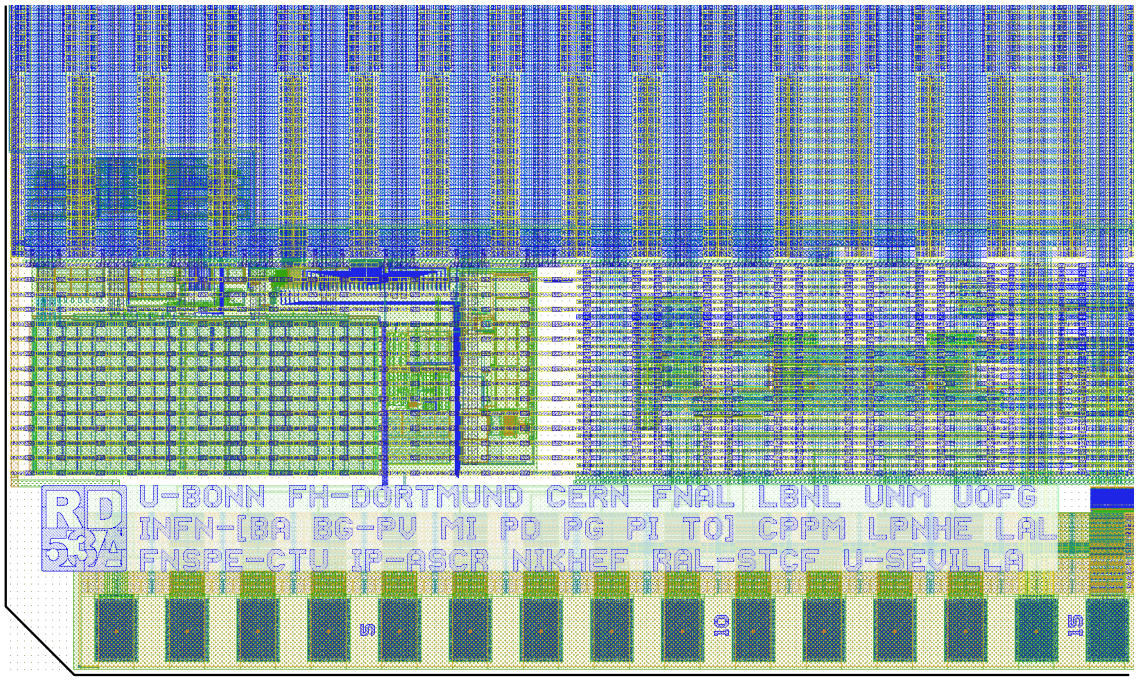
CERN-RD53-PUB-17-001

Version 3.14, November 7, 2017

The RD53A Integrated Circuit

ABSTRACT: Implementation details for the RD53A pixel readout integrated circuit designed by the RD53 Collaboration. This is a companion to the specifications document and will eventually become a reference for chip users. RD53A is not intended to be a final production IC for use in an experiment, and contains design variations for testing purposes, making the pixel matrix non-uniform. The chip size is 20.0 mm by 11.8 mm.



Contents

1. Introduction

The RD53A integrated circuit specifications [1] were approved in Fall 2015 after review by the ATLAS, CMS, and RD53 collaborations. Chip design to meet these specifications began in 2016. This document describes the design and implementation choices and is expected to eventually become a chip users' guide. RD53A is intended to demonstrate in a large format IC the suitability of the chosen 65nm CMOS technology for HL-LHC upgrades of ATLAS and CMS, including radiation tolerance (Sec. A), stable low threshold operation, and high hit and trigger rate capabilities. RD53A is not intended to be a final production IC for use by the experiments, and will contain design variations for testing purposes, making the pixel matrix non-uniform (Sec. 5). It will form the basis for the production designs of ATLAS and CMS. The design revisions from RD53A to production will involve selecting one of the front end variants and uniformly using it everywhere, increasing the pixel matrix size as specified by each experiment, and tailoring the digital functionality to according to the experiment specifications. These should all be achieved by changing parameters and high level description code and re-running the RD53A design flow, which is developed with such flexibility in mind (App. B). RD53A will be fabricated on an engineering wafer run shared with the CMS MPA chip in order to control prototyping cost.

The physical size and pad locations are shown in Fig 1. Sec. 4 describes the pads in detail. The width of RD53A is 20 mm, as it expected that the final production chips will be 20 mm or wider and can therefore reuse the bottom of chip and pad frame. The height of the RD53A chip is constrained by available space on the shared reticle submission. The RD53A pixel matrix is 400 pixels wide by 192 pixel tall. Production chips are expected to increase the number of rows and remove the top row of test pads. The power and bias distribution have been designed for a larger number of rows, up to 384. The labels top, bottom, left, and right refer to the orientation as shown in Fig 1. Alignment crosses are included on the top left and right sides, while at the bottom there are small crosses (the size of one bump bond pad) described in Sec. 4.

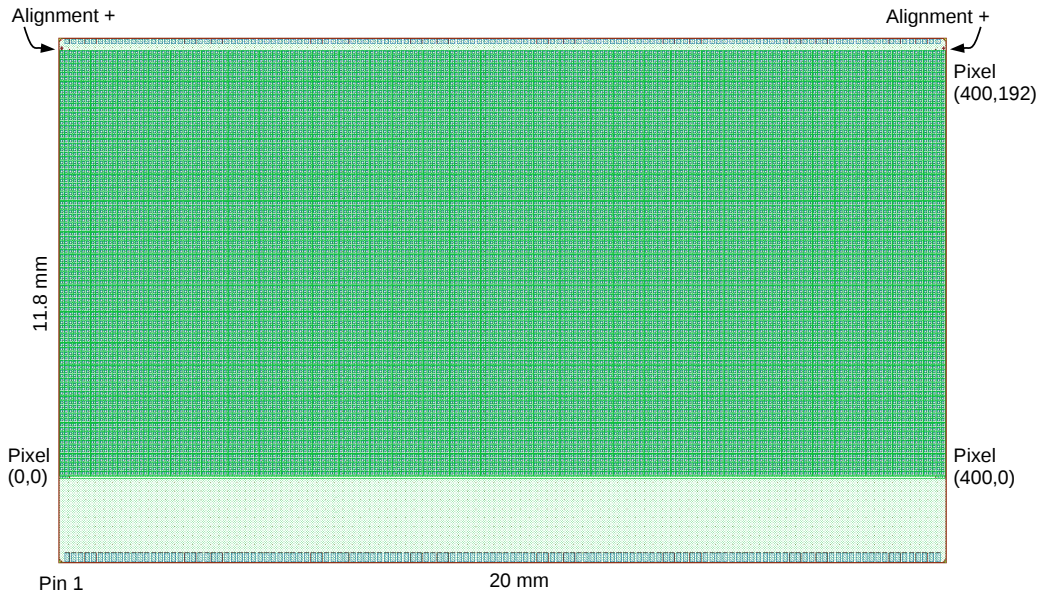


Figure 1: Top view of RD53A layout. The width is 20mm and 400 pixels, and the height is 11.8mm and 192 pixels.

2. Floorplan and Organization

RD53A uses a 9 metal layer stack, consisting of 7 thin, 1 thick and 1 ultra-thick metal layers. In addition, the 28K AP layer is also used for power lines distribution. In Fig. 2 and Fig. 3 the layout and functional view of RD53A floorplan are shown. The sensitive area of the chip is placed at the top of the chip and is arranged as a matrix of 192 x 400 pixels of 50 μm $\times$ 50 μm. At the top is a row of test pads for debugging purposes, which should be removed in a production chip. The peripheral circuitry is placed at the bottom of the chip and contains all global analog and digital circuitry needed to bias, configure, monitor and readout the chip. The wire bonding pads are organized as a single row at the bottom chip edge and are separated from the first row of bumps by 1.7 mm in order to allow for wire bonding after sensor flip-chip (Sec. 4). However, the wire bond pads are also designed compatible with Thru-Silicon Via post processing.

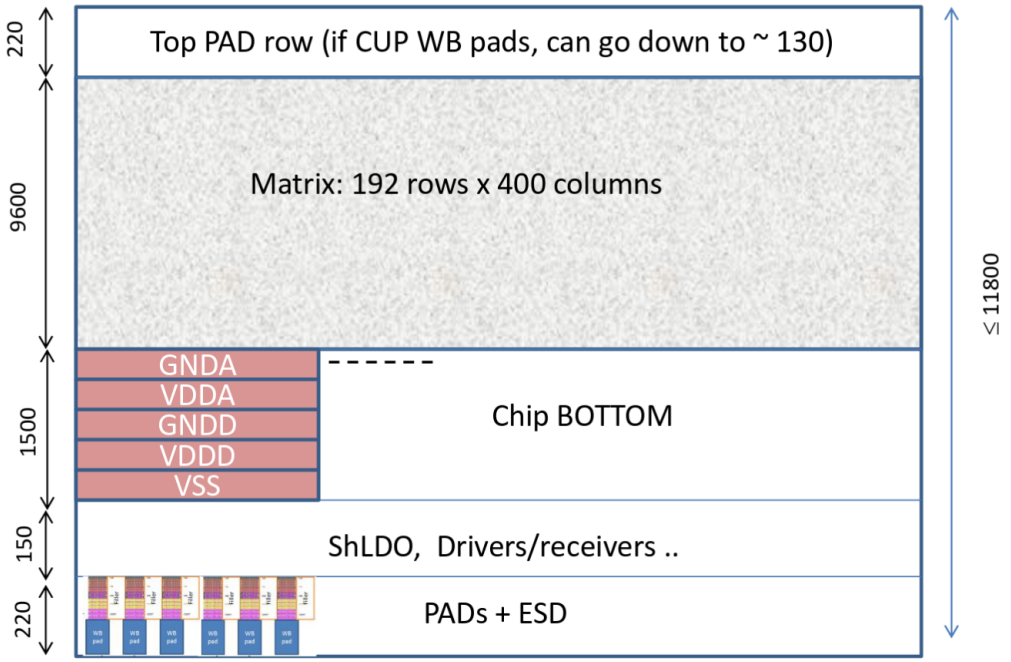


Figure 2: RD53A floorplan, layout view.

The pixel matrix is built up of 8 by 8 pixel *cores*. The 64 front ends within a core are placed as 16 so-called analog islands with 4 fronts ends each, which are embedded in a flat digital synthesized "sea" as shown in Fig. 4. The circuitry around each island is not identical but depends on the placement of gates by the synthesis tool. Prototype tests have shown that, within the digital/analog isolation scheme used, this approach does not introduce any visible systematic differences between islands. Furthermore, a core is small enough that it can be checked with a transistor level analog simulation. In the chip periphery, all the analog building blocks are grouped in a macroblock called Analog Chip Bottom (ACB), which is fully assembled and characterized in an analog environment (Sec. 8). All the building blocks have been previously prototyped, tested and characterized in radiation environment at least up to 500 Mrad TID. The ACB block is surrounded by a synthesized

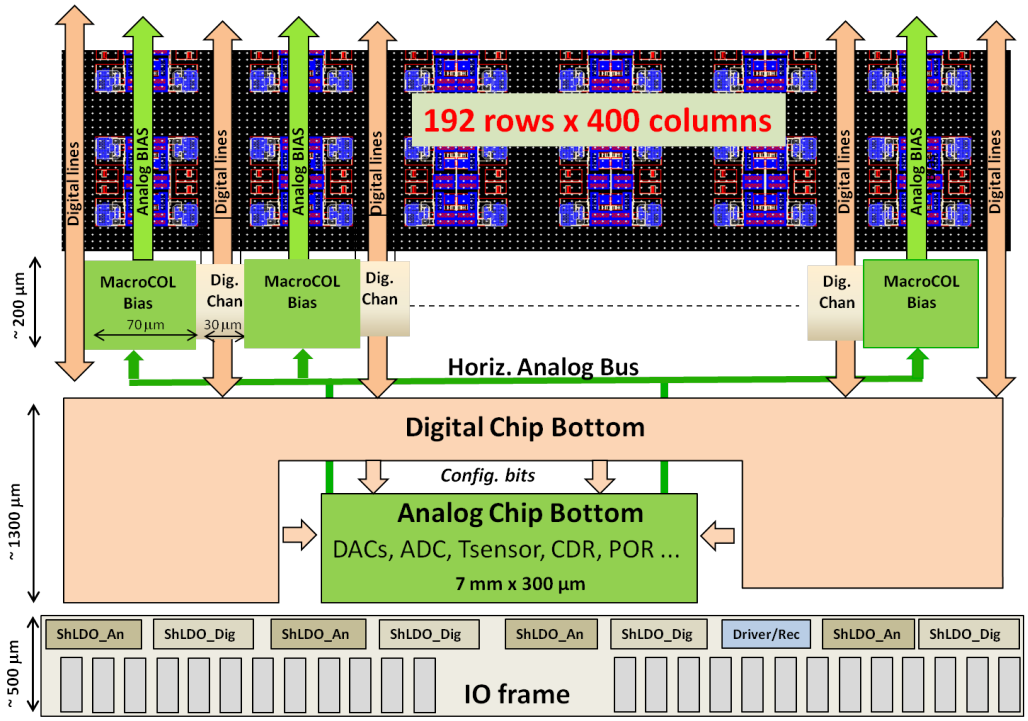


Figure 3: RD53A floorplan, functional view.

130 block, called Digital Chip Bottom (DCB), which implements the Input, Output and Configuration digital logic, as described in (Sec. 7) and (Sec. 9).

