10.4f", ((trthick/2)-0.001));

```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 118 of 241

```
index = -1;
add_surface(0,label,mnemonic,lu8,lu9,&index,value
,ptr_surface_usage,nsurface);
strcpy(label,"Ambiguity Surface for Sheath Corner");
search_surface_usage_list(1,label,&index,"","",
,ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"0*** F a t a l E r r o r *** Function");
    fprintf(nout," build_control_blade --\n");
    fprintf(nout," Surface Not Found in Linked List label =");
    fprintf(nout," %s\n",label);
    abort();}
index = -index;
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
,nsurface);
strcpy(label,"Top of Absorber Column");
search_surface_usage_list(1,label,&index,"","",
,ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"0*** F a t a l E r r o r *** Function");
    fprintf(nout," build_control_blade --\n");
    fprintf(nout," Surface Not Found in Linked List label =");
    fprintf(nout," %s\n",label);
    abort();}
index = -index;
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
,nsurface);
strcpy(label,"Bottom of Absorber Column");
search_surface_usage_list(1,label,&index,"","",
,ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"0*** F a t a l E r r o r *** Function");
    fprintf(nout," build_control_blade --\n");
    fprintf(nout," Surface Not Found in Linked List label =");
    fprintf(nout," %s\n",label);
    abort();}
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
,nsurface);
fprintf(lu8," ):( ");
strcpy(label,"Inner Surface for Sheath Corner");
search_surface_usage_list(1,label,&index,"","",
,ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"0*** F a t a l E r r o r *** Function");
    fprintf(nout," build_control_blade --\n");
    fprintf(nout," Surface Not Found in Linked List label =");
    fprintf(nout," %s\n",label);
    abort();}
index = -index;
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 119 of 241

```

    ,nsurface);
strcpy(label,"Top of Absorber Column");
search_surface_usage_list(1,label,&index,"","",""
    ,ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"0*** F a t a l E r r o r *** Function");
    fprintf(nout," build_control_blade --\n");
    fprintf(nout," Surface Not Found in Linked List label =");
    fprintf(nout," %s\n",label);
    abort();}
index = -index;
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
    ,nsurface);
strcpy(label,"Bottom of Absorber Column");
search_surface_usage_list(1,label,&index,"","",""
    ,ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"0*** F a t a l E r r o r *** Function");
    fprintf(nout," build_control_blade --\n");
    fprintf(nout," Surface Not Found in Linked List label =");
    fprintf(nout," %s\n",label);
    abort();}
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
    ,nsurface);
fprintf(lu8," )\n");
fprintf(lu8,"          u= %i fill= %i imp:n= 1.0\n",uwing,us);
/* - Region Outside of Blade Wing */
/* - Cell */
strcpy(label,"Bypass Water");
search_usage_list(1,label,&index,ptr_material_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"0*** F a t a l E r r o r *** Function");
    fprintf(nout," build_control_blade --\n");
    fprintf(nout," Material Not Found in Linked List label =");
    fprintf(nout," %s\n",label);
    abort();}
strcpy(label,"Region Outside of Blade Wing");
add_cell(label,"Bypass Water",lu8,ncell,index,-bypass_density
    ,&uwing);
/* - Surfaces */
fprintf(lu8," #i #i",cs,cis);
fprintf(lu8,"\n          imp:n=1.0 u= %i\n",uwing);
sprintf(buffer,"          #i #i",cs,cis);
bufferpad(buffer,strlen(buffer),59);
ptr_buf = buffer+59*sizeof(char);
sprintf(ptr_buf,"Definition formed by Complement Function\n");
lines(1);
fprintf(nout,buffer);
/* - Window for Wing of Blade */
/* - Cell */
strcpy(label,"Bypass Water");

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 120 of 241

```

search_usage_list(1,label,&index,ptr_material_usage);
if(index == 0){
  lines(3);
  fprintf(nout,"0*** F a t a l E r r o r *** Function");
  fprintf(nout," build_control_blade --\n");
  fprintf(nout," Material Not Found in Linked List label =");
  fprintf(nout," %s\n",label);
  abort();}
strcpy(label,"Window for Wing of Blade");
add_cell(label,"Bypass Water",lu8,ncell_tr,index,-bypass_density
, nuniverse);
cbw = *ncell_tr;
/* - Surfaces */
strcpy(label,"XMIN for Blade Window");
strcpy(mnemonic,"px");
sprintf(value,"%10.4E", (trspan+0.01));
index = 1;
add_surface(0,label,mnemonic,lu8,lu9,&index,value,ptr_surface_usage
,nsurface);
strcpy(label,"XMAX for Blade Window");
strcpy(mnemonic,"px");
sprintf(value,"%10.4E", (cbspan+0.01));
index = -1;
add_surface(0,label,mnemonic,lu8,lu9,&index,value,ptr_surface_usage
,nsurface);
strcpy(label,"YMIN for Blade Window");
strcpy(mnemonic,"py");
sprintf(value,"%10.4E", (- (trthick+wsthick+0.01)));
index = 1;
add_surface(0,label,mnemonic,lu8,lu9,&index,value,ptr_surface_usage
,nsurface);
strcpy(label,"YMAX for Blade Window");
strcpy(mnemonic,"py");
sprintf(value,"%10.4E", (trthick+wsthick+0.01));
index = -1;
add_surface(0,label,mnemonic,lu8,lu9,&index,value,ptr_surface_usage
,nsurface);
fprintf(lu8," fill= %i",uwing);
fprintf(lu8,"\n      imp:n=1.0 u= %i\n",ublade);
lines(1);
fprintf(nout,"      Filling Universe = %i\n",uwing);
/* - Materials */
/* - Bypass Water should already exist as a material */
/* - Wing Cloned to North Position - - - - - */
/* - Cell */
(*ncell_tr)++;
cchn = *ncell_tr;
strcpy(label,"Wing Cloned to North Position");
fprintf(lu8,"c      %s\n",label);
fprintf(lu8,"%5i",*ncell_tr);
fprintf(lu8," like %i but",cbw);
fprintf(lu8
," *trcl=( 0 0 0 90 0 90 180 90 90 90 0)\n");
fprintf(lu8,"      u= %i\n",ublade);

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 121 of 241

```

sprintf(buffer, " %5i", *ncell_tr);
bufferpad(buffer, strlen(buffer), 59);
ptr_buf = buffer+59*sizeof(char);
sprintf(ptr_buf, "%s\n", label);
lines(2);
fprintf(nout, buffer);
fprintf(nout, "      Cloning Directive:");
fprintf(nout, " like %i but", cbw);
fprintf(nout
, " *trcl=( 0 0 0 90 0 90 180 90 90 90 90 0)\n");
/* - Wing Cloned to West Position - - - - - */
/* - Cell */
(*ncell_tr)++;
ccbws = *ncell_tr;
strcpy(label, "Wing Cloned to West Position");
fprintf(lu8, "c      %s\n", label);
fprintf(lu8, "%5i", *ncell_tr);
fprintf(lu8, " like %i but", cbw);
fprintf(lu8
, " *trcl=( 0 0 0 180 90 90 -90 180 90 90 90 0)\n");
fprintf(lu8, "      u= %i\n", ublade);
sprintf(buffer, " %5i", *ncell_tr);
bufferpad(buffer, strlen(buffer), 59);
ptr_buf = buffer+59*sizeof(char);
sprintf(ptr_buf, "%s\n", label);
lines(2);
fprintf(nout, buffer);
fprintf(nout, "      Cloning Directive:");
fprintf(nout, " like %i but", cbw);
fprintf(nout
, " *trcl=( 0 0 0 180 90 90 -90 180 90 90 90 0)\n");
/* - Wing Cloned to South Position - - - - - */
/* - Cell */
(*ncell_tr)++;
ccbs = *ncell_tr;
strcpy(label, "Wing Cloned to South Position");
fprintf(lu8, "c      %s\n", label);
fprintf(lu8, "%5i", *ncell_tr);
fprintf(lu8, " like %i but", cbw);
fprintf(lu8
, " *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0)\n");
fprintf(lu8, "      u= %i\n", ublade);
sprintf(buffer, " %5i", *ncell_tr);
bufferpad(buffer, strlen(buffer), 59);
ptr_buf = buffer+59*sizeof(char);
sprintf(ptr_buf, "%s\n", label);
lines(2);
fprintf(nout, buffer);
fprintf(nout, "      Cloning Directive:");
fprintf(nout, " like %i but", cbw);
fprintf(nout
, " *trcl=( 0 0 0 -90 180 90 0 -90 90 90 90 0)\n");
/* - Region Outside of Blade - - - - - */
/* - Cell */

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 122 of 241

