**CBC3 Technical Specification**

Author: Mark Prydderch

Issue: 1.3

Published: 25/05/16

Contents

[Document Change Log 3](#_Toc428275344)

[Top Level Architecture 4](#_Toc428275345)

[Analogue Front end 4](#_Toc428275346)

[Channel Gain 4](#_Toc428275347)

[Noise 5](#_Toc428275348)

[Pulse shaping 5](#_Toc428275349)

[Overload recovery 5](#_Toc428275350)

[Leakage Current 5](#_Toc428275351)

[Dual polarity 5](#_Toc428275352)

[‘Hit’ Comparator Time-walk 5](#_Toc428275353)

[‘Hit’ Comparator threshold trim 5](#_Toc428275354)

[Test Pulse Generator 5](#_Toc428275355)

[Digital Logic 6](#_Toc428275356)

[Channel Mask 6](#_Toc428275357)

[Hit Detect Logic 6](#_Toc428275358)

[OR254 8](#_Toc428275359)

[Layer swapping logic 8](#_Toc428275360)

[Stub-finding logic 8](#_Toc428275361)

[Stub Gathering logic 12](#_Toc428275362)

[Bend Look-up Table 15](#_Toc428275363)

[Pipeline Memory and Output Buffer 16](#_Toc428275364)

[L1 Counter 16](#_Toc428275365)

[Data Assembly and Output Logic 16](#_Toc428275366)

[Serial Fast Command 17](#_Toc428275367)

[Clock Domains 18](#_Toc428275368)

[40 Mhz Test Mode 18](#_Toc428275369)

[Delay Locked Loop (DLL) 19](#_Toc428275370)

[Logic SEU immunity 19](#_Toc428275371)

[External Interfaces 19](#_Toc428275372)

[Clocks 19](#_Toc428275373)

[Serial Fast Command 19](#_Toc428275374)

[Slow Control 19](#_Toc428275375)

[Inter-chip signals 20](#_Toc428275376)

[Analogue Biases 21](#_Toc428275377)

[Master Reference Current Generator (CBC\_IREF) 21](#_Toc428275378)

[Comparator Threshold Reference (Vcth) 21](#_Toc428275379)

[Post-Amp Reference Voltages (VPLUS & VPLUS2) 21](#_Toc428275380)

[Bandgap Reference 21](#_Toc428275381)

[Power 21](#_Toc428275382)

[Pads 21](#_Toc428275383)

# Document Change Log

|  |  |  |
| --- | --- | --- |
| **Change Log** | | **Page** |
| Aug 25th 2015 | Contents page added. | 1 & 2 |
| Page Numbers Added. | All |
| Block diagram updated to include Layer Swap Logic & identify different signal types. | 3 |
| Layer Swapping Logic: Diagram added & text re-written for clarity. | 7 |
| Stub-finding Logic: Additional diagram added & text re-written for clarity. | 8 |
| Stub gathering Logic: Diagram updated & extra one added for clarity. Text added regarding priority. | 11 & 12 |
| Bend LUT: Note added | 14 |
| L1 Counter: Diagram removed. | 15 |
| Data Assembly and Output Logic: Text modified to expand on error flags. Diagram of serial data stream added. Note added regarding gaps in serial data. | 15 & 16 |
| Section added for Serial Fast Command Interface. | 16 |
| Section added for Clock Domains. | 17 |
| 40MHz Test Mode: Text updated for clarity. | 18 |
| Delay Locked Loop: Text updated to include clock domains. | 18 |
| SEU Immunity: Section title changed & text updated for clarity. | 18 |
| External Interfaces: Serial Fast Command section modified. Added text & diagram to Inter-chip signals section. | 18 & 19 |
| Analogue Biases: Section moved to later in document. Sections added for Post-Amp reference voltages and Bandgap. | 20 |
| July 17th 2015 | Note on detector capacitance assumptions | 4 |
| March 9th 2015 | Updated to include statement relating to the bend code definition, where a spare binary code 10000 is assigned to null event coding.  Correction of Power Specification Target to 450 µW/Channel. | 8 |
| January 25th 2016 | Some corrections, clarifications (figure 3 added), and updates. | ~ all |
| May 25th 2016 | Updated to reflect modifications to generation of 40 MHz from serial command interface (figs. 1 & 13), order of data output package (caption to fig. 11), and order of channels in triggered data (fig.12). Text associated with figure 11 (pages 17 and 18) has been modified. Hard reset specified. | 4,17,18,19 |

# Top Level Architecture

The CBC3 design will build on the existing CBC2 architecture with added features and functionality.



*Figure 1. Block Diagram of CBC3 architecture.*

# Analogue Front end

***Note:*** *These specifications are related to the design of the analogue front end which is the responsibility of Imperial College.*

## Channel Gain

The target gain of the Pre-amplifier and Post Amplifier combined is 50 mV / fC.

## Noise

The noise target of the Pre-amplifier and Post Amplifier combined is <1000 electrons for 5 cm strips with a leakage current up to 1µA. Design studies show that this can be achieved for an external capacitance (sensor + stray) up to 10 pF, for an input FET power of 240 uW, at an operating temperature of 0 deg. C.

## Pulse shaping

The amplifier pulse shape peaking-time should be <20 ns with a return to the baseline within 50 ns.

## Overload recovery

An individual channel should respond to normal size signals < 2.5 µs following a hip-type (Heavy Ionizing Particle) signal of up to 4 pC.

## Leakage Current

The choice of sensor provides AC-coupled strips, so there is no requirement for the preamplifier to be able to source or sink any significant leakage current.

## Polarity

The strip sensors will be n-on-p (electrons readout).

## ‘Hit’ Comparator Time-walk

Specified as < 16 ns.

Defined as the maximum time difference between the 50% amplitude points of the comparator digital output signals measured with a 1.25 fC and a 10 fC input signal, for a comparator threshold of 1 fC.

## ‘Hit’ Comparator threshold trim

The comparator threshold voltage will be provided by a 10 bit resistor ladder (monotonic) with mV resolution. This will be provided by a resistor ladder between VDDA and GND with no output buffering.

## Test Pulse Generator

The Test Pulse Generator will be reused from the CBC2. No modification is envisaged.