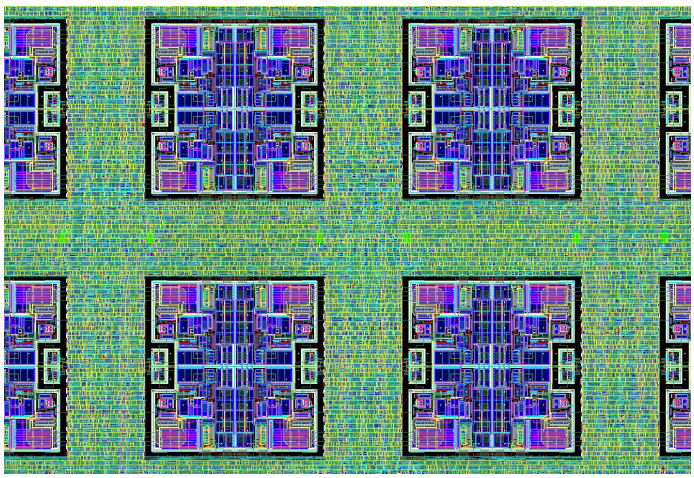


Figure 4: Layout of analog island concept.

3. Power Distribution

RD53A is designed for single supply serial powered operation with connections from the bottom of chip only. However, because it is a prototype that must have flexibility for a variety of measurements, it is also be possible to power the chip by directly supplying the internal rail voltages, bypassing the regulation. There are two internal voltage rails for powering the chip core, analog (VDDA) and digital (VDDD). In addition, two blocks (serializer/CML cable driver and PLL) have their individual power pads: VDD_CML/GND_CML and VDD_PLL/GND_PLL, respectively. User must texternally connect these power pins to one of the two rails. This was done to allow testing of power sensitivity of these circuits. The ESD protection scheme uses additional rails as discussed in Sec. 3.3. The separation between local analog ground, digital ground, and substrate is discussed in Sec. 3.2. The operation of a serial powered detector is new, and system design requires a simplified load module of the chip to understand how to design the power distribution and cooling and how the system responds to transients, faults, etc. Such a model is given in Sec. 3.4. The chip has a Power On Reset (POR) covered in Sec. 7.2. When power is applied it should be ramped up fasteer than the POR delay for proper reset: this delay is nominally 50 ms. Table 1 gives the power supply limits.

Parameter	**Typ.**	**Max.**
Core direct supply voltage	1.2 V	1.32 V
ShuLDO input voltage	1.5 V	2.0 V
Per pixel analog current	4 μA	8 μA
Per pixel digital current	4 μA	6 μA
RD53A Periphery analog current	30 mA	60 mA
RD53A Periphery digital current	30 mA	60 mA
Output drivers (each)	20 mA	30 mA
Total RD53A current (4 outputs)	0.75 A	1.3 A
400x384 chip periphery analog current	30 mA	60 mA
400x384 chip periphery digital current	60 mA	120 mA
Total 400x384 chip current (4 outputs)	1.3 A	2.3 A

Table 1: Power supply limits

To meet the above needs, RD53A uses "active power pads" as shown in Fig. 5. The pad has an active mode and a passive mode. The passive mode is activated by tying the external supply and the V_rail pins together and disabling the Shunt transistor, in which case the user externally controls the corresponding internal voltage rail. In the active mode the V_rail pin can be used to connect external decoupling capacitors, but no DC current flows on it. Special voltage regulators called ShuLDOs (short for Shunt-LDO) are used. The ShuLDO control circuit ensures that the external supply pad sinks a constant input current independent of the internal circuit consumption, and generates a regulated output voltage (V_rail) to the chip interior. The voltage rail (ShuLDO output) can be adjusted through the chip configuration (Table 22). The ShuLDO control circuit is common to all the power pads feeding the same rail and is described in Sec. 3.1. Any excess current

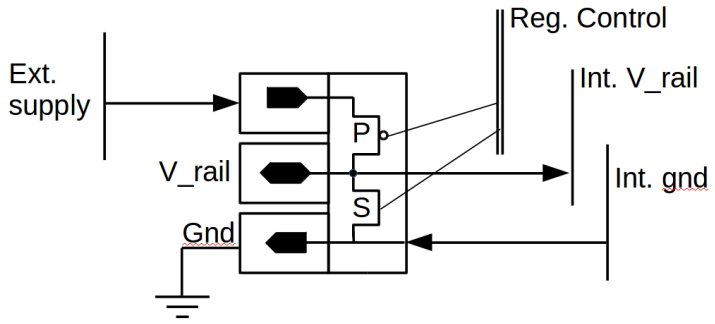


Figure 5: Active power pad as described in the text. The labels P and S stand for Pass and Shunt transistors, which are actively controlled by the ShuLDO regulator circuit.

not needed by the chip internal circuits is shunted to ground, with the sum of chip core and shunt currents remaining constant. Use of a common control circuit permits having multiple active pads in parallel, distributed along the chip bottom. This avoids having a single power device (regulator) where all the current for a given rail must be concentrated, which would present reliability challenges. The shunted current to ground will be evenly distributed among all the parallel active pads, as this is a feature of the ShuLDO design. The chip contains two ShuLDO control circuits, one for the analog and another for the digital internal rails, and 4 active pads associated with each. As the absolute maximum current consumption of a full size (400 pixels by 384 pixels) chip would be 1.5 A each analog and digital, this means that each active pad must be rated for 375 mA plus some derating for current sharing and failures. Each active pad has therefore been designed for 500 mA maximum current, and assigned 5 input and 5 ground wire bonding pads. The absolute maximum current rating for the full chip is therefore 4.0 A.

3.1 ShuLDO Voltage Regulators

The Shunt-LDO regulator (ShuLDO) is a combination of a low-drop linear voltage regulator and a shunt regulator. The regulator can be configured as a pure Low Drop-Out linear voltage regulator for usage in a conventional voltage based supply scheme. In addition, the regulator provides dedicated shunt circuitry which can be enabled for application in a current based serially powered supply scheme. The power device of each regulator, namely the pass, and the shunt device, are each split into four parts and spread across the pad frame to balance power distribution, but there is a single control circuit per regulator.

A simplified circuit of the Shunt-LDO regulator is shown in Fig. 6 and the inputs and outputs are listed in Table 2. The LDO regulator part is formed by the error amplifier A1, the PMOS pass transistor M1 and the voltage divider formed by the resistors R1 and R2. In a voltage based supply scheme, the unregulated input voltage is applied to the V_{IN} pin while GND corresponds to local ground. The regulator generates an output voltage VDD=$2 \times V_{REF}$ where V_{REF} is the reference voltage which is provided to the inverting input of the error amplifier A1.

In a constant current scheme the supply current is flowing into the V_{IN} pin. The pass transistor

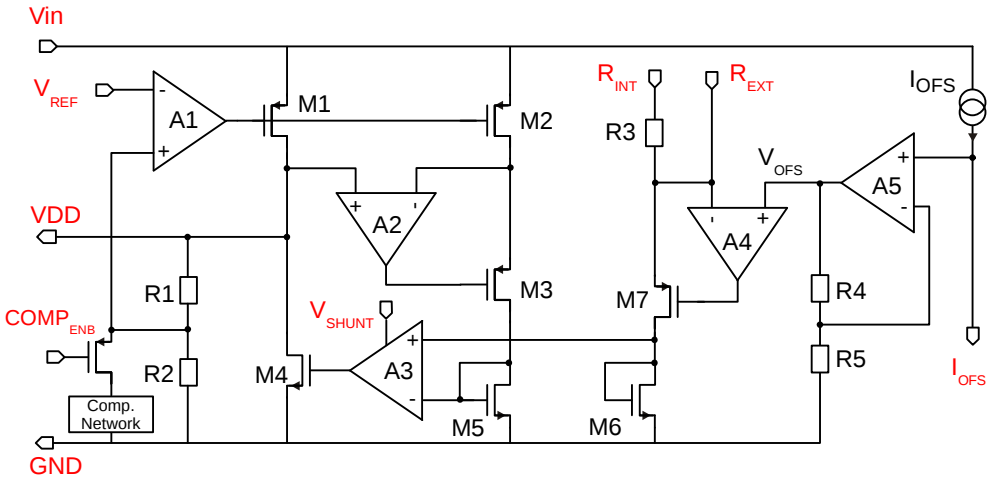


Figure 6: LDO regulator with shunt capability (Shunt-LDO)

Pin	Type	Min.	Typ.	Max.	Description
V_{IN}	Power	1.4 V		2.0 V	external power input (voltage)
	Power	0	0.5 A	2.0 A	external power input (current)
V_{SHUNT}	Power	1.4 V		2.0 V	supply voltage of shunt circuitry
GND	Ground				local ground and shunt current output
VDD	Power	1.0 V	1.2 V	1.32 V	regulator output
V_{REF}	Analog	500 mV	600 mV	660 mV	reference voltage (VDD=2V_{REF})
R_{INT}	Analog		V_{IN}		supply/enable internal ref. R
R_{EXT}	Analog	300 Ω			external ref. R to V_{IN}
I_{OFS}	Analog		200 kΩ		external offset R to GND
$COMP_{ENB}$	Digital		GND		GND to enable compensation

Table 2: Shunt-LDO regulator circuit pins

M1 is steered to create a voltage drop V_{DS} between the regulator input V_{IN} and the output voltage terminal, VDD, such that the desired output voltage is generated with respect to local ground. The shunt transistor M4 is added to provide an additional current path to GND: it is controlled to drain all current which is not drawn by the load connected to VDD. For this purpose the current flow through transistor M1 is compared with a reference current which is defined by use of resistor R3, amplifier A4 and transistor M7. Due to the negative feedback loop, the amplifier A4 is applying the offset voltage V_{OFS} to the lower terminal of resistor R3, while the V_{IN} potential is connected to the upper terminal. If the voltage at the V_{IN} terminal is higher than the offset voltage V_{OFS}, the reference current is flowing through resistor R3 and given by:

$$I_{ref} = \frac{V_{IN} - V_{OFS}}{R3} \tag{3.1}$$

If the voltage at the V_{IN} terminal is lower than the offset voltage V_{OFS}, the reference current equals

to zero. While the reference current is drained into the gate-drain connected transistor M6, a fraction of the current flowing through transistor M1 given by the aspect ratio k of the current mirror formed by transistor M1 and M2, is drained into the gate-drain connected transistor M5. The value of k is 1000 and thus the total current in shunt mode is 1000 times the reference current. The amplifier A2 and the cascode transistor M3 are added to improve the mirroring accuracy. The reference current is compared to the fraction of current flowing through transistor M1 by use of the differential amplifier A3. If the current drained to transistor M6 is smaller than the reference current, the shunt transistor M4 is steered to draw more current and vice versa. By this means, a constant current independent of the regulator load is flowing through transistor M1 with a value defined by:

$$I_{IN} = k\frac{V_{IN} - V_{OFS}}{R3} \tag{3.2}$$

The resistor R3 is integrated internally and has a design resistance of 600 Ω. R3 sets the reference current if R_{INT} is connected externally to V_{IN}. However the reference current can be set to any desired value by using an external resistor instead to R3, placed between R_{EXT} and V_{IN}. In this case R_{INT} is kept floating (or if it is not then the external and internal resistors add in parallel). Note that for a reasonable offset voltage of around 1 V the total current set by R3 will be fairly large: 667 mA at V_{IN}=1.4 V, which significantly exceeds the RD53A nominal analog or digital current, and is instead of the order of a full chip analog or digital current. Therefore, for efficient RD53A operation R_{EXT} should be used instead of R_{INT} (alternatively a series resistor can be added on R_{INT} and R_{EXT} can be left unconnected).

For the generation of the offset voltage V_{OFS}, the bias current I_{OFS} of 2 μA is drained to an external resistor which is connected between the I_{OFS} pin and ground. The voltage is amplified by factor two by the non-inverting amplifier A5 and fed to A4. The generated offset voltage is $V_{OFS} = 2I_{OFS}R_{OFS}$, where R_{OFS} is the external resistor connected to the I_{OFS} pin.

A compensation circuit for stable operation of the regulator circuit with low ESR output capacitors is enabled by the digital port $COMP_{ENB}$, which is active low. This circuit should be used ($COMP_{ENB}$) for normal operation. Shunt operation is disabled by shorting the Rext and VDDShunt ports to the local ground port GND. The amplifier A3 has a dedicated power pin, V_{SHUNT}, so that the shunt transistor M4 can be completely off in case of using passive mode of the power pins to directly control the internal supply rails.

3.2 Power and Ground Network and Substrate Isolation

Digital and Analog power and digital and analog return (ground) are independently distributed throughout the pixel matrix. The digital and analog grounds are expected to be connected together with low impedance off chip. The ESD rail force the connection of analog and digital grounds at a single point on-chip, in the wire bond pad frame (see Sec. 3.3).

All circuits (both analog and digital) have been placed in deep N-wells for maximum possible substrate isolation. The only exceptions are the ESD protection structures of the wire bond pads, as descried in Sec. 3.3.

3.3 ESD Protection and Safe Wire Bonding

In the bottom pad frame the four power domains (VDDA, VDDD, VDD_CML, and VDD_PLL) are

isolated by power-cut cells. Within a power domain group, the IO pad ESD devices are connected to the respective power rails. To allow an ESD path between power domains a common ESD bus connects to every power domain's ground rail via sets of anti-parallel diodes. This ESD bus is also used to connect the global substrate of the chip (VSUB). Because the IO pads use ESD devices connecting to the power rails, care must be taken to not drive signals to the chip while it is not powered, as this would supply parasitic power. There are a few pads which have an over-voltage tolerant ESD protection without a current path to the power rail. These pads are used where the input voltage can exceed the VDDA/VDDD rail potentials (input- and bias pads of the shunt regulator blocks, for example). Where low capacitance is mandatory (CML driver output pads) a path to the VDDA/VDDD rails was also omitted.