```

strcpy(label,"Bypass Water");
search_usage_list(1,label,&index,ptr_material_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"0*** F a t a l E r r o r *** Function");
    fprintf(nout," build_control_blade --\n");
    fprintf(nout," Material Not Found in Linked List label =");
    fprintf(nout," %s\n",label);
    abort();}
strcpy(label,"Region Outside of Blade");
add_cell(label,"Bypass Water",lu8,ncell,index,-bypass_density
    ,&ublade);
/* - Surfaces */
fprintf(lu8," %i %i %i %i %i",ctr,cbw,ccbn,ccb,ccbs);
fprintf(lu8,"\n      imp:n=1.0 u= %i\n",ublade);
sprintf(buffer,"          %i %i %i %i %i",ctr,cbw,ccbn,ccb,ccbs);
bufferpad(buffer,strlen(buffer),59);
ptr_buf = buffer+59*sizeof(char);
sprintf(ptr_buf,"Definition formed by Complement Function\n");
lines(1);
fprintf(nout,buffer);
*uch = ublade;
}

```

Function build_control_cells

```

#include <stdio.h>
#include <string.h>
#include <math.h>

typedef char ascii_string[133];
typedef struct ascii_record{
    struct ascii_record *last;
    ascii_string line;
    struct ascii_record *next;
} a_record;

typedef struct s_material{
    struct s_material *last;
    int atomic_number;
    int mass_number;
    float weight_percentage;
    char library_suffix[5];
    struct s_material *next;
} ll_material;

typedef struct u_list{
    struct u_list *last;
    int index;
    ascii_string label;
    struct u_list *next;
} usage_list;

typedef struct su_list{
    struct su_list *last;

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 123 of 241

```

        int index;
        ascii_string label;
        ascii_string value;
        char mnemonic[4];
        ascii_string equivalent_label;
        struct su_list *next;
    } surface_usage_list;

void add_cell(char[],char[],FILE *,int *,int,float,int *);
void add_surface(int,char[],char[],FILE *,FILE *,int *
, char[],surface_usage_list *,int *);
usage_list *load_usage_list(char[],int,usage_list *);
surface_usage_list *load_surface_list(char[],int,char[]
, char[],char[],surface_usage_list *);
void search_usage_list(int,char[],int *,usage_list *);
void search_surface_usage_list(int,char[],int *,char[]
, char[],char[],surface_usage_list *);
void bufferpad(char[],int,int);
ll_material *material_match(a_record *,char[],float *,int *);
void add_material(FILE *,int *,char[],int,ll_material *);
int *memory_integer(int,int,int *);
void abort();
void lines(int);
int mchar(int *,char[]);

void build_control_cells(int *ncell,int *nuniverse,int ncolcc
, int nrowcc,int *ccmap,int *ccmapw,int ncc,FILE *lu8,FILE *lu9
, float blade_window_offset,surface_usage_list *ptr_surface_usage
, float apitch,int ucb,float cbspan,float cb_stroke,char migt[]
, usage_list *ptr_material_usage,a_record *ptr_core_mtls
, int *nmaterial,int *ptr_ufa,FILE *lu10,float bypass_density
, float dtid,float dtod,int nu_cc,int *fau_fill,int *nsurface){
/* -----
- - *   b u i l d _ c o n t r o l _ c e l l s   *   creates universes for
- -                                           unique control cells
- - -----
- - Argument(s):
- -     ncell - Number of Cells in MCNP Model           (input)
- -     nuniverse - Number of "Universes" in MCNP Model (input)
- -     ncolcc - Number of Columns in Control Cell Map   (input)
- -     nrowcc - Number of Rows in Control Cell Map     (input)
- -     ccmap - Map of Control Cell Universe Numbers    (input)
- -     ccmaw - Map Containing Contents of each Control Cell (input)
- -     ncc - Number of Unique Control Cells            (input)
- -     lu8 - Stream Pointer for Scratch File for Cell Def- (input)
- -           initions
- -     lu9 - Stream Pointer for Scratch File for Surface (input)
- -           Definitions
- -     blade_window_offset
- -           - sizing of windows for control blades and fuel (input)
- -             assemblies in control cell template
- -     ptr_surface_usage
- -           - linked list for surfaces that have already (input)
- -             been defined

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV **Page 124 of 241**

```

--      apitch - assembly pitch                (input)
--      ucb - control blade universe           (input)
--      cbspan - control blade wing span (cm)  (input)
--      cb_stroke - control blade stroke (cm)  (input)
--      migt - guide tube material             (input)
--      ptr_material_usage
--          - linked list for materials that have already (input)
--          been defined
--      ptr_core_mtls
--          - linked list for core materials         (input)
--      nmaterial - number of material definitions (input)
--      ptr_ufa - vector of MNCP "universes" assigned to each (input)
--          fuel assembly
--      lul0 - Stream Pointer for Scratch File for Material (input)
--          Definitions
--      bypass_density
--          - density for water in bypass           (input)
--      dtid - inner diameter of guide tube (cm)  (input)
--      dtod - outer diameter of guide tube (cm)  (input)
--      nu_cc - "universe" index for first control cell (input)
--      fau_fill - array providing fuel assembly universe num- (output)
--          bers filling each unique control cell uni-
--          verse
-----
--
-- Variable Declarations -----
-- Integer Variable(s)
--      i - loop variable for columns in map
--      j - loop variable for rows in map
--      n - utility loop variable
--      index - index for cells and surfaces
--      ptr_map - pointer to ccmmap array
--      ptr_mapw - pointer to ccmmapw array
--      pcc - pointer to list of unique control cells to track whether
--          they have already been defined
--      pc - rotating pointer for pcc
--      n_entries - number of entries in material definition linked list
*/
short int i,j,n;
int index, n_entries;
int *ptr_map = ccmmap, *ptr_mapw = ccmmapw;
int *pcc, *pc;
/* - Float Variable(s)
--      density - material density
--      delta - distance from maximum radial extent of curved corner
--          of fuel assembly window to edge of window
--      delta_0 - offset for fuel assembly box from control cell
--          boundary
--      R - radius of curved corner of fuel assembly window
--      Xc - x-coordinate of cylinder defining curved corner of
--          fuel assembly window
--      Yc - Y-coordinate of cylinder defining curved corner of
--          fuel assembly window
*/

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 125 of 241

```

float density, delta, delta_0 = 0.1, R, Xc, Yc;
/* - Character Variable(s)
- - label - label for cells and surfaces
- - material - label for material
- - mnemonic - MCNP surface mnemonic
- - value - surface definition vector
- - buffer - string variable for output management
- - ptr_buf - pointer to buffer
*/
char mnemonic[4];
char *ptr_buf;
ascii_string label, material, value, buffer;
/* - FILE Variable(s)
- - nout - stream pointer to output file
*/
extern FILE *nout;
/* - Structured Variable(s)
- - ptr_mtl - pointer to linked list for a specific material
- - ptr_ml - pointer to material usage list
*/
ll_material *ptr_mtl;
usage_list *ptr_ml;
/* - Fix Up Guide Tube Material Definition for FORTRAN processing - - */
{ int len = 6, length;
length = mchar(&len,migt);
migt[length] = '\0';
}
/* - Compute coordinates for curved corner of fuel assembly window - - */
delta = -delta_0+(sqrt(2.0)/2.0)*((dtod/2)+delta_0);
/* - 2.4142 is equivalent to tand(67.5) */
R = sqrt(2.0)*delta*2.4142;
Xc = (apitch - delta_0) - R;
Yc = -(apitch - delta_0) + R;
/* - Set Up Tracking for Control Cell Creation - - - - - */
pcc = memory_integer(1,ncc,pcc);
/* - Sweep Over Map - - - - - */
for(j = 1;j <= ncolcc;j++){
for(i = 1;i <= nrowcc;i++,ptr_map++){
/* - Check for Null Control Cell */
if(*ptr_map == 0){
for(n = 1;n <= 6;n++) ptr_mapw++;
continue;}
pc = pcc;
for(n = 1;n < (*ptr_map-nu_cc+1);n++) pc++;
/* - Check whether the Control Cell that has already been Modeled */
if(*pc == 1) {
for(n = 1;n <= 6;n++) ptr_mapw++;
continue;}
*pc = 1; (*nuniverse)++;
/* - Build Components of Control Cell and Fill Appropriately - - - - - */
/* - Fuel Assembly in Northwest Quadrant */
/* - Cell */
sprintf(label,"Fuel Assembly in Northwest Quadrant, CC%i"
, *ptr_map);

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 126 of 241