# Digital Logic

## Channel Mask

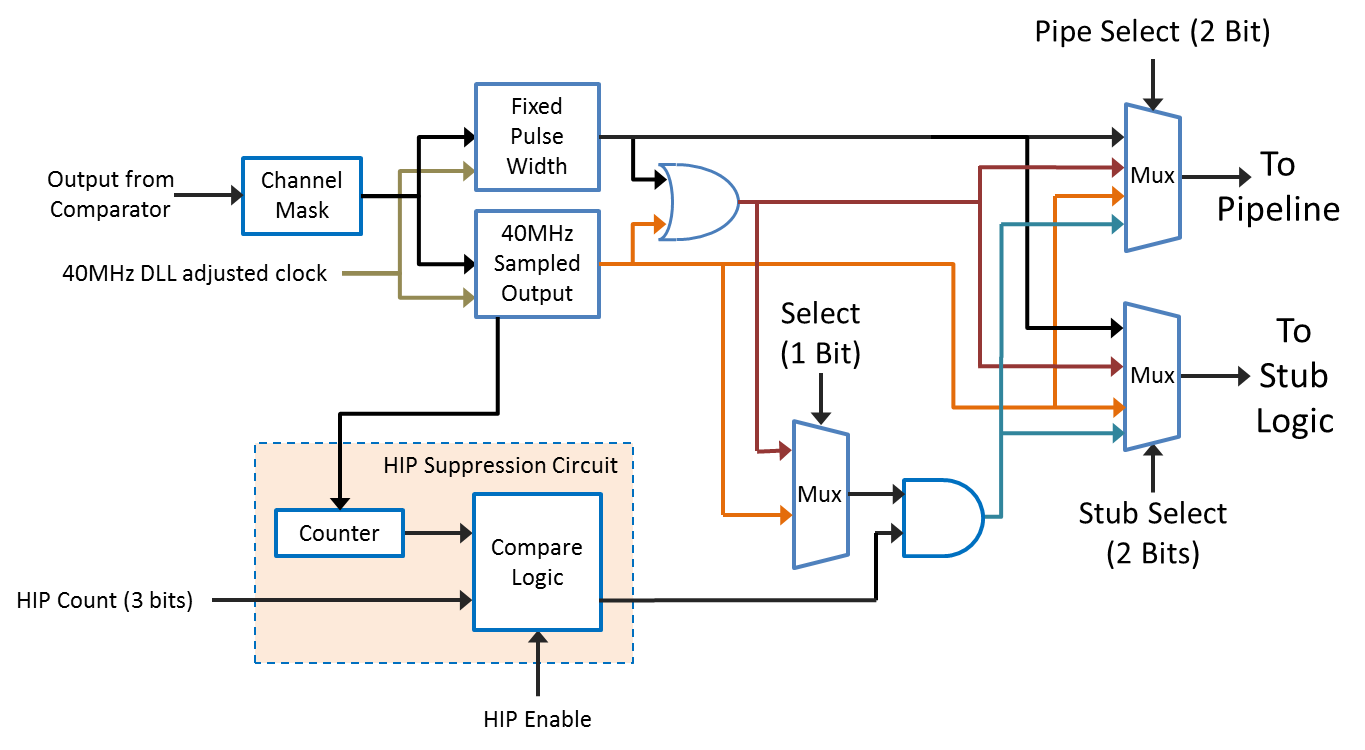
The Channel Mask is a programmable I2C register, the outputs of which are used to disable the input to the Hit Detect circuits on a channel by channel basis. The Hit Detect output to the OR254 block, the Stub-finding logic and the SRAM pipeline from the masked channel will be inhibited. In this mode there will be no L1 data output from the ‘masked’ channel and there will be no ‘stubs’ as a result of this channel.

## Hit Detect Logic

The Hit detect logic processes the output signal from the Hit Comparator circuit. There are four possible outputs to select for the Stub Logic and Pipeline:

1. **Fixed Pulse Width:** The output from the Hit comparator is latched for a full 25ns clock period. Any comparator transition is captured, regardless of its relationship to the 40MHz clock. The output is a fixed 25ns pulse, regardless of the width of the comparator output pulse.
2. **40MHz Sampled Output:** The output from the comparator is sampled using the 40MHz clock from the Delay Locked Loop. Only comparator outputs present on the rising edge of the clock will be captured and the output will only return to zero on the first rising clock edge following the comparators return to zero. The minimum width of output pulse is one clock cycle.
3. **Logical OR Output:** The outputs from (1) and (2) are passed through a logical OR to provide a combined result.
4. **HIP Suppressed Output:** Associated with (2) is a selectable Highly Ionizing Particle (HIP) suppression circuit. This circuit will check the length of the pulse and if it exceeds a pre-programmed number of clock cycles, the circuit will force the output to return to zero. The number of clock cycles for which the pulse can remain high will be programmed by setting 3 bits in a register. This suppression can be applied to either the 40MHz Sampled Output or the Logical OR Output by appropriate multiplexer selection.

The conceptual diagram for the circuit is shown in Figure 2, and Figure 3 illustrates the functionality.



*Figure 2. Hit Detect block diagram.*

NOTE: In the case of 1 and 2 above, hits following one after another in subsequent clock cycles will be captured.

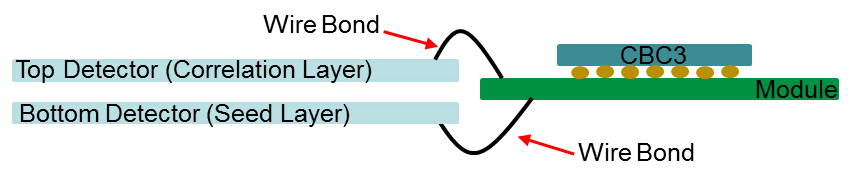


*Figure 3. Hit Detect functionality.*

## OR254

The CBC2 chip contained a large logical OR network at the output of the Hit Detect. The circuit produced an output if any of the channels were hit. This functionality will be retained for the CBC3 and the output will appear in the output data packet.

## Layer swapping logic

The CBC3 is designed to be used in conjunction with two detectors, as illustrated in figure 4.