The top test pads have an independent power domain (VDD_TOP/GND_TOP) which is not connected to the global ESD bus at the chip bottom. Within the top row the IO pads are protected but since there is no connection to the global ESD bus, ESD events between top and bottom pads should be avoided. The IO pads in the top pad frame are all over-voltage tolerant and therefore output signals are not clamped if the top row is not powered.

The wire bonding sequence to avoid ESD problems is as follows: Start with the VSUB pads (14, 88 and 184) followed by all GND pads in any order. Finally bond the remaining pads in any order. The top row pads can be left floating, but if wire bonded then start with the GND pads (T2, T51 and T97) followed by the VDD pads (T1, T50 and T96) and then all other pads.

3.4 Load Model of RD53A for Serial Power System Design

The ShuLDO regulator as seen by the external system can be modelled as an effective resistance, $R_{\text{effective}}$, in series with a voltage offset, V_{offset}. The internal chip core load is not directly visible to the external user. As shown in Fig. 7, the two parameters $R_{\text{effective}}$ and V_{offset} can be typically chosen by considering two extreme operating points: the lowest and highest current and voltage, and connecting them by a straight line. The minimum voltage needed for regulation (1.4 V in the case of running with 1.2 V core voltage) should occur at the minimum required operating current, which is given by the current consumption of the chip core, without any additional current shunted in the regulators (Table 1. Clearly, at least this current must be supplied for the chip to work properly. For reliable operation the serial current supply setting will exceed this minimum current by some margin, and the module voltage will therefore be higher than 1.4 V according to the curve in Fig. 7. How much current margin to add on top of the core current (10%, 20%, etc.) is a user choice. Of course, the core current is also a user choice, as it is given by the selected analog biases, output modes and regulator output voltage settings. At the other extreme, the absolute maximum voltage should result from the the maximum current. Assuming a current higher than the absolute maximum will never be supplied, this guarantees that the absolute maximum voltage will never be exceeded. The comparison of these choices for RD53A and a full size chip is shown in Fig. 7. The effective resistance values in the figure are (2.0 V-1.4 V)/(4.0 A-0.75 A) = 185 mΩ for RD53A and 230 mΩ for a full-size chip. The slope is lower (less resistive) for RD53A because the offset voltage can be higher.

It should be clear from the above that the ShuLDO effective resistance $R_{\text{effective}}$ is not the load resistance of the chip core (core current over core voltage). Because of the voltage offset, the power consumption also has a constant offset $P = I_{\text{in}}^2 R_{\text{effective}} + I_{\text{in}} V_{\text{offset}}$. A low value of ShuLDO $R_{\text{effective}}$

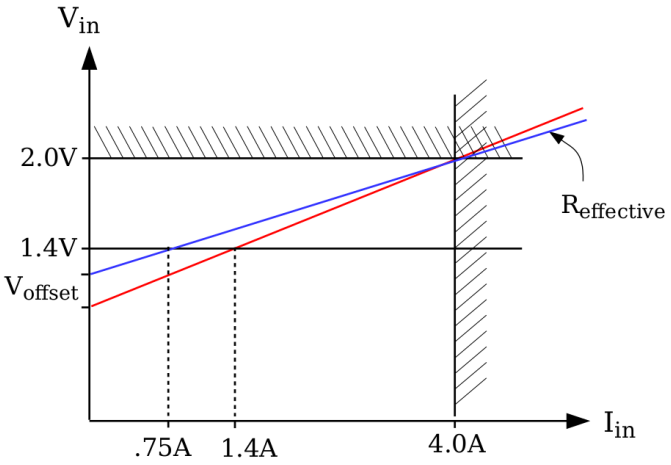


Figure 7: Voltage vs. current behavior of chip load model. The shaded regions show the absolute maximum ratings: 2.0 V maximum voltage and 4.0 A per chip maximum current. The horizontal line at 1.4 V shows the minimum voltage needed for operation with 1.2 V regulated chip core voltage. The slopes represent the effective resistances of the RD53A chip (blue) and an expected full size chip (red), respectively. The dashed vertical lines at 0.75 A and 1.4 A are the nominal currents (not including shunt regulator margin) for the RD53A and expected full size chip.

is desirable because it leads to small change in power with supply current, and so a small regulation overhead. For RD53A, with $R_{\text{effective}}$=185 mΩ, a 20% margin in the serial current results in a power overhead of 22%, while for a full size chip with $R_{\text{effective}}$=230 mΩ a 20% current overhead results in a power overhead of 26%. It is also possible to purposely choose lower $R_{\text{effective}}$ values to reduce the power overhead. This is in fact also desirable to provide safety margin against faults as discussed below. However, non-zero value is needed for multiple regulators to work well in parallel, and there is furthermore a lower limit to the achievable $R_{\text{effective}}$ from transistor performance. the estimated lower limit per RD53A chip (8 regulators in parallel) is ???. With this value of $R_{\text{effective}}$ a 20% current overhead translates to a ???% power overhead.

In a serial powered system, several chips are wired in parallel within a module, and several modules in series, as shown in Fig. 8. It is instructive to work through an example of 8 modules in series with 4 full size chips per module. We start by assuming the above $R_{\text{effective}}$=230 mΩ per chip and V_{offset}=1.08 V. Thus at a current of 1.4 A per chip (I_{supply}=5.6 A), V_{module}=1.4 V. To have some operating margin, we set the serial current 20% higher, I_{supply}=6.7 A. Now V_{module}=1.47 V. The total chain voltage drop is 11.8 V because there are 8 modules. There is also resistance in the 100 m cable runs. Let us assume a 2 Ω round trip cable resistance, and so the power supply output voltage is 25 V. Note 50% of the supplied power is lost in the cables: while this may sound bad, it is an improvement over present pixel detectors. Just as we would set a current limit on constant voltage supplies, we set a voltage limit on serial power supplies, so we limit them at 27 V. The maximum power that can be delivered to the detector is thus 27 V×6.7 A minus the cable loss, or 11.5 W per module, while the operating power is 9.8 W per module.

Now suppose one chip in one modules fails (since each chip has two independent regulator controllers, the more probable scenario is for half of a chip to fail). Since the total current is sill the

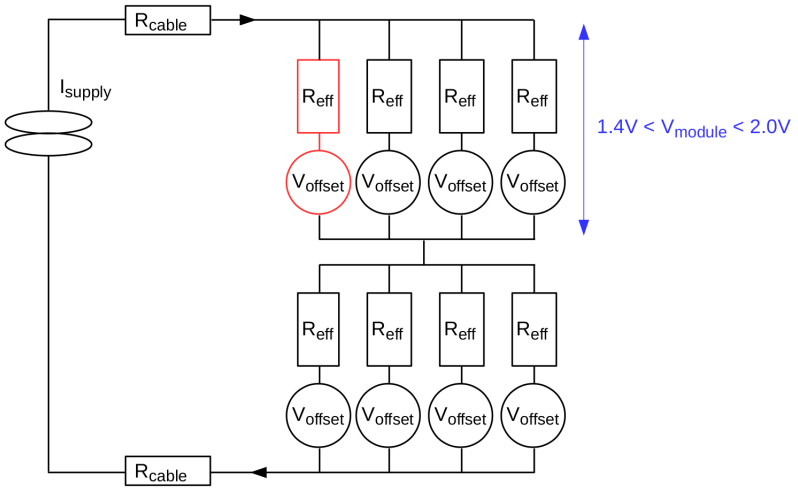


Figure 8: Simple model of multi-chip modules in a serial power chain. The chip model in red is referred to in the text to discuss the impact of failures.

same (forced by the supply), each of the three remaining chips must pass a current of 6.7/3=2.23 A. Therefore V_{module}=1.6 V. The power in this one module is now 10.7 W- a 9% increase. For the total 8-module chain power the increase is just 1%. The power supply output voltage will increase slightly to deliver the additional power, still well below the 27 V safety limit. In fact, even with 1 disabled chip in every module the supply will stay below the safety limit, since it is set to deliver up to 11.5 W per module. If 2 chips fail on a single module, then the per chip current is double the normal, 3.36 A, which is still below the absolute maximum rating of 4.0 A. The module voltage will be 1.85 V and the module power 12.4 W. While this is higher than the per module power supply limit, the full chain power remains well below the power supply limit and the system should continue to operate normally.

Finally we consider operation near the maximum ratings, should that be necessary. A single point failure "runaway" condition could be reached if the failure of one chip is able to trigger the failure of the rest of the module. In order to be safe against this scenario, the supply current must always be compatible with flowing through 3 chips instead of 4. Since the absolute maximum rated per chip current is 4 A, this means that the maximum safe supply current should be 12 A, which in a normal module translates to 3 A per chip. At this current, with our example $R_{\mathrm{effective}}$ and V_{offset}, V_{module}=1.77 V. The module power would thus be 21.2 W and the supply voltage, including cable drop, 38 V. If a chip then fails the module power rises to 24 W at V_{module}=2 V.

4. Bump Pads, Wire Bond Pads, and Alignment Crosses.

The bump pad shape and dimensions are shown in fig. 9. The passivation opening is square shape with 45 degree corners. The figure also shows the expected height profile across the center of a bump pad as derived from the metal stack.

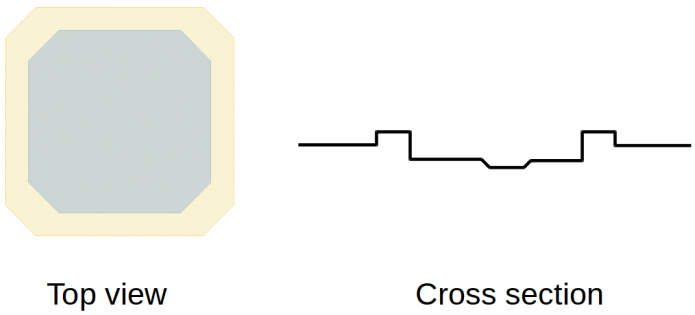


Figure 9: Bump pad layout for RD53A (left) and expected surface profile (right). The dark central area of the layout is the passivation opening, which is 12 μm wide, while the borders shows the pad metal extending an additional 1.5 μm on all sides under the passivation. The exposed metal is not completely flat and can have depressions less than 1 μm deep due to vias below. The profile shows such a depression.

The wire bond pads at the chip bottom are rectangular shape, with 58 μm wide by 86 μm tall passivation opening on a 100 μm pitch (pinouts can be found in Sec. 11). This size is considered large enough to allow for wafer probing followed by reliable wire bonding with the possibility of bond repair. Only the bottom of chip pads need to be connected for normal operation. The top of the chip has test pads for dedicated measurements. The top test pads use the same layout but placed on a 200 μm pitch. Figure 10 shows top and bottom layout details where the wire bond pads are visible. In the top left (right) corner there is a small alignment cross placed 50 μm above the top left (right) bump pad. There are also alignment crosses at the bottom of the matrix shown in Sec. 4.1. In the bottom left corner is the RD53A logo listing the collaborating institutes.

The bottom of chip pad layout is shown graphically in Fig. 11, including typical wire bond connections to as needed for realistic detector module operation (refer to Sec. 11 for pinout). There are four power groups, each with an analog and a digital Shunt-LDO regulator next to each other. The number of non-power wire bonds is small as only the serial control input and 1 to 4 high speed differential outputs are needed. A detail showing one bottom pad (oriented sideways) and ESD protection structures as well as the metal stack in each region, is showing in Fig. 12. Wire bond pads use most metal layers for compatibility with thru-silicon vias. The lowest two layers, M1 and M2 are excluded from the stack to avoid a high capacitance to the substrate, which would be problematic for high speed pads. This means TSV processing needs to drill through additional oxide to reach M3, which is considered an acceptable compromise. The bottom of chip pin assignments are given in Table 12 with pin 1 on the left side. The top test pin assignments are given in Table **??** with pin T1 on the left side.

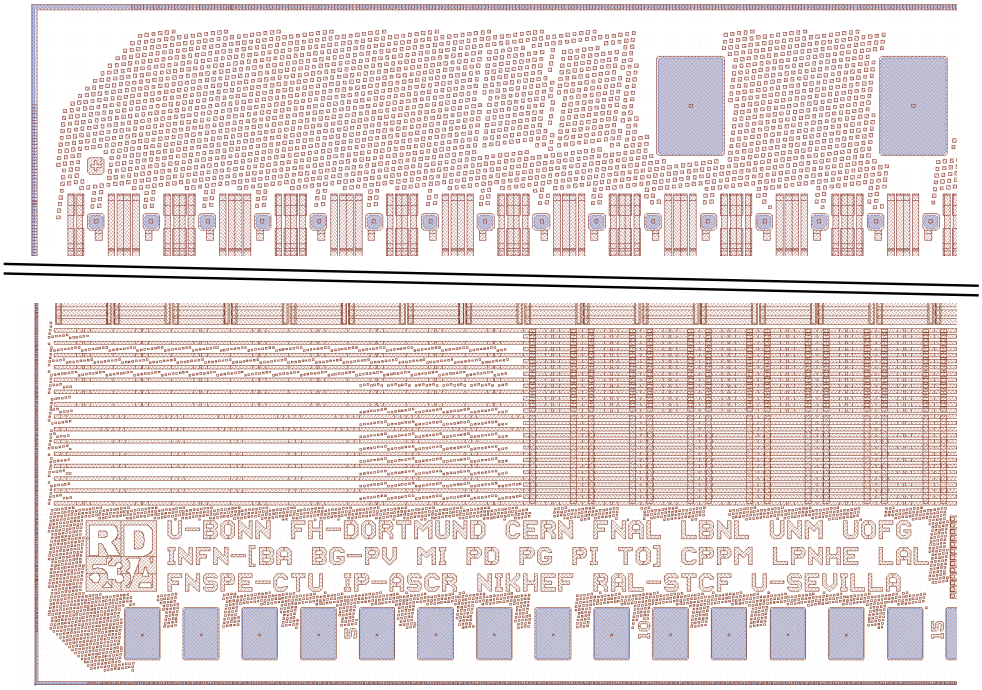


Figure 10: Top metal and passivation layout details of top left and bottom left corners of RD53A. The magnification is different for top and bottom: the wire bond pads are all the same size, but the pitch is 200 μm on at the top of the chip and 100 μm at the bottom. In the top left an alignment cross is also visible which is placed 50 μm above the top left bump pad. Metal fill (small peppered squares) is also visible.

