Project Specification
Project Name: ABCD3T ASIC
Version: V1.2

1. SCOPE

This document describes the requirements and target design specifications for the front-end ASIC to be used in the binary readout architecture of silicon strip detectors in the ATLAS Semiconductor Tracker (SCT). The ABCD3T chip designed for the DMILL technology comprises all functions required for the binary readout architecture in a single chip. The ABCD3T design is based upon the ABCD2T prototype chip. Some modififications and adjustment have been implemented in the ABCD3T design in order to improve rediation resistance of the new design, while the architecture and all core blocks of the ABCD2T design remain unchanged. This document is based on the specification documents for the SCT128B, CAFE-M, ABC and ABCD2T designs and includes additional information specific for the ABCD3T design.

The first prototype of the ABCD showed unsatisfactory performance with respect to the channel-to-channel matching in the front-end circuit and with respect post-radiation speed margins in some digital blocks. Two new versions of the front-end circuit addressed the issue of matching using two different approaches: (a) following the circuit concept implemented in the first ABCD prototype along with improving performance and tuning the circuit to the new process parameters, (b) implementing a new architecture with individual threshold correction per channel. Following the two versions of the front-end circuit, the two versions of the ABCD chip were designed: ABCD2NT (No Trim DACs) and ABCD2T (Trim DACs). The top level architecture and most of the of the parameters were identical for the two version. Based upon the test results of the prototype modules using the ABCD2T chips and upon the irradiation tests the ABCD2T architecture with individual threshold correction per channel has been selected for further prototyping.

2. REFERENCE DOCUMENTS

1. SCT128B - Binary Readout Chip for the ATLAS SCT, Project description, Rev. 1.1.
2. CAFE-P, Project Specification, Version 4.00.
3. Atlas Binary Chip (ABC), Project Specification, Version V4.03.
4. ABCD Chip, Project Specification, Version V1.1
5. W. Dabrowski, et al., “Fast Bipolar Front-end for Binary Readout of Silicon Strip Detectors”, Nuclear Instruments and Methods A 350 (1994), 548.
6. W. Dabrowski, et al., “Performance of the Binary Readout of Silicon Strip Detectors Using the Radiation Hard SCT128B Chip”, IEEE, Trans. Nucl. Sci. Vol. 45, No.3 (1998), 310.
7. W. Dabrowski, et al., “The ABCD Binary Readout Chip for Silicon Strip Detectors in the Atlas Silicon Tracker”, Proc. of the Fourth Workshop on Electronics for LHC Experiments, Rome, September 21-25, 1988, CERN/LHCC/98-36, p. 175.
8. W. Dabrowski et al., “Radiation hardness of the ABCD chip for the binary readout of silicon strip detectors in the ATLAS Semiconductor Tracker”, Proc. of the Fifth Workshop on Electronics for LHC Experiments, Snowmass, Colorado, USA, 20-24 September 1999, CERN/LHCC/99-33, 29 October 1999, p.113.
9. I. Kipnis, “CAFE: A complementary Bipolar Analog Front End Integrated Circuit for the ATLAS SCT”, 1995.
10. E. Spencer et al., “A Fast Shaping Low Power Amplifier-Comparator Integrated for Silicon Strip Detectors”, IEEE Nuclear Science Symposium, 1994, (SCIPP 94/32).
11. T. Dubbs et al. “The development of the CAFE-P/CAFE-M bipolar chips for the ATLAS Semiconductor Tracker”, Proc. of the Fifth Workshop on Electronics for LHC Experiments, Snowmass, Colorado, USA, 20-24 September 1999, CERN/LHCC/99-33, 29 October 1999, p.123.
12. D. Campbell et al. “The development of the rad-hard CMOS chip for the binary readout of the ATLAS Semiconductor Tracker” Proc. of the Fifth Workshop on Electronics for LHC Experiments, Snowmass, Colorado, USA, 20-24 September 1999, CERN/LHCC/99-33, 29 October 1999, p.152.
13. RAL 214 /DORIC3 Project Specification, V1.01.
14. RAL 216/LDC Project Specification, V 1.01.
15. A.R. Weidberg, Dead time calculations for SCT readout architectures. ATLAS-INDET-124.
16. W. Dabrowski, A proposal to implement individual threshold correction per channel in the ABCD chip.
17. ABCD2T/ABCD2NT ASIC, Project Specification V 2.1.

3. TECHNICAL ASPECTS

3.1 Requirements

3.1.1 General

The chip must provide all functions required for processing of signal from 128 strips of a silicon strip detector in the ATLAS experiment employing the binary readout architecture. The simplified block diagram of the chip is shown in figure 3.1. The main functional blocks are: front-end, input register, pipeline, derandomizing buffer, command decoder, readout logic, threshold&calibration control.

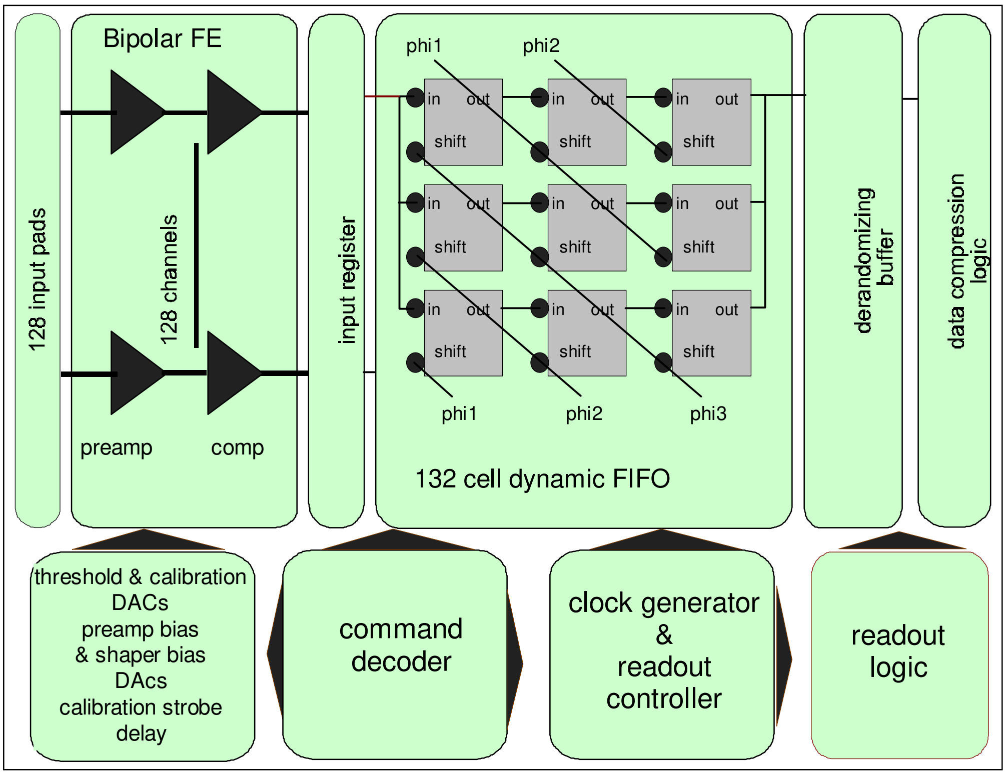


Figure 3.1. Block diagram of the ABCD chip.

3.1.2 Signal processing.

The chip must contain following functions:

1. Charge integration
2. Pulse shaping
3. Amplitude discrimination. The threshold value for the amplitude discrimination is provided as a differential voltage either from internal programmable DAC or from an external source.
4. The outputs of the discriminators must be latched either in the edge sensing mode or in the level sensing mode.
5. At the start of each clock cycle the chip must sample the outputs from the discriminators and store these values in a pipeline until a decision can be made whether to keep the data.
6. Upon receipt of a L1 Trigger signal the corresponding set of values together with it's neighbours are to be copied into the readout buffer serving as a derandomizing buffer.
7. The data written into the readout buffer is to be compressed before being transmitted off the chip.
8. Transmission of data from the chip will be by means of token passing and must be compatible with the ATLAS protocol.

9. The chip is required to provide reporting of some of the errors that occur:
   a) Attempt to read out data from the chip when no data is available.
   b) Readout Buffer Overflow: The readout buffer is full and data from the oldest event(s) has been overwritten.
   c) Readout Buffer Error: The readout buffer is no longer able to keep track of the data held in it. (Chip reset required).
   d) Configuration error (ChipID sent).
10. The chip shall incorporate features that will enable the system to continue operating in the event of a single chip failure.
11. It is a system requirement that the fraction of data which is lost due to the readout buffer on the chip being full is less than 1%. This assumes that on average only 1% of the silicon strip detectors are hit during any particular beam crossing.

3.1.3 Calibration and testability.

Each channel has an internal Calibration Capacitor connected to its input for purposes of simulating a "hit" strip. The Calibration Capacitors are charged by an internal chopper circuit which is triggered by a command. Every fourth channel can be tested simultaneously with group selection determined by two binary coded Calibration Address inputs (CALD0, CALD1). The strobe and the address signals are delivered from the control circuitry. The voltage applied to the Calibration Capacitors by the chopper is determined by an internal DAC. The four calibration bus lines, each of which connects the calibration capacitors of every fourth channel, are also brought out to pads which can be directly driven with an AC coupled voltage step. This is intended for use during IC testing. A tuneable delay of the calibration strobe with respect to the clock phase covering at least two clock periods must be provided. The chip must incorporate such features that will enable to test and calibrate it either on the wafer level or in situ.

3.1.4 Compatibility

The design of the ABCD3T chip must be functionally and electrically compatible with the ABCD2T prototype. It is required that the physical dimesions and the pad layout must be identical with the ABCD2T design.

3.2 Specification

3.2.1 Detector parameters

The parameters of the analogue front-end part are specified for the electrical parameters of 12.8 cm long p-type silicon strip detector. The assumed detector parameters are listed in Table 3.1.

Table 3.1: Assumed detector electrical parameters.

	Unirradiated	Irradiated
Coupling type to amplifier	AC	AC
Coupling capacitance to amp Total for 12 cm strips	20 pF/cm 240 pF	20 pF/cm 240 pF
Capacitance of strip to all neighbour strips	1.03 pF/cm	1.40 pF/cm
Capacitance of strip to backplane	0.30 pF/cm	0.30 pF/cm
Metal strip resistance	15 Ω/cm	15 Ω/cm
Bias Resistor	0.75 MΩ	0.75 MΩ
Max leakage current per strip for shot noise	2.0 nA	2.0 μA
Charge collection time	< 10 ns	< 10 ns

3.2.2 Front-end

3.2.2.1 Electrical Requirements:

Note that notation convention for currents used in the entire specification is "+" for current going into (sunk by) the chip and "-" for current going out of (sourced from) the chip.

3.2.2.2 Input Characteristics:

Input Signal Polarity: Positive signals from p-type strips.

Crosstalk: < 5% (via detector interstrip capacitance)

Input Protection: Must sustain voltage step of 450 V of either polarity with a cumulative charge of 5 nC in 25 ns.

Open Inputs: Any signal input can be open without affecting performance of other channels.

Max Parasitic Leakage Current: 100 nA DC per channel with < 10 % change in gain at 1 fC input charge.

3.2.2.3 Preamplifier-Shaper Characteristics

Gain at the discriminator input: 50 mV/fC for the nominal shaper current of 20 μA and the nominal process parameters

Linearity: better than 5% in the range 0 - 4 fC

Peaking time: 20 ns

Noise: Maximum rms noise for nominal components as measured on fully populated modules

<= 1500 electrons rms for unirradiated module

<= 1800 electrons rms for irradiated module

Gain Sensitivity to VCC for 1 fC signal: 1%/100mV

Power Supply Rejection Ratio at:
(not design targets but simulation results of the circuit)

10 Hz - 100 Hz	60 dB
10 kHz - 100 kHz	20 dB
10 MHz - 60 MHz	-14 dB

3.2.2.4 Comparator Stage:

A threshold is applied as a differential voltage offset to the comparator stage. This threshold voltage is applied from an internal DAC in the normal operation mode or can be applied from the external pads for test purposes.

Threshold setting range:	0 fC to 12.8 fC , nominal setting at 1 fC
Threshold setting step:	0.05 fC of input charge around nominal threshold of 1 fC
Threshold variation at 1 fC:	(1 sigma) channel to channel matching within one chip vs Range Set of TrimDACs by 2 bits in the Configuration Register (See Table 3.1a min (00) 2.5% x2 (01) 5.0% x3 (10) 7.5% x4 (11) 10%

3.2.2.5 Timing Requirements:

Timewalk:	<= 16 ns. This specification depends on the precision of the digital acquisition latch edge. Good alignment, 1 or 2 ns over a common clocked array of channels implies a longer timewalk assignment to the rising edge of the shaped signal.
Timewalk defined:	The maximum time variation in the crossing of the time stamp threshold over a signal range of 1.25 to 10 fC, with the comparator threshold set to 1 fC.
Double Pulse Resolution:	<= 50 ns for a 3.5 fC signal followed by a 3.5 fC signal

Max recovery time for a 3.5 fC signal following a 80 fC signal: 1 µs

3.2.2.6 Threshold Generation Circuit

Differential voltage for the discriminator threshold is generated by an internal DAC circuit (Threshold DAC). The threshold voltages generated by the internal circuit are applied to the same pads VTHP and VTHN to which the external threshold is applied. When the external threshold is not applied the internal threshold voltage can be measured at pads VTHP and VTHN.

Range:	0 - 640 mV
Step value:	2.5 mV
Absolute accuracy:	1%

3.2.2.7 Calibration Circuit Characteristics

Calibration signal can be applied to one of the four calibration lines via the external pads or from the internal calibration circuit. In the later case the address of the calibration line, the amplitude of the calibration signal and its delay is set via the control logic.

Calibration Capacitors: 100 fF ±20% (3 sigma) over full production skew ±2% (3 sigma) within one chip.

Calibration signal:

amplitude range: 0 - 160 mV (charge range: 0 - 16 fC)

amplitude step: 0.625 mV (charge step: 0.0625 fC)

Absolute accuracy of amplitude: 5% (full process skew)

Relative accuracy of amplitude: < 2 % (for known values of calibration capacitors, amplitude range 0.8 to 4 fC, across one chip, including switching pickup, etc.)

Relative accuracy of amplitude: < 10 % (for known values of calibration capacitors, amplitude range 0.8 to 8 fC, across one chip, including switching pickup, etc.)

Calibration Strobe signal pickup at comparator should be less than 0.1 fC equivalent sensor input.

For test purposes, a voltage step can be applied directly to any one of the four groups of calibration capacitor via the input pads (CAL0, CAL1, CAL2, CAL3). When not used, these four pads must be left floating.

3.2.2.8 Threshold Correction Circuit

In order to compensate channel-to-channel variation of the discriminator offset each channel is provided with a trim DAC of 4-bit resulution. Each channel can be addressed individually and the threshold correction can be applied channel by channel. The range of the trim DAC can be selected with two bits in the configuration register (see Table 3.10b). This is to cover the offset spread which is expected to increase after irradiation.

Range of the trim DAC: four selectable ranges (see Table 3.1a)

Step value: see Table 3.1a.