*Figure 4. Illustration of 2 sensor layers wire-bonded to a module hybrid on which a CBC3 is bump-bonded.*

Strips on the bottom detector (Seed Layer) are connected to the CBC3’s odd numbered channels, starting with channel 1. Conversely the strips of the top detector (Correlation Layer) are connected to the CBC3’s even numbered channels. In some arrangements of the modules it is anticipated that the top and bottom detectors will swap position, such that the top detector becomes the Seed layer. Due to the asymmetric nature of the Stub-finding logic, it is necessary to reconfigure the logic to accommodate this change of seed layer. A multiplexer circuit at the output of the Hit Detect circuit will perform a swap of all odd and even channel signals (before the signals are transferred between chips). The selection of the swap will be controlled by a bit in an I2C register and acts on all channels on the chip.

**NOTE**: All chips on one module will have the same layer set up, so the inter-chip logic does not need account for different layer set ups on neighbouring chips.

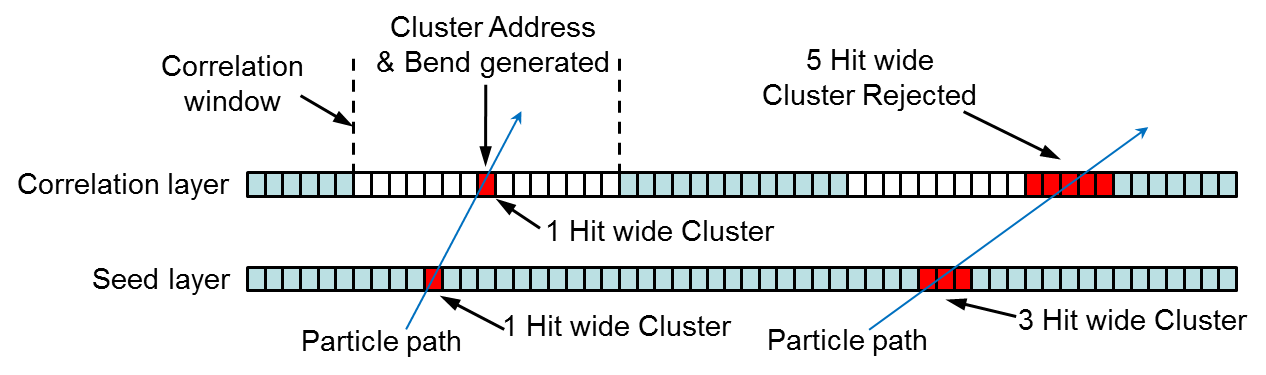
## Stub-finding logic

Hit Detect signals pass through the layer swapping logic, and are then processed in order to find correlations between the hits in both layers. The signals are processed differently depending on whether they are associated with the Seed layer or the Correlation layer.

A hit in either layer is referred to as a Cluster. Clusters may be one channel wide, or greater if adjacent channels on the same layer have also produced a hit. The first stage of the Stub-finding logic examines the width of the Clusters in each layer, in order to reject unsuitable Clusters, i.e. those clusters over a predetermined number of channels in width. This Cluster Width Discrimination logic, compares the width of Clusters on the same sensor layer against a programmable value, and suppresses them if they are found to be wider than this value. The rejection width is programmable up to a width of four adjacent channels. Clusters five adjacent channels wide or more will always be rejected.

**NOTE:** If a Cluster of multiple channels falls across a masked channel, this will appear to the logic as if there were two separate Clusters, and individually these smaller Cluster sizes might not be rejected.

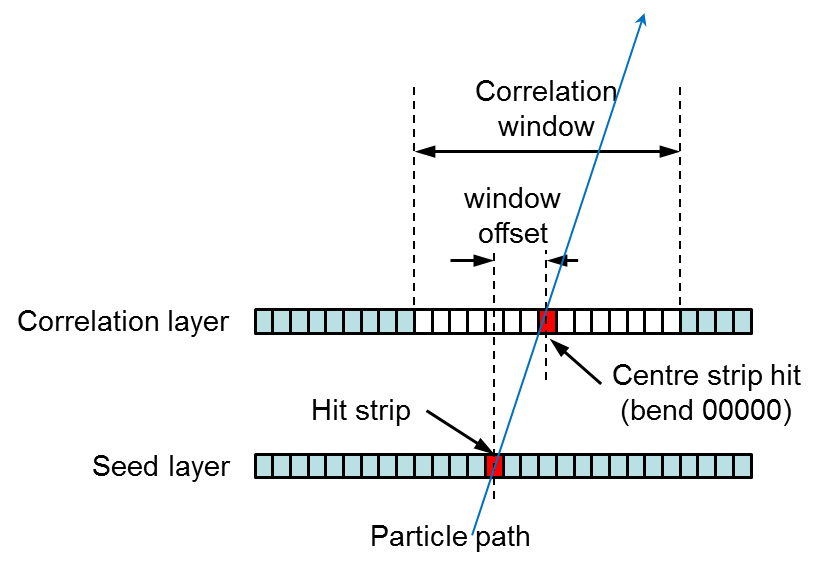
An illustration of a valid correlation and a rejected Cluster is shown in figure 5.



*Figure 5. Illustration of a valid correlation between layers and a rejected Cluster.*

Having rejected any oversized Clusters, the next stage of logic tries to match Clusters on the Seed layer with Clusters on the Correlation layer. The matching is restricted to a ‘window’ defined in the Correlation layer. This ‘window’ is essentially a group of channel positions in the Correlation layer, with which the Cluster in the Seed layer can be compared. The Correlation Window size is programmable in half channel steps, up to +7 channels about a centre channel, as illustrated in figure 5. To correct for the geometrical offset due to the position of the detector in the r-φ plane, the position of the Correlation window will be adjustable in half channel steps up to ±3 channels, as illustrated in figure 6. This positional programmability is on a regional basis only, with four independently programmable regions per chip. The offset value for each region will be programmed into a register via the I2C interface, requiring two I2C registers for four regions with four bits per region.

To summarise, I2C programmability will be provided for maximum cluster width (up to 4 channels, applied to both seed and correlation sensor layers), correlation width (up to +7 channels about a centre channel in the correlation layer), and centre channel position in the correlation layer (up to ±3 channels for 4 independently programmable regions per chip).