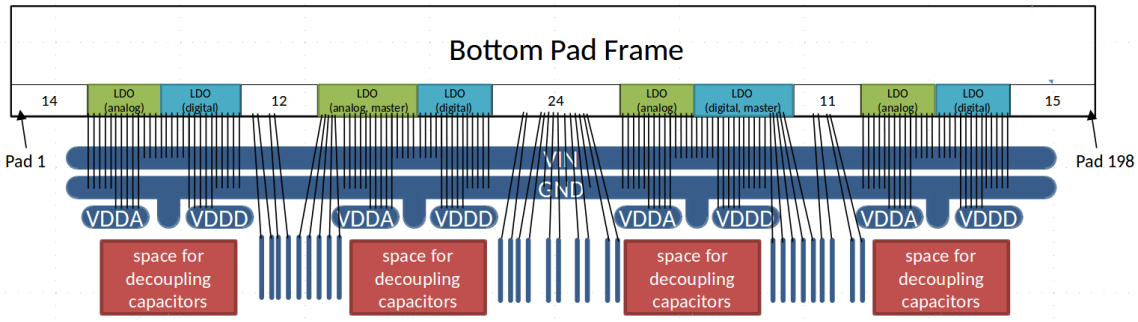


Figure 11: Organization of bottom of chip pad frame. Typical wire bond connections needed for realistic detector module operation are also shown. The number of pads in each group is indicated.

4.1 Sensor Guard/Bias Ring Bumps

Special bumps for access to sensor guard rings or bias grids will be included at the bottom of the regular bump matrix. These will be connected by metal, without antenna diodes, to dedicated wire bond pads. Note these are not rated to carry high voltage, but expected to be at near ground reference potentials. Four such special bumps at each end will be included as shown in Fig. 13. On each side of the chip, the group of four bumps is connected together to the first/last wire bond pad

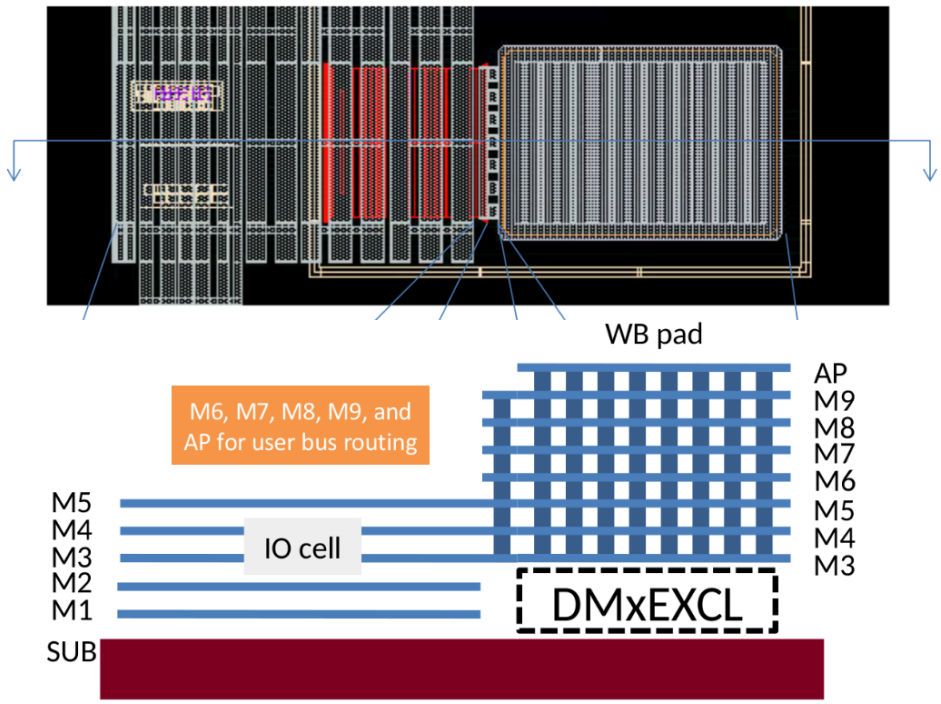


Figure 12: Detail of wire bond pad, ESD structures, and metal stack.

(see Table 12).

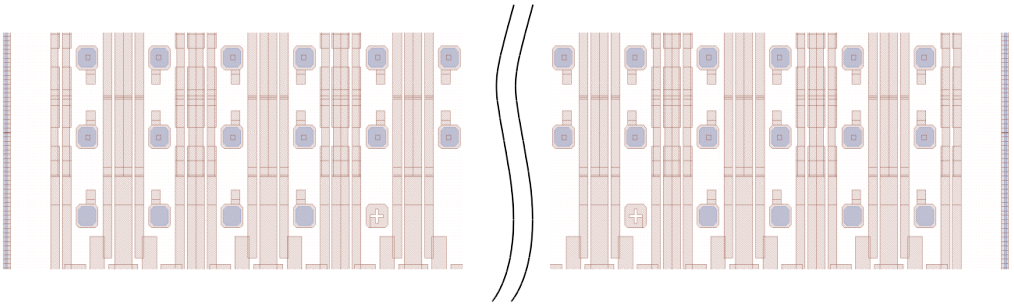


Figure 13: Location of guard ring / bias grid bumps. The placement is equivalent to an extra bottom row, but only with the first 4 and last 4 columns populated. In the fifth location from both ends is a miniature alignment mark made of floating metal covered with passivation.

5. Analog Front Ends

RD53A contains three different front end designs to allow detailed performance comparisons. These are not small variants of one common design, but substantially different ground-up designs. They are identified as Differential, Linear and Synchronous. The Differential front end uses a differential gain stage in front of the discriminator and implements a threshold by unbalancing the two branches. The Linear front end implements a linear pulse amplification in front of the discriminator, which compares the pulse to a threshold voltage. The Synchronous front end uses a baseline "auto-zeroing" scheme that requires periodic acquisition of a baseline instead of pixel-by-pixel threshold trimming. The designs are individually described in Sec. 5.1, 5.2, and 5.3, respectively. They share some common constraints and features, the main being the layout area and the bump bond pads, which are the same for all designs, making them easily interchangeable on the pixel matrix layout. The analog *quad* layout area is 70 μm by 70 μm, containing four front ends and four bump pads on a 50 μm by 50 μm grid. Also common to all front ends is the calibration injection circuit (Sec. 5.4), as this is important for direct performance comparisons. The bias distribution follows the same organization for all 3 flavors as described in Sec. 5.5.

The front end designs share the pixel matrix area as shown in Fig. 14. It was not possible to have equal area for all designs because the 400-pixel wide matrix is built of 8×8 pixel synthesized *cores*. Therefore two designs must have a 17 core width while one must have a 16 core width. The 16 core width was assigned to the Sync. FE flavor because it is simulated to use slightly more power. The Diff. and Lin. FE designs were placed next to each other because they have the most similar functionality, allowing for the largest possible area with uniform response, as desired for sensor characterization in test beams. They also are read out by the same digital region architecture, whereas the Sync. design implements a different architecture, as indicated in Fig. 14. The output data format is identical for both readout architectures (See Sec. 6 for more details). The Diff. FE and Distributed Buffer (DB) region architecture have been prototyped in the FE65-P2 64×64 pixel demonstrator, while the Lin. and Sync. FE and Central Buffer (CB) architecture have been prototyped in the Chipix 64×64 pixel demonstrator.

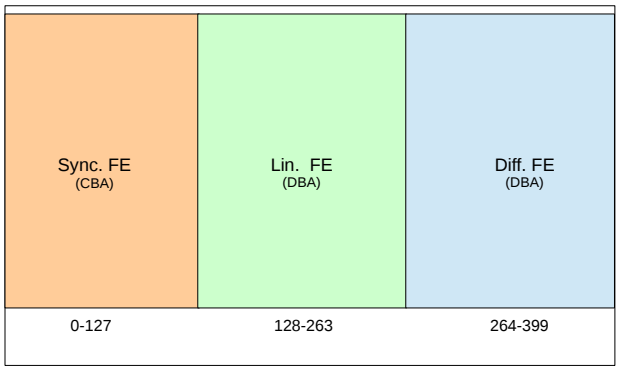


Figure 14: Arrangement of front end flavors in RD53A. The pixel column number range of each flavor is shown along the bottom. The digital readout architecture used is also indicated: Central Buffer Architecture (CBA) or Distributed Buffer Architecture (DBA).

5.1 Differential Front End Design

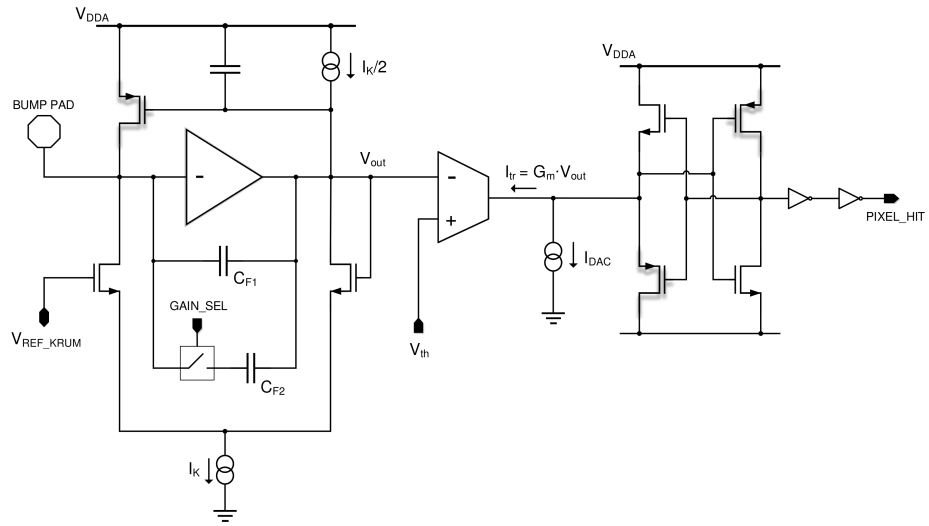


Figure 15: Schematic of the Linear analog front end flavor

The Linear front-end is shown in figure 15. The readout chain includes a charge sensitive amplifier (CSA) featuring a Krummenacher feedback complying with the expected large radiation induced increase in the detector leakage current. The choice of a single amplification stage in the front-end channel has been simply dictated by power consumption and area constraints. The signal from the CSA is fed to a high-speed, low power current comparator that, combined with the time-over-threshold (ToT) counter, is exploited for time-to-digital conversion. Channel to channel dispersion of the threshold voltage is addressed by means of a local circuit for threshold adjustment, based on a 4-bit binary weighted DAC generating the current I_{DAC}. The front-end chain has been optimized for a maximum input charge equal to 30000 electrons and features an overall current consumption close to 4 μA. The CSA can be operated in high gain (floating C_{F2}) and low gain mode (C_{F1} in parallel with C_{F2}) by acting on the GAIN_SEL bit, whereas the recovery current, I_K/2, in the Krummenacher feedback network, can be set by means of a peripheral DAC. I_K current equal to 25 nA results in a ToT close to 400 ns for an input charge equal to 30000 electrons in high gain configuration. The designed sensitivity in the high gain configuration is 15 mV/ke- , and 7.5 mV/ke- in the low gain mode. The core element of the charge sensitive amplifier is the gain stage shown in figure 16. This is a folded cascode architecture including two local feedback networks, composed by the M4-M5 and M7-M8 pairs, boosting the signal resistance seen at the output node. With a current flowing in the input branch equal to 3 μA and a current in the cascode branch close to 200 nA, the CSA is responsible for most of the power consumption in the analog front-end. The DC gain and the -3dB cutoff frequency of the open loop response, as obtained from simulations, are 76 dB and 140 kHz respectively. The noise performance of the charge preamplifier is mainly determined by the contributions from the CSA input device and from the PMOS transistor part of the feedback network. The simulated equivalent noise charge, for a detector capacitance of 50 fF, is equal to 87 electrons. The front-end channel includes a high-speed,

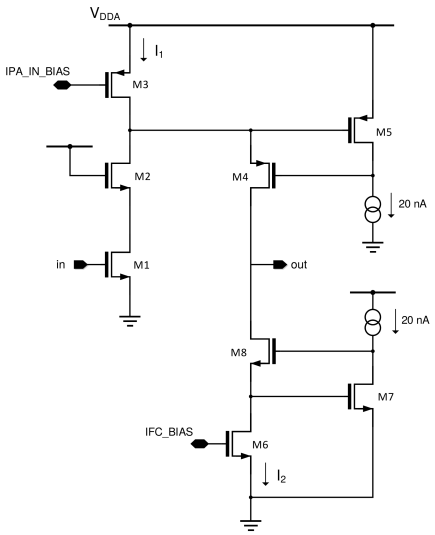


Figure 16: Linear front end charge sensitive amplifier forward gain stage.