```

strcpy(material,"Bypass Water");
search_usage_list(1,material,&index,ptr_material_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"0*** F a t a l E r r o r *** Function");
    fprintf(nout," build_control_cells --\n");
    fprintf(nout," Material not found in Usage List, label = %s"
        ,material);
    abort();}
add_cell(label,material,lu8,ncell,index,-bypass_density
    ,nuniverse);
/* - Surfaces */
fprintf(lu8," (");
sprintf(label,"Window for Fuel Assembly (max X), CC%i"
    ,*ptr_map);
search_surface_usage_list(1,label,&index,"",""
    ,ptr_surface_usage);
if(index == 0){
    strcpy(mnemonic,"px");
    sprintf(value,"%10.4E", (apitch-0.1));
    index = -1;
    add_surface(0,label,mnemonic,lu8,lu9,&index,value
        ,ptr_surface_usage,nsurface);}
else{
    index = -1;
    add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
        ,nsurface);}
sprintf(label,"Window for Fuel Assembly (min X), CC%i"
    ,*ptr_map);
search_surface_usage_list(1,label,&index,"",""
    ,ptr_surface_usage);
if(index == 0){
    strcpy(mnemonic,"px");
    sprintf(value,"%10.4E", (blade_window_offset+0.1));
    add_surface(0,label,mnemonic,lu8,lu9,&index,value
        ,ptr_surface_usage,nsurface);}
else{
    index = -1;
    add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
        ,nsurface);}
sprintf(label,"Window for Fuel Assembly (max Y), CC%i"
    ,*ptr_map);
search_surface_usage_list(1,label,&index,"",""
    ,ptr_surface_usage);
if(index == 0){
    strcpy(mnemonic,"py");
    sprintf(value,"%10.4E", -(blade_window_offset+0.1));
    index = -1;
    add_surface(0,label,mnemonic,lu8,lu9,&index,value
        ,ptr_surface_usage,nsurface);}
else{
    index = -1;
    add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
        ,nsurface);}

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 127 of 241

```

sprintf(label,"Window for Fuel Assembly (min Y), CC%i"
, *ptr_map);
search_surface_usage_list(1,label,&index,"","",""
, ptr_surface_usage);
if(index == 0){
    strcpy(mnemonic,"py");
    sprintf(value,"%10.4E",-(apitch-0.1));
    add_surface(0,label,mnemonic,lu8,lu9,&index,value
, ptr_surface_usage,nsurface);}
else{
    add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
, nsurface);}
fprintf(lu8," )\n      (#(");
sprintf(label,"Window for Fuel Assembly (max X), CC%i",*ptr_map);
search_surface_usage_list(1,label,&index,"","",""
, ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"*** F a t a l E r r o r *** Function");
    fprintf(nout," build_control_cells --\n");
    fprintf(nout," Surface not Found in Usage List, label = %s"
, label);
    abort();}
index = -index;
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
, nsurface);
sprintf(label,"Window for Fuel Assembly (min Y), CC%i",*ptr_map);
search_surface_usage_list(1,label,&index,"","",""
, ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"*** F a t a l E r r o r *** Function");
    fprintf(nout," build_control_cells --\n");
    fprintf(nout," Surface not Found in Usage List, label = %s"
, label);
    abort();}
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
, nsurface);
sprintf(label,"Curved Corner in Window for Fuel Assembly, CC%i"
, *ptr_map);
search_surface_usage_list(1,label,&index,"","",""
, ptr_surface_usage);
if(index == 0){
    strcpy(mnemonic,"c/z");
    sprintf(value,"%10.4E %10.4E %10.4E",Xc,Yc,R);
    index = 1;
    add_surface(0,label,mnemonic,lu8,lu9,&index,value
, ptr_surface_usage,nsurface);}
else
    add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
, nsurface);
sprintf(label
, "Y Ambiguity Surface for Fuel Assembly Window, CC%i"
, *ptr_map);

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 128 of 241

```

search_surface_usage_list(1,label,&index,"","",""
,ptr_surface_usage);
if(index == 0){
    strcpy(mnemonic,"py");
    sprintf(value,"%10.4E",Yc);
    index = -1;
    add_surface(0,label,mnemonic,lu8,lu9,&index,value
,ptr_surface_usage,nsurface);}
else{
    index = -index;
    add_surface(1,label,"",lu8,lu9,&index,""
,ptr_surface_usage,nsurface);}
sprintf(label
,"X Ambiguity Surface for Fuel Assembly Window, CC%i"
,*ptr_map);
search_surface_usage_list(1,label,&index,"","",""
,ptr_surface_usage);
if(index == 0){
    strcpy(mnemonic,"px");
    sprintf(value,"%10.4E",Xc);
    index = 1;
    add_surface(0,label,mnemonic,lu8,lu9,&index,value
,ptr_surface_usage,nsurface);}
else
    add_surface(1,label,"",lu8,lu9,&index,""
,ptr_surface_usage,nsurface);
fprintf(lu8," )");
fprintf(lu8,"\n      u= %i imp:n= 1.0\n",*ptr_map);
/* - Define Translation */
{ float trans;
  trans = apitch+blade_window_offset;
  fprintf(lu8,"      *trcl=( 0.0 0.0 0.0 "
,-trans,trans);
  fprintf(lu8," 180 90 90 -90 180 90 90 90 0 )\n");
  lines(1);
  fprintf(nout,"      *trcl=( %10.4E %10.4E 0.0 "
,-trans,trans);
  fprintf(nout," 180 90 90 -90 180 -90 90 90 0 )");
}
/* - Determine Lattice with which to Fill Window */
if(*ptr_mapw != 0){
    { short int nn;
      int *p = ptr_ufa;
/*      for(nn = 1; nn < (*ptr_mapw-nu_cc+1);nn++,p++) */
      for(nn = 1;nn < *ptr_mapw;nn++,p++)
          continue;
      fprintf(lu8,"      fill= %i\n",*p);
      lines(1);
      fprintf(nout,"\n      fill= %i\n",*p);
      *fau_fill = *p;
      fau_fill++;
    }
}
else{

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 129 of 241

```

        fprintf(lu8,"          fill= 0\n");
        fprintf(nout," fill= 0\n");
        *fau_fill = 0;
        fau_fill++;}
    ptr_mapw++;
/* - Fuel Assembly in Northeast Quadrant */
/* - Cell */
    sprintf(label,"Fuel Assembly in Northeast Quadrant, CC%i"
        ,*ptr_map);
    fprintf(lu8,"c          %s\n",label);
    (*ncell)++;
    fprintf(lu8,"%5i like %i but",*ncell,((*ncell)-1));
    lines(2);
    sprintf(buffer," %5i",*ncell);
    bufferpad(buffer,strlen(buffer),59);
    ptr_buf = buffer+59*sizeof(char);
    sprintf(ptr_buf,"%s",label);
    fprintf(nout,"%s\n",buffer);
    fprintf(nout
        ,"          Cloning Directive: like %i but *trcl=( 0 0 0"
        ,((*ncell)-1));
    fprintf(nout," 90 0 90 180 90 90 90 90 0)\n");
    if(*ptr_mapw != 0)
/* - Determine Lattice with which to Fill Window */
    { short int nn;
      int *p = ptr_ufa;
/*
      for(nn = 1; nn < (*ptr_mapw-nu_cc+1);nn++,p++) */
      for(nn = 1;nn < *ptr_mapw;nn++,p++)
        continue;
      fprintf(lu8," fill= %i\n",*p);
      lines(1);
      fprintf(nout,"          fill= %i\n",*p);
      *fau_fill = *p;
      fau_fill++;
    }
    else{
      fprintf(lu8," fill= 0\n");
      fprintf(nout," fill= 0\n");
      *fau_fill = 0;
      fau_fill++;}
    fprintf(lu8
        ,"          *trcl=( 0 0 0 90 0 90 180 90 90 90 0)\n");
    fprintf(lu8,"          u= %i imp:n= 1.0\n",*ptr_map);
    ptr_mapw++;
/* - Fuel Assembly in Southeast Quadrant */
/* - Cell */
    sprintf(label,"Fuel Assembly in Southeast Quadrant, CC%i"
        ,*ptr_map);
    fprintf(lu8,"c          %s\n",label);
    (*ncell)++;
    fprintf(lu8,"%5i like %i but",*ncell,((*ncell)-1));
    lines(2);
    sprintf(buffer," %5i",*ncell);
    bufferpad(buffer,strlen(buffer),59);

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 130 of 241