*Figure 6. Example of Correlation Window & Offset Correction*

**NOTE:** If there is more than one distinct Cluster within a valid coincidence window, then only the Cluster closest to the matching Cluster from the Seed layer, will be selected (i.e. the one giving the least amount of bend and therefore, in theory, the highest pT track).When the logic identifies a correlation between Clusters in the two layers (called a stub), the Seed Cluster location is output to the Stub Gathering Logic using a Correlation Bit. For a Cluster consisting of an even number of channels, the logic will generate a signal to indicate that the Cluster centre is in-between the two centre channels, giving what’s known as half-strip resolution. For an odd number of channels, the position of the actual centre channel is output.

The logic will also calculate the offset between the centre of the Seed Cluster and the centre of the Correlation Cluster, and output a representative 5 bit code to the Stub Gathering Logic. The latter is known as Bend information, and 5 bits allow for the bend to be calculated to half-strip resolution. For the hardware defined bend codes from the Stub-finding logic see figure 7. These 5 bit codes are only used internally to the chip and will be reduced to 4 bit codes by a Look-up Table (see Bend Look-up Table section), before they are output from the chip.

**NOTE:** The bend code will be calculated after the offset correction, so that it is independent of the geometrical position in the module or the Tracker.

For Stub-finding coverage to be un-interrupted across a module, signals need to be passed between chips in order to identify Stubs that straddle the border. The Stub-finding Logic will contain additional circuitry to accept and process the inputs from neighbouring chips.

|  |  |
| --- | --- |
| **Cluster Centre** | **Bend Code**  **NOTE:**-  Centre strip = centre strip of window which may be offset.  Centre assigned +ve sign |
| -7 | 10010 |
| -6.5 | 10011 |
| -6 | 10100 |
| -5.5 | 10101 |
| -5 | 10110 |
| -4.5 | 10111  Direction of Lower Channel Numbers |
| -4 | 11000 |
| -3.5 | 11001 |
| -3 | 11010  🡨 Channel 1 |
| -2.5 | 11011 |
| -2 | 11100 |
| -1.5 | 11101 |
| -1 | 11110  CBC3 Face Down  on the Module  (bumps underneath) |
| -0.5 | 11111 |
| 0 -- Centre | 00000 |
| +0.5 | 00001 |
| +1 | 00010 |
| +1.5 | 00011 |
| +2 | 00100  🡨 Channel 254 |
| +2.5 | 00101 |
| +3 | 00110  Direction of Higher Channel Numbers |
| +3.5 | 00111 |
| +4 | 01000 |
| +4.5 | 01001 |
| +5 | 01010 |
| +5.5 | 01011 |
| +6 | 01100 |
| +6.5 | 01101 |
| +7 | 01110 |

*Figure 7. Definition of Bend Codes generated by the Stub-finding logic.*

**NOTE:** The spare binary code 10000 is assigned to null event coding, specifically when the cluster center fall outside the coincidence window.

## Stub Gathering logic

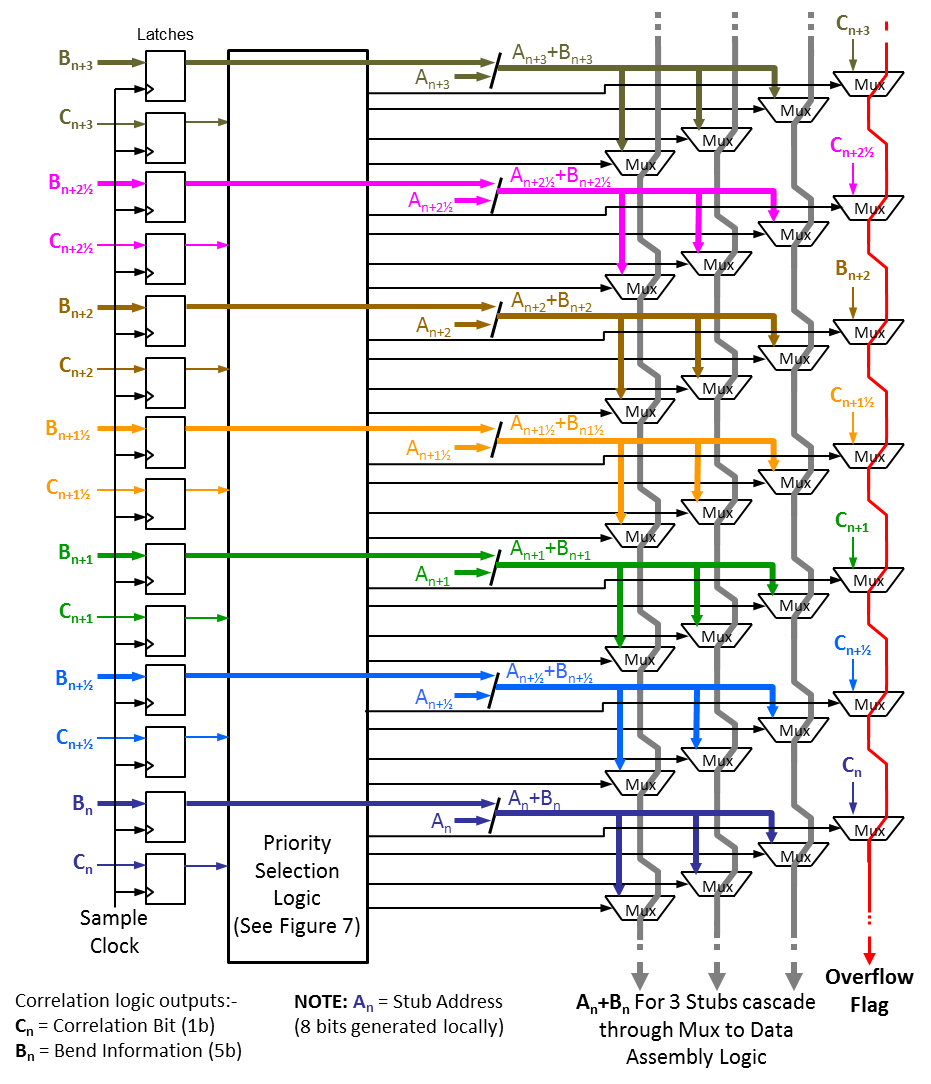
The Stub Gathering Logic generates an eight bit address for every Correlation Bit output by the Stub-finding logic. This is known as the Stub address and can represent a whole or half strip. Due to data rate limitations, the Stub Gathering logic must limit the amount of stub data output, to a maximum of three stubs per bunch crossing. If there are more than three stubs per chip per bunch crossing, only the 3 stubs with the lowest numbered addresses will be output.