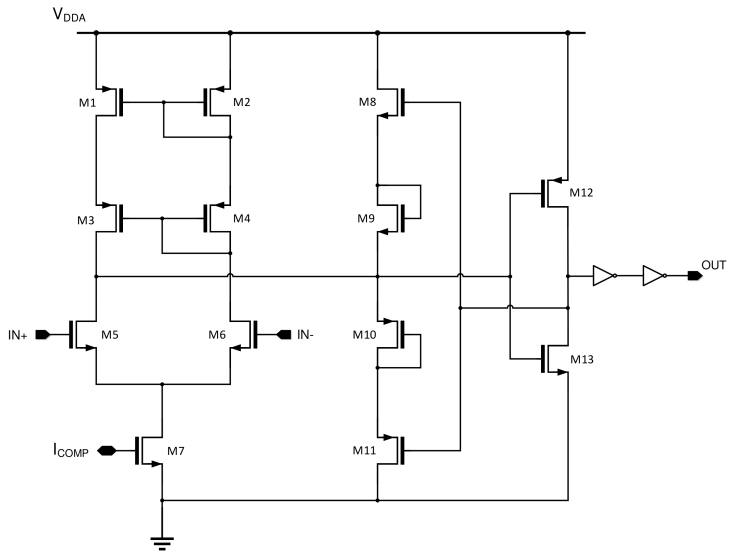


Figure 17: Linear front end threshold discriminator schematic diagram.

low power threshold discriminator, based on current comparison, connected at the preamplifier output. Shown in figure 17, it includes a transconductance stage whose output current is fed to the input of a transimpedance amplifier providing a low impedance path for fast switching. A couple of inverters is used after the transimpedance amplifier in order to consolidate the logic levels. The performance of the analog front-end in terms of threshold dispersion are greatly improved by the use of a 4 bit, in-pixel binary weighted DAC providing local threshold trimming. From circuit simulations, threshold dispersion before correction is equal to 380 electrons, reduced down to 35 electrons after tuning.

5.2 Linear Front End Design

The Differential front end is a pure analog circuit: it contains no memory latches, flip-flops or counters. Static configuration values are provided by the digital core, which receives only the comparator out signal from the analog part. The design is a small-area low-power free-running front-end, for negative input charge. The ADC function is implemented entirely in the digital core, by digitizing the time-over-threshold (ToT) of the comparator pulse. Fig. 18 shows the Diff. FE block diagram.

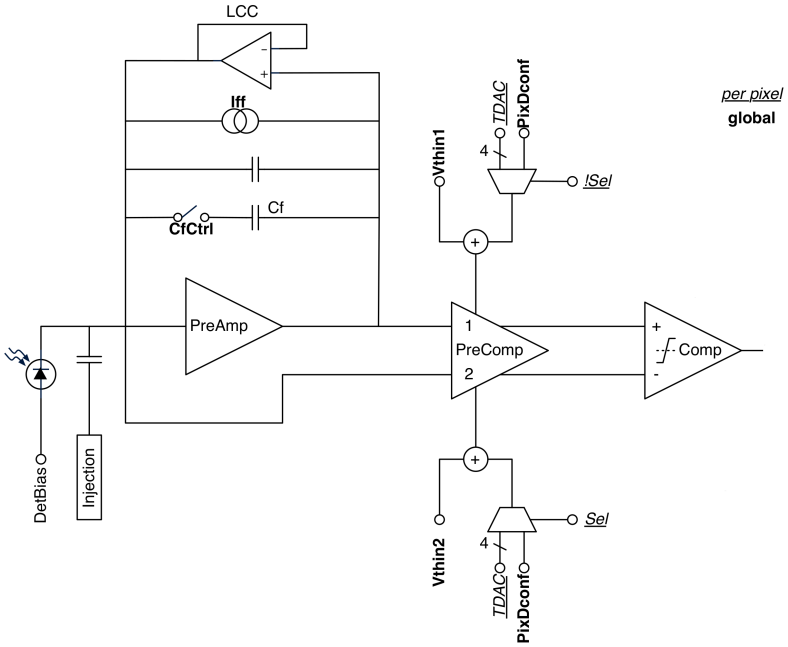


Figure 18: Schematic of Differential front end flavor.

The pre-amplifier or first stage (Fig. 19) has a simple straight regulated cascode architecture with NMOS input transistor in weak inversion. It has a continuous reset and adjustable gain by choosing between two possible values of feedback capacitance (this choice is made globally, not per pixel). The preamp can operate at very low currents and has three bias currents: the main preamp bias (input transistor current), voltage follower, and continuous feedback current. The feedback current is set globally and cannot be trimmed in each individual pixel. Prototype measurements showed that fall time dispersion (which leads to a ToT dispersion) has an acceptable level without any need for per-pixel trimming. The preamp is single ended, but the feedback ensures that, in the absence of signal, input and output are at the same potential. Input and output are thus taken as a differential input to the next stage.

The DC-coupled pre-comparator or second stage (Fig. 20) provides additional gain in front of the comparator and also acts as a differential threshold circuit. The global threshold is adjustable through two distributed voltages (VTH1 and VTH2) which introduce an offset between the two branches of the pre-comparator. The threshold is trimmed in each pixel using one 4-bit resistor ladder in each pre-comparator branch. The branch current is turned into additional voltage offset by these resistor ladders. An effective 5-bit adjustment is obtained by adjusting one branch resistance

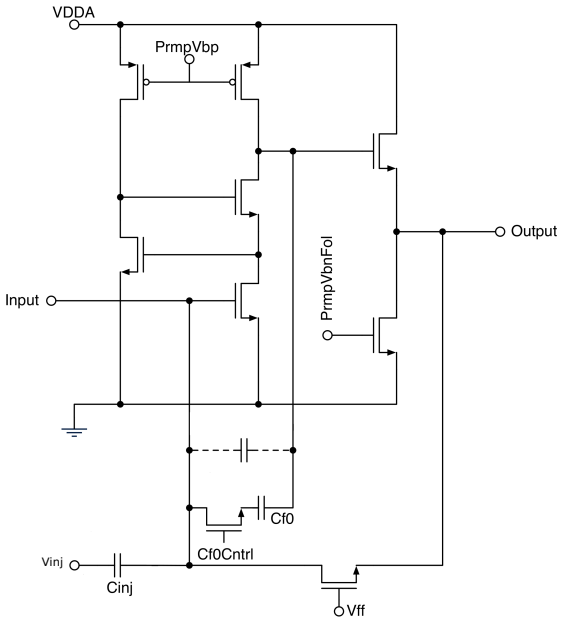


Figure 19: LBNL First stage (preamp) schematic.

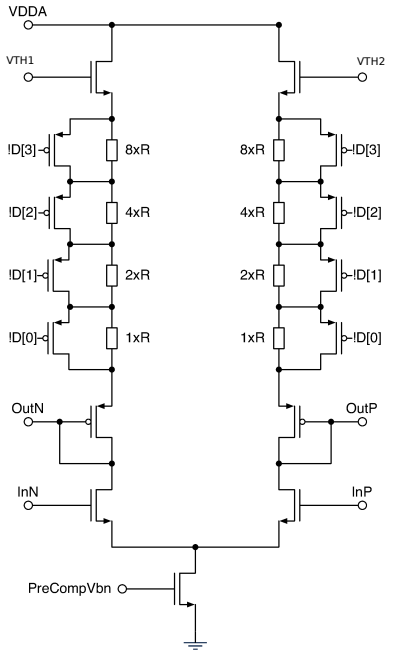


Figure 20: LBNL Second stage (pre-comparator) schematic.

or the other using a single 4-bit value. The 5^{th} bit is thus a select bit, which determines which branch is adjusted. The branch that is not adjusted is set to all 1 or all 0, depending on a global configuration value. This design is optimized for low-threshold operation. The pseudo-differential design reduces variation due to mismatch and provides improved power supply rejection. The pre-comparator stage is followed by a classic continuous time comparator stage with output connected to the digital pixel region through logic gates. Designed for 500e$^-$ threshold operation, the analog pixel has a current consumption of 4 μA/pixel at 50 fF detector load and 10 nA leakage current, but can operate up to 100 fF load and 20 nA leakage, resulting in higher threshold and higher power. It can also meet performance specifications with 3 μA/pixel current consumption and 50 fF load. For leakage currents below 10 nA, the continuous feedback is capable of preventing the input from saturating. For leakage of order 10 nA or higher, a the leakage current compensation circuit shown in Fig. 18 is required to prevent saturation. This circuit can be globally enabled or disabled. When disabled, the circuit is disconnected from the input, which improves phase margin and noise. It is expected that the Diff. FE will be able to operate without the leakage current compensation circuit for planar sensors even after significant radiation damage.

5.3 Synchronous Front End Design

A block diagram of the Synchronous front end is shown in Fig. 21. It features a single stage Charge Sensitive Amplifier (CSA) with a Krummenacher feedback AC coupled to a synchronous discriminator composed of a Differential Amplifier (DA) and a positive feedback latch.

The CSA (Fig. 22) contains a telescopic cascode stage with current splitting to minimize noise contributions and a source follower to improve the driving strength. The Krummenacher feedback

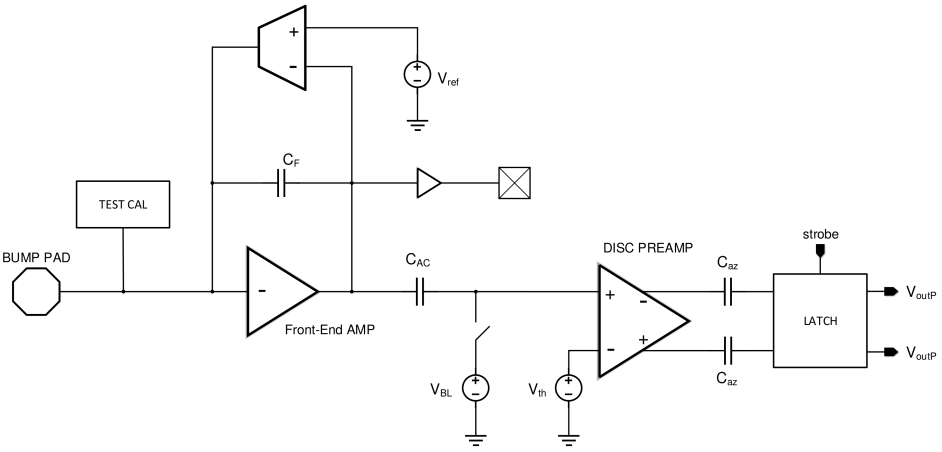


Figure 21: Schematic of the Synchronous front end flavor.

(Fig. 23) is designed to provide both the sensor leakage current compensation and the constant current discharge of the feedback capacitor. The larger the current the faster the preamplifier signal returns to the baseline. As a reference, a 10 nA current results in a 400 ns long signal for an input charge of 10ke-, which is reduced to around 100 ns for a 40 nA current. Two capacitors, equal to 2.5 and 4 fF, respectively, have been included in order to implement different gain values. The calibration injection circuit is described in Sec. 5.4. The preamplifier open loop gain is around 60 dB.

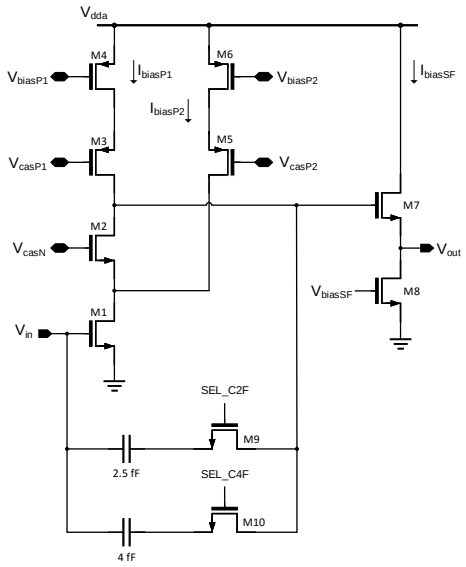


Figure 22: Sync. FE Charge Sensitive Amplifier (CSA) schematic.

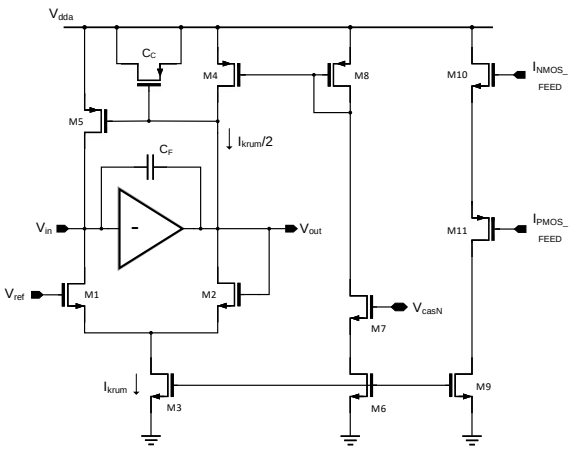


Figure 23: Sync. FE Krummenacher feedback schematic.