Absolute accuracy: 10%

Table 3.1a: Trim DAC range selection

Trim DAC range Bit 1	**Trim DAC range Bit 0**	**Trim DAC range**	**Trim DAC step**
0	0	0 mV - 60 mV	4 mV
0	1	0mV -120 mV	8 mV
1	0	0mV -180 mV	12 mV
1	1	0mV -240 mV	16 mV

3.2.3 Input Register and Mask Register

The functions of the input register and mask register will be implemented in a single functional block. The input/output connections to this block are shown in Fig 3.2.

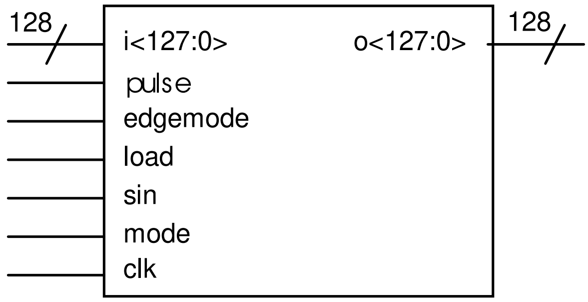


Figure 3.2. Input Register Inputs/Outputs

Table 3.2: Input Register I/O Signal Definitions

Signal Name	**Active State/Edge**	**Function**
i<127:0>		Hit Inputs (from input translators)
load	Active High	Load Mask Register
sin		Configuration Data Inputs
mode	see Table 3.3	
clk	Pos Edge	
o<127:0>		Data Outputs (to pipeline)
edgemode	High	Enables edge detection logic
pulse	High	Pulse all outputs simultaneously

3.2.3.1 Input Register

This register latches the incoming data, delivering a well defined pulse width to the pipeline.

3.2.3.2 Edge Detection Circuitry

The function of this block is to detect a high to low transition in the data entering the pipeline, and for each of such transition found the circuit generates a pulse of duration 1 clock cycle irrespective of the length of the incoming pulse. The effect of this block is that only a single ‘1’ is written into the pipeline for every hit detected regardless of the response time of the discriminator. This circuitry can be turned on or off by setting the appropriate bit in the configuration register.

3.2.3.3 Channel Masking Register

This register serves a dual purpose. Firstly, the Channel Mask register enables any bad or noisy channels to be turned off thus preventing them from increasing the data rate due to false hits. Secondly, it can be used during chip testing to apply a set of test patterns to the pipeline. In the test mode the inverted test pattern appears at the output of the pipeline. The contents of this register can be changed by sending the appropriate control command to the chip. A channel is masked with ‘0’. The test pulse is masked by the mask register as well.

Table 3.3: Masking Register Modes of Operation

mode	**Mode of Operation**
0	Normal Data Taking (Contents of register used to "Mask Inputs")
1	Test Mode (Contents of mask register are used to supply test values to pipeline)

3.2.4 Pipeline

The binary pipeline is realised as a multiplexed FIFO circuit. An array of *12x12* dynamic memory cells is controlled by *12* non-overlapping clock signals. In each clock cycle only *12* cells out of *12x12* are switched while the effective delay provided by such a block is equal *12x(12-1)* clock cycles. The pipeline block comprises 128 channels, each 132-bit deep. The hit pattern from the input register is shifted through the pipeline during 132 clock cycles. When a L1 trigger arrives, the hit-pattern from the last three time bins are copied into the readout buffer. Incorporated into the pipeline is the accumulator register. This function can be enabled and disabled by setting a bit in the configuration register. With a command “**reset**” the clock generator is reset while the contents of the pipeline remains unchanged. The input/output connections to the pipeline block are shown in Fig. 3.3.

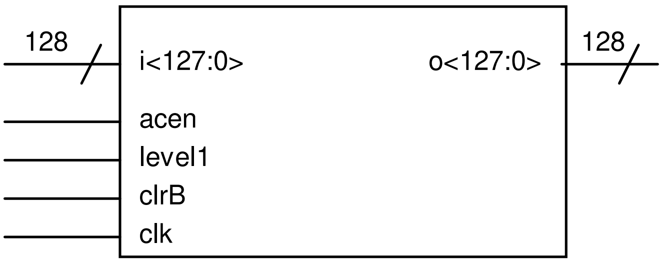


Figure 3.3. Pipeline Input/Outputs.

Table 3.4: Pipeline Input/Output Signal Definitions.

Signal Name	**Active State/Edge**	**Function**
i<127:0>		Data Input
acen	High	Enables Accumulator Register
level1	High	Reads Value out of pipeline
clrB	Low	Initialises pipeline pointers and clears accumulator register
clk	Pos edge	Clock input
o<127:0>		Data Output

3.2.5 Readout Buffer

Data corresponding to each L1 trigger will be held in a Readout buffer pending readout. This data buffering is needed to remove the statistical fluctuations in the arrival time of L1 triggers. Data compression and read out will be started only when this buffer is not empty. Three bits of data will be stored in this buffer for each channel per L1 trigger. These bits represents the three beam crossings centred on the L1 trigger time and are set if the input was above threshold during the corresponding crossings. In the case when the Accumulator Register has been enabled, the contents of this register will be copied into the buffer 3 times resulting in the same amount of data being written into the readout buffer regardless of the operating mode. This buffer will be 128 bits wide by 24 locations deep. This is sufficient to hold the data from eight L1 triggers. This satisfies the ATLAS specification of maintaining <= 1% data loss at a L1 trigger rate of 100 kHz and a strip occupancy of up to 1%.

3.2.5.1 FIFO

This buffer will be implemented as a "barrel store", i.e. it will be addressed by 2 cycling pointers, a write pointer and a read pointer. Once a pointer has reached the end of the block of RAM, it will return to the beginning of the block of RAM the next time it is incremented.

The write pointer will be allowed to go past the read pointer and over-write data that has not yet been read out. However, if this happens the Overflow flag will be set to indicate that data has been over-written. The read pointer will not be allowed to pass the write pointer and if the read pointer should catch up with the write pointer the EMPTY flag will be set. This is to prevent attempts to readout the buffer when there is no data in it.

3.2.5.2 Accumulator Register

The Accumulator Register is implemented in the Readout Buffer. This register marks all channels that have been hit since it was last cleared. If the Accumulator Mode is selected in the configuration register, a L1 trigger results in the transfer of the contents of this accumulator into the readout buffer instead of the appropriate time bin of the pipeline. This accumulator column is cleared by a power-up or soft ”**reset”** command. As the content of the pipeline is not cleared with the ”**reset”** and the accumulator register reads the data from the pipeline two ”**reset”** signals separated at least by 132 clock cycles are required to clear the Accumulator Register completely.

3.2.5.3 Overflow Counter

A counter will be used to track the number of events that have been over-written in the buffer. This counter will be incremented for every time an event is written into the buffer while the buffer is full. The outputs from this counter represents the number of events from which data has been lost. This counter is decremented for every event that is readout of the buffer, until it's value reaches zero. Once its value has reached zero, it is no longer decremented. This is because in this state all the events from which data have been lost will have been cleared and none of the data in the buffer will have been overwritten. Should this counter overflow, the ERROR flag will be set. This will occur after 16 events have been overwritten. This flag will remain set until either a software reset, or power-up resets has been issued to the readout buffer and associated logic.

3.2.5.4 Flag Logic

Three signals DATA_AVAIL, OVERFLOW and ERROR are produced by the readout buffer. DATA_AVAIL indicates when there is data in the buffer to be readout. This signal is used by the Data Compression logic to determine when to start a readout cycle. OVERFLOW indicates when data in the buffer has been overwritten and hence data lost. OVERFLOW occurs when the buffer contains more than 8 events i.e. 24 samples . This signal is sent to the readout logic which results in the readout logic sending an error message to say that data from the current event being readout has been lost. Finally, the ERROR signal is generated when the buffer has overflowed and it has also lost track of the number of events from which data has been lost. This simulation occurs if the data from more than 16 events have been lost. This flag can only be cleared by issuing a reset to the chip.

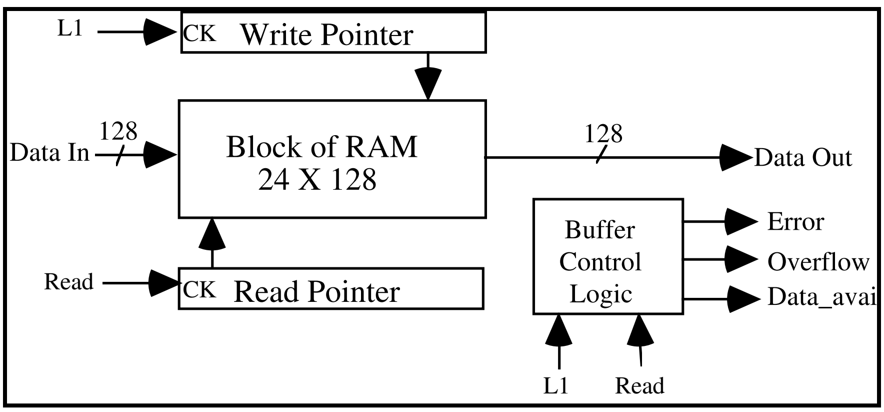


Figure 3.4. Block Diagram of the Readout Buffer.

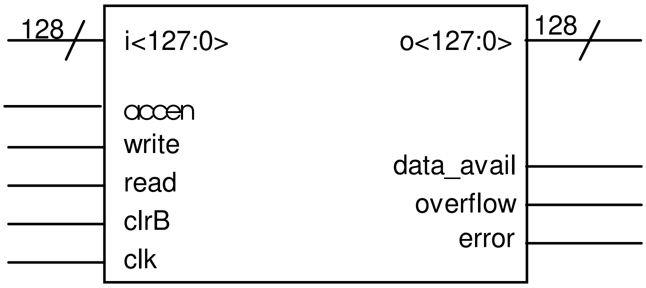


Figure 3.5. Readout Buffer Input/Outputs.

Table 3.5: Readout Buffer Input/Output Signal Definitions.

Signal Name	Active State/Edge	Function
i<127:0>		Data Input
write	High	Write value into readout buffer
read	High	Reads value from readout buffer
clrB	Low	Resets buffers pointers and counter
clk	Pos edge	Clock input
o<127:0>		Data Output
data_avail	High	Data available in buffer
overflow	High	Buffer Overflow
error	High	Buffer Error

3.2.6 Data compression logic

It is anticipated that on any event very few channels will contain hits. This fact can be used to reduce the number of bits of data that have to be read out of the chip for each event. The data compression logic works by examining in turn the 3 bits of data that make up the hit pattern for each channel. Each group of 3 bits is compared against one of 4 selectable criteria. If the pattern meets the criteria, then the hit pattern from that channel is sent to the readout circuitry for transmission, if not, no data is sent from that channel and the hit pattern from the next channel is examined. This process is repeated until the hit patterns from all 128 channels have been examined. The following table shows the 4 selection criteria (currently there are only plans to use 3, the 4th is for chip testing only).

Table 3.6a: Data Compression Criteria.

mode(1:0)	Name of Selection Criteria	Hit Pattern (Oldest data bit 1st)	Usage
00	Hit	1XX or X1X or XX1	Detector alignment
01	Level	X1X	Normal Data Taking
10	Edge	01X	Normal Data Taking
11	Test	XXX	Test Mode

N.B. X = Don't care state.

This block operates as follows.

As soon as the chip receives a L1 trigger, the three 128-bit words that make up an event are written into the read out buffer. This results in the empty flag on the readout buffer being negated, indicating that there is data to be processed. The data compression logic monitors the state of this flag until it finds that there is data available. Providing that it is not already processing data, it then proceeds to read in the three 128-bit words that make up an event from the readout buffer.

The next thing that happens is that the data compression logic re-arranges the order of the data from being 3 128-bit words into 128 3-bit words. The reason for this is that the data compression algorithm requires all 3 samples of an event to be examined in parallel.

The data compression logic then starts to scan through all the channels in turn until it finds one which has a pattern of hits which matches the data selection criteria. If it finds such a pattern of hits, it asserts the "datavalid" signal and places the pattern of hit bits on the "hit<2:0>" outputs and places the address of the hit channel on the address outputs "ch<6:0>". The logic then waits until the readout logic signals it to proceed by asserting the "next" input. The data compression logic responds to "next" by presenting the address and data for the next hit found if any. If the next hit found is on the next adjacent channel, the "adj" is asserted with the data from the previous channel. If no more hits are found, the "end" signal is asserted.

In certain situations, e.g. in the case of a overflow of the readout buffer, it is not necessary for the data compression logic to process the data from the readout buffer but it is still necessary for it to read the 3 values from the buffer in order to flush them from the readout buffer. There are 3 cases when this happens, these are listed below in order of priority.

1) When the chip is in its SEND_ID mode of operation.

2) When the Readout Buffer error flag has been set.

3) When the Readout Buffer overflow flag has been set.

The following table shows how the datavalid, end and overflowout outputs are used to indicate the status of the data compression logic.

Table 3.6b: Data Compression Logic Output States.

datavalid	**end**	**overflowout**	**condition**
low	low	low	no events available to be read out i.e. readout buffer empty.
high	low	low	data from hit channel waiting to be read out. (not last channel)
high	high	low	data from last hit channel waiting to be read out.
low	high	low	all hits read out or no hits found
low	low	high	data for event lost due to readout buffer overflow

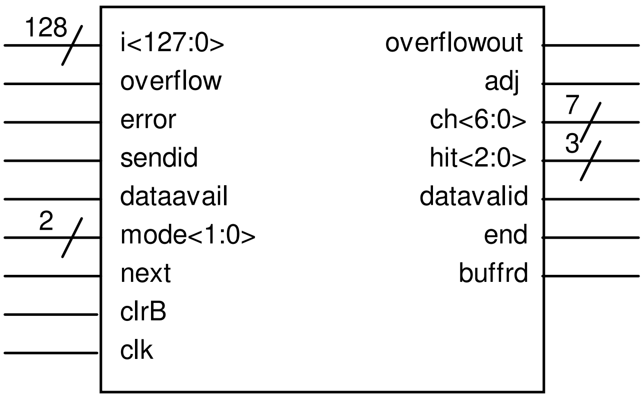


Figure 3.6. Data Compression Logic Input/Outputs.

Table 3.6c: Data Compression Logic Input/Output Signal Definitions.

Signal Name	**Input/Output**	**Active State/Edge**	**Function**
i<127:0>	input		Data Input
overflow	input		Overflow output from readout buffer
error	input		Error output from readout buffer
sendid	input		indicates chips mode of operation
dataavail	input	High	Data available to be readout
mode<1:0>	input		Selects data compression mode
next	input	High	Find next hit channel.
clrB	input	Low	Resets logic
clk	input	Pos Edge	Clock Input
overflowout	output	High	Overflow output to readout circuitry
adj	output	High	Next Hit found on adjacent channel
ch<6:0>	output		Channel address of Hits
hit<2:0>	output		Hit Data pattern
datavalid	output	High	Hit Data outputs valid
end	output	High	Last Channel scanned
buffrd	output	High	Reads Value out of Readout Buffer

3.2.7 Readout Circuitry

The readout circuitry will be responsible for capture and release of the token and outputting data from the chip. The readout circuit always waits until the token arrives. On arrival of the token, it checks if any hits have been found by the data compression logic. If so, it then outputs the appropriate header information. It then proceeds to output the address of the hit channel together with the data from that channel. Once the readout circuitry has finished sending the data from one channel, it proceeds to output the data from then next channel. In the situation where one or more neighbouring channels are to be read out, only the address of the first channel is output., but the data from all hit channels is sent. This process continues until the data compression logic indicates that all channels have been examined by asserting "end". Once all the data corresponding to a single event has been read out, a token is sent out to the next chip in the readout chain. This token will be sent out ahead of the last bit of data sent out. If the chip has no data to be readout, circuitry sends out a "No hit data" code and passes the token on to the next chip in the chain.