Figure 8 on the next page shows the proposed circuit architecture for achieving the Stub Gathering function. Here, the outputs from the Stub finding Logic are latched at the end of the 25ns bunch crossing period. A following stage of asynchronous logic uses the correlation bits output from the Stub-finding Logic, to addresses one of three possible multiplexers (one for each Stub). For a given correlation location, the multiplexer selected will depend on the presence or absence of stubs in its higher priority neighbours. If the stub has sufficient priority, then its eight bit address and its associated five bit bend information will be routed onto one of the three, thirteen bit wide multiplexer output buses. The data will cascade through the multiplexer tree to the output where it is latched by the 40MHz clock at the start of the next bunch crossing period. If three higher priority stubs exist, then the stub data will not be routed through any of the multiplexers.

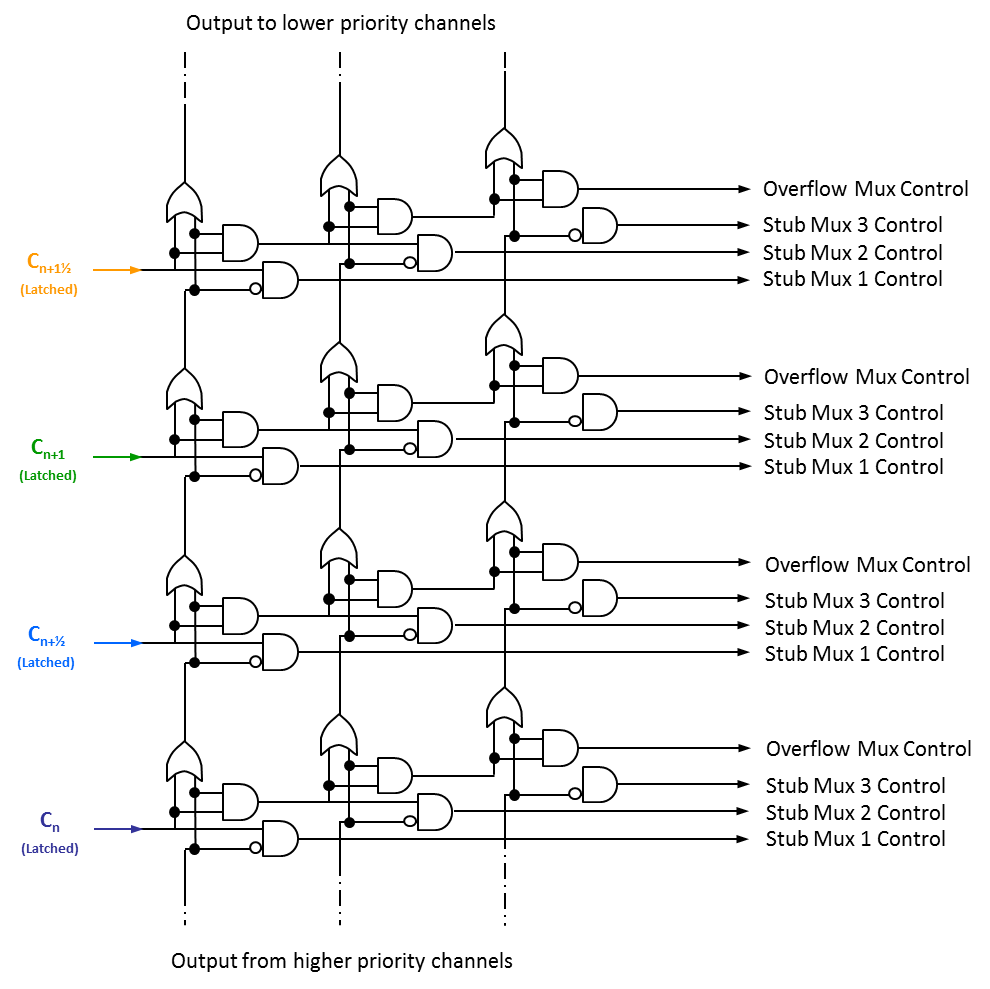
Prioritisation of the Stubs is based on their location in the chip, with lower addresses having priority over higher addresses, i.e. hits in the lower numbered channels will be given priority over hits in the higher numbered channels. In order to meet speed requirements it might be necessary to limit the size of the multiplexers and have further stages of latches and multiplexing to produce a tree structure.

The presence of more than 3 stubs will be indicated by an additional branch of logic to pass the 4th Correlation bit out of the circuit as a flag.

**IMPORTANT:** The gathered Stubs **will not** be prioritised on the basis of their bend information. If there are more than three stubs per chip per bunch crossing, it is not guaranteed that the Stubs read out will be the most desirable ones.