Due to mismatch effects, considerably relevant in deep submicron technologies like 65nm, the output baseline of the first stage is subject to quite large fluctuations (of the order of tens

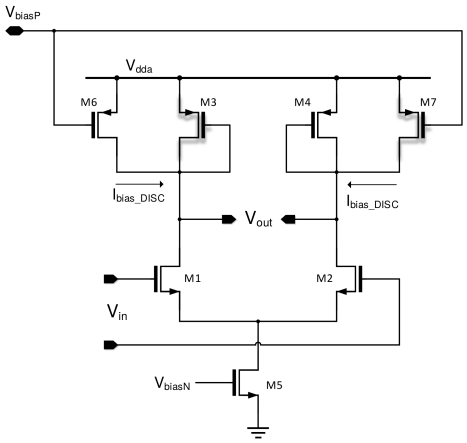


Figure 24: Sync. FE discriminator schematic.

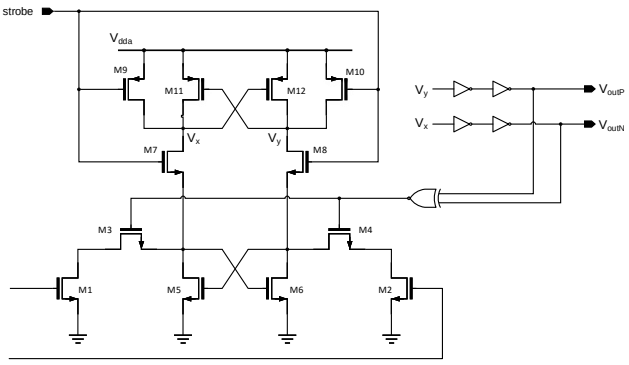


Figure 25: Sync. FE latch schematic.

of mV) between different channels. In order be immune to such differences an AC coupling to the discriminator (DA) has been implemented. The DA (Fig. 24) provides a further small gain (around 2). However, transistor mismatch results in an offset of the DA output voltage between pixels, that has traditionally been compensated with trim DACs (as for the other front end flavors). Instead, the Sync. FE compensates offsets using internal capacitors to implement an "auto-zeroing" mechanism. This requires acquisition of a baseline every 100 μs or less. The baseline acquisition takes 100 ns and is intended to take place during LHC abort gaps. During collisions, the difference signal (signal minus stored baseline) is fed into the positive feedback latch stage (Fig. 25), which performs the comparison and generates the discriminator output. This stage has been designed to minimize mismatch effects causing a dynamic offset resulting in an additional threshold dispersion. Furthermore, the latch can be turned into a local oscillator up to 800 MHz using an asynchronous logic feedback loop, and so used to perform a fast Time-over-Threshold (ToT) counting.

5.4 Calibration Injection Circuit

The calibration injection circuit uses two distributed DC voltages plus in-pixel switches to chop them and generate steps fed to an injection capacitor. Having two voltages allows a precision differential voltage that will be independent of local ground drops in the chip, as well as two consecutive injections into the same pixel. The circuit topology is shown in Fig. 26. Every pixel (regardless of front end flavor) contains the same circuit. The control signals, S0 and S1, are generated in the digital domain as explained in Sec. 5.4.1 and can be phase shifted from bunch crossing clock with a fine delay, which is global for the whole chip. The enable bit is programmable for each pixel. The cascaded "inverter" configuration makes for simple control without the need to synchronize non-overlapping edges. Injection takes place only for cal enabled pixels when either S0 or S1 switch from low to high. Analog injection must therefore be primed by setting at least one control signal low, prior to being able to inject. This priming is not automatic, so that the user is able to control the amount of settling time allowed prior to injection. The CAL command is used for both functions: prime and inject (see Sec. 9.2).

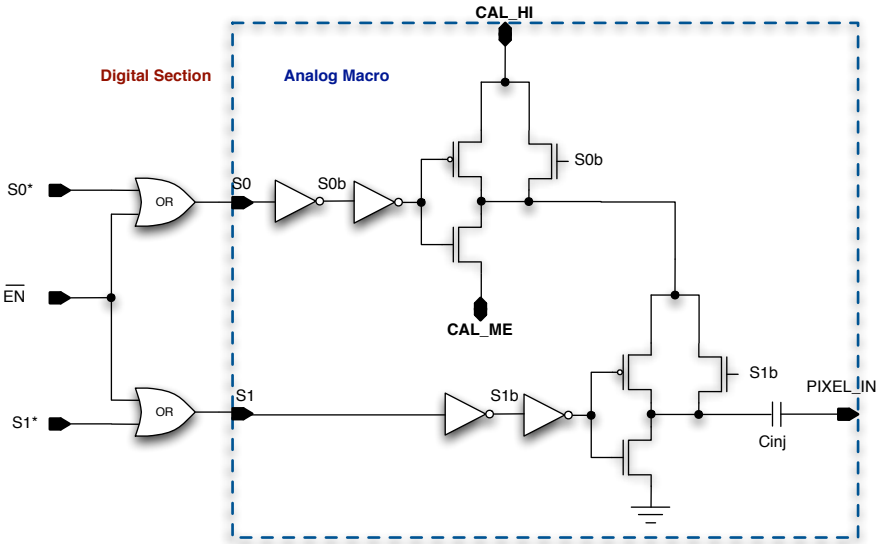


Figure 26: Calibration injection circuit in each pixel.

Just as in a common inverter, there will be a switching transient when a control signal switches from low to high. Simulations show these transients to have a negligible impact on the distributed CAL_HI and CAL_MED voltages. Note that at the top of each inverter there is an NMOS transistor in parallel with the PMOS, which switches first (before the PMOS) when injecting. This allows the switches to operate for any choice of voltages CAL_HI>CAL_MED>GND, but since the top NMOS switches first, it does not contribute transients during injection. During priming, on the other hand, the top NOMS switches first, while the bottom NMOS is still conducting, resulting in a short circuit lasting one inverter delay. This will cause a transient on the CAL_HI and CAL_MED voltages, and the user must therefore allow some settling time between priming and injection. In addition to this transient, priming injects a positive polarity pulse into each enabled front end, so one must allow for the front end to settle in any case. The two-voltage injection circuit allows injection of two successive pulses without priming in between, and with arbitrary delay between these pulses. This is a new feature not found in previous chips. The two voltages also mean that the charge injected by S0 is given by a differential voltage and not affected by local ground potential differences. Leaving S1 set and only toggling S0 will result in single pulse differential injection. The two voltage system also makes it possible to inject different amount of charge simultaneously in neighboring pixels by changing the meaning of S0 and S1 in different pixels (see Sec. 5.4.1.

5.4.1 Generation of S0 and S1 signals

The signals S0 and S1 of Sec.5.4 exist locally in each pixel but are derived from different internal signals produced by the command decoder and distributed to the array. This two-step scheme is necessary in order to have either sequential or simultaneous injection, and more importantly to avoid having to distribute two switching signals with precise timing. Since the calibration input is used to study and calibrate timing, it must occur simultaneously in all enabled pixels, just as is the case for the bunch crossing clock (here, simultaneously means within a 2 ns window). Two control

signals are distributed: CAL_edge and CAL_aux. As the name implies, CAL_edge needs to be simultaneous in all pixels, while CAL_aux does not. CAL_edge has a fine phase adjustment relative to the beam crossing clock, which is called CAL_delay. In fact CAL_edge looks like a traditional injection pulse, with user controlled leading edge time and duration. In *uniform* injection mode (which allows injecting two pulses close in time into all selected pixels), S0 and S1 are derived from CAL_edge and CAL_aux identically for all pixels:

$$\mathrm{S0} = \mathrm{CAL\_edge}\ \ OR\ \ \mathrm{CAL\_aux} \tag{5.1}$$

$$\mathrm{S1} = \overline{\mathrm{CAL\_edge}}\ \ AND\ \ \mathrm{CAL\_aux} \tag{5.2}$$

The rising edge of CAL_edge throws the S0 switch, while the falling edge throws the S1 switch. The CAL_aux starts low and then goes high after CAL_edge, but not with precise timing. In uniform mode the injection switches can only be thrown in that order. Either only use S0 for single pulse, differential voltage injection, of use S0, then S1, for double pulse injection.

In order to allow injection of different size pulses simultaneously into adjacent pixels, there is an *alternating* injection mode that can be selected instead of the default uniform mode. In this mode the S0 and S1 signals are derived as above only for *even* pixels, but swapped for *odd* pixels:

$$\mathrm{S1}_{odd} = \mathrm{CAL\_edge}\ \ OR\ \ \mathrm{CAL\_aux} \tag{5.3}$$

$$\mathrm{S0}_{odd} = \overline{\mathrm{CAL\_edge}}\ \ AND\ \ \mathrm{CAL\_aux} \tag{5.4}$$

where an even (odd) pixel is one for which the sum of row + column is an even (odd) number. Thus, for example, in single injection mode the CAL_edge rising edge throws S0 for even pixels, but S1 for odd pixels. The S0 and S1 assignment options are independent of the cal enable bit in each pixel.

5.5 Analog Bias

The Front-end bias circuitry is organized in a 2-stage structure, as shown in Fig. 27: the global voltage/currents are globally provided by a set of 10-bit current steering Digital-to-Analog Converters housed in the ACB macroblock. After some current scaling to obtain the requested range and resolution, the biases are distributed to the so-called MacroColumnBias cells, where the final bias/voltage is produce and distributed in parallel to the 384 pixels (768 in the production chip) of two adjacent columns, called Macrocolumn. The MacroColumnBias cell is housed just below the first pixel row and is replicated 200 times. The main advantage of this 2-stage scheme is that a failure in a pixel involving the bias lines will affect only 2 columns, thus improving the overall reliability of the chip. Moreover, the degradation caused by the expected increase of leaking current with radiation is much more limited respect to the scheme where the full matrix is biased in parallel. The main drawback of this concept is the possible increase of bias mismatch among pixels, but since the first stage of current mirroring (between Global Bias and Macrocolumn Bias) can be realized with large transistors, Monte Carlo simulations show that this worsening is negligible. The reference current for the DACs is generated using a Bandgap Reference Voltage circuit and can be finely tuned using a 4-bit DAC to compensate for process variations. The optimal value of this 4-bit DAC will be found during the chip characterization and hard-wired with bonding pads. It's also worthwhile to mention that all the DACs currents can be read back using the 12-bit Monitoring ADC.

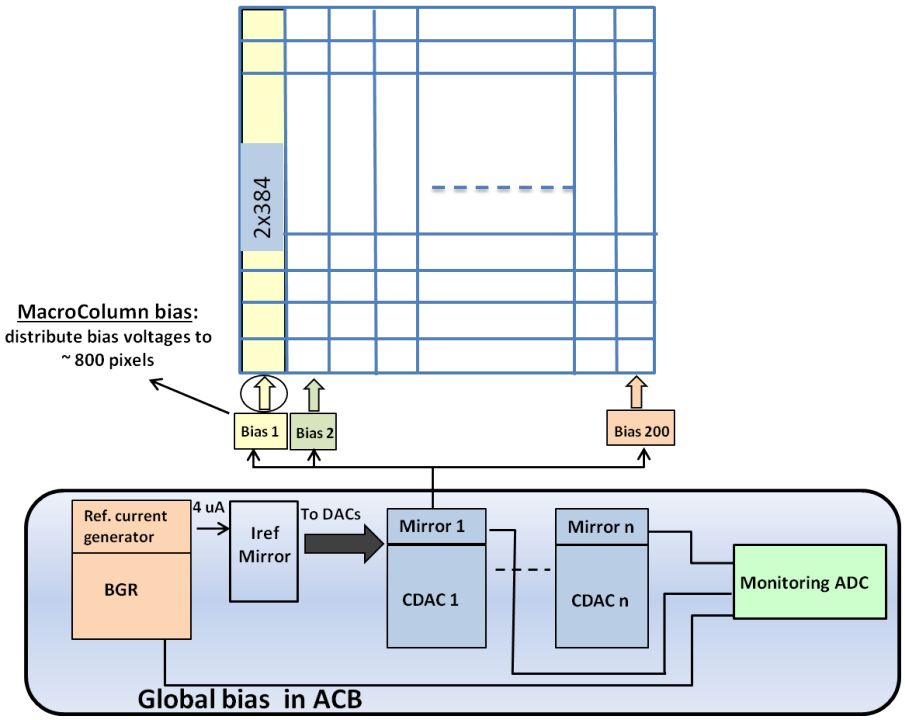


Figure 27: Block diagram of front-end bias scheme.

6. Digital Matrix

The pixel matrix is built up of digital *cores* with 8 by 8 pixel channels each, contained in 4 by 4 analog islands. The core is synthesized as one digital circuit. It provides static configuration bits to the analog islands, and receives four binary outputs from each island. The digital core handles all processing of the binary outputs, including masking, digital injection, ToT counting, storage of ToT values, latency timing, triggering and readout.

Each core receives all input signal from the previous core closer to the chip bottom, and regenerates the signals for the next core. The timing critical clock and injection edge signals are internally delayed within the core relative to the regenerated ones so that when all the cores are integrated there is uniform timing (within 2 ns) of those signals within all cores. To achieve this uniformity there are multiple core delay values used in RD53A, with the largest (smallest) delay used in the top (bottom) third of the matrix. Other than this delay value, all cores of the same flavor are identical.