If the chip is in "send-id" mode or the readout buffer has overflowed or generated an error condition, the readout circuitry sends the appropriate error packet or configuration data packet. In these cases the readout circuitry is still required to signal to the data compression logic that it has processed an event by asserting the "next" signal. This operation is needed so that a correct count of the number of events waiting to be read out can be maintained.

In the case of an error condition occurring, e.g. attempt to readout data and no data available, the appropriate error code will be sent by the readout logic. If the chip is in the "send_id" mode of operation, no data or error codes are output from the chip but instead a special packet of data containing information about the chips current configuration is sent.

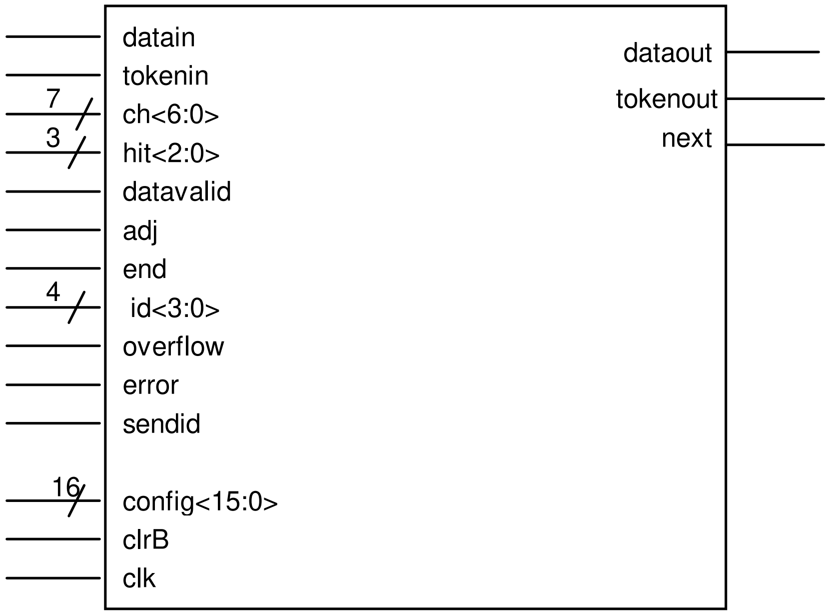


Figure 3.7. Connections to Readout Circuitry.

Table 3.7: Readout Logic Input/Output Signal Definitions.

Signal Name	**Input/Output**	**Active State/Edge**	**Function**
datain	input		Data Input
tokenin	input	High	Token Input
ch<6:0>	input		Address of Hit Channel
hit<2:0>	input		Hit data pattern
datavalid	input	High	Data available for sending
adj	input	High	Hit found on adjacent channel
end	input	High	End of data to be sent
id<3:0>	input		LS 4 bits of chip address
overflow	input	High	Readout buffer Overflow
error	input	High	Readout Buffer Error
sendid	input		Chip mode of operation
config<15:0>	input		Data from config-reg
clrB	input	Low	Resets circuit
clk	input	Pos Edge	Clock input
dataout	output		Data output
tokenout	output		Token Output
next	output	High	Scan Next Channel

3.2.8 Readout Controller Block

This block is to control the readout of data from several ABCD chips connected together in a token chain. This block is enabled by placing the chip in "Master Mode". This block has to detect when a L1 trigger has been received, issue a token to all the ABCD chips connected to it, collect all the data from the chips and tag the data with the beam crossing number from which it came and the number of L1 Trigger. This block then has to transmit this data serially to the LED driver chip.

3.2.8.1 L1 Counter

This a 4-bit binary counter which is incremented every time the chip receives a L1 trigger. The counter is zeroed by either a hardware reset or a software reset.

3.2.8.2 Beam Crossing Counter

This is an 8-bit binary counter which is incremented on every clock cycle. This counter is zeroed by either a hardware reset, a software reset, or a special BC Reset Command

3.2.8.3 Event FIFO

This is a 24 location deep, 12-bit wide FIFO. Each time the chip receives a L1 trigger, the output of the L1-Counter and the Beam crossing counter are loaded into the FIFO prior to the counters being incremented. These values are read from the FIFO every time an Event is readout and are used to tag the data with 12-bits of information about which trigger number and beam crossing number the data came from.

3.2.8.4 Token Generation Logic

The purpose of the token generation logic is to detect when the chip has received an L1 trigger and to generate a token to initiate the read out of data from that L1 trigger. This logic waits until the Event FIFO becomes not empty and it then issues a token. It then monitors the data passing through it from all the chips in the chain looking for a "Trailer" bit pattern. It waits until this trailer is detected before checking to see if the Event FIFO is empty. If the Event FIFO is still not empty it repeats the cycle.

3.2.8.5 Data Formatting Logic

The purpose of this logic is to attach the header information to the packets of data output from the chip on the Serial Data Output.

3.2.8.6 Serial Data Output Driver

This is the output from the chip outputs its data to the DAQ system.

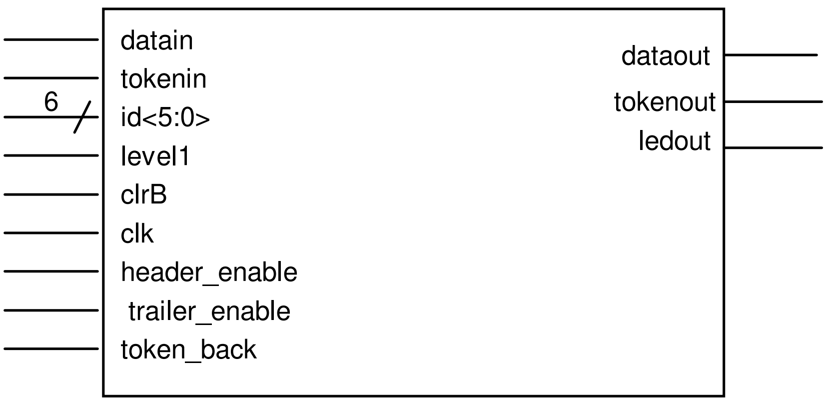


Figure: 3.8. Connections to Readout Controller Circuitry.

Table 3.8: Readout Controller Input/Output Signal Definitions.

Signal Name	Active State/Edge	Function
datain		Serial Data Input
tokenin	High	Token Input
id<5:0>		Address of chip
level1	High	L1 Trigger
clrB	Low	Resets block
clk	Pos Edge	Clock input
dataout		Serial data output
tokenout	High	Token Output
header_enable	High	Enables Generation of Packet Header
trailer_enable	High	Enables Generation of Packet Trailer
token_back	High	Input for Token output from ROL
ledout		Serial data out to LED driver

3.2.9 Command Decoding

The command and control information all comes into the chip on the command input pins. There are two main classes of information which arrive here, L1 Triggers Commands and Control Commands. These are distinguished by a 3-bit code. Furthermore two types of Control Commands are possible, Fast Control Commands and Slow Control Commands. Depending on which class arrives, further information may follow. This further information will also need decoding, formatting and sending to the appropriate functional blocks of the chip. More detailed information is contained in 3.2.20 and the actual data fields of the commands are listed in Tables 3.20 and 3.21. The two classes of Commands and two types of Control Commands are:

3.2.9.1 L1 Trigger Command

If the 3-bit code indicating this command is received, the control logic writes 3 samples from the pipeline or the accumulator register, into the Readout Buffer.

3.2.9.2 Control Commands

If the 3-bit code indicating a Control Command is received, the second field of 4 bits is decoded to determine if it is a Fast Control Command or a Slow Control Command. If a Fast Control Command is decoded, the appropriate command is executed. No address or data fields are included in these commands..

3.2.9.3 Slow Control Command

If the second field of the command is decoded to be a Slow Control Command, the third, forth, fifth and possibly sixth field is decoded to determine the full action required. These Slow Control Commands are of variable length and the contents of the third field determines the total number of bits to be processed.

The command decoder block is required to decode the command and send the relevant instruction and data to other parts of the chip. The input/output connections of the Command Decoder are shown in Fig. 3.9.

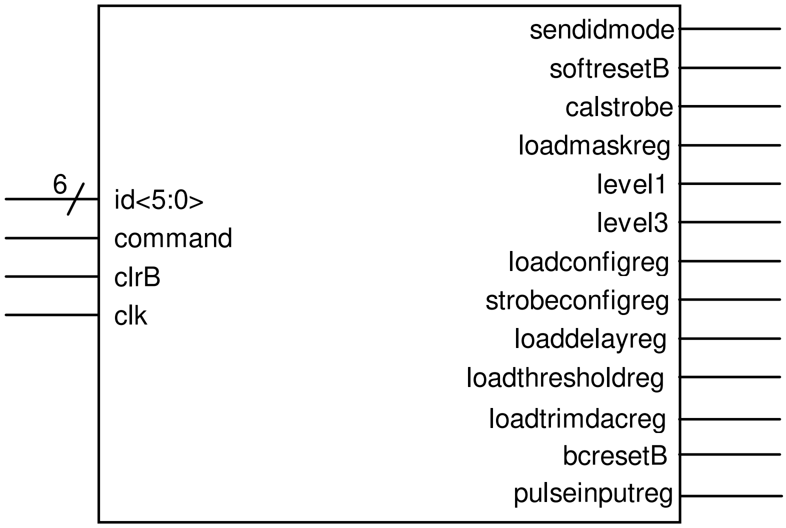


Figure 3.9. Command Decoder Inputs/Outputs.

Table 3.9: Command Decoder Input/Output Signal Definitions

Signal Name	**Active State/Edge**	**Function**
id<5:0>		Chip ID
command		Command Data Input
clrB	Low	Reset Block
clk	Pos Edge	Clock Input
sendidmode		Sets chips mode of operation
softresetB	Low	Software controlled reset
calstrobe	High	Send Calibration pulse
loadmaskreg	High	Load Mask Register
level1	High	L1 Trigger received
level3	High	Same as level1 but High for 3 cycles
loadconfigreg	High	Loads Configuration reg.
strobeconfigreg	High	Strobes data into configuration reg
loaddelayreg	High	Load Strobe Delay Register
loadthresholdreg	High	Load Threshold Register
loadtrimreg	High	Load TrimDac Register
bcresetB	Low	Beam crossing reset

3.2.10 Configuration Register

This is a 16-bit register which is used to hold information about the chip's current configuration. The following table defines the usage of the bits in this register. The power up value of this register will be zero. The input/output connections to this block are shown in Fig. 3.10. Data is shifted into this register MS bit first. The contents of this register are not effected by a software reset command.

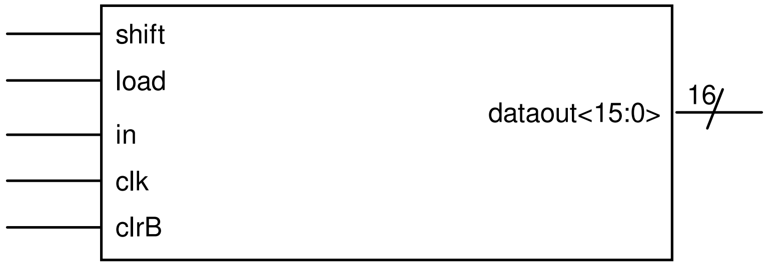


Figure 3.10. Configuration Register Inputs/Outputs.

Table 3.10a: Configuration Register Input/Output Signal Definitions.

Signal Name	**Active State/Edge**	**Function**
load	Pos Edge	Transfers data to register outputs
in		Serial Data input
shift	High	Enables data to be shifted into reg
clrB	Low	Resets register to default values
dataout<15:0>		Data Outputs See section 3.2.5

Table 3.10b: Configuration Register Contents.

Bit	**Name**	**Function**
0-1	Readout Mode	Selects the data compression Criteria (see Table 3.6a)
2-3	Cal_Mode<1:0>	Selects the Calibration code (see Table 3.11). The state of these two bits also determines which channels are tested when Test Mode is enabled.
4-5	Trim DAC range <1:0>	Selects the range of the trim DAC (see Table 3.1a)
6	Edge_Detect	Enables the edge detection circuitry in the input stage is enabled.
7	Mask	When this bit is set the input register is disabled and the contents of the mask register are routed into the L1 pipeline.
8	Accumulate	When this bit is set the Accumulate function is enabled.
9	Input_Bypass	This bit determines which set of token/data inputs are active.
10	Output Bypass	This bit determines which set of token/data outputs are active.
11	Master	When clear the chip acts as a Master providing the masterB input pin has be asserted. This bit is "ored" with the value on the "masterB" input. If the result is "0", the chip is placed into master mode. Otherwise, it is placed into slave mode.
12	End	When set this bit configures the chip as the end of a readout chain.
13	Feed_Through	When clear the chip outputs a 20MHz clock signal but only is the chip has been configured as a Master (see above)
14-15	not used	

3.2.11 Calibration Logic

The calibration logic produces a calibration strobe signal for the front-end calibration circuit. This strobe is produced in response to a control command when the "Cal Enable" bit is also set in the configuration Register. A two-bit calibration code is also sent to the calibration circuit which selects one of the four possible patterns in the front-end. The two-bit calibration code outputs are single-ended CMOS levels. The calibration Strobe signal must be sent to the front-end a fixed number of clock pulses after receipt of the control command. The delay from the rising edge of the clock signal to the rising edge of the strobe signal is determined by the value loaded into the Strobe Delay Register. This delay can be adjusted in 64 equal steps over a range of 50 ns.

Table 3.11: Calibration Codes.

Cal enable Bit	Cal Mode Bit 1	Cal Mode Bit 0	Channels of front-end pulsed
0	X	X	Calibration disabled
1	0	0	in3, in7 in11,...in127
1	0	1	in2, in6 in10,...in126
1	1	0	in1, in5 in9,...in125
1	1	1	in0, in4 in8,...in124

3.2.12 Strobe Delay Register

The Strobe Delay Register is a 8 bit register or which only the least significant 6 bits are used. The value stored in this register determines the relative delay between the rising edge of the Calibration Strobe output and the rising edge of the clock input. This delay can be set in 64 steps, of approximately 0.8ns ±0.2ns each. Thus enabling the delay of Calibration Strobe to be swept thought at complete clock cycle at 40MHz. Data is shifted into the register with the MS bit first.

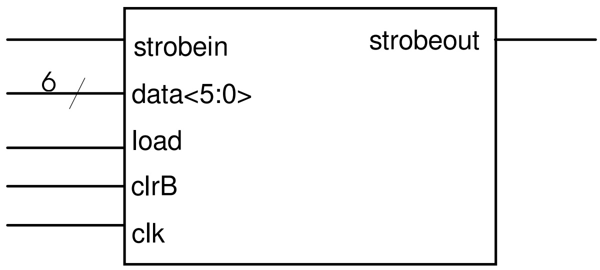


Figure 3.11. Strobe Delay Register Inputs/Outputs.