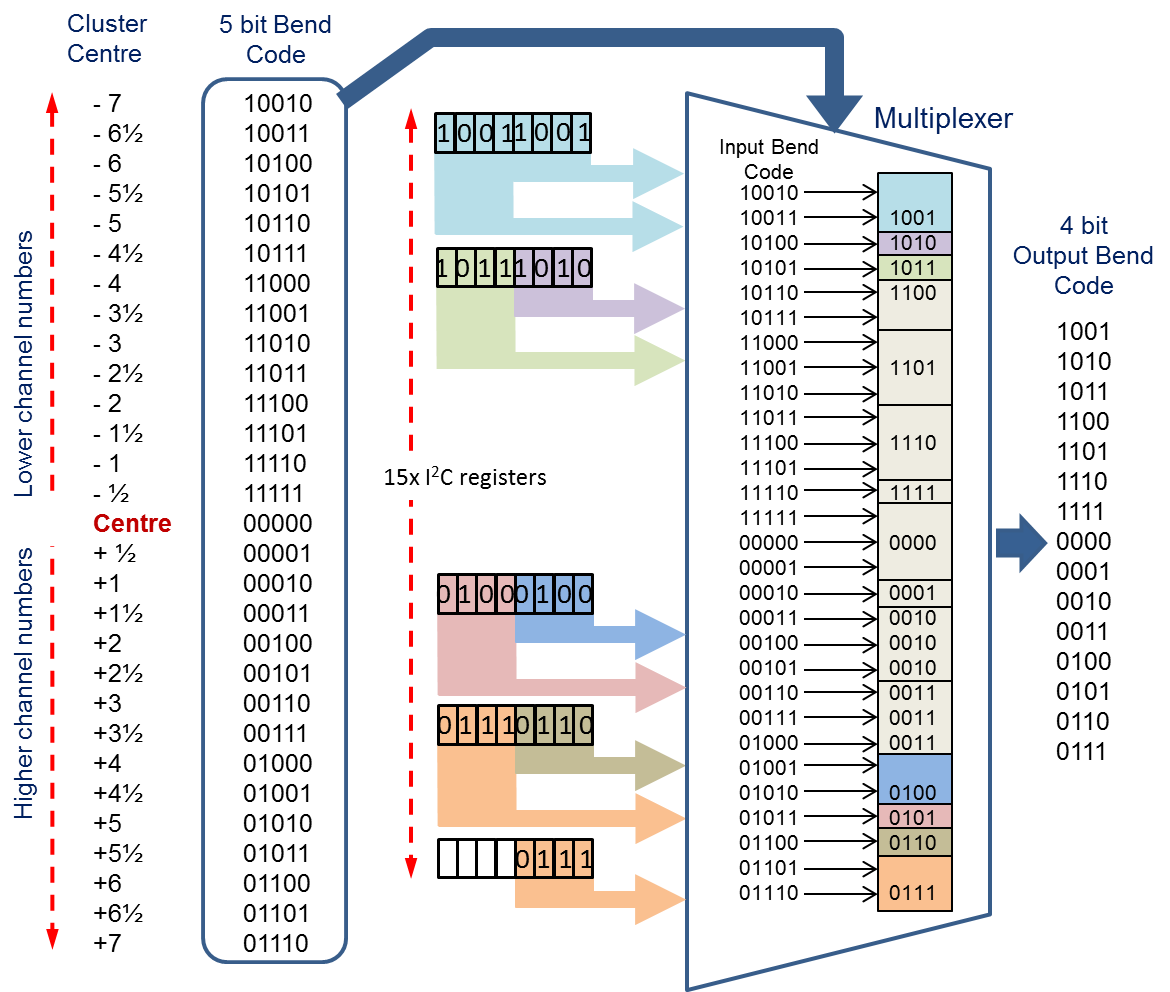
*Figure 8. Design for transferring up to 3 Stub addresses.*



*Figure 9. Example design for the priority selection of Stubs.*

## Bend Look-up Table

In order to make optimum use of the output data bandwidth, a look-up table has been added to convert the 5 bit bend information generated by the Stub-finding logic into 4 bits for output from the CBC3. An I2C programmable register block will be used to define how the 5 bits map to 4 bits. This has the advantage of allowing the mapping to be agreed at a later date and even programmed differently for different regions of the tracker, or for different runs. We can also reverse the bend polarity using this method, which might be useful in the case of modules that are flipped on the rod/stave. The programming is achieved using 15 I2C registers, where each register contains two sets of 4 bit bend codes corresponding to two of the original 5 bit codes from the Correlation logic. Figure 10 illustrates how different bend resolutions can be programmed.



*Figure 10. Illustrating the mapping of 5 bit bend information to 4 bit output.*

**NOTE:** The default value in the I2C registers will be set as the four most significant bits of the original 5 bit bend code associated with each register.

## Pipeline Memory and Output Buffer

* **Pipeline Memory:** The pipeline memory will consist of dual-port SRAM cells modified to improve their radiation hardness over those in CBC2. The depth of the pipeline will double to 512 bunch crossings to match the latency requirement of 12.8µs. The buffering of control signals and data write/read will be increased in drive strength where necessary, to match the larger pipeline size.
* **Output Buffer:** Triggered data from the pipeline will be buffered in a FIFO as with previous versions. The depth of the FIFO will remain at 32 L1 triggered events, but the readout rate will increase to match the requirement for a 320 Mbps data output rate.

## L1 Counter

A triple redundant 9 bit counter will be added to the CBC3 to count the number of L1 triggers received (up to 511). The counter will be reset by either the ‘fast’ reset OR its own dedicated reset (e.g. every orbit). The count will be output as part of the ‘Triggered’ data stream (figure 12).

## Data Assembly and Output Logic

The Data Assembly logic will take the address (8 bits for ½ strip resolution) and bend information (4 bits) for up to three Stubs and combine it with the Information Flags, for output onto five SLVS differential outputs operating at 320 Mbps. Figure 11 shows one possible arrangement of the data packet.

The ‘Stub Overflow’ flag (1 bit) generated by the Stub Gathering Logic, will indicate the presence of more than 3 stubs. The OR254 flag (1 bit) will indicate a hit on any of the 254 channels. The Error Flags (1 bit) is the logical OR of the Latency Error Flag and the FIFO Overflow Flag, both of which also appear separately in the L1 Triggered Data strea, and which are also stored in an I2C register which can be read. A latency error occurs if the circuit monitoring the pipeline operation detects a discrepancy between the actual latency and programmed latency. The FIFO Overflow Flag will be set if a burst of triggers occurs such that the FIFO overflows. Both errors will be reset by a fast reset (see Serial Fast Command section). The Synchronisation bit (1 bit) will be used to provide the Concentrator chip with logic “1” for synchronization purposes.