```

ptr_buf = buffer+59*sizeof(char);
sprintf(ptr_buf,"%s",label);
fprintf(nout,"%s\n",buffer);
fprintf(nout
, "      Cloning Directive: like %i but trcl=( 0 0 0)\n"
, ((*ncell)-1));
if(*ptr_mapw != 0)
/* - Determine Lattice with which to Fill Window */
{ short int nn;
  int *p = ptr_ufa;
/*
  for(nn = 1; nn < (*ptr_mapw-nu_cc+1);nn++,p++) */
  for(nn = 1;nn < *ptr_mapw;nn++,p++)
    continue;
  fprintf(lu8," fill= %i\n",*p);
  lines(1);
  fprintf(nout,"      fill= %i\n",*p);
  *fau_fill = *p;
  fau_fill++;
}
else{
  fprintf(lu8," fill= 0\n");
  fprintf(nout," fill= 0\n");
  *fau_fill = 0;
  fau_fill++;}
fprintf(lu8
, "      trcl=( 0 0 0 )");
fprintf(lu8," u= %i imp:n= 1.0\n",*ptr_map);
ptr_mapw++;
/* - Fuel Assembly in Southwest Quadrant */
/* - Cell */
sprintf(label,"Fuel Assembly in Southwest Quadrant, CC%i"
,*ptr_map);
fprintf(lu8,"c      %s\n",label);
(*ncell)++;
fprintf(lu8,"%5i like %i but",*ncell,((*ncell)-3));
lines(2);
sprintf(buffer," %5i",*ncell);
bufferpad(buffer,strlen(buffer),59);
ptr_buf = buffer+59*sizeof(char);
sprintf(ptr_buf,"%s",label);
fprintf(nout,"%s\n",buffer);
fprintf(nout
, "      Cloning Directive: like %i but *trcl=( 0 0 0"
, ((*ncell)-2));
fprintf(nout," -90 180 90 0 -90 90 90 0)\n");
if(*ptr_mapw != 0){
/* - Determine Lattice with which to Fill Window */
{ short int nn;
  int *p = ptr_ufa;
/*
  for(nn = 1; nn < (*ptr_mapw-nu_cc+1);nn++,p++) */
  for(nn = 1;nn < *ptr_mapw;nn++,p++)
    continue;
  fprintf(lu8," fill= %i\n",*p);
  lines(1);

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 131 of 241

```

        fprintf(nout, " fill= %i\n", *p);
        *fau_fill = *p;
        fau_fill++;
    }
}
else{
    fprintf(lu8, " fill= 0\n");
    fprintf(nout, " fill= 0\n");
    *fau_fill = 0;
    fau_fill++;}
fprintf(lu8
, "          *trcl=( 0 0 0 -90 180 90 0 -90 90 90 0)");
fprintf(lu8, " u= %i imp:n= 1.0\n", *ptr_map);
ptr_mapw++;
/* - Control Blade Window */
/* - Cell */
sprintf(label, "Blade Window, CC%i", *ptr_map);
strcpy(material, "Bypass Water");
search_usage_list(1, material, &index, ptr_material_usage);
if(index == 0){
    lines(3);
    fprintf(nout, "*** F a t a l E r r o r *** Function");
    fprintf(nout, " build_control_cells -- \n");
    fprintf(nout, " Material not found in Usage List, label = %s"
, material);
    abort();}
add_cell(label, material, lu8, ncell, index, -bypass_density, nuniverse);
/* - Surfaces */
/* - Horizontal Part of Cross */
fprintf(lu8, " (");
strcpy(label, "Surface 1 for Control Blade Window");
search_surface_usage_list(1, label, &index, "", "", ""
, ptr_surface_usage);
if(index == 0){
    strcpy(mnemonic, "py");
    sprintf(value, "%10.4E", -blade_window_offset);
    index = 1;
    add_surface(0, label, mnemonic, lu8, lu9, &index
, value, ptr_surface_usage, nsurface);}
else
    add_surface(1, label, "", lu8, lu9, &index, ""
, ptr_surface_usage, nsurface);
strcpy(label, "Surface 2 for Control Blade Window");
search_surface_usage_list(1, label, &index, "", "", ""
, ptr_surface_usage);
if(index == 0){
    strcpy(mnemonic, "py");
    sprintf(value, "%10.4E", blade_window_offset);
    index = -1;
    add_surface(0, label, mnemonic, lu8, lu9, &index
, value, ptr_surface_usage, nsurface);}
else{
    index = -index;
    add_surface(1, label, "", lu8, lu9, &index, ""

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 132 of 241

```

        ,ptr_surface_usage,nsurface);}
strcpy(label,"Surface 3 for Control Blade Window");
search_surface_usage_list(1,label,&index,"","",""
        ,ptr_surface_usage);
if(index == 0){
    strcpy(mnemonic,"px");
    sprintf(value,"%10.4E",-(cbspan+0.1));
    index = 1;
    add_surface(0,label,mnemonic,lu8,lu9,&index
        ,value,ptr_surface_usage,nsurface);}
else
    add_surface(1,label,"",lu8,lu9,&index,""
        ,ptr_surface_usage,nsurface);
strcpy(label,"Surface 4 for Control Blade Window");
search_surface_usage_list(1,label,&index,"","",""
        ,ptr_surface_usage);
if(index == 0){
    strcpy(mnemonic,"px");
    sprintf(value,"%10.4E", (cbspan+0.1));
    index = -1;
    add_surface(0,label,mnemonic,lu8,lu9,&index
        ,value,ptr_surface_usage,nsurface);}
else{
    index = -index;
    add_surface(1,label,"",lu8,lu9,&index,""
        ,ptr_surface_usage,nsurface);}
fprintf(lu8," ):(");
/* - Vertical Part of Cross */
strcpy(label,"Surface 5 for Control Blade Window");
search_surface_usage_list(1,label,&index,"","",""
        ,ptr_surface_usage);
if(index == 0){
    strcpy(mnemonic,"px");
    sprintf(value,"%10.4E",-blade_window_offset);
    index = 1;
    add_surface(0,label,mnemonic,lu8,lu9,&index
        ,value,ptr_surface_usage,nsurface);}
else
    add_surface(1,label,"",lu8,lu9,&index,""
        ,ptr_surface_usage,nsurface);
strcpy(label,"Surface 6 for Control Blade Window");
search_surface_usage_list(1,label,&index,"","",""
        ,ptr_surface_usage);
if(index == 0){
    strcpy(mnemonic,"px");
    sprintf(value,"%10.4E",blade_window_offset);
    index = -1;
    add_surface(0,label,mnemonic,lu8,lu9,&index
        ,value,ptr_surface_usage,nsurface);}
else{
    index = -index;
    add_surface(1,label,"",lu8,lu9,&index,""
        ,ptr_surface_usage,nsurface);}
strcpy(label,"Surface 7 for Control Blade Window");

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 133 of 241

```

search_surface_usage_list(1,label,&index,"","",""
,ptr_surface_usage);
if(index == 0){
    strcpy(mnemonic,"py");
    sprintf(value,"%10.4E",- (cbspan+0.1));
    index = 1;
    add_surface(0,label,mnemonic,lu8,lu9,&index
,value,ptr_surface_usage,nsurface);}
else
    add_surface(1,label,"",lu8,lu9,&index,""
,ptr_surface_usage,nsurface);
strcpy(label,"Surface 8 for Control Blade Window");
search_surface_usage_list(1,label,&index,"","",""
,ptr_surface_usage);
if(index == 0){
    strcpy(mnemonic,"py");
    sprintf(value,"%10.4E", (cbspan+0.1));
    index = -1;
    add_surface(0,label,mnemonic,lu8,lu9,&index
,value,ptr_surface_usage,nsurface);}
else{
    index = -index;
    add_surface(1,label,"",lu8,lu9,&index,""
,ptr_surface_usage,nsurface);}
fprintf(lu8,")\n");
fprintf(lu8,"      u= %i imp:n= 1.0\n",*ptr_map);
if(*ptr_mapw >= 0){
    { int notch_fi=48;
      float trans;
      trans =
        -(*ptr_mapw)*(cb_stroke/((float) notch_fi));
      fprintf(lu8
        ,"      fill= %i trcl=( 0.0 0.0 %10.4E )\n"
        ,ucb,trans);
      lines(1);
      fprintf(nout,"      fill= %i trcl= ( 0.0 0.0 %10.4E)\n"
        ,ucb,trans);
    }
}
ptr_mapw++;
/* - Guide Tube */
if(*ptr_mapw != 0){
    { short int ng, ngt;
      int val;
      if(*ptr_mapw > 9)
          ngt = 2;
      else
          ngt = 1;
      for(ng = 1;ng <= ngt;ng++){
          if(ngt == 1) val = *ptr_mapw;
          else
              if(ng == 1) val = (*ptr_mapw)/10;
              else val = (*ptr_mapw)-(10*val);
      }
}
/* - Cell */

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 134 of 241