Table 3.12: Strobe Delay Register Input/Output Signal Definitions.

Signal Name	Active State/Edge	Function
strobein		Strobe input
data		Data input to register
clrB	Low	Resets register
clk	Pos Edge	Clock input
load	High	Loads delay value into register
strobeout		Delayed version of STROBE_IN

The value of delay is determined according to the following formula.

delay = min_delay + (register_value x step_value)

Where :

min_delay is the delay produced when the register is set to zero .

register_value is the value written into the delay register (least significant 6 bits only)

step_value is the increase in delay produced by incrementing the contents of the delay register (typically 0.8 ns.)

3.2.13 DAC Registers

There are two 16 bit register which holds 2 8-bit values each.

The Threshold/Calibration register holds a threshold value and a calibration value. The threshold value is held in the MS byte of this register and a calibration amplitude value is held in the LS byte of this register. The outputs of this register are used to control 2 separate 8-bit DACs. The outputs from these DACs supply 2 independent DC current levels to the threshold generation circuit and the calibration circuit. The MS bit is shifted into this register first.

Figure 3.12. Threshold/Calibration DAC Register Inputs/Outputs.

Table 3.13. Threshold & Calibration DAC Register - Input/Output Signal Definitions.

Signal Name	**Active State/Edge**	**Function**
load	Pos Edge	Load Values into register
data		Data input to register
clrB	Low	Resets register
clk		clock input to register
threshold		Binary value of Threshold Output from 1st DAC
cal amp		Binary value of Calibration Amplitude Output from 2nd DAC

The Bias register holds a bias current value in the input transistor and a shaper bias current value. Bits 4:0 of the register are used to set the DAC which controls the shaper bias current. Bits 12:8 are used to set the DAC which controls the bias current in the input transistor. These Bit ranges have been chosen to align the data for both DACs on byte boundaries. The MS bit is sent first.

Figure 3.13. Bias DAC Register Inputs/Outputs.

Table 3.14. Bias DAC Register Input/Output Signal Definitions.

Signal Name	**Active State/Edge**	**Function**
load	Pos Edge	Load Values into register
data		Data input to register
clrB	Low	Resets register
clk		clock input to register
ish		Binary value of Shaper bias current from 1st DAC
ipre		Binary value of Bias current in the input transistor from 2nd DAC

3.2.13A Trim DAC Register

The TrimDac register holds an address of the channel and a threshold correction value for the given channel. Bits 3:0 of the register are used to set the DAC which controls the offset of the comparator for the given channel. Bits 10:4 are used to set address of the channel . The MS bit is sent first.

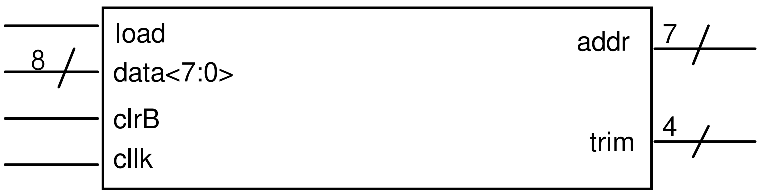


Figure 3.13A. Trim DAC Register Inputs/Outputs.

Table 3.14A. Trim DAC Register Input/Output Signal Definitions.

Signal Name	**Active State/Edge**	**Function**
load	Pos Edge	Load Values into register
data		Data input to register
clrB	Low	Resets register
clk		clock input to register
addr		Channel address to be trimmed
trim		Binary value of the threshold correction value from the TrimDac

3.2.14 DACs

On the ABCD chips there are:

two 8-bit DACs which are used to set a threshold and calibration values. The reference current for these DACs is generated internally in the chip. This reference current will be scaled at the output of the DAC by a value of 0 to 255 depending on the setting of the DAC register.

two 5-bit DACs for setting the current in the input transistor the shaper bias current which will allows to control the shaper gain. The reference current for these DACs is generated internally in the chip. This reference current will be scaled at the output of the DAC by a value of 0 to 31 depending on the setting of the DAC register.

Table 3.15. Threshold & Calibration DACs - range and resolution

	Range	**Resolution**
Calibration signal amplitude	0 - 160 mV	0.625 mV/step
Calibration charge injected via 100 fF cap	0 - 16 fC	0.0625 fC/step
Threshold voltage	0 - 640 mV	2.5 mV/step

Table 3.16. Bias DACs - range and resolution

	Range	**Resolution**
Input transistor current	0 - 294.4 μA	9.2 μA/step
Shaper current	0 - 38.4 μA	1.2 μA/step

3.2.15 Clock and Command Inputs

Two sets of clock and command inputs will be provided in order to make the system in which the "ABCDs" will be used fault tolerant and to provide an additional method of setting up the timing of the system. Each chip will be supplied with two independent sources of clock and commands. In the event of the fall out of one of these sources, the alternative source can be used. An external input to the chip "select" will be used to determine which pair of inputs will be used by the chip.

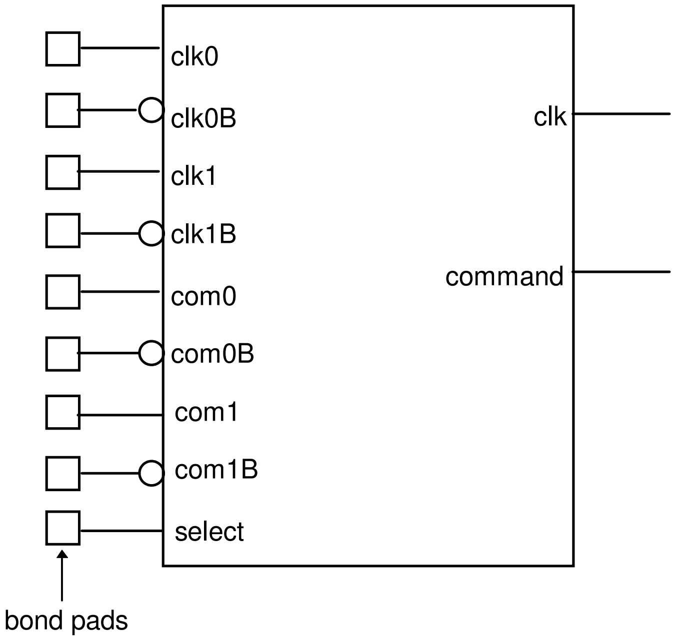


Figure 3.14. Clock & Command Data Inputs.

Table 3.17a. Clock Input/Output Signal Definitions.

Signal Name	**Active State/Edge**	**Function**
clk0		default Clock Input
clk0B		Complement of above
clk1		reserve Clock Input
clk1B		Complement of above
com0		default Command Input
com0B		Complement of above
com1		reserve Command Input
com1B		Complement of above
select		Selects which pair of inputs to use
clk		Clock output
command		Command Data Output

Table 3.17b: Clock Input Modes of Operation.

select	**clk**	**command**
Low	clk0	com0
High	clk1	com1

3.2.16 Chip ID

To enable a chip to be individually addressed five inputs ID(4:0) will be used to implement a geographical addressing scheme. This is because there will be a total of 12 chips on each module (6 per side), and under certain conditions it may be necessary to address all the chips on 2 modules using the same control line. These inputs will be wire bonded to a unique set of logic levels on each chip mounted on the detector module. This set of logic levels will form a geographical address which will enable individual chips on the module to be addressed. The chips will be bonded according to the following scheme. For historical reasons the command decoder on the chip has been designed to decode a 6-bit geographical address however the MS bit of the chips geographical address is always set to a 1. The sixth bit ID(5) is not brought out of the chip but instead set to a logic 1 level. The bit ID(4) is set to a logic 0 by internal pull-down and will be tied to the same line on the hybrid as Select. When module is working with the primary fibre Select and ID(4) will be set to a logic 0. When Select is activated to a logic 1 simultaneously the ID(4) will be set a logic 1. ID(3:0) when left unconnected will be forced to go to o logic 1 by internal pull-ups.

Table 3.18a: Geographical Addresses ID(5-0).

ID<5:0>	**Type of Chip Selected**	**Module**
10aaaa	ABCD	Served by primary fibre
11aaaa	ABCD	Served by adjacent fibre
1x1111	All ABCD chips	Both

N.B aaaa is the 4 bit address of the ABCD chip on the hybrid see Table 3.18b

Table 3.18b: ABCD Geographical Addresses (ID3-0).

Chip Position on Hybrid *	**ID(3)**	**ID(2)**	**ID(1)**	**ID(0)**
M0	LOW	LOW	LOW	LOW
S1	LOW	LOW	LOW	HIGH
S2	LOW	LOW	HIGH	LOW
S3	LOW	LOW	HIGH	HIGH
S4	LOW	HIGH	LOW	LOW
E5	LOW	HIGH	LOW	HIGH
M8	HIGH	LOW	LOW	LOW
S9	HIGH	LOW	LOW	HIGH
S10	HIGH	LOW	HIGH	LOW
S11	HIGH	LOW	HIGH	HIGH
S12	HIGH	HIGH	LOW	LOW
E13	HIGH	HIGH	LOW	HIGH

* Refer to diagram of Hybrid layout

3.2.17 Token Input/Output Circuit

In order to provide some measure of fault tolerance in the system, a token and data bypass circuit will be built into each chip. The purpose of this circuit is to enable a chip to source or send it's token and data to another chip other then it's direct neighbours. Each chip will have 2 token and data inputs. It will also have two token or data outputs. Pairs of inputs and pairs of outputs will be connected to different chips enabling it to send or receive data from one of two chips. In this way, should one of chip's neighbours fail, an alternative chip can take its place. Commands are used to direct each chip to use its normal or bypass inputs and outputs.

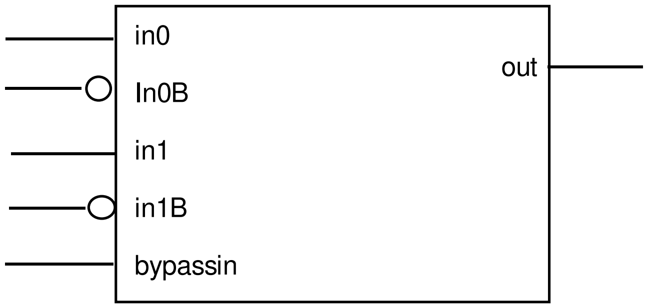


Figure 3.15. Token and Data Inputs circuit

Table 3.19a: Token and Data Input Signal Definitions

Signal Name	Active State/Edge	Function
in0		1st Data/Token Input
in0B		Complement of above
in1		2nd Data/Token Input
in1B		Complement of above
bypassin		When High in1 is selected else in0
out		Token /Data Output

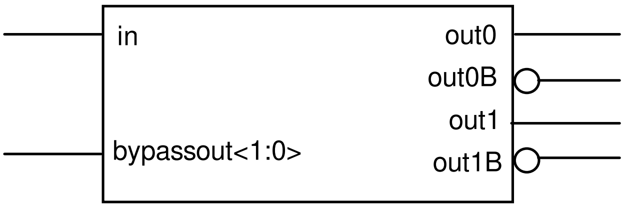


Figure 3.16. Token and Data Outputs circuit

Table 3.19b: Token and Data Output Signal Definitions.

Signal Name	Active State/Edge	Function
in		Token/Data in
bypassout<1:0>		When High out1 is enabled else out0
out0		1st Data/Token Output
out0B		Complement of above
out1		2nd Data/Token Output
outB		Complement of above

3.2.18 Test Circuitry

To simplify the testing of the chip during production and module assembly, it is proposed that additional test pads have been included on the chip to enable selected parts of the chip to be tested easily

3.2.19 Readout Protocols

Output data from the ABCD can be grouped into one of five classes:

3.2.19.1 Module Data

This type of data packet is only output from chips which have been configured as masters. The packet consists of 2 elements, a 13-bit header generated by the Master ABCD chip and a string of physics data packets, from the all ABCD chips daisy-chained together including the Master ABCD chip.

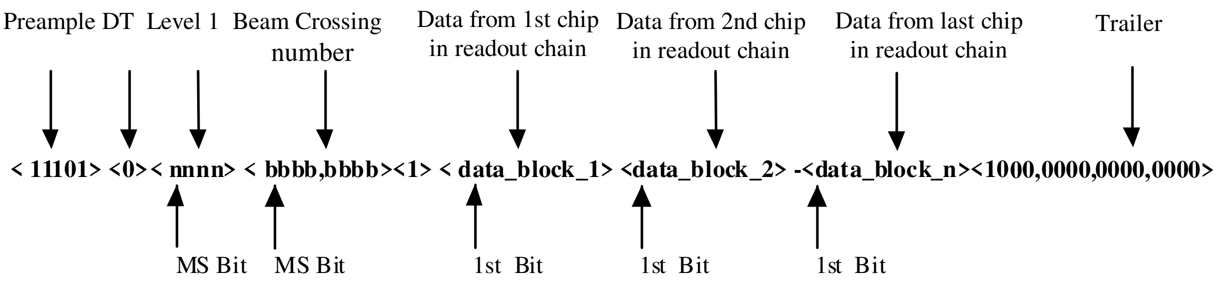


Figure 3.17a Module Data Format.

3.2.19.2 DT(Data Type)

The value of this bit determines the type of data which follows. This can either be L1 Trigger Data (DT=0) or Information Data (DT=1). In the case of the ABCD chip only L1 trigger Data is Sent and hence this field is always set to '0'.

3.2.19.3 L1

Current count of L1 Trigger modulo 16 since the last system reset. This field can be used for event building by the DAQ and also to monitor for lost data.

3.2.19.4 Beam Crossing Number

Current count of Beam Crossing modulo 256 since the last system reset or BC Reset command. It is intended to monitor for clock pulses lost by the on-detector electronics and can be used to tag one beam crossing out of the complete ring of the LHC.

3.2.19.5 Data Block

This is the data packet set from each chip including the master chip. This data block can be any of the four following types, Physics Data, No-Hit Data, Error Data or Configuration Data.

3.2.19.6 Physics Data

This type of data packet is used to send the compressed hit data from the detector. The format of this data is a series of one or more data packets.

<data_packet_1><data_packet_2> - - - <data_packet_n><data_packet_n+1>

There are 2 types of data_packet, isolated hit packet and non-isolated hit packet. A physics data packet can consist of any combination of these 2 types of packet.

3.2.19.7 Isolated Hit Data-Packet.

This type of packet is used to send the hit information from a hit channel on a chip when non of it's neighbouring channels have been hit.

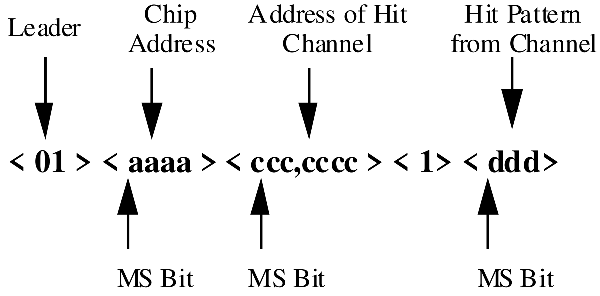


Figure 3.17b. Isolated hit data packet.