Readout of L1 triggered data from the pipeline memory will be via a dedicated SLVS PAD at the rate of 320Mbps. Figure 12 illustrates the format of this serial data stream.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Stub1<0> | Stub2<0> | Stub3<0> | Bend1<0> | Bend3<0> | L1 Triggered Data |
| Stub1<1> | Stub2<1> | Stub3<1> | Bend1<1> | Bend3<1> | L1 Triggered Data |
| Stub1<2> | Stub2<2> | Stub3<2> | Bend1<2> | Bend3<2> | L1 Triggered Data |
| Stub1<3>  25ns | Stub2<3> | Stub3<3> | Bend1<3> | Bend3<3> | L1 Triggered Data |
| Stub1<4> | Stub2<4> | Stub3<4> | Bend2<0> | S Overflow | L1 Triggered Data |
| Stub1<5> | Stub2<5> | Stub3<5> | Bend2<1> | OR254 | L1 Triggered Data |
| Stub1<6> | Stub2<6> | Stub3<6> | Bend2<2> | Error Flags | L1 Triggered Data |
| Stub1<7> | Stub2<7> | Stub3<7> | Bend2<3> | Sync = 1 | L1 Triggered Data |

Six Differential SLVS Outputs operating at 320 Mbps

*Figure 11. Illustration of the proposed data packet. The direction of time flow is* ***bottom*** *to* ***top****, i.e. Stub1<7> is output first.*



*Figure 12. Illustration of the serial L1 triggered data stream.*

**NOTE:** There will naturally be gaps between the frames of L1 Triggered Data due to the mismatch in trigger rate and data transfer rate. It is also expected that there will be a gap between consecutive frames due to the change from 40MHz clock domain to the 320MHz clock domain. The exact length of this gap will be determined by the circuit design. All gaps will be filled with zeros.

## Fast Control

There will be a serial command input operating at 320 MHz (differential SLVS) for fast commands (figure 13). The 40 MHz reference clock to the CBC3 will be derived from the serial command input by detecting the 110 sync pattern (bits 7, 6 and 5) present in every 8-bit frame. The four fast command bits follow the sync pattern.

CBC3 Commands are: Fast Reset (bit 4), Trigger (bit 3), Test Pulse Trigger (bit 2) and Orbit Reset (bit 1)

The last trailing bit (bit 0) in the frame is set to 1, to prevent the 110 sync symbol being confused with a test pulse trigger coinciding with an orbit reset. In normal operation no other combination of fast control bits can ever reproduce the 110 sync pattern.



*Figure 13. Illustrating the proposed serial command interface data.* *Direction of time flow left to right (i.e. bit 7 arrives before bit 6).*

## Clock Domains

Unlike the CBC2, the CBC3 will have split clock domains, as shown in figure 1. The 40 MHz clock domain is used for most of the circuits where the timing of the bunch-crossing period is the important factor, so circuits such as the Hit Detect, Stub-finding and Stub gathering logic, and the pipeline logic, all use the 40 MHz clock. This 40 MHz clock domain will be supplied by a programmable Delay Locked Loop (DLL) circuit, which is fed by the 40 MHz reference clock derived from the serial command interface data. The programmable DLL will allow for CBC3s to be tuned to the arrival time of the particles, thus adjusting for time-of-flight from the interaction point.

The 320 MHz clock domain is used for the Serial Fast Command interface and the high speed readout of the data. Data produced in the 40 MHz clock domain will have to be re-synchronised to the 320MHZ domain. Re-synchronisation will occur at the input of the Data Assembly Logic and the input to the Parallel-in-serial-out (PISO) shift register on the output of the pipeline buffer. Similarly, serial fast commands that pertain to circuits in the 40 MHz domain, must be re-synchronised to the 40 MHz domain. The latter may inevitably lead to latency between the receipt of the command and it taking effect on the chip.

The re-synchronisation will be performed using FIFO architectures designed to write and read on different clock domains.

**NOTE:** Due to the need to pass hits between neighbouring CBC3s on a module, as part of the Stub finding operation, the DLL circuits of each CBC should be tuned to have the same delay. Small differences of one or two nanoseconds should have little impact, but the greater the difference, the more likely it is that correlations across chip boundaries will be missed and potentially false correlations generated, as hits from one bunch-crossing are delayed into the next bunch-crossing.

## 40 MHz Test Mode

There is a requirement that the Bend Look-up Table, Fast Control Interface,the Data Assembly Logic and the PISO shift register must all be able to cope with operation of the 320 MHz clock domain at 40MHz during the wafer-level testing. This test mode will be programmed by a bit in the I2C and used for functionality tests during wafer probing.

## Delay Locked Loop (DLL)

A DLL is currently used to adjust the timing of the calibrate edge. This circuit will remain in use and will be reused for the adjustment of the 40 MHz clock for the 40MHz clock domain.

## Logic SEU immunity

The SEU hardening will remain as for the CBC2.

I2C registers containing data for the configuration of the chip will be modified to use SEU robust Whitaker Flip-Flops. These flip-flops are already used throughout the CBC2 in important areas such as the I2C control and the pipeline control registers.

Any synthesised logic circuits will use triple redundant logic where SEU upset would cause problems.

It remains the case that corruption of data in the pipeline memory can be tolerated for a reasonable SRAM cross-section.

# External Interfaces

## Clocks

There will be a single 320MHz SLVS clock input which must be terminated externally.

## Fast Control

There will be a single serial command input operating at 320 MHz (differential SLVS, must be terminated externally) for fast commands.

## Slow Control

The baseline option is to reuse the I2C slave circuit from the CBC2 which operates at 1MHz.