```

    if (ng == 1)
        sprintf(label, "Guide Tube Segment, CC%i", *ptr_map);
    else
        sprintf(label, "Guide Tube Segment (2), CC%i", *ptr_map);
    strcpy(material, migt);
    search_usage_list(1, migt, &index, ptr_material_usage);
    if (index == 0) {
        ptr_mtl = material_match(ptr_core_mtls, migt, &density
            , &n_entries);
        (*nmaterial)++;
        ptr_ml = ptr_material_usage;
        while (ptr_ml->next != NULL) ptr_ml = ptr_ml->next;
        index = (ptr_ml->index)+1;
        ptr_ml = load_usage_list(migt, index, ptr_ml);
    }
    else {
        (void) material_match(ptr_core_mtls, migt, &density
            , &n_entries);
        n_entries = 0;
    }
    add_cell(label, migt, lu8, ncell, index, -density, nuniverse);
/* - Surfaces */
    strcpy(label, "Outer Surface of Guide Tube");
    search_surface_usage_list(1, label, &index, "", "", ""
        , ptr_surface_usage);
    if (index == 0) {
        strcpy(mnemonic, "c/z");
        sprintf(value, "%10.4E %10.4E %10.4E", apitch, -apitch
            , (dtod/2.0));
        index = -1;
        add_surface(0, label, mnemonic, lu8, lu9, &index, value
            , ptr_surface_usage, nsurface);
    }
    else {
        index = -index;
        add_surface(1, label, "", lu8, lu9, &index, ""
            , ptr_surface_usage, nsurface);
    }
    strcpy(label, "Inner Surface of Guide Tube");
    search_surface_usage_list(1, label, &index, "", "", ""
        , ptr_surface_usage);
    if (index == 0) {
        strcpy(mnemonic, "c/z");
        sprintf(value, "%10.4E %10.4E %10.4E", apitch, -apitch
            , (dtid/2));
        index = 1;
        add_surface(0, label, mnemonic, lu8, lu9, &index, value
            , ptr_surface_usage, nsurface);
    }
    else
        add_surface(1, label, "", lu8, lu9, &index, ""
            , ptr_surface_usage, nsurface);
/* - Material */
    if (n_entries != 0) {
        add_material(lu10, nmaterial, migt, n_entries, ptr_mtl);
        n_entries = 0;
    }
    switch (val) {
        case 1:
            fprintf(lu8, "\n          trcl=( %10.4E %10.4E 0.0) "

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 135 of 241

```

        ,-(2*apitch), (2*apitch));
    lines(1);
    fprintf(nout, "        trcl=( %10.4E %10.4E 0.0)\n"
        ,-(2*apitch), (2*apitch));
    break;
case 2:
    fprintf(lu8, "\n        trcl=( 0.0 %10.4E 0.0)"
        , (2*apitch));
    lines(1);
    fprintf(nout, "        trcl=( 0.0 %10.4E 0.0)\n"
        , (2*apitch));
    break;
case 3:
    break;
case 4:
    fprintf(lu8, "\n        trcl=( %10.4E 0.0 0.0)"
        ,-(2*apitch));
    lines(1);
    fprintf(nout, "        trcl=( %10.4E 0.0 0.0)\n"
        ,-(2*apitch));
}
    fprintf(lu8, " u= %i imp:n= 1.0\n", *ptr_map);
/* - Inside of Guide Tube */
/* - Cell */
    if (ng == 1)
        sprintf(label, "Inside Guide Tube Segment, CC%i"
            , *ptr_map);
    else
        sprintf(label, "Inside Guide Tube Segment (2), CC%i"
            , *ptr_map);
    strcpy(material, "void");
    add_cell(label, material, lu8, ncell, 0, 0.0, nuniverse);
/* - Surfaces */
    strcpy(label, "Inner Surface of Guide Tube");
    search_surface_usage_list(1, label, &index, "", "", ""
        , ptr_surface_usage);
    if (index == 0) {
        lines(3);
        fprintf(nout, "0*** F a t a l E r r o r *** Function");
        fprintf(nout, " build_control_cells --\n");
        fprintf(nout
            , "0Surface not Found in Usage List, label = %s", label);
        abort();
    }
    index = -index;
    add_surface(1, label, "", lu8, lu9, &index, ""
        , ptr_surface_usage, nsurface);
    switch (val) {
    case 1:
        fprintf(lu8, "        trcl=( %10.4E %10.4E 0.0)"
            ,-(2*apitch), (2*apitch));
        lines(1);
        fprintf(nout, "        trcl=( %10.4E %10.4E 0.0)\n"
            ,-(2*apitch), (2*apitch));
        break;

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 136 of 241

```

        case 2:
            fprintf(lu8,"          trcl=( 0.0 %10.4E 0.0) "
                ,(2*apitch));
            lines(1);
            fprintf(nout,"          trcl=( 0.0 %10.4E 0.0)\n"
                ,(2*apitch));
            break;
        case 3:
            break;
        case 4:
            fprintf(lu8,"          trcl=( %10.4E 0.0 0.0) "
                ,- (2*apitch));
            lines(1);
            fprintf(nout,"          trcl=( %10.4E 0.0 0.0)\n"
                ,- (2*apitch));
        }
        fprintf(lu8," u= %i imp:n= 1.0\n",*ptr_map);
    }
}

/* - Balance of Universe */
/* - Cell */
    sprintf(label,"Balance of Control Cell, CC%i",*ptr_map);
    strcpy(material,"Bypass Water");
    search_usage_list(1,material,&index,ptr_material_usage);
    if(index == 0){
        lines(3);
        fprintf(nout
            ,"0*** F a t a l E r r o r *** Function");
        fprintf(nout," build_control_cells --\n");
        fprintf(nout
            ," Material not Found in Usage List, label = %s\n"
            ,material);
        abort();
    }
    (*ncell)++;
    fprintf(lu8,"c          %s\n",label);
    sprintf(buffer," %5i",*ncell);
    bufferpad(buffer,strlen(buffer),49);
    ptr_buf = buffer+49*sizeof(char);
    sprintf(ptr_buf,"%4i",*nuniverse);
    bufferpad(buffer,strlen(buffer),59);
    ptr_buf = buffer+59*sizeof(char);
    sprintf(ptr_buf,"%s",label);
    lines(1);
    fprintf(nout,"%s\n",buffer);
/* - Defintion by Complement */
    { short int nn;
      int nloc;
      if(*ptr_mapw == 0)
          nloc = (*ncell)-5;
      else
          if(*ptr_mapw < 10)
              nloc = (*ncell)-7;
    }

```

Title: Listing of Routines and Functions for BLINK, Version 0**Document Identifier** B00000000-01717-0210-00010 REV 01 Attachment XIV Page 137 of 241

```
        else
            nloc = (*ncell)-9;
            fprintf(lu8,"%5i",*ncell);
            fprintf(lu8," %i %10.4E",index,-bypass_density);
            lines(1);
            fprintf(nout,"          ");
            for(nn = nloc;nn < *ncell;nn++){
                fprintf(lu8," #%i",nn);
                fprintf(nout," #%i",nn);}
        }
        fprintf(nout,"\n");
        fprintf(lu8,"\n          u= %i imp:n= 1.0\n",*ptr_map);
/* - End of Control Cell Construction - - - - - */
        ptr_mapw++;
    }
}
pcc = memory_integer(-1,ncc,pcc);
return;
}
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 138 of 241