3.2.19.8 Non Isolated Hit Data-Packet

This type of packet is used to send data from a group of 2 or more adjacent channels which have been hit. Only the channel address of the 1st channel in the group is sent. It should be noted that this will also be the lowest numbered channel in the group. The chip address and channel address's of the other channels can be derived from that of the 1st and hence are not sent.

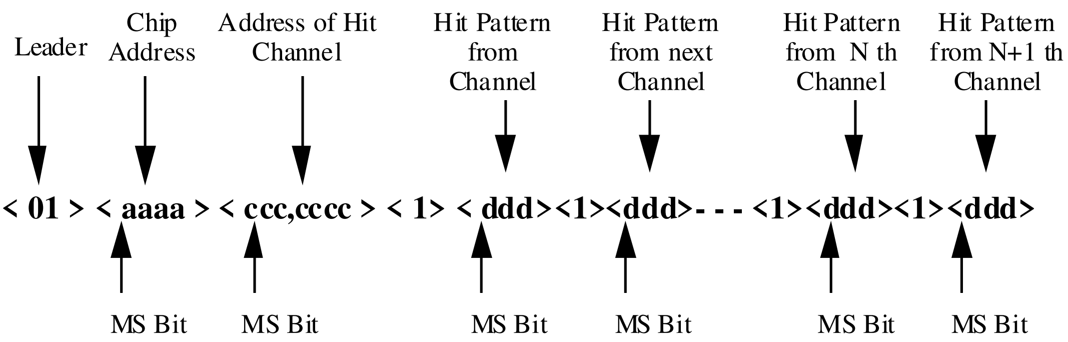


Figure 3.17c. Non isolated hit data packet (readout data format).

where

aaaa LS 4 bits of the chips geographical address

ccc,cccc 7-bit address of the channel on which the hit or 1st channel in a groups of hits was found

(See Figure 3.25 for the physical location of channel addresses.)

ddd Is the 3 bit hit pattern read out from the hit channel. (Previous, Current, Next)

Example

The following physics data packet would be send out from a chip with a geographical address of $2D_H$ and hits on channels 3, 5 and 6.

<01><1101><000,0011><1><ddd_3><01><1101><000,0101><1><ddd_5><1><ddd_6>

where ddd_3 = data from channel 3, ddd_5 = data from channel 5, ddd_6 = data from channel 6

3.2.19.9 No Hit Data

If a chip has received the event currently being read out but has not found any hit channels, it outputs a No Hit Data Packet.

< 001>

Figure 17d. No Hit Data Packet.

3.2.19.10 Configuration Data

Configuration data is sent by the chip in response to a L1 trigger when the chip is in its Send_ID mode of operation, i.e. the chip is not sending data. A packet of data is sent from the chip which contains the chips address and the contents of the chips configuration register.

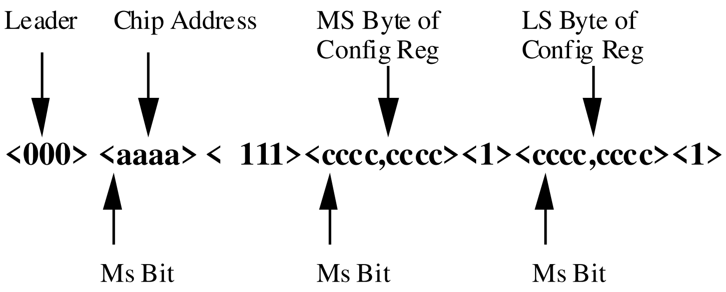


Figure 17e. Configuration Data Packet.

3.2.19.11 Error Data

Error data is only sent if the chip detects an error, e.g. Buffer overflow. In this cases a data packet of the following format is sent:

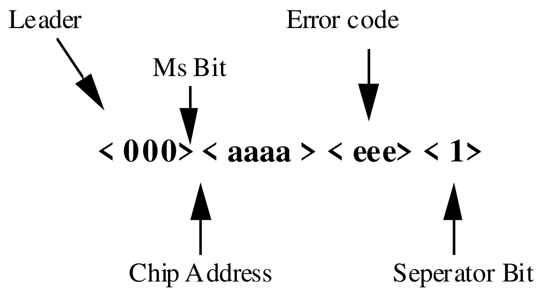


Figure 3.17f. Error Data Format.

3.2.19.12 Error Codes:

3 codes have been defined :

eee = 001	No Data Available (The chip has not received an L1 command)
eee = 010	Buffer Overflow
eee = 100	Buffer Error (Soft Reset needed)

N.B. Error messages are only sent if the chip is Data_Taking Mode. (see section 3.2.19)

3.2.20 Control Protocol

There are two main classes of commands, L1 Trigger Commands and Control Commands, and there are two types of Control Commands, Fast Control Commands and Slow Control Commands. It is not expected that the Slow Control Commands will be issued during data taking operation.

Table 3.20: Commands

Type	Field 1	Field 2	Field 3	Description
Level 1	110	---	---	L1 Trigger
Fast	101	0100 or 0010		Soft Reset BC Reset
Slow	101	0111	Command	Slow Control Command see Table 3.21

3.2.20.1 L1 trigger Command:

This is the most frequently issued packet and hence the smallest. All ABCD chips that receive this packet act on it. There is no addressing. If this command is received 3 samples are readout out of the pipeline and written into the readout buffer.

3.2.20.2 Fast Control Command:

This type of command is sent when a command has to be issued to the chip more quickly than can be achieved by sending a slow command to the chip. In the case of the ABCD chip, only two commands of this type have been defined, i.e. the Soft Reset and BC Reset commands. It is expected that these commands will be sent to the chip at regular intervals during periods of time when no L1 Triggers will be sent to the chip. The purpose of these commands is to perform a limited reset of the chip. (see section 3.2.19 for details)

3.2.20.3 Control commands

These are long packets that enable the operation of the chip to be controlled. While they are being sent, it is not possible to send a L1 trigger. Only the addressed ABCDs will act on the packet, unless the address sent equals '111111', in which case all chips will act on the packet. All chips that receive the packet must decode it, even if they do not act on it. This is to avoid un-addressed ABCDs erroneously decoding parts of the data field as the start of packets. To ensure that the chip does not respond to erroneous commands the chip will be placed out of taking mode for any command it receives which effects the configuration of the chip, i.e. all commands in which the 1st bit of field 5 is '0'. Hence it will be necessary to issue a command to the chip to enable data taking after issuing a command to change its configuration. When the chip is not in data taking mode, it will send its ID instead of real data in response to a L1 Trigger. This is the power-on default state.

Table 3.21: Control Commands

Field 3	Field 4	Field 5	Field 6	Description
0001,1100	aaaaaa	000 000	dddd,dddd,dddd,dddd	Write to Configuration Register
1000,1100	aaaaaa	001 000	d---,----,----,---d	Write to Mask Register
0001,1100	aaaaaa	010 000	dddd,dddd,dddd,dddd	Write to Strobe Delay Register
0001,1100	aaaaaa	011 000	dddd,dddd,dddd,dddd	Write to Threshold Registers
0000,1100	aaaaaa	100 000	-------------------	Pulse Input_Reg
0000,1100	aaaaaa	101 000	-------------------	Enable Data taking Mode
0000,1100	aaaaaa	110 000	-------------------	Issue Calibration Pulse
0001,1100	aaaaaa	111 000	dddd,dddd,dddd,dddd	Load Bias DAC
0001,1100	aaaaaa	000 100	dddd,dddd,dddd,dddd	Load TrimDac

N.B.

xxx = don't care state.

aaaaaa = 6 bit chip address(MS bit first)

dddd = data value for register (MS bit first)

Field 3

This is an 8 bit count of the number of bits in the following instruction.

Field 4

This is the 6-bit address of the chip for which the command is intended. (See Section on Geographical Address.)

N.B. To Address an ABCD chip the MS bit must always be set.

Field 5

This 6 bit field is used to determine into which register on the chip the data contained in the following field will be written or which command sequence is to be executed.

Field 6

This field holds the data that is to be written into the selected register. With the exception of instructions which load the mask register, this field will be 16-bits long.

3.2.21 Chip Initialisation and Configuration

The chip has 2 modes of operation, "Send_ID Mode" and "Data_Taking Mode". After a Power-up reset the chip is placed into Send_ID mode.

3.2.21.1 Send_ID Mode

In this mode of operation the chip sends its ID and Configuration data in response to a L1 trigger. There is no command which explicitly places the chip into this mode of operation, however, any attempt to alter the contents of the chip's various registers automatically results in the chip being placed into Send_ID mode.

3.2.21.2 Data_Taking Mode.

When the chip is not in Send_ID mode it is in Data_Taking Mode and visa-versa. In this mode of operation the chip sends out any physics data that it has. The chip may be placed in this mode of operation by sending a command to the chip to enable data taking. The chip may be taken out of this mode of operation and placed into Send_ID mode by either a Power_up reset or any attempt to change the contents of the chip's registers.

3.2.21.3 Clock Feed Through

If the clock feed through bit in the configuration register has been cleared and the chip has been configured as a Master, the chip outputs the chips system clock divided down by 2 from it's data output pins. This feature has been included to simplify system testing.

3.2.22 Resets

There are three kinds of reset in the system.

3.2.22.1 Power up reset

The power-up reset is an asynchronous (i.e. clock independent) reset that sets the value of the chips command register to zero, it's default value, and clears all the buffers in the chip, thus placing the chip into a well defined state. This type of reset is issued automatically when power is first applied to the chip. Provision will be made to enable this signal to be supplied externally to the chip.

3.2.22.2 Soft Reset

This type of reset is sent to the chip via a command instruction. Its purpose is to clear all the buffers in the chip, while leaving the configuration of the chip unaffected. This type of reset will be issued to the chip periodically during data taking to eliminate synchronisation errors.

1) Upon receipt of the reset command, the ABCD chip resets all internal counters, clears tokens and sets itself to the no-data state. If it was transmitting data, it terminates this immediately.

2) The external system must wait a time consistent with any data in the serial chain at the reset clock cycle to flush through the chain. This is one clock cycle per chip in the read-out chain, or 0.3µs for a 12 chip ring.

N.B. It should be noted that the off-detector system must then be able to determine the last complete event transmitted before the reset and discard it (Complete in the sense that all read-out chains supply a header and trailer). With either reset it must also be able to recognise and discard partial events since there is no guarantee that different read-out chains will be reading the same event when the periodic reset arrives. This section will define how the chip behaves on power up, i.e. default state of registers etc. latch-up prevention measures needed , and any special power cycling or power ramping required.

3.2.22.3 BC Reset

This type of reset is sent to the chip via a command instruction. Its purpose is to zero the Beam Crossing counter. It has no effect on the operation of any other part of the chip.

The following sequence of instructions should normally be sent to the chip after power-up

1) Send command to load the configuration register with the appropriate settings.

2) Send a command to load the mask register

3) Send a series of commands to load the DAC register/s and Delay registers

4) Send a command to place the chip into data taking mode.

The chip will now be in a state to receive L1 trigger command and send data.

3.2.23 Data Readout and Redundancy

The figure below shows the data and token interconnections on a typical silicon detector module. The module has 6-ABCD chips on each side. The datalink outputs of 2 of these chips are connected to a fibre-optic interface and are configured to act as Masters in controlling the readout of data from each side of the module. On the diagrams the Master chips are denoted by a “M” and all the other chips are configured to act as slaves as denoted by a “S” or "E' on the diagram.

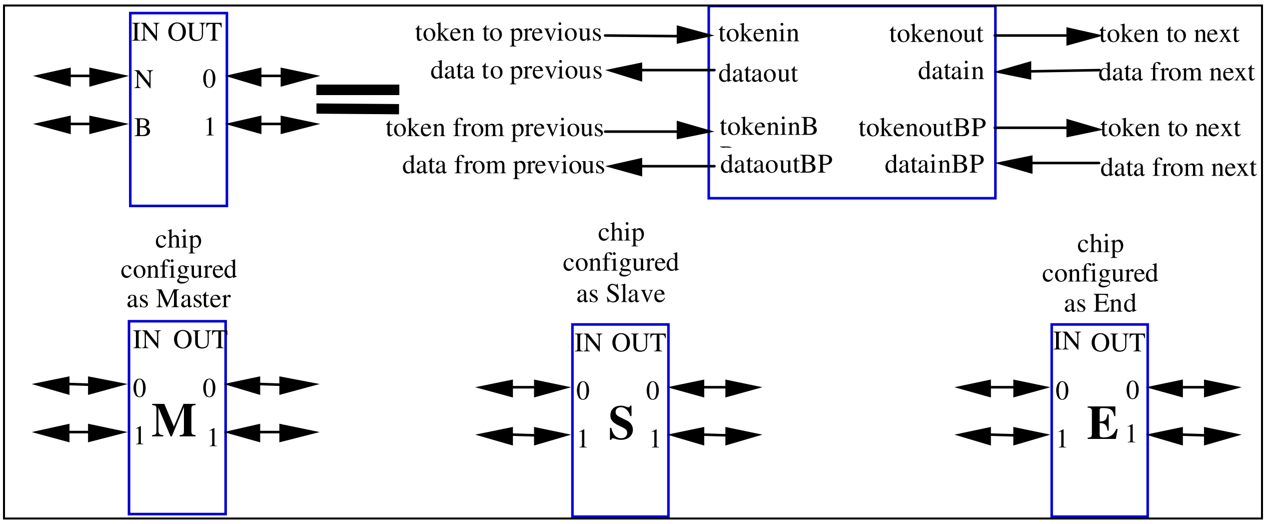


Figure 3.18a. Key to symbols used in following Diagrams.

After the receipt of a L1 Trigger, the Master chip initiates a readout cycle by sending the pre-amble bits at the start of each data block to the optical link driver. It then appends its data bits to the output stream sent to the optical link driver. A few clock cycles before the last bit has been sent, it sends a token to the slave chip on it's right. The slave chip on the right responds by sending its data packet to the Master which in turn is appended to the pre-amble and data bits from the Master already sent to the optical link driver.

N.B. each chip always sends at least 3 bits of data even if it hasn't found any hit channels in the event being read out.