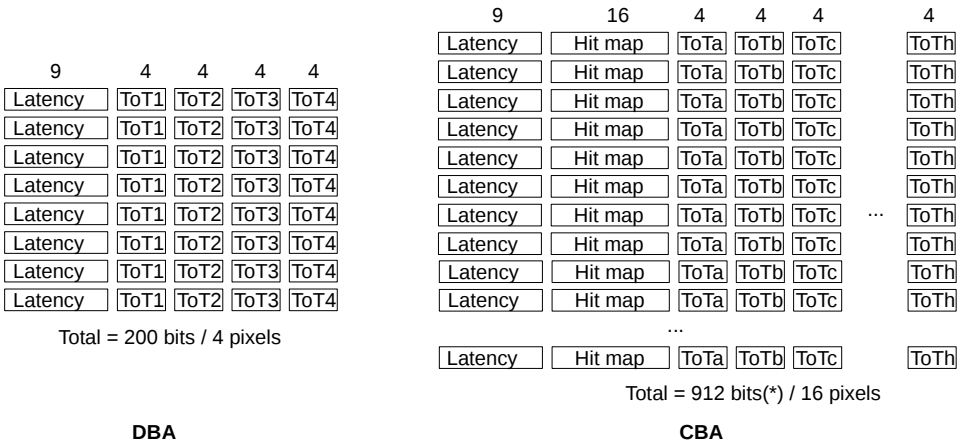


Figure 28: Diagram of memory maps for DBA (left) and CBA (right) regions.(*)The number of ToT storage locations per latency timer for the CBA has been maximized to fill the available layout space and does not reflect the minimum number needed for physics.

There are two core flavors implementing two readout architectures, called DBA (Distributed Buffer Architecture) and CBA (Central Buffer Architecture) covering the column ranges indicated in Fig. 14. In both cases the basic unit is called a *region*, which can be thought of as a macropixel with internal structure. The DBA (CBA) region size is 4 (16) pixels. In both cases, timing information (when a hit occurred) is stored by region, not by pixel. In the DBA region ToT information is stored by pixel, while in the CBA region there is a common region memory where ToT information is stored. In this way, the CBA suppresses zero ToT values, but at the cost of having to store a hit map, while the DBA "wastes" memory storing zero ToT values, but needs no hit map. The DBA is efficient as long as multiple pixels are typically hit (high region occupancy- few zero ToT values), which is the case for a small region. The CBA architecture is efficient as long as region occupancy is low (most pixels hit is very rare), which is the case for larger regions. The memory usage of

both architectures is illustrated in Fig. 28, which is further explained if Sections 6.1 and 6.2. The total storage in the DBA region is 50 bits per pixel, while the RD53A CBA region has 57 bits per pixel, because the amount of ToT storage has been maximized to fill available space. The different architectures have a negligible impact on the digital chip bottom design, because the output data formats have been kept identical. The advantage of one of another in terms of region layout area and resources depends on the size of the latency timers and the ToT registers. For 9 bit timers and 4 bit ToT they are similar. A 2 bit ToT would clearly favor DBA, while a 6 bit ToT would clearly favor CBA.

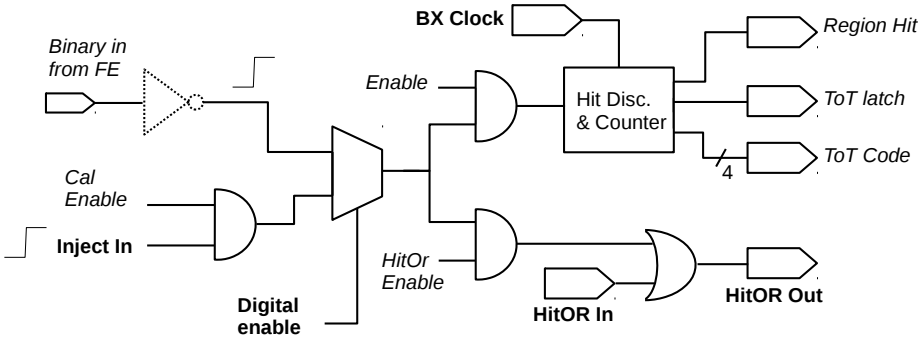


Figure 29: Digital processing chain for a single binary output from an analog island.The bold text indicates global signals, while the italic text indicates local pixel signals.The dotted line inverter is only present for the Diff. FE.

6.1 Distributed Buffer Architecture (DBA)

Each core is logically divided into 16 4-pixel regions, which have identical functionality, but differ in address and are not identical in terms of layout because they are all synthesized together in one flat layout (dubbed digital sea). Each region spans 4 columns and one row and therefore covers an effective area of $50\,\mu$m $\times$ $200\,\mu$m. In terms of a sensor this means 1×4 50x50 pixels or 2×2 25x100 pixels. Because the beam spot is extended in Z, clusters in the detector barrel have a rectangular rather than square shape and are more efficiently recorded with $50\,\mu$m $\times$ $200\,\mu$m regions than they would be with $100\,\mu$m $\times$ $100\,\mu$m regions. Note that there is no one-to-one correspondence between 4-pixel analog islands and 4-pixel digital regions.

Each individual pixel output is processed as shown in Fig. 29 The pixel output feeds two data paths: the triggered DAQ path (Sec. **??**), and the prompt path, also called Hit OR, where a hard-wired OR of many pixel outputs is made available in real time to output pads and/or bottom of chip processing. These outputs can be independently masked for each pixel, via the Enable and HitOR pixel configuration bits (see Sec. 6.3).

The Hit Discriminator and Counter block in Fig. 29 is responsible for determining if an in-time hit is present and what is the ToT value to be stored. The block does *not* contain a method to separate "small hits" from "large hits" as in the FE-I4 chip. All hits fire the region hit output, which in turn starts a latency counter for the whole pixel region. If multiple pixels fire in the same

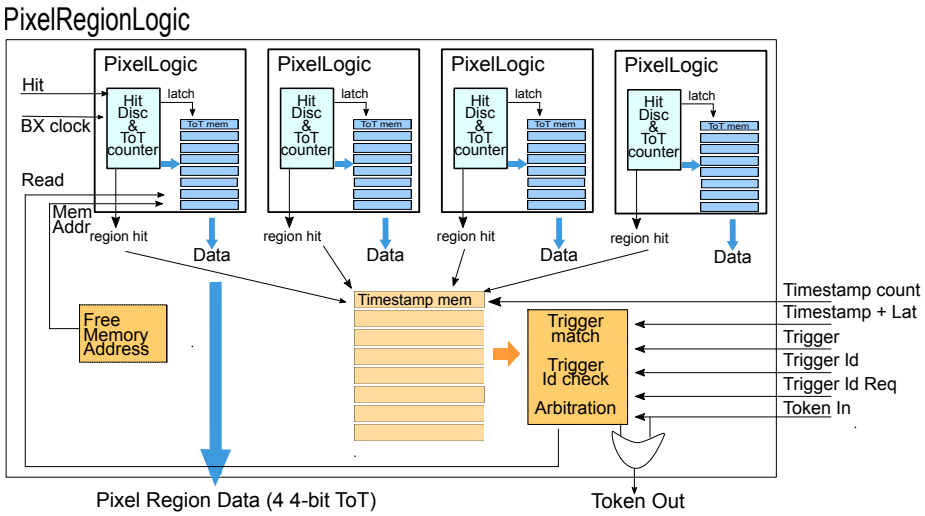


Figure 30: 4-Pixel region with 8 shared latency timers and 8 ToT memories (4 bits each) per pixel.

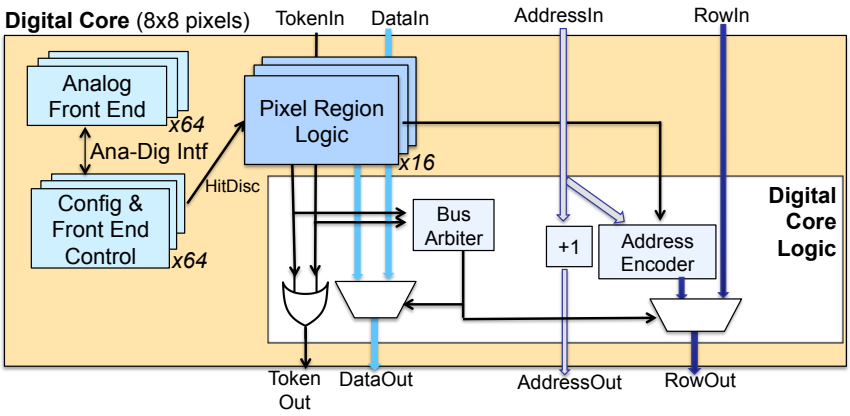


Figure 31: Block diagram of digital core with multiple regions.

region in the same crossing, still only one latency timer starts. ToT is counted independently for each pixel using the 40 MHz clock. If the counter reaches maximum while the pixel comparator output is still high the counting concludes and the maximum ToT is recorded (Any ToT value 15 or greater is encoded as ToT code 0xF).

Each 4-pixel region has 8 latency timers and each pixel has 8 ToT storage registers, 4 bits each (Figs. 28, 30). The latency timers are common to the whole region, but the ToT storage is at the single pixel level- hence the name Distributed Buffer Architecture. Each of the 8 ToT registers is associated with one latency timer (hard-wired). This way when one pixel is counting ToT it does not prevent the other pixels in the region from being hit: no region dead time.

The latency timers are 9 bits wide, allowing for a programmed latency of up to 511 bunch crossings. The timers are idle until a hit arrives and with each new region hit (the OR of all 4 pixel hits) a new timer is started. Only one timer can start in any given bunch crossing regardless of how many pixels were hit. When a timer reaches the programmed latency and no trigger is present, it is reset and the associated ToT values erased. If a trigger is present, then the ToT values are not erased

and a trigger tag is stored to label the hits for later readout. Readout proceeds with column-parallel token chain, one trigger tag at-a-time. The organization of a core is shown in Fig. 31.

6.2 Central Buffer Architecture (CBA)

A CBA region is 4 by 4 pixels. It contains region latency timers as described in Sec. **??**, but it has 16 instead of 8, since more hits can occur in the larger region. The difference is in the ToT storage, which uses one shared buffer rather than individual buffers for each pixel. Since each latency timer serves 16 pixels, larger clusters can be stored more efficiently and fewer clusters will straddle a region boundary. Instead of storing the full ToT for each pixel, a region hit map (16 bits) is also stored with each timer. Then the 4-bit ToT value for up to 8 hits is stored in a shared buffer (see Fig. 28). A fixed priority queue is used to uniquely assign each stored ToT to a given hit pixel. If more than 8 pixels are hit (which is very unlikely) the ToT information for the additional pixels is lost, but the full hit map is always preserved. A diagram of the CBA organization is shown in Fig. 32.

The ToT zero suppression requires the ToT digitization to be be completed. While in CHIPIX this was achieved by using fixed dead-time counters in the pixels, in RD53A, an intermediate "Staging Buffer" has been added. The Staging Buffer saves a copy of the hit map and holds it until all pixels have completed ToT. A new hit arriving in the region during this time can be accepted and will create a new hit map. The Staging Buffer can host up to 4 events: if there are more than 4 hits in a period of time equal to the Write Synchronization Time, there is a hit loss.

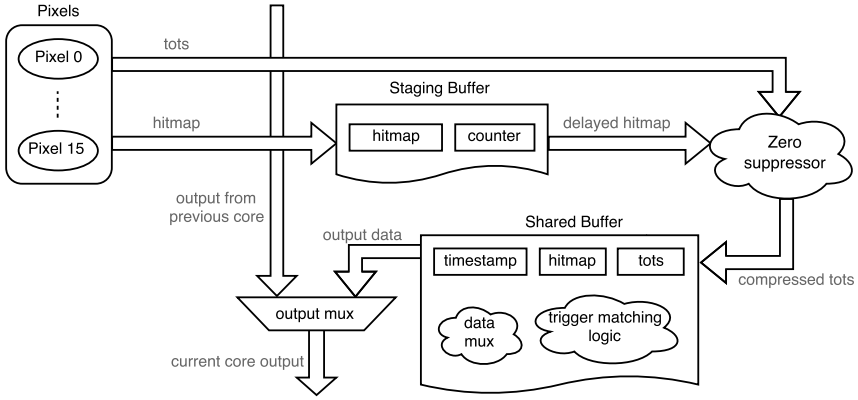


Figure 32: Block diagram of Central Buffer Architecture region logic.

Every pixel generates a "Hit Present" pulse when the output of the discriminator is high. A region-wide OR of these signals is used to trigger the Staging Buffer. All the Hit Presents are written in a row to create the hit map and a counter is started. When the counter reaches the Write Synchronization Time (configurable), it feeds the hit map to the ToT zero suppression stage. The Write Synchronization Time should therefore be fixed at the maximum expected number of 40MHz clocks required to compute the ToT in the pixels. Given that CBA features the SYNC FE, the minimum value of this configuration register depends on the frequency of the Fast clock for fast ToT computation. The hit map and the 8 zero suppressed ToT values are only written to the shared buffer of Fig. 28 once the Write Synchronization Time has elapsed.

The trigger logic that selects hits for readout is the same as for the DBA. A dedicated adapter stage along side the column read control at the chip periphery constructs DBA-like output packets and feeds them to the data concentrated in the digital chip bottom. This makes the data identical for both region architectures. The only difference in the DBA and CBA output data is in the meaning of ToT code 0xE. In CBA data this code is not a ToT value 14, but rather a flag indicating that some ToT were lost due to zero suppression. Any ToT value 14 or greater is encoded as ToT code 0xF.