Subroutine ccmgen

```

SUBROUTINE CCMGEN(NROWP, NCOLP, MMAP, NROWCC, NCOLCC, CCMAP, CCMAPW
2 , CORE_F, NCOLBP, NROWBP, BLADEP, NCC, ICLOC, NROWB, NCOLB)
C - - - - -
C - - * CCMGEN * Creates Map of Unique Control Cells
C - - - - -
C - - Argument(s):
C - - NROWP - Maximum Number of Rows of Fuel Assemblies (input)
C - - in Core
C - - NCOLP - Maximum Number of Columns of Fuel Assem- (input)
C - - blies in Core
C - - MMAP - Map of Fuel Assembly Material Types in the (input)
C - - Core
C - - NROWCC - Number of Rows in Control Cell Map (input)
C - - NCOLCC - Number of Columns in Control Cell Map (input)
C - - CCMAP - Map of Unique Control Cell Types (output)
C - - CCMAPW - Scratch Array for Identifying Each Con- (scratch)
C - - trol Cell in Terms of Constituent Fuel
C - - Assemblies and Control Blade Position
C - - (1 - Fuel Assembly Type in the Northwest
C - - Quadrant
C - - 2 - Fuel Assembly Type in the Northeast
C - - Quadrant
C - - 3 - Fuel Assembly Type in the Southeast
C - - Quadrant
C - - 4 - Fuel Assembly Type in the Southwest
C - - Quadrant
C - - 5 - Blade Position
C - - 6 - Location of Incore Detector in Control
C - - Cell
C - - 0 - No Guide Tube Present,
C - - 1 - Guide Tube in Northwest Quadrant
C - - 2 - Guide Tube in Northeast Quadrant
C - - 3 - Guide Tube in Southeast Quadrant
C - - 4 - Guide Tube in Southwest Quadrant
C - - If there is a second Guide Tube in
C - - Control Cell, then 10 Times the
C - - Index above is Added to the Value)
C - - CORE_F - Fraction of Core Represented (input)
C - - (0 - Full Core,
C - - 2 - Half Core,
C - - 4 - Quarter Core)
C - - NCOLBP - Number of Columns in Control Blade Map (input)
C - - NROWBP - Number of Rows in Control Blade Map (input)
C - - BLADEP - Control Blade Insertion Map (input)
C - - NCC - Number of MCNP "Universes" (i&o)
C - - INLOC - Locations of Incore Detectors (input)
C - - NROWB - Number of Rows of Control Blade Locations in (input)
C - - the Whole Core
C - - NCOLB - Number of Columns of Control Blade Locations (input)
C - - in the Whole Core
C - - - - -
C - -

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 139 of 241

```

C - - Type Statement(s) - - - - -
C - -  EVENX - Flag to Indicate that Peripheral Fuel Assemblies not
C - -          Resident in Control Cells Exist at the X extreme(s) of
C - -          the Map
C - -  EVENY - Flag to Indicate that Peripheral Fuel Assemblies not
C - -          Resident in Control Cells Exist at the Y extreme of
C - -          the Map
C - -  NULL - Flag to Indicate that a Control Cell Contents Vector
C - -          Empty, Indicating a Null Cell
C - -  MATCH - Flag to Indicate that a Given Control Cell matches a
C - -          Reference Control Cell
          LOGICAL EVENX,EVENY,NULL,MATCH
          INTEGER CORE_F
          INTEGER CCMAP(NCOLCC,NROWCC),BLADEP(NCOLBP,NROWBP)
          INTEGER CCMAPW(6,NCOLCC,NROWCC)
C - - Dimension Statement(s) - - - - -
          DIMENSION MMAP(NCOLP,NROWP),ICLOC(NCOLB,NROWB)
C - - Determine Whether there are Peripheral Assemblies not in Control
C - - Cells at the Extremes of the Material Map
          IF(CORE_F .EQ. 4) THEN
              IF(FLOAT(NROWP/2) .EQ. (FLOAT(NROWP)/2.)) THEN
                  EVENX = .TRUE.
                  EVENY = .TRUE.
              ELSE
                  EVENX = .FALSE.
                  EVENY = .FALSE.
              ENDIF
          ELSEIF(CORE_F .EQ. 1) THEN
              IF(FLOAT(NROWP/4) .EQ. (FLOAT(NROWP)/4.)) THEN
                  EVENX = .TRUE.
                  EVENY = .TRUE.
              ELSE
                  EVENX = .FALSE.
                  EVENY = .FALSE.
              ENDIF
          ELSEIF(CORE_F .EQ. 2) THEN
              IF(FLOAT(NROWP/4) .EQ. (FLOAT(NROWP)/4.)) THEN
                  EVENX = .TRUE.
              ELSE
                  EVENX = .FALSE.
              ENDIF
              IF(FLOAT(NCOLP/2) .EQ. (FLOAT(NCOLP)/2.)) THEN
                  EVENY = .TRUE.
              ELSE
                  EVENY = .FALSE.
              ENDIF
          ENDIF
C - - Sweep through Map Loading Identifications for Control
C - - Cells
          IXCC = 0
          IYCC = 1
          CALL INVALIDI((NCOLCC*NROWCC),CCMAP,0)
          CALL INVALIDI((6*NCOLCC*NROWCC),CCMAPW,0)
C - - Set up Starting and Ending Limits for Processing Map

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 140 of 241

```

      IF(EVENY) THEN
        JSTART = 2
        JEND = NROWP-1
      ELSE
        JSTART = 1
        JEND = NROWP
      ENDIF
      IF(EVENX) THEN
        ISTART = 2
        IEND = NCOLP-1
      ELSE
        ISTART = 1
        IEND = NCOLP
      ENDIF
      IF(CORE_F .EQ. 4) THEN
        IEND = NCOLP-1
        JEND = NROWP-1
      ENDIF
      IF(CORE_F .EQ. 2) JEND = NROWP-1
C - - Load Locations for Extra Row at the Top of the Problem
      IYCC = 1
      IF(EVENY) THEN
        DO I = ISTART,IEND,2
          IXCC = IXCC+1
          CCMAPW(4,IXCC,IYCC) = MMAP(I,1)
          CCMAPW(3,IXCC,IYCC) = MMAP((I+1),1)
        ENDDO
        IF(CORE_F .EQ. 4) THEN
          IXCC = IXCC+1
          CCMAPW(4,IXCC,IYCC) = MMAP(NROWP,1)
        ENDIF
      ENDIF
C - - Process Bulk of Map
      DO J = JSTART,JEND,2
        IXCC = 0
C - - Process First Column if Extra Column
        IF(EVENX) THEN
          IXCC = IXCC+1
          CCMAPW(2,IXCC,IYCC) = MMAP(1,J)
          CCMAPW(3,IXCC,IYCC) = MMAP(1,(J+1))
        ENDIF
C - - Process Bulk of Map
        DO I = ISTART,IEND,2
          IXCC = IXCC+1
          CCMAPW(1,IXCC,IYCC) = MMAP(I,J)
          CCMAPW(2,IXCC,IYCC) = MMAP((I+1),J)
          CCMAPW(4,IXCC,IYCC) = MMAP(I,(J+1))
          CCMAPW(3,IXCC,IYCC) = MMAP((I+1),(J+1))
          CCMAPW(5,IXCC,IYCC) = BLADEP(IXCC,IYCC)
          IF(ICLOC(IXCC,IYCC) .EQ. 1) THEN
            IF(CCMAPW(6,IXCC,IYCC) .EQ. 0) THEN
              CCMAPW(6,IXCC,IYCC) = 3
            ELSE
              CCMAPW(6,IXCC,IYCC) = CCMAPW(6,IXCC,IYCC)+30
            
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 141 of 241