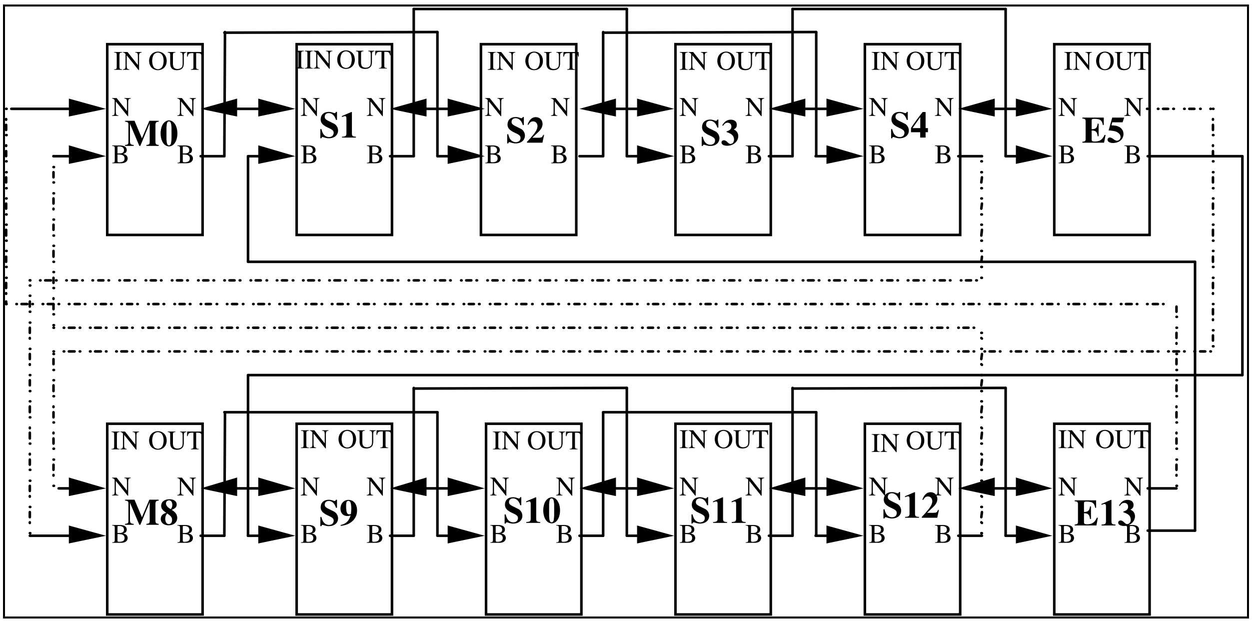


Figure 3.18b. Diagram Showing the Interconnection of ABCD chips on a Silicon Detector Module.

Once this slave chip has finished sending its data, it also passes on the token to the next chip on the right. The next chip on the right passes its data onto the previous chip on the left which in turn passes it back to the Master chip for transmission to the LED driver. This process continues until the last chip in the chain has sent it's data.

A bit is set in the last chip in the chain to inform it that it is the last chip (these chips are shown as 'E' on the diagrams). When this chip has sent it's data it appends a trailer to the end of the data stream. While the Master chip is outputting data, it is constantly looking for the trailer pattern which has been carefully chosen to be distinct from the data. Once it finds the trailer pattern, it knows that all the data from the event has been sent and it can start processing the next event.

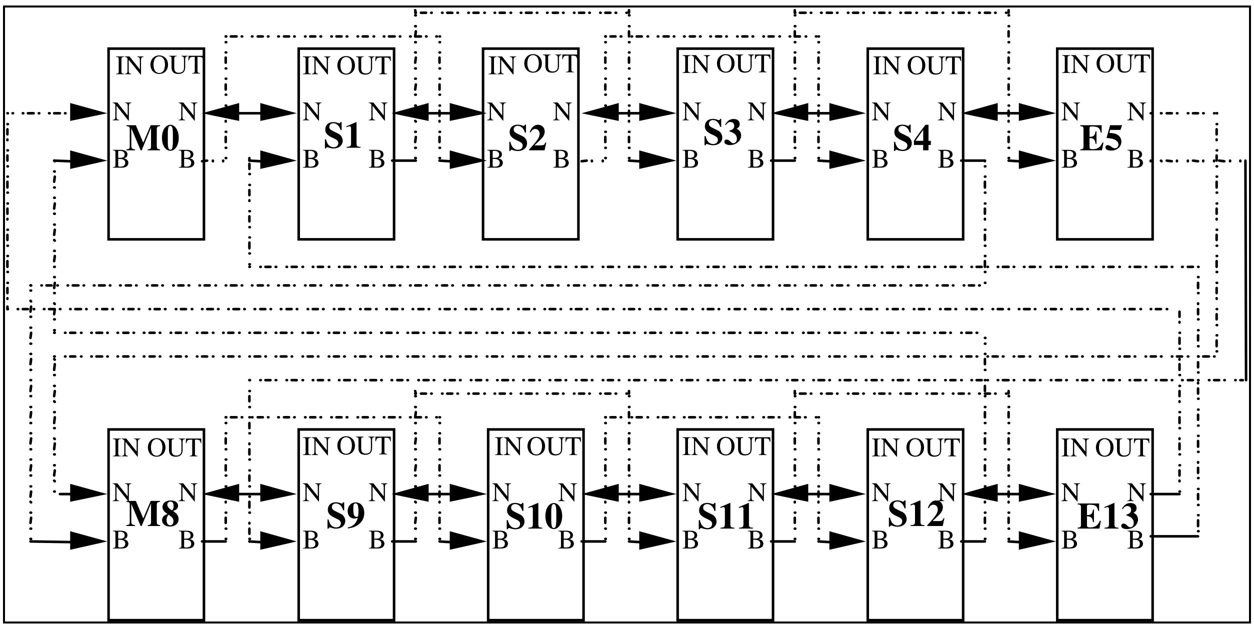


Figure 3.18c. Diagram Showing the normal flow of Data and Tokens between chips.
(Active links are highlighted with solid lines.)

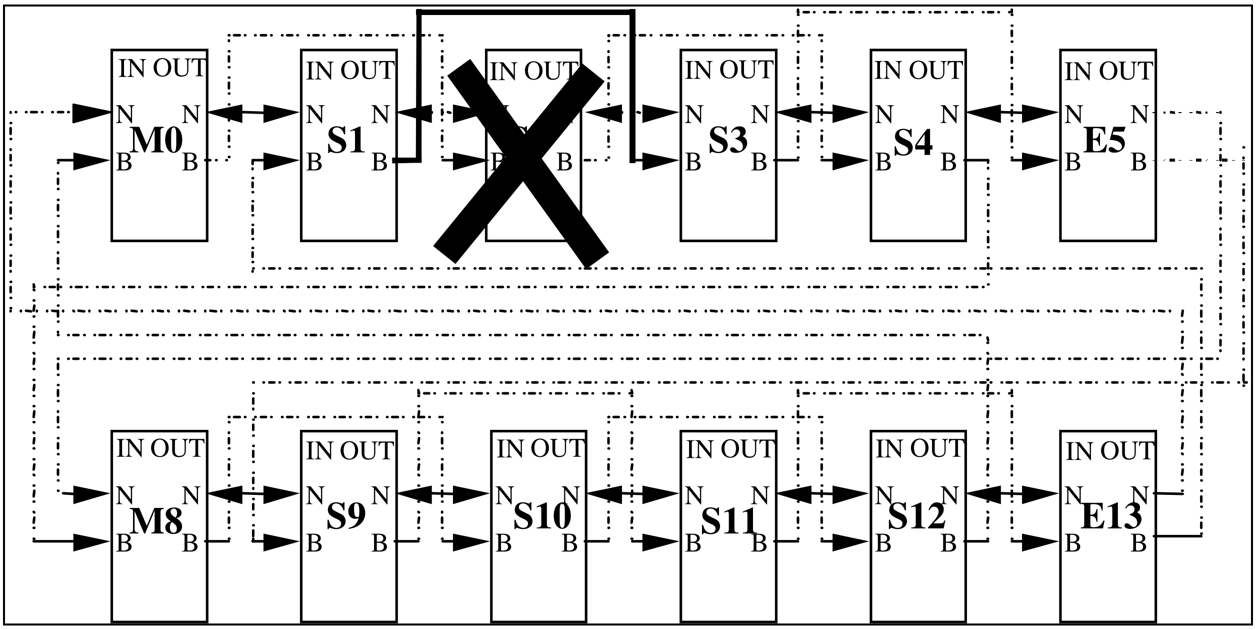


Figure 3.18d. Diagram Showing the flow of Tokens and Data in the event of the failure of a Slave ABCD chip.

In the event of the failure of one of the Slave chips, the previous and next slave chips in the chain are programmed to route their data and tokens around the failed chip. If the last chip in the chain should fail, then the penultimate chip in the chain is programmed to perform the operation of the "End chip".

In the event of the failure of a Master chip in the chain, the data and tokens from the chain with the failed Master chip are routed to the working master chip as shown in the next diagram .

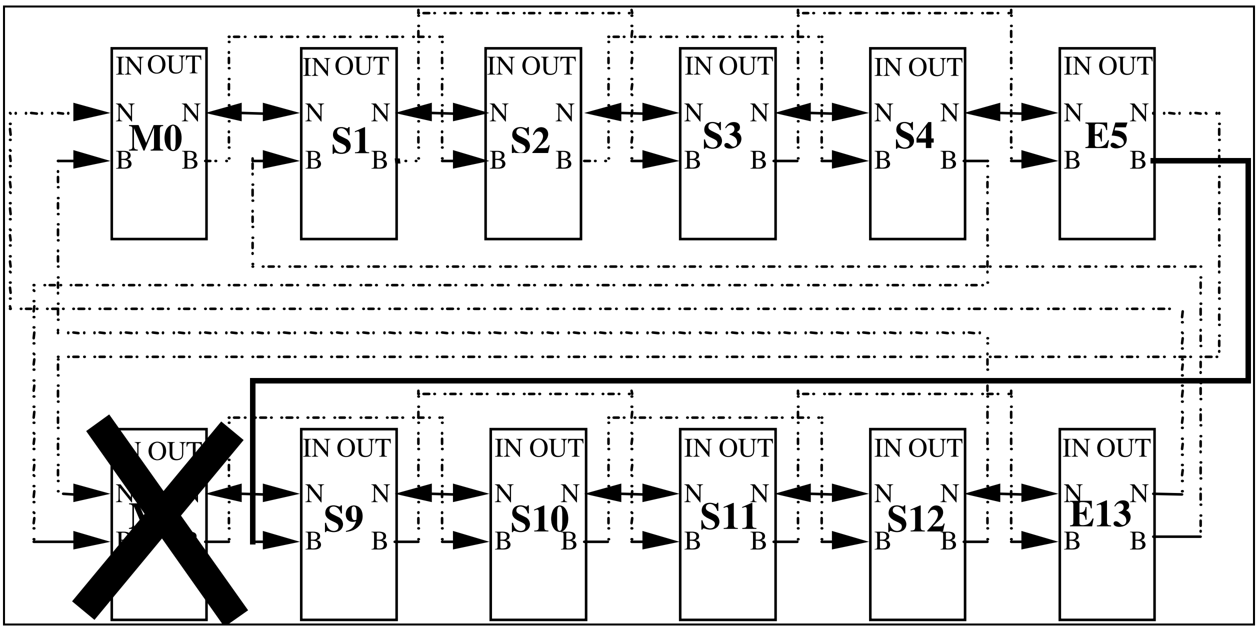


Figure 3.18e. Diagram Showing the flow of Tokens and Data in the event of the failure of a Master ABCD chip.

3.2.24 Default Register Values

On power up, the contents of the configuration register will be set to zero. This results in the following configuration.

Read Out Mode is set to Detector alignment mode

Calibration Mode is disabled

Send_ID mode is enabled.

tokenin, datain inputs are enabled

tokenout, dataout outputs are enabled.

Input test mode is disabled.

Edge Detection Mode disabled

Clock Feed Through Mode is Enabled if the chip is acting as a Master.

Chip will not be configured as the end of a readout chain.

The chip will be configured as a Master if masterB is asserted, else it will be configured as a slave.

3.2.24.1 Master/Slave Selection

The default state of the chip on power up is determined by the state on the masterB input pin. If this pin has been left unconnected or tied high, the chip will power-up as a Slave. If this pin has been tied to ground, the chip will power up as a Master. If the chip is configured as a Mater on power up it may be re-configured as a slave. However if the chip has been configured as a slave on power up it may not be re configured as a master.

3.2.25 Input/Output Connections

The following tables describes the names and function of the various Input/Output connections to the chip.

Table 3.22a: Input Signals.

Name	**Function**	**Type**
clk0 & clk1	Clock input	LVDS
clk0B & clk1B	Complement of above signal	LVDS
com0 & com1	Command Input	LVDS
com0B & com1B	Complement of above signal	LVDS
tokenin & tokeninBP	Token Input	Current Mode
tokeninB & tokeninBPB	Complement of above signal	Current Mode
datain & datainBP	Data Input	Current Mode
datainB & datainBPB	Complement of above signal	Current Mode
ID<5:0>	Geographical address of chip	CMOS
masterB	Sets chip default to master	CMOS
select	Selects clock/command inputs	CMOS
resetB	Resets Chip	CMOS

Table 3.22b: Default settings of CMOS input signals

Name	**Function**	**Default setting**
ID<4>	Geographical address of chip	Low, pull-down with 300 kOhm
ID<3:0>	Geographical address of chip	High, pull-up with 100 kOhm
masterB	Sets chip default to master	High, pull-up with 100 kOhm
select	Selects clock/command inputs	Low, pull-down with 300 kOhm
resetB	Resets Chip	High, pull-up with 300 kOhm

Table 3.22c: Output Signals.

Name	**Function**	**Type**
tokenout, tokenoutBP	Token Output	Current Mode
tokenoutB, tokenoutBPB	Complement of above	Current Mode
dataout, dataoutBP	Data Output	Current Mode
dataoutB, dataoutBPB	Complement of above	Current Mode
datalink	Data Output to optical link driver	LVDS
datalinkB	Complement of above	LVDS

3.2.26 DC Supply and Control Characteristics:

Table 3.23: DC supply voltages.

	Pad Name	Absolute Min	Min	Nominal	Max	Absolute Max
Analogue Supply	VCC	0 V	3.3 V	3.5 V	3.7 V	5.5 V
Analogue Ground	GNDA			0 V		
Input transistor current*	Set by internal DAC	0 μA	100 μA	200 μA	300 μA	400 μA
Input transistor current monitor	IP_PR	$V_{ip} = I_p$ x 250 Ω				
Shaper current *	Set by internal DAC	0 μA	10 μA	15 μA	30 μA	50 μA
Shaper current monitor	IS_PR	$V_{is} = I_{sh}$ x 10 kΩ				
Discriminator threshold	VTHP	0 V	3.3 V	3.5 V	3.7 V	VCC
Discriminator threshold	VTHN	0 V	3.25 V	3.45 V	3.65 V	VCC
(VTHP - VTHN)		0 V	0 V	0.05 V	0.5 V	1.0 V
Digital Supply **	VDD	0 V	3.8 V	4.0 V	4.2 V	5.5 V
Digital Ground	DGND			0 V		

* The Min/Max values define the range of the bias currents for which the front-end circuit will be biased correctly and will amplify the input signal. The absolute Min/Max values are the values which can be delivered from the internal DACs. The absolute Max values cover the worst case combination of corner process parameters and operating conditions.

** For the on chip power-up reset to operate correctly the VDD power supply must be ramped up to 90% of its final value in less than 10 ms.

The current draw at each DC input is as follows.

Table 3.24: DC supply currents for the nominal voltage supplies (VCC=3.5V, VDD=4.0V) and nominal operating conditions.

		Min	Nominal	Max
Analogue Supply	VCC	50 mA	75 mA	100 mA
Analogue Ground	GNDA	-50 mA	-75 mA	-100 mA
Digital Supply*	VDD	32 mA	36 mA	41 mA
Digital Ground	DGND	-32 mA	-36 mA	-41 mA
Discriminator threshold**	VTHP	0.3 μA	0.5 μA	1.5 μA
Discriminator threshold**	VTHN	0.3 μA	0.5 μA	1.5 μA

*In the Master chip the current draw at VDD power supply will be approximately 20 mA higher compared to the values shown in the table.

