

# ZPD Memory Organization – Blocks, Addresses, and Formats

1 April 2003

## Memory Block Organization

Block	Finders	Block	Fitters	Block	Miscellaneous
0x2	Finder 0	0x4	Fitter 0	0x1	Sergio (segment I/O)
0x8	Finder 1	0x10	Fitter 1	0x4000	Clock Manager reset (temp)
0x20	Finder 2	0x40	Fitter 2	0x8000	Decision Module
0x80	Finder 3	0x100	Fitter 3		
0x200	Finder 4	0x400	Fitter 4		
0x800	Finder 5	0x1000	Fitter 5		
0x2a	SL10 Finders	0x54	SL10 Fitters		
0xa80	SL7 Finders	0x1500	SL7 Fitters		
0xaaa	All Finders	0x1554	All Fitters		

## Diagnostic Memories

Block	Address	Name
Sergio 0x1	4000 – 7F9F	TSF Segment Input; XXA0 – XXFF reserved <sup>1</sup>
Finders	4000 – 7F9F	Megabus; XXA0 – XXFF reserved <sup>1</sup>
Finders	1000 – 17FB	Finder Results; 1XXC – 1XXF unused
Fitters	4000 – 41FF	Fitter Results
Decision Module 0x8000	4000 – 4BFF	Fitter Results
Decision Module 0x8000	3000 – 303F	Output

## DAQ Memories (read only<sup>2</sup>)

Block	Address	Name
Sergio 0x1	2000 – 2049	Segment Masks output buffer 0
“	2100 – 2149	Segment Masks output buffer 1
“	2200 – 2249	Segment Masks output buffer 2
“	2300 – 2349	Segment Masks output buffer 3
“	2400 – 267F	Circular Buffer ( <b>16 bit data</b> )
“	2A00	offset
Decision Module 0x8000	2000 – 20C7	Fit Results and output buffer 0
“	2100 – 21C7	Fit Results and output buffer 1
“	2200 – 22C7	Fit Results and output buffer 2
“	2300 – 23C7	Fit Results and output buffer 3
“	2400 – 273F	Circular Buffer ( <b>32 bit data</b> )
“	2A00	offset

<sup>1</sup> Reserved addresses in the segment/Megabus diagnostic memories will return 0xBADD if read.

<sup>2</sup> Offsets (between the pointer of the circular memory and of the daq memory) are writable.

## Miscellaneous – All Blocks

Address	Name
0	Version (read only)
1	Control (unused)
2	Status (read only)
3 – F	Reserved

## Look Up Tables (LUTs)

Block	Address	Name
Finders	8000 – A7FF	Seed phi SL conversion
“	C000 – E7FF	Expected phi position ( <b>32 bit data</b> )
Fitters	100 – 108	phiconv
“	200 – 208	twistzero
“	300 – 41F	curvcorr
“	500 – 508	wr2
“	600 – 71F	wr2dpdr
“	800 – 83F	sumr2
“	900 – 10FF	sumr2dpdr
“	1100 – 18FF	sumrd2pdr2
“	1900 – 20FF	denomrp
“	2100 – 21BF	dphidrho
“	2200 – 23FF	fitok
“	2400 – 25FF	hitax
“	2600 – 283F	useax
“	2900 – 293F	rh5
“	2A00 – 2A3F	rh3
“	2B00 – 2B05	rstero
“	2C00 – 2C05	sigma2z
“	2D00 – 2D2F	dsigma2z
“	2E00 – 2E2F	sums2
“	2F00 – 30FF	sumds2
“	3100 – 32FF	sumd2s2 ( <b>32 bit data</b> )
“	3300 – 34FF	denomzt
“	3500 – 36FF	z0err
Decision Module	0xy0 <sup>3</sup>	$\rho$ min
“	0xy1	$\rho$ max
“	0xy2	tandip min
“	0xy3	tandip max
“	0xy4	z0 min
“	0xy5	z0 max
“	0xy6	z0err max
“	0xF0	4 bit mask

<sup>3</sup> The value of  $x$  runs from 1 to 8 for eight decision bits and the value of  $y$  is 0 for A10 and 1 for A7 tracks.

## TSF Segment Input and Megabus Diagnostic Memory Format

*NB: Although the memories are 16 bits wide, only the lowest 14 bits are sent across the Megabus*

Each event is sent across the Megabus in 32 clk120 ticks of 5 segments each. The segment order in the TSF Segment Input and Megabus Diagnostic memories are identical and if all is well, their contents will be identical.

The address for a given segment is:

$$\text{Addr} = 0x4000 + 0x100 * \text{event} + 0x20 * \text{segment} + \text{clock\_tick}.$$

Each sector ( $2\pi/16$  radians) of each superlayer (SL) has 3 segments sent in consecutive clock ticks. The SL:sector to Megabus segment:tick mapping is as follows:

MB	MB Segment				
	4	3	2	1	0
<b>0 - 2</b>	1:1	1:3	1:5	10:1	10:3
<b>3 - 5</b>	1:2	1:4	1:0	10:2	10:4
<b>6 - 8</b>	2:1	2:3	2:5	7:1	7:3
<b>9 - B</b>	2:2	2:4	2:0	7:2	7:4
<b>C - E</b>	3:1	3:3	3:5	5:1	5:3
<b>F - 11</b>	3:2	3:4	5:0	5:2	5:4
<b>12 - 14</b>	9:1	9:3	9:5	4:1	4:3
<b>15 - 17</b>	9:2	9:4		4:2	4:4
<b>18 - 1A</b>	6:1	6:3	6:5	8:1	8:3
<b>1B - 1D</b>	6:2	6:4		8:2	8:4
<b>1E - 1F</b>					

The segment format is:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		M	cell[3:0]			phi[5:0]					dphi[2:0]				

*M=mask bit (0 or 1)*

## Finder Results Diagnostic Memory Format per Track (2 per clk4)

$$\text{Addr} = 0x1000 + 0x20 * \text{event} + 0x10 * \text{seednum}(0 \text{ or } 1)$$

Addr	15	Value[15:0]										0
0											hitmask[9:0]	
1										dipbin[5:0]		rhobin[5:0]
2												segphi SL 1
3												segphi SL 2
...												...
A												segphi SL 9
B												segphi SL 10
C – F												Unused

## Fitter Results Diagnostic Memory Format per Track (2 per clk4)

*NB: Although the memory is 16 bits wide, only the lowest 12 are sent from the Fitters to the Decision Module.*

$$\text{Addr} = 0x4000 + 8 * \text{event} + 4 * \text{seednum}(0 \text{ or } 1)$$

Addr	15	Value[15:0]										0
0											hitmask[9:0]	
1										rho[7:0]		z0err[3:0]
2												z0[7:0]
3												dip[7:0]

The Decision Module Fitter Results have an identical format. Addresses 0x4000-0x41FF cover Fitter 0, addresses 0x4200-0x431F cover Fitter 1, etc.

*NB: This format will likely soon change; keep your code flexible...*

## Output Diagnostic Memory Format

[Upcoming]

## DAQ Memory Format

[Upcoming]

## LUT Formats

[Upcoming]