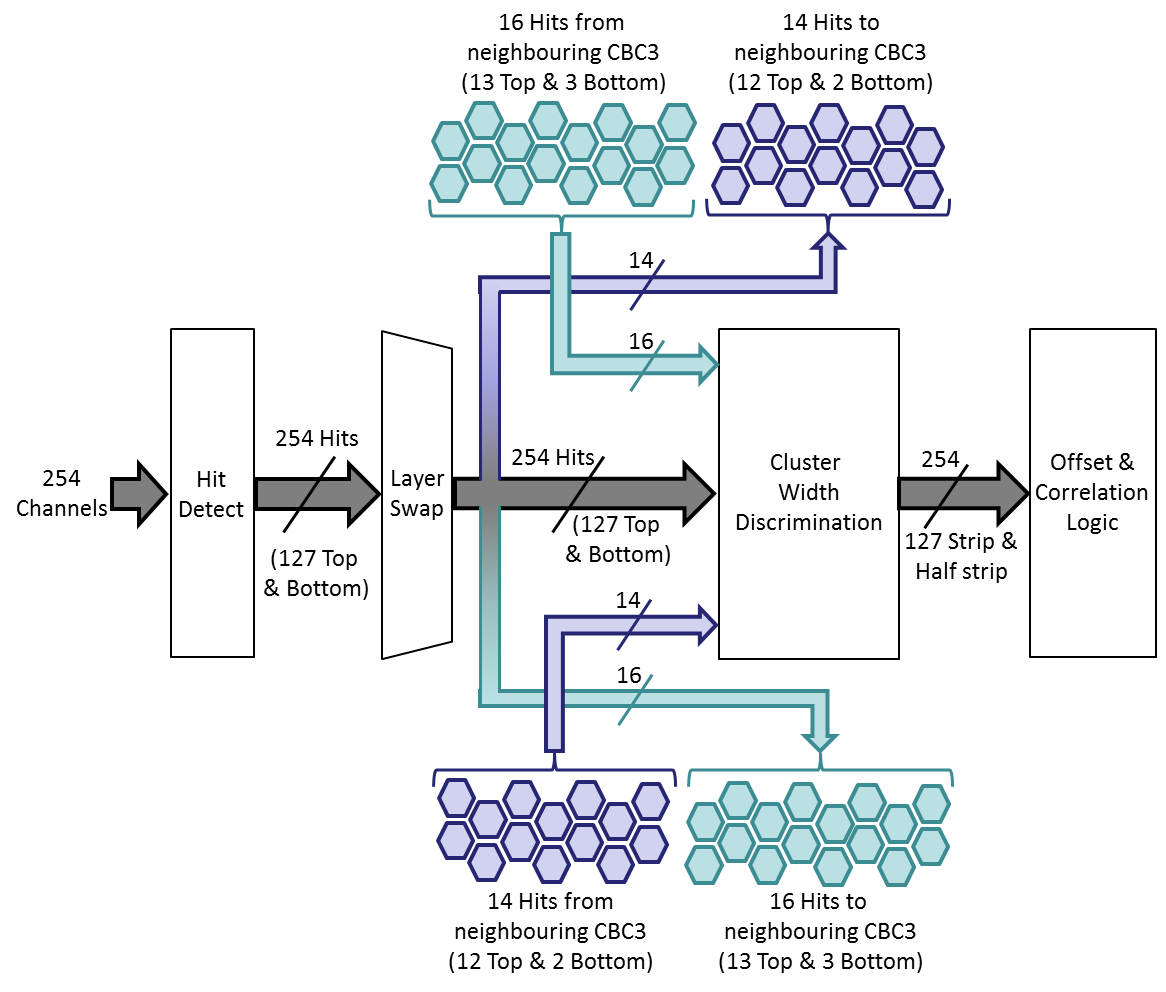
## Hard Reset

A CMOS level input signal (VDDD level) can be supplied to reset the chip to it’s power-on default state. The signal will be active high.

## 

## Inter-chip signals

As with the CBC2, signals must be passed between chips for the correlation to be performed. Unlike with the CBC2, these signals will originate from the Hit Detect circuit and not the output of the Cluster Width Discrimination logic. This requires extra logic in the cluster width discrimination and correlation circuits, but reduces the number of inter-chip signals required. A total of 60 pads are required for the inter-chip signals. Each CBC3 will output the hits from the twelve highest numbered channels assigned to the Correlation layer (Top), along with the two highest numbered channels assigned to the Seed layer (Bottom). Each chip will input these same signals from its neighbour for comparison with hits from its own lower numbered channels. Similarly, each CBC3 will output the hits from the thirteen lowest numbered channels assigned to the Correlation layer (Top), along with the three lowest numbered channels assigned to the Seed layer (Bottom). Each chip will input these same signals from its neighbour for comparison with hits from its own higher numbered channels. Figure 14 illustrates the inter-chip signals.



*Figure 14. Illustrating the inter-chip signals required for finding stubs.*

# Analogue Biases

## Master Reference Current Generator (CBC\_IREF)

The output circuit of the opamp that takes the bandgap voltage and uses it to generate a master reference current Ibias of CBC\_IREF will be modified so that Ibias is determined by a resistor to ground rather than to VDDA. The voltage at the top end of the resistor will be controlled to be the bandgap reference voltage VBG in a similar fashion to the current circuit.

## Comparator Threshold Reference (Vcth)

The existing circuit is not sufficiently monotonic. The circuit will be modified to be monotonic and provide 1 mV resolution (10 bit). This will be achieved with a resistor string DAC.

## Post-Amp Reference Voltages (VPLUS & VPLUS2)

The Post-Amp circuit has been modified and it now requires two reference voltages that will be set equal, but separately sourced to avoid cross-talk. These will be implmented by two four bit resistor string DACs. They will be programmable via I2C and their default setting will be the midpoint value.

## Bandgap Reference

Radiation has an effect on the bandgap, resulting in output variation. This has a knock-on effect for other circuits. A solution is to use a bandgap with a PMOS reference transistor which is robust to radiation, but suffers from process variation. In order to cope with the latter, the bandgap can be made adjustable. This trimming of the circuit can be done at wafer testing and then either hard wired by blowing e-fuses, or recorded for later programming.

# Power

Unlike the previous versions of the CBC chip, there will be no DC-DC converter on the chip. VDDD (1.2V +/- 10%) will be provided and will be filtered through an on chip LDO to provide a lower voltage rail to the analogue sections. The analogue front-end will be adjusted (by Imperial College) to make sure it works comfortably at 1V, to accommodate the minimum possible value of VDDD.

The target power consumption for 5cm strips (~8pF) is 450 µW/Channel, assuming 350 µW/Channel analogue and 100 µW/Channel digital power consumption.

# Pads

Similar to CBC2: 254 inputs, with no active circuitry placed under the pads to avoid noise injection and mechanical-stress failures. Input pads and other signal pads to be separated by at least 2 columns of dummy pads (with balls grounded on the hybrid side) to limit parasitic capacitance (may consider the use of gnd pads instead).

The last column of pads will be wire-bondable pads for wafer-probing, as for the CBC2.

Redundancy will be built in by doubling pads where possible, to increase yield in bumping on the module.

Bumps will be C4 lead-free bump bonding using the same 5on10 arrangement (i.e. 250um pitch) used on CBC2.