** If threshold voltages applied from external pads and the TrimDACs are set to zero.

Table 3.24a: Current draws at power supply inputs: nominal and absolute min/max values which may occur in non-standard operating conditions, e.g. all bias DACs set at zero or to full range, clock not supplied to the chips.

		Absolute Min	Nominal	Absolute Max
Analogue Supply	VCC	2 mA	75 mA	110 mA
Digital Supply	VDD	10 mA (30 mA)*	36 mA (56 mA)*	43 mA# (63 mA)* #

#Maximum current expected for the VDD power supply of 4.2 V

*Current draws by the Master chip

Table 3.24b: Current draws by the fully loaded module: nominal and min/max values are specified for the normal operating conditions and nominal supply voltages, absolute min/max values may occur in non-standard operating conditions, e.g. all bias DACs set at zero or to full range, clock not supplied to the chips.

		Absolute Min	Min	Nominal	Max	Absolute Max
Analogue Supply	VCC	24 mA*	600 mA	900 mA	1200 mA	1320 mA
Digital Supply	VDD	140 mA*	360 mA	400 mA	450 mA	470 mA#
Total power		0.58 W*	3.54 W	4.75 W	6.0 W	7.0 W

* For the absolute minimum values it is assumed that all 12 chips on the module are connected to the power lines and none of them is damaged. The power consumption is calculated for the minimum supply voltages: VCC = 3.3V and VDD = 3.8V

#Maximum current expected for the VDD power supply of 4.2 V

3.2.27 Input/Output Levels

Table 3.25: Input Levels for CMOS Inputs (select, masterB, ID<4:0>, resetB)

Parameter	**Conditions**	**Minimum**	**Maximum**
Input low voltage		0 V	0.4V
Input high voltage		VDD-0.4V	VDD
Input impedance	See Table 3.25a	0.8*NOM	1.2*NOM

Table 3.25a: Input Resistances for CMOS Inputs (select, masterB, ID<4:0>, resetB)

Name	**NOMINAL RESISTOR (NOM)**
ID<4>	Low, pull-down with 300 kOhm
ID<3:0>	High, pull-up with 100 kOhm
masterB	High, pull-up with 100 kOhm
select	Low, pull-down with 300 kOhm
resetB	High, pull-up with 300 kOhm

Table 3.26: Input Levels for LVDS Inputs (Clock, Control).

Parameter	**Conditions**	**Minimum**	**Maximum**
Input Voltage Range V_i		0 mV	2400 mV
Input Voltage Common mode V_{icm}		50 mV	2350 mV
Differential high input threshold $+V_{idth}$	$R_{load} = 100\Omega$		100 mV
Differential high input threshold $-V_{idth}$	$R_{load} = 100\Omega$	-100 mV	
Receiver input impedance		100 kΩ	

N.B. No internal terminating resistor is built into these inputs and consequently an external terminating resistor is required.

Table 3.27: Input Levels for Token and Data Inputs (Tokenin, Datain).

Parameter	**Minimum**	**Typical**	**Maximum**
Input voltage range	1.0V	1.7V	2.0V
Input differential voltage	40mV	80mV	160mV
Receiver input impedance	80 Ω	100 Ω	120 Ω

Table 3.27a: Input Levels for Token and Data Bypass Inputs (TokeninBP, DatainBP).

Parameter	**Minimum**	**Typical**	**Maximum**
Input voltage range	1.0V	1.7V	2.0V
Input differential voltage	40mV	80mV	160mV
Receiver input impedance	80 Ω	100 Ω	120 Ω

Table 3.28: Output Levels for Token and Data Outputs (Tokenout, Dataout)

Parameter	Conditions	Minimum	Typical	Maximum
Output common mode voltage	C_L=10pF R_{load}=100Ω	1.02V	1.7V	1.98V
Output differential voltage	C_L=10pF R_{load}=100Ω	40mV	80mV	160mV

Table 3.28a: Output Levels for Token and Data Bypass Outputs (TokenoutBP, DataoutBP)

Parameter	Conditions	Minimum	Typical	Maximum
Output common mode voltage	C_L=50pF R_{load}=100Ω	1.02V	1.7V	1.98V
Output differential voltage	C_L=50pF R_{load}=100Ω	40mV	80mV	160mV

Table 3.29: Output Levels for LED Outputs.

Parameter	Conditions	Minimum	Maximum
Output Voltage low V_{OL}	R_{load} =100Ω	1000 mV	
Output Voltage High V_{OH}	R_{load} =100Ω		1400 mV
Output offset Voltage	R_{load} =100Ω	1125 mV	1275 mV
Output Differential Voltage	R_{load} =100Ω	250 mV	400 mV
Output impedance		40 Ω	280 Ω

Table 3.30: Requirements for the resetB input signal.

Parameter	Conditions	Minimum	Maximum	Absolute Max
Output Voltage low V_{OL}	CMOS input	0 V	0.4 V	
Output Voltage High V_{OH}	CMOS input	3.6 V	4.4 V	5.5 V
Rise/Fall time	CMOS input	0 ns	1 ms	
Pulse duration	CMOS input	100 ns	no limit	

Table 3.31: Requirements for the select input signal.

Parameter	Conditions	Minimum	Maximum	Absolute Max
Output Voltage low V_{OL}	CMOS input	0 V	0.4 V	
Output Voltage High V_{OH}	CMOS input	3.6 V	4.4 V	5.5 V
Rise/Fall time	CMOS input	static signal		
Pulse duration	CMOS input	static signal		

3.2.28 Input/Output Timings

Remarks:

a) Timing windows are described for all process and conditions variations, including post-radiation effects.

b) Delays on outputs may be larger than one clock period after total dose. In order to guarantee timing consistency after irradiation it will be required that only the chips irradiated up to the same dose should be connected together.

Table 3.32: CMOS Inputs (Select*, MasterB*, ID<4:0>*, ResetB)

Parameter	**Minimum**	**Typ**	**Maximum**
resetB pulse width	100 ns		
resetB Rise and fall time	1 ns		1 ms

*: select, masterB, ID<4:0> are static inputs only.

Table 3.33: Timing for Clock inputs.

Parameter	**Minimum**	**Typ**	**Maximum**
Pulse width TclW	10 ns	12.5 ns	15 ns
Rise and fall time TclRF	0.5 ns	1 ns	2 ns
Phase Jitter TclASY	-2 ns	0 ns	2 ns

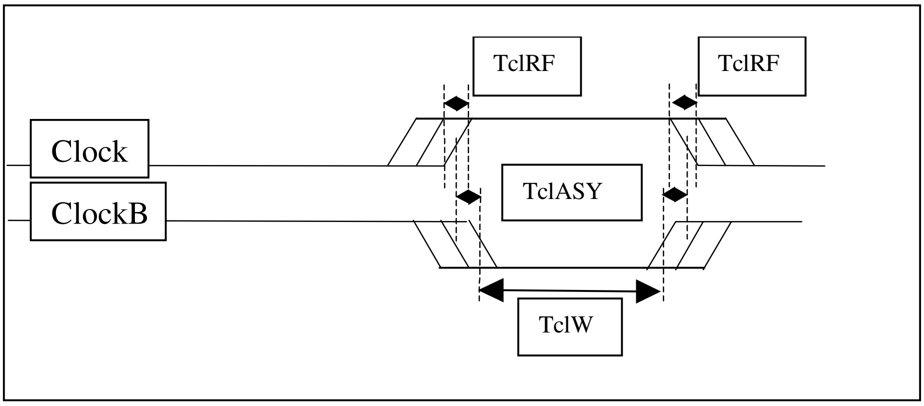


Table 3.34: Timing for Control versus Clock inputs.

Parameter	**Minimum**	**Typ**	**Maximum**
Setup Time TcomST	9 ns	12.5 ns	20 ns
Hold Time TcomH	5 ns	12.5 ns	16 ns
Pulse width TcomW	14 ns	25 ns	36 ns
Rise and fall time TcomRF	0.5 ns	1 ns	2 ns
Phase Jitter TcomASY	-2 ns	0 ns	2 ns

Time is referred to the 50% point of the rising voltage swing of the clk0 (resp. clk1) input to the chip.

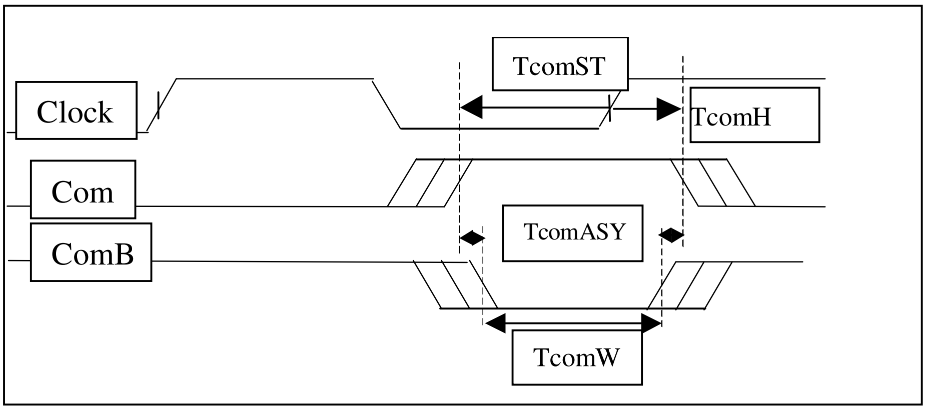


Table 3.35: Timing for Datain, DatainB, DatainBP, DatainBPB inputs.

Parameter	**Minimum**	**Typ**	**Maximum**
Setup Time TdtiST	0 ns	12.5 ns	15 ns
Hold Time TdtiH	10 ns	12.5 ns	24 ns
Pulse width TdtiW	10 ns	25 ns	39 ns
Rise and fall time TdtiRF	0.5 ns	1 ns	2 ns
Phase Jitter TdtiASY	-2 ns	0 ns	2 ns

Time is referred to the 50% point of the rising voltage swing of the clk0 (resp. clk1) input to the chip.

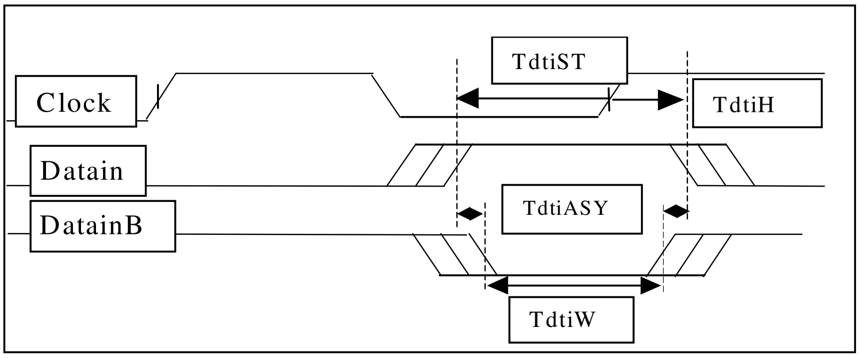


Table 3.36: Timing for Tokenout, TokenoutB, TokenoutBP, TokenoutBPB outputs.
Load : 100Ω , 10pF (Tokenout, TokenoutB)
100Ω , 50pF (TokenoutBP, TokenoutBPB)

Parameter	**Minimum**	**Typ**	**Maximum**
Delay Time TtkoDL	14.5 ns	21 ns	39 ns note a)
Pulse width TtkoW	20 ns	23 ns	25 ns
Rise and fall time TtkoRF	Note b)	Note b)	Note b)
Phase Jitter TtkoASY	-2 ns	0 ns	2 ns

Time is referred to the 50% point of the rising voltage swing of the clk0 (resp. clk1) input to the chip.
Note a) : The delay is in excess of one clock period. This is functional if all chips communicating through token and data lines are in the same process conditions (i.e. = accumulated dose, temperature, power supply)
Note b) : Rise and Fall time are mainly fixed by the RC time constant of the load.
For Tokenout, TokenoutB : TtkoRF typ. = 1ns with 100Ω , 10pF load
For TokenoutBP, TokenoutBPB : TtkoRF typ. = 5ns with 100Ω , 50pF load

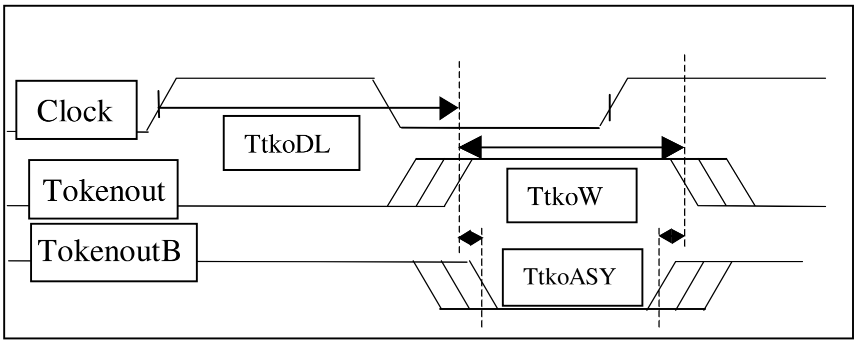


Table 3.37: Timing for Tokenin, TokeninB, TokeninBP, TokeninBPB inputs.

Parameter	**Minimum**	**Typ**	**Maximum**
Setup Time TtkiST	0 ns	5 ns	9 ns
Hold Time TtkiH	16 ns	20 ns	31 ns
Pulse width TtkiW	16 ns	25 ns	40 ns
Rise and fall time TtkiRF	0.5 ns	1 ns	2 ns
Phase Jitter TtkiASY	-2 ns	0 ns	2 ns

Time is referred to the 50% point of the rising voltage swing of the clk0 (resp. clk1) input to the chip.