```

      ENDIF
      IF(CCMAPW(6,(IXCC+1),IYCC) .EQ. 0) THEN
        CCMAPW(6,(IXCC+1),IYCC) = 4
      ELSE
        CCMAPW(6,(IXCC+1),IYCC) = CCMAPW(6,(IXCC+1),IYCC)+40
      ENDIF
      IF(CCMAPW(6,(IXCC+1),(IYCC+1)) .EQ. 0) THEN
        CCMAPW(6,(IXCC+1),(IYCC+1)) = 1
      ELSE
        CCMAPW(6,(IXCC+1),(IYCC+1)) =
2      CCMAPW(6,(IXCC+1),(IYCC+1))+10
      ENDIF
      IF(CCMAPW(6,IXCC,(IYCC+1)) .EQ. 0) THEN
        CCMAPW(6,IXCC,(IYCC+1)) = 2
      ELSE
        CCMAPW(6,IXCC,(IYCC+1)) = CCMAPW(6,IXCC,(IYCC+1))+20
      ENDIF
    ENDIF
  ENDDO
C - - Process Last Column, if Extra Column or Quarter Core
  IF(EVENX .OR. (CORE_F .EQ. 4)) THEN
    IXCC = IXCC+1
    CCMAPW(1,IXCC,IYCC) = MMAP(NCOLP,J)
    CCMAPW(4,IXCC,IYCC) = MMAP(NCOLP,(J+1))
    CCMAPW(5,IXCC,IYCC) = BLADEP(IXCC,IYCC)
  ENDIF
  IYCC = IYCC+1
ENDDO
C - - Load Locations for Extra Row at Bottom of Problem, if Present
C - - or Quarter or Half Core
IF(EVENY .OR. (CORE_F .EQ. 2) .OR. (CORE_F .EQ. 4)) THEN
  IXCC = 0
  DO I = ISTART,IEND,2
    IXCC = IXCC+1
    CCMAPW(1,IXCC,IYCC) = MMAP(I,NROWP)
    CCMAPW(2,IXCC,IYCC) = MMAP((I+1),NROWP)
    CCMAPW(5,IXCC,IYCC) = BLADEP(IXCC,IYCC)
  ENDDO
  IF(CORE_F .EQ. 4) THEN
    IXCC = IXCC+1
    CCMAPW(1,IXCC,IYCC) = MMAP(NCOLP,NROWP)
    CCMAPW(5,IXCC,IYCC) = BLADEP(IXCC,IYCC)
  ENDIF
ENDIF
C - - Search for Unique Control Cells - - - - -
CALL INVALID((NCOLCC*NROWCC),CCMAP,0)
DO J = 1,NROWCC
  DO I = 1,NCOLCC
C - - Check for non-fuel Cells
  NULL = .TRUE.
  DO N = 1,4
    IF(CCMAPW(N,I,J) .NE. 0) NULL = .FALSE.
  ENDDO
  IF(.NOT. NULL) THEN

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 142 of 241

```

C - - Match Found
      IF(CCMAP(I,J) .EQ. 0) THEN
          NCC = NCC+1
          CCMAP(I,J) = NCC
C - - Examine Balance of Core for Matches with Current Control Cell
      IF(I .NE. NCOLCC) THEN
          DO II = (I+1),NCOLCC
              MATCH = .TRUE.
              DO N = 1,6
                  IF(CCMAPW(N,II,J) .NE. CCMAPW(N,I,J))
2                     MATCH = .FALSE.
              ENDDO
              IF(MATCH) CCMAP(II,J) = NCC
              ENDDO
          ENDDIF
          IF(J .NE. NROWCC) THEN
              DO JJ = (J+1),NROWCC
                  DO II = 1,NCOLCC
                      MATCH = .TRUE.
                      DO N = 1,6
                          IF(CCMAPW(N,II,JJ) .NE. CCMAPW(N,I,J))
2                             MATCH = .FALSE.
                      ENDDO
                      IF(MATCH) CCMAP(II,JJ) = NCC
                      ENDDO
                  ENDDO
              ENDDIF
          ENDDIF
      ENDDO
      ENDDO
C - - End of Normal Processing - - - - -
      RETURN
      END

```

Function core_lattice_generation

```

#include <stdio.h>
#include <string.h>

typedef char ascii_string[133];
typedef struct ascii_record{
    struct ascii_record *last;
    ascii_string line;
    struct ascii_record *next;
} a_record;

typedef struct s_material{
    struct s_material *last;
    int atomic_number;
    int mass_number;
    float weight_percentage;
    char library_suffix[5];
    struct s_material *next;
} ll_material;

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 143 of 241

```

typedef struct u_list{
    struct u_list *last;
    int index;
    ascii_string label;
    struct u_list *next;
} usage_list;

typedef struct su_list{
    struct su_list *last;
    int index;
    ascii_string label;
    ascii_string value;
    char mnemonic[4];
    ascii_string equivalent_label;
    struct su_list *next;
} surface_usage_list;

void search_usage_list(int,char[],int *,usage_list *);
void search_surface_usage_list(int,char[],int *,char[],char[]
, char[],surface_usage_list *);
void add_cell(char[],char[],FILE *,int *,int,float,int *);
void add_surface(int,char[],char[],FILE *,FILE *,int *,char[]
,surface_usage_list *,int *);
void add_symmetry_surfaces(int,FILE *,FILE *,surface_usage_list *
,int);

void core_lattice_generation(int *ncell,int *nsurface,float apitch
,float bypass_density,surface_usage_list *ptr_surface_usage
,usage_list *ptr_material_usage,int *ccmap,int ncolcc,int nrowcc
,FILE *lu8,FILE *lu9,int core_f,int *nuniverse,float afl){
/* -----
- - *   c o r e _ l a t t i c e _ g e n e r a t i o n   *
- -   Creates Control Cell Lattice Field
-----
- - Argument(s):
- -     ncell - Total Number of Cells in MCNP Model           (i|o)
- -     apitch - Fuel Assembly Pitch (cm)                     (input)
- - bypass_density
- -     - water density in bypass region (g/cc)               (input)
- - ptr_surface_usage
- -     - pointer to linked list for surface labels and       (input)
- -     indices
- - ptr_material_usage
- -     - pointer to linked list for material labels and       (input)
- -     indices
- -     ccmap - pointer to control cell map                     (input)
- -     ncolcc - number of columns in control cell map         (input)
- -     nrowcc - number of rows in control cell map            (input)
- -     lu8 - pointer to scratch file containing MCNP cell     (input)
- -     definitions
- -     lu9 - pointer to scratch file containing MCNP sur-     (input)
- -     face definitions
- -     core_f - flag for fraction of core modeled             (input)

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 145 of 241

```

index = -index;
add_surface(1,label,mnemonic,lu8,lu9,&index,value,ptr_surface_usage
,nsurface);
/* - Symmetry Surfaces, if present) */
if((core_f == 2) || (core_f == 4))
    add_symmetry_surfaces(core_f,lu8,lu9,ptr_surface_usage,0);
/* - Top of Active Fuel */
strcpy(label,"Top of Active Fuel");
search_surface_usage_list(1,label,&index,"","",""
,ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"0*** F a t a l E r r o r *** Function ");
    fprintf(nout," core_lattice_generation --\n");
    fprintf(nout," Surface Not Found in Linked List,label = %s\n"
,label);
    abort();}
index = -index;
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
,nsurface);
/* - Bottom of Active Fuel */
strcpy(label,"Bottom of Active Fuel");
search_surface_usage_list(1,label,&index,"","",""
,ptr_surface_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"0*** F a t a l E r r o r *** Function ");
    fprintf(nout," core_lattice_generation --\n");
    fprintf(nout," Surface Not Found in Linked List, surface = %s\n"
,label);
    abort();}
add_surface(1,label,"",lu8,lu9,&index,"",ptr_surface_usage
,nsurface);
/* - Filling Universe and Neutron Importance of Cell */
(*nuniverse)++;
fprintf(lu8,"\n      fill=%i\n",*nuniverse);
fprintf(lu8,"      imp:n= 1.0\n");
/* - Create Control Cell Lattice - - - - - */
/* - Cell */
strcpy(label,"Core Lattice");
strcpy(material,"Bypass Water");
search_usage_list(1,material,&index,ptr_material_usage);
if(index == 0){
    lines(3);
    fprintf(nout,"0*** F a t a l E r r o r *** Function ");
    fprintf(nout," core_lattice_generation --\n");
    fprintf(nout," Material Not Found in Linked List, material = %s\n"
,label);
    abort();}
add_cell(label,material,lu8,ncell,index,-bypass_density,nuniverse);
/* - Surfaces */
strcpy(label,"Edge of Control Cell (max X)");
sprintf(value,"%10.4E",apitch);
index = -1;

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 146 of 241