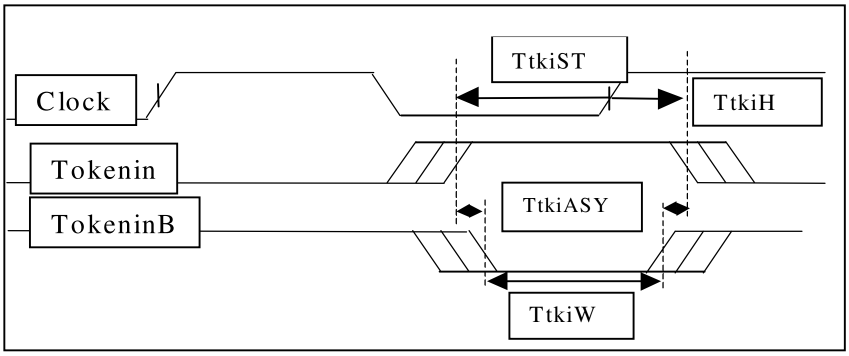


Table 3.38: Timing for Dataout, DataoutB, DataoutBP, DataoutBPB outputs.
Load : 100Ω , 10pF (Dataout, DataoutB)
100Ω , 50pF (DataoutBP, DataoutBPB)

Parameter	**Minimum**	**Typ**	**Maximum**
Delay Time TdtoDL	12 ns	20 ns	33 ns note a)
Pulse width TdtoW	23 ns	25 ns	27 ns
Rise and fall time TdtoRF	Note b)	Note b)	Note b)
Phase Jitter TdtoASY	-2 ns	0 ns	2 ns

Time is referred to the 50% point of the rising voltage swing of the clk0 (resp. clk1) input to the chip.
Note a) : The delay is in excess of one clock period. This is functional if all chips communicating through token and data lines are in the same process conditions (i.e. = accumulated dose, temperature, power supply)
Note b) : Rise and Fall time are mainly fixed by the RC time constant of the load.
For Dataout, DataoutB : TdtoRF typ. = 1ns with 100Ω , 10pF load
For DataoutBP, DataoutBPB : TdtoRF typ. = 5ns with 100Ω , 50pF load

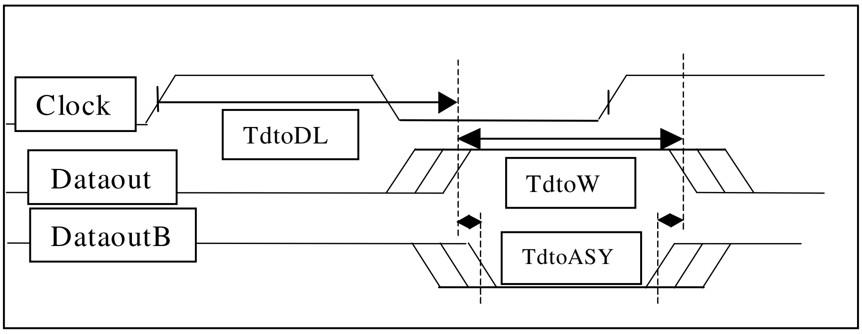
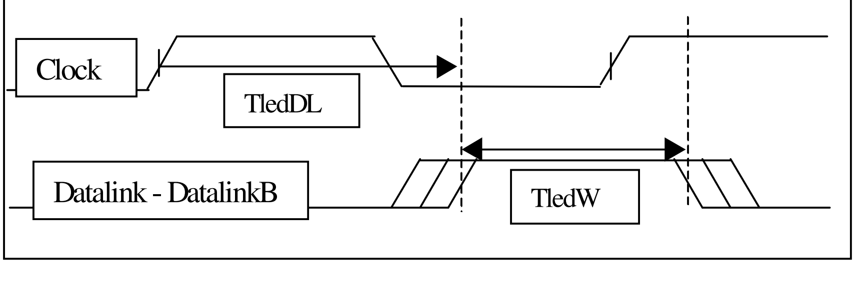


Table 3.39: Timing for Datalink, DatalinkB outputs.
Load : 100Ω , 50pF (Datalink, DatalinkB)

Parameter	**Minimum**	**Typ**	**Maximum**
Delay Time TledDL	16 ns	22 ns	40 ns note a)
Pulse width TledW	20 ns	25 ns	30 ns
Rise and fall time TledRF	5.8 ns	6.9 ns	12.5ns

Time is referred to the 50% point of the rising voltage swing of the clk0 (resp. clk1) input to the chip.
The timing is specified on the differential signal across 100 ohms load.
Note a) : The delay is in excess of one clock period.



3.2.29 Physical Requirements

The die size of the ABCD3T chip is 6550 μm x 8400 μm.

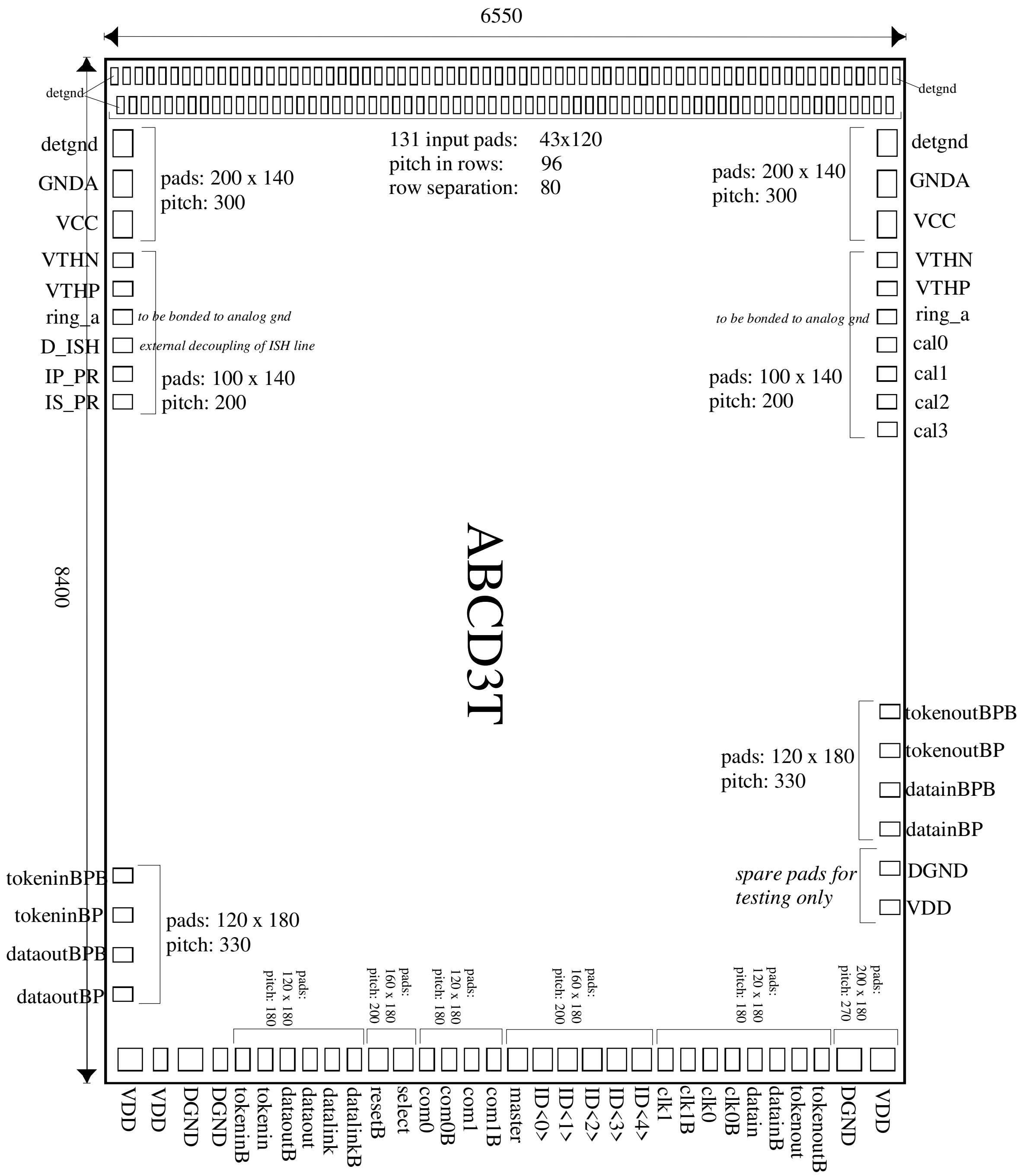


Figure 3.19. ABCD3T Pad Layout

3.2.30 Bond Pad Arrangement

Table 3.40: Bond Pad Position with respect to the origin at the lower left corner (detgnd pad).

Pad Name	Pad Centre (x,y)	Pad Size (x x y)	Pad Name	Pad Centre (x,y)	Pad Size (x x y)
INPUT PADS					
detgnd	(65,6266)	120x43	in<127>	(266,6218)	120x43
in<126>	(65,6170)	120x43	in<125>	(266,6122)	120x43
in<124>	(65,6074)	120x43	in<123>	(266,6026)	120x43
in<122>	(65,5978)	120x43	in<121>	(266,5930)	120x43
in<120>	(65,5882)	120x43	in<119>	(266,5834)	120x43
in<118>	(65,5786)	120x43	in<117>	(266,5738)	120x43
in<116>	(65,5690)	120x43	in<115>	(266,5642)	120x43
in<114>	(65,5594)	120x43	in<113>	(266,5546)	120x43
in<112>	(65,5498)	120x43	in<111>	(266,5450)	120x43
in<110>	(65,5402)	120x43	in<109>	(266,5354)	120x43
in<108>	(65,5306)	120x43	in<107>	(266,5258)	120x43
in<106>	(65,5210)	120x43	in<105>	(266,5162)	120x43
in<104>	(65,5114)	120x43	in<103>	(266,5066)	120x43
in<102>	(65,5018)	120x43	in<101>	(266,4970)	120x43
in<100>	(65,4922)	120x43	in<99>	(266,4874)	120x43
in<98>	(65,4826)	120x43	in<97>	(266,4778)	120x43
in<96>	(65,4730)	120x43	in<95>	(266,4682)	120x43
in<94>	(65,4634)	120x43	in<93>	(266,4586)	120x43
in<92>	(65,4538)	120x43	in<91>	(266,4490)	120x43
in<90>	(65,4442)	120x43	in<89>	(266,4394)	120x43
in<88>	(65,4346)	120x43	in<87>	(266,4298)	120x43
in<86>	(65,4250)	120x43	in<85>	(266,4202)	120x43
in<84>	(65,4154)	120x43	in<83>	(266,4106)	120x43
in<82>	(65,4058)	120x43	in<81>	(266,4010)	120x43
in<80>	(65,3962)	120x43	in<79>	(266,3914)	120x43
in<78>	(65,3866)	120x43	in<77>	(266,3818)	120x43
in<76>	(65,3770)	120x43	in<75>	(266,3722)	120x43
in<74>	(65,3674)	120x43	in<73>	(266,3626)	120x43
in<72>	(65,3578)	120x43	in<71>	(266,3530)	120x43
in<70>	(65,3482)	120x43	in<69>	(266,3434)	120x43
in<68>	(65,3386)	120x43	in<67>	(266,3338)	120x43
in<96>	(65,3290)	120x43	in<65>	(266,3242)	120x43
in<64>	(65,3194)	120x43	in<63>	(266,3146)	120x43
in<62>	(65,3098)	120x43	in<61>	(266,3050)	120x43

in<60>	(65,3002)	120x43	in<59>	(266,2954)	120x43
in<58>	(65,2986)	120x43	in<57>	(266,2858)	120x43
in<56>	(65,2810)	120x43	in<55>	(266,2762)	120x43
in<54>	(65,2714)	120x43	in<53>	(266,2666)	120x43
in<52>	(65,2618)	120x43	in<51>	(266,2570)	120x43
in<50>	(65,2522)	120x43	in<49>	(266,2474)	120x43
in<48>	(65,2426)	120x43	in<47>	(266,2378)	120x43
in<46>	(65,2330)	120x43	in<45>	(266,2282)	120x43
in<44>	(65,2234)	120x43	in<43>	(266,2186)	120x43
in<42>	(65,2138)	120x43	in<41>	(266,2090)	120x43
in<40>	(65,2042)	120x43	in<39>	(266,1994)	120x43
in<38>	(65,1946)	120x43	in<37>	(266,1898)	120x43
in<36>	(65,1850)	120x43	in<35>	(266,1802)	120x43
in<34>	(65,1754)	120x43	in<33>	(266,1706)	120x43
in<32>	(65,1658)	120x43	in<31>	(266,1610)	120x43
in<30>	(65,1562)	120x43	in<29>	(266,1514)	120x43
in<28>	(65,1466)	120x43	in<27>	(266,1418)	120x43
in<26>	(65,1370)	120x43	in<25>	(266,1322)	120x43
in<24>	(65,1274)	120x43	in<23>	(266,1226)	120x43
in<22>	(65,1178)	120x43	in<21>	(266,1130)	120x43
in<20>	(65,1082)	120x43	in<19>	(266,1034)	120x43
in<18>	(65,986)	120x43	in<17>	(266,938)	120x43
in<16>	(65,890)	120x43	in<15>	(266,842)	120x43
in<14>	(65,794)	120x43	in<13>	(266,746)	120x43
in<12>	(65,698)	120x43	in<11>	(266,650)	120x43
in<10>	(65,602)	120x43	in<9>	(266,554)	120x43
in<8>	(65,506)	120x43	in<7>	(266,458)	120x43
in<6>	(65,410)	120x43	in<5>	(266,362)	120x43
in<4>	(65,314)	120x43	in<3>	(266,266)	120x43
in<2>	(65,218)	120x43	in<1>	(266,170)	120x43
in<0>	(65,122)	120x43	detgnd	(266,74)	120x43
detgnd	(65,26)	120x43			
FRONT-END SERVICE PADS					
detgnd	(596,74)	200x140	detgnd	(596,6221)	200x140
GNDA	(896,74)	200x140	GNDA	(896,6221)	200x140
VCC	(1196,74)	200x140	VCC	(1196,6221)	200x140
VTHN	(1446,74)	100x140	VTHN	(1446,6221)	100x140
VTHP	(1646,74)	100x140	VTHP	(1646,6221)	100x140

ring_a	(1846,74)	100x140	ring_a	(1846,6221)	100x140
D_ISH	(2046,74)	100x140	cal0	(2046,6221)	100x140
IP_PROBE	(2246,74)	100x140	cal1	(2246,6221)	100x140
IS_PROBE	(2446,74)	100x140	cal2	(2446,6221)	100x140
			cal3	(2646,6221)	100x140
BYPASS PADS					
			tokenoutBPB	(5211,6211)	100x160
			tokenoutBP	(5543,6211)	100x160
			datainBPB	(5875,6211)	100x160
			datainBP	(6207,6211)	100x160
tokeninBPB	(6522,84)	100x160	**Spare digital bias pads**		
tokeninBP	(6854,84)	100x160	DGND	(6540,6211)	100x160
dataoutBPB	(7186,84)	100x160	VDD	(6872,6211)	100x160
dataoutBP	(7518,84)	100x160			
OUTPUT PADS					
VDD	(8069,6187)	160x180			
DGND	(8069,5914)	160x180			
tokenoutB	(8069,5688)	160x100			
tokenout	(8069,5508)	160x100			
datainB	(8069,5328)	160x100			
datain	(8069,5148)	160x100			
clk0B	(8069,4968)	160x100			
clk0	(8069,4788)	160x100			
clk1B	(8069,4608)	160x100			
clk1	(8069,4428)	160x100			
ID<4>	(8069,4238)	160x140			
ID<3>	(8069,4038)	160x140			
ID<2>	(8069,3838)	160x140			
ID<1>	(8069,3638)	160x140			
ID<0>	(8069,3438)	160x140			
masterB	(8069,3238)	160x140			
com1B	(8069,3048)	160x100			
com1	(8069,2868)	160x100			
com0B	(8069,2688)	160x100			
com0	(8069,2508)	160x100			
select	(8069,2318)	160x140			
resetB	(8069,2118)	160x140			
datalinkB	(8069,1928)	160x100			

datalink	(8069,1748)	160x100			
dataout	(8069,1568)	160x100			
dataoutB	(8069,1388)	160x100			
tokenin	(8069,1208)	160x100			
tokeninB	(8069,1028)	160x100			
DGNG	(8069,848)	160x100			
DGND	(8069,621)	160x180			
VDD	(8069,395)	160x100			
VDD	(8069,169)	160x180			