```

add_surface(0,label,"px",lu8,lu9,&index,value,ptr_surface_usage
,nsurface);
strcpy(label,"Edge of Control Cell (min X)");
sprintf(value,"%10.4E",(-apitch));
index = 1;
add_surface(0,label,"px",lu8,lu9,&index,value,ptr_surface_usage
,nsurface);
strcpy(label,"Edge of Control Cell (min Y)");
sprintf(value,"%10.4E",(-apitch));
index = 1;
add_surface(0,label,"py",lu8,lu9,&index,value,ptr_surface_usage
,nsurface);
strcpy(label,"Edge of Control Cell (max Y)");
sprintf(value,"%10.4E",apitch);
index = -1;
add_surface(0,label,"py",lu8,lu9,&index,value,ptr_surface_usage
,nsurface);
fprintf(lu8,"\n");
fprintf(lu8,"          lat=1 u=%i imp:n= 1.0\n",*nuniverse);
/* - Populate Lattice */
fprintf(lu8,"          fill=");
if(core_f == 4)
    fprintf(lu8," %i:0 %i:0 0:0\n",-ncolcc,-nrowcc);
if(core_f == 1)
    if(ncolcc%2 != 0)
        fprintf(lu8," %i:%i %i:%i 0:0\n"
,(-(ncolcc/2)),(ncolcc/2),(-(nrowcc/2)),(nrowcc/2));
    else
        fprintf(lu8," %i:%i %i:%i 0:0\n"
,(-(ncolcc/2)+1),(ncolcc/2)+1)
,(-(nrowcc/2)+1),(nrowcc/2)+1));
if(core_f == 2){
    if(ncolcc%2 != 0)
        fprintf(lu8," %i:%i",(-(ncolcc/2)),(ncolcc/2));
    else
        fprintf(lu8," %i:%i"
,(-(ncolcc/2)+1),(ncolcc/2)+1));
    fprintf(lu8," %i:0 0:0 \n",(-(ncolcc/2)));}
{ short int i,j;
int *loc;
loc = ccmmap;
fprintf(lu8,"          ");
for(i = -1;i < ncolcc;i++){
    fprintf(lu8," %3i",*nuniverse);
    fprintf(lu8,"\n");
    for(j = 0;j < nrowcc;j++){
        fprintf(lu8,"          ");
        fprintf(lu8," %3i",*nuniverse);
        for(i = 0;i < ncolcc;i++){
            if(*loc != 0)
                fprintf(lu8," %3i",*loc);
            else
                fprintf(lu8," %3i",*nuniverse);
            loc++;}
    }
}

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV **Page 147 of 241**

```
        fprintf(lu8, "\n");  
    }  
}  
}
```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV **Page 148 of 241**

Function echo_MCNP_deck

```

#include<stdio.h>
#include<string.h>

void echo_MCNP_deck(FILE *lmcnp) {
/* -----
- - * e c h o _ M C N P _ d e c k * Copies the MCNP File to the
- -                               Output Stream
-----
- - Argument(s):
- -   lmcnp - File Pointer to MCNP Input Deck           (input)
-----
- - Variable Declaration(s) -----
/* - Character Variable(s)
- -   c - single character used to read from input file
- -   p - pointer to character variable
- -   buffer - string variable used to copy files from input file
- -   case_title - title for analysis
*/
  unsigned char c;
  char *p;
  char buffer[133] = "";
/* - FILE Variable(s)
- -   nout - output file
*/
  extern FILE *nout;
/* - - Rewind File before Processing - - - - - */
  rewind(lmcnp);
  lines(4);
  fprintf(nout,"0Input to MCNP\n");
  fprintf(nout,"\n");
/* - Copy Line-by-line to Output File - - - - - */
  do
  { strcpy(buffer,"");
    p = buffer;
    do
    { c = fgetc(lmcnp);
      *p = c;
      p++;}
    while((c != 10) && (c != 255));
    if(c != 255) fprintf(nout," ");
    if(c != 255){
      p = buffer;
      c = *p;
      while(c != 10){
        fputc(*p,nout);
        p++;
        c = *p;
      }
      fprintf(nout,"\n");
      lines(1);
    }
  }
}

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 149 of 241

```

}
while(c != 255);
fclose(lmcp);
return;
}

```

Function edit_materials

```

#include <stdio.h>
#include <string.h>

typedef char ascii_string[133];

typedef struct u_list{
    struct u_list *last;
    int index;
    ascii_string label;
    struct u_list *next;
} usage_list;

void header();
void lines(int);

void edit_materials(usage_list *ptr_material_usage){
/* -----
- - * e d i t _ m a t e r i a l s *  edits the descriptions of mater-
- -                               i a l s  used in the problem
-----
- - Argument(s):
- -   material_usage_list
- -   - linked list of material descriptions           (input)
-----
- - Variable Declarations -----
- - Character Variable(s)
- -   buffer - buffer for processing lines of output
- -   ptr_buf - pointer to buffer
*/
    ascii_string buffer;
    char *ptr_buf;
/* - Structured Variable(s)
- -   ptr - pointer to surface usage linked list
*/
    usage_list *ptr = ptr_material_usage;
/* - FILE Variable(s) -----
- -   nout - pointer to output stream
*/
    extern FILE *nout;
/* - Edit List of Universes Assigned to Unique Lattices ----- */
    header();
    lines(4);
    fprintf(nout,"0Description of Materials used in the Problem\n");
    fprintf(nout,"0Index Label\n\n");
    while(ptr->next != NULL){
        lines(1);

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 150 of 241

```

        fprintf(nout, " %3i %s\n", ptr->index, ptr->label);
        ptr = ptr->next;
    }
/* - Last Member of Linked List */
    lines(l);
    fprintf(nout, " %3i %s\n", ptr->index, ptr->label);
}

```

Function edit_surfaces

```

#include <stdio.h>
#include <string.h>

typedef char ascii_string[133];

typedef struct su_list{
    struct su_list *last;
    int index;
    ascii_string label;
    ascii_string value;
    char mnemonic[4];
    ascii_string equivalent_label;
    struct su_list *next;
} surface_usage_list;

void header();
void lines(int);
void bufferpad(char [], int, int);

void edit_surfaces(surface_usage_list *ptr_surface_usage){
/* - - - - -
- - * e d i t _ s u r f a c e s * edits the descriptions of surfaces
- -                               used in the problem
- - - - -
- - Argument(s):
- -   surface_usage_list
- -       - linked list of surface descriptions           (input)
- - - - -
- - Variable Declarations - - - - -
- -   seq_num - sequence number for surfaces in surface definition
- -       linked list
*/
    int seq_num = 0;
/* - Character Variable(s)
- -   buffer - buffer for processing lines of output
- -   ptr_buf - pointer to buffer
*/
    ascii_string buffer;
    char *ptr_buf;
/* - Structured Variable(s)
- -   ptr - pointer to surface usage linked list
*/
    surface_usage_list *ptr = ptr_surface_usage;
/* - FILE Variable(s) - - - - -

```

Title: Listing of Routines and Functions for BLINK, Version 0

Document Identifier B00000000-01717-0210-00010 REV 01 Attachment XIV Page 151 of 241

```

- - nout - pointer to output stream
*/
extern FILE *nout;
/* - Edit List of Universes Assigned to Unique Lattices - - - - - */
header();
lines(7);
fprintf(nout,"0Description of Surfaces used in the Problem\n");
fprintf(nout,"0Seq. No. Index          Label\n");
fprintf(nout,"          Mnemonic Definition\n");
fprintf(nout,"          Equivalent Label\n\n");
while(ptr->next != NULL){
    ptr_buf = buffer;
    seq_num++;
    ptr = ptr->next;
    if(strlen(ptr->equivalent_label) != 0)
        lines(3);
    else
        lines(2);
    fprintf(nout,"  %3i      %3i          %s\n",seq_num,ptr->index
,ptr->label);
    sprintf(buffer,"          %s",ptr->mnemonic);
    ptr_buf = buffer;
    bufferpad(buffer,strlen(buffer),26);
    ptr_buf += 25*sizeof(char);
    sprintf(ptr_buf,"%s",ptr->value);
    fprintf(nout,"%s\n",buffer);
    if(strlen(ptr->equivalent_label) != 0)
        fprintf(nout,"          %s\n"
,ptr->equivalent_label);
}
}

```

Function edit_universes

```

#include <stdio.h>
#include <string.h>

typedef char ascii_string[133];

void header();
void lines();

int search_fau_list(int,int,int *,int);

void edit_universes(int ncolcc,int nrowcc,int *ccmap,int *fau_fill
,int core_f,int ncolp,int nrowp,int nu_cc,int nlatticm,int nbundlm
,int *ptr_ufl,int *ptr_ufa){
/* - - - - -
- - * e d i t _ u n i v e r s e s * edits the universe indices for
- -                               control cells and fuel assemblies
- - - - -
- - Argument(s):
- -   ncolcc - Number of Columns in Control Cell Map      (input)
- -   nrowcc - Number of Rows in Control Cell Map         (input)
- -   ccmap  - Map of Control Cell Universe Indices       (input)

```