6.3 Hit OR

In RD53A there are actually four independent Hit OR networks, each one fed by one quarter of the pixels. Fig. 33 shows graphically how the 64 pixels in one core are grouped into the 4 OR networks. The figure also indicates two possible sensor formats of 50 μm $\times$ 50 μm (50x50) or 25 μm $\times$ 100 μm (25x100) pixels. It can be seen that in the 50x50 case, a given pixel in network 1 has its two up-down neighbors on network 3, and its left-right neighbors on 2 and 4. Conversely, a given 25x100 pixel on network has has its left-right neighbors in network 3 and its up-down neighbors on 2 and 4.

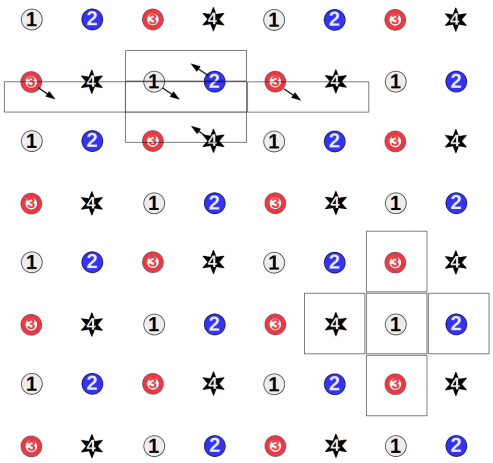


Figure 33: The four Hit Or networks in a 64 pixel core.

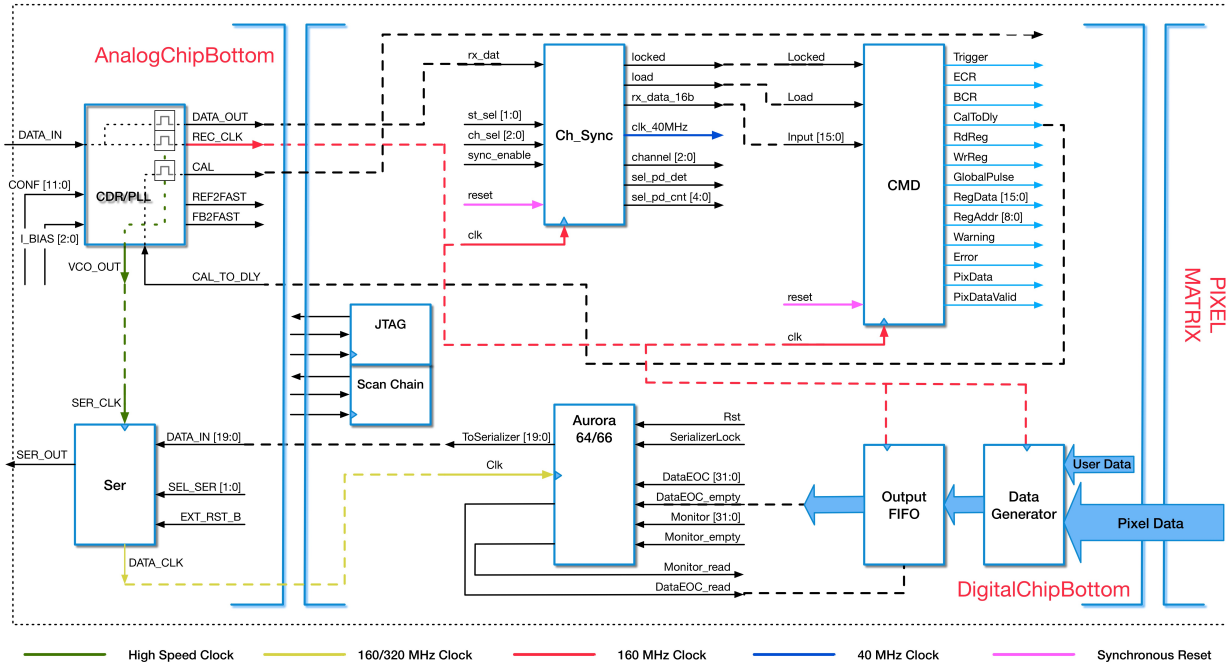


Figure 34: Block diagram of digital bottom of chip showing the command input and data output paths

7. Digital Bottom of Chip and Clocks

The Digital Chip Bottom (DCB) contains the blocks that implement all control and processing functionality. The entire digital bottom is high level description code synthesized together during top level integration. This includes SEU tolerant configuration memory, which uses a special synthesis flow described in Se. B. However, it is driven by clocks generated in custom layout blocks that are part of the Analog Chip Bottom (ACB), but nevertheless described in this section. Fig. 34 shows the relationship between the various blocks. The programming and I/O protocol details are given in Sec. 9, while chip start up and reset considerations are included in this section.

To provide failsafe functionality and to allow for special tests (such as response to clock jitter), the standard control path of RD53A can be fully bypassed. All bypass functions are controlled from wire bond pads, so there is no possibility they are accidentally enabled or disabled. The chip can be entirely operated without the serial input and clock and data recovery, or one can bypass individual elements in almost any combination.

7.1 Clocks and CDR/PLL

RD53A needs four clocks, most easily identified by their nominal frequencies. Three of them are normally generated by the Clock and Data Recovery (CDR) block (Fig. 35): the 160 MHz clock (CMD_CLK), the 1.28 GHz clock (SER_CLK) and the 640 MHz fine delay clock. If the CDR is bypassed by wire bonding the Bypass_CDR pin to VDD, the 160 MHz must be supplied externally, and will be used to generate the 1.28 GHz and the 640 MHz clocks. The 1.28 GHz clock can also be supplied externally independently of the Bypass_CDR state, and in this case this externally

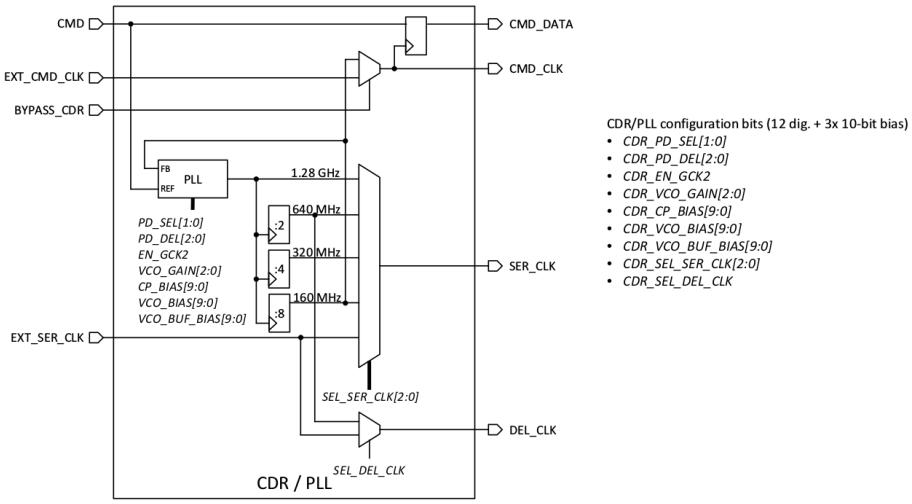


Figure 35: Diagram of the CDR (Clock and Data Recovery) block operation options

supplied clock will also be used as the fine delay clock. Selection an external clock source is done by a configuration register. These three clocks do not need to be synchronous nor have any particular ratio between their frequencies. Of course, when they are internally generated they will all be derived from a primary 1.28 GHz clock as shown in Fig. 35.

The CDR is is a custom analog block with an internal Voltage Controlled Oscillator (VCO) and a Phase Locked Loop (PLL) to lock to the incoming 160 Mbps control serial stream. The VCO produces primary 1.28 GHz clock locked to the phase of the command input stream transitions. This primary clock is divided down to produce the needed outputs. In the absence of a control input the CDR will still generate clocks with some frequency of order the nominal values, but arbitrary.

The 40 MHz clock (BX_CLK), is derived from the 160 MHz clock by the Channel Synchronizer (CS) block and it is the only clock distributed to the pixel matrix (Fig. 34). when dividing the 160 MHz one can choose 4 possible phase shifts for the 40 MHz clock. The CS picks the correct

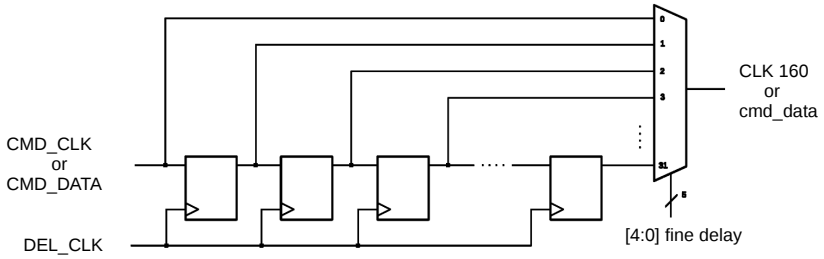


Figure 36: Schematic of clock and command phase adjustment, which permits synchronizing every chip's internal clock to the LHC bunch crossings. There is a selectable 2 inverter delay of DEL_CLK (not shown).

phase using Sync symbols in the command stream as explained in Sec. 9.2. However, before commands are sent to the chip, or if the command stream is interrupted, the the CS must still produce a 40 MHz clock to send to the pixel matrix, as this clock must never stop for proper startup and operation of the matrix. Therefore, upon start up, the CS produces a 40 MHz clock with arbitrary phase and later on switches to the correct phase without ever stopping.

In operation, the 40 MHz BX_CLK must be synchronized with the accelerator bunch crossings. For this purpose there is a programmable phase delay that can shift the BX_CLK phase in nominally 1.6 ns steps. This delay is not applied directly to the 40 MHz, but rather to both the 160 MHz clock and the serial command stream, as shown in Fig. 36. Thus it has the same effect as adding a cable delay to the clock/command serial stream. To guard against the possibility of a race condition there is a 2-inverter delay (2INV) that can be selected via configuration for the phase shifter clock.

7.2 Start Up and Reset

Upon start up several actions an initializations must be performed. Some are carried out automatically by a Power On Reset (POR) while others use set/reset commands. For commands to work the chip input path must first initialize and lock, which happens automatically with a supplied input command serial stream, or this default control path must be bypassed as explained in Sec. 7.3. The clock and data recovery lock signal is available on the STATUS output wire bond pad (Table 20): this can be monitored for debugging purposes or could for example drive an LED on a test board. Different parts of the chip follow different reset procedures. Most of the chip logic is in the data path, both within the pixel matrix and in the digital bottom of chip. All this logic is reset with a command called ECR (see Sec. 9.2). There is a dedicated reset for the JTAG interface, which remains in reset if not used, driven by the DEBUG_EN wire bond pad (Table 20).

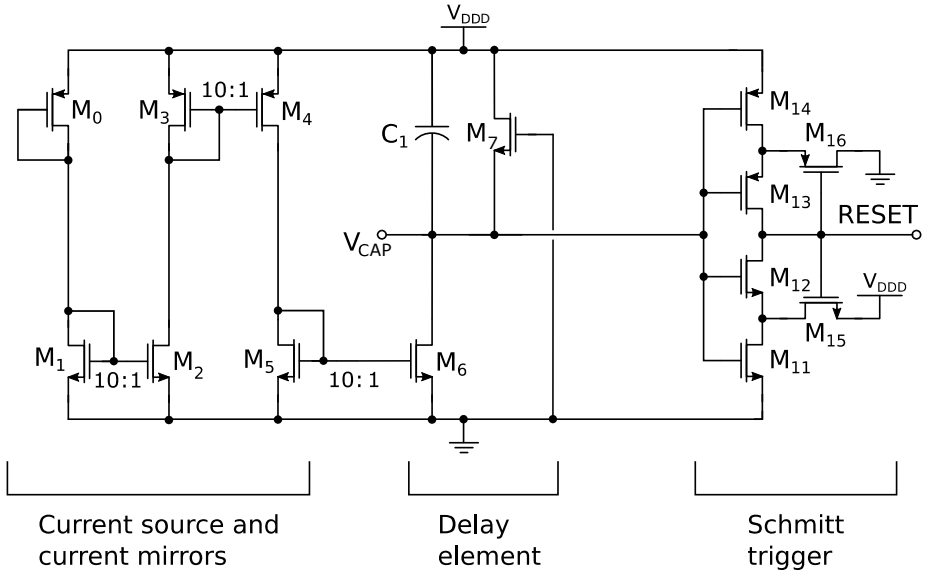


Figure 37: Power-on reset schematic. Capacitor C1 must be loaded externally with a nominal value of 5 nF. The POR signal, which is active low, is labeled RESET.

RD53A contains an active low POR generator (low is reset, high is not reset). The main function of the POR is to ensure that the chip has a reasonable configuration immediately after startup and stored logic states are well defined. The POR schematic is shown in Fig. 37. This circuit is itself powered from the digital power internal rail, which is the output, not the input, of the digital voltage regulator. The voltage regulators themselves do not have a reset: they are purely analog circuits. The POR begins with the reset condition asserted (POR output low). A Schmitt trigger fires when VDD exceeds 0.6 V, and 10 ms later the reset condition is removed (POR output goes high). The 10 ms delay assumes a 5 nF value external capacitor to VDDD has been loaded. The delay is linear with the capacitor value. When power is applied it should ramped up faster than this delay. There is no brown-out detection feature in RD53A. The POR output is available on a wire bond pad (Table 12) and can be forced externally if necessary, as shown in Fig. 38. An internal resistor is used to ensure a high output impedance of the POR circuit so that it can be safely forced form the bonding pad if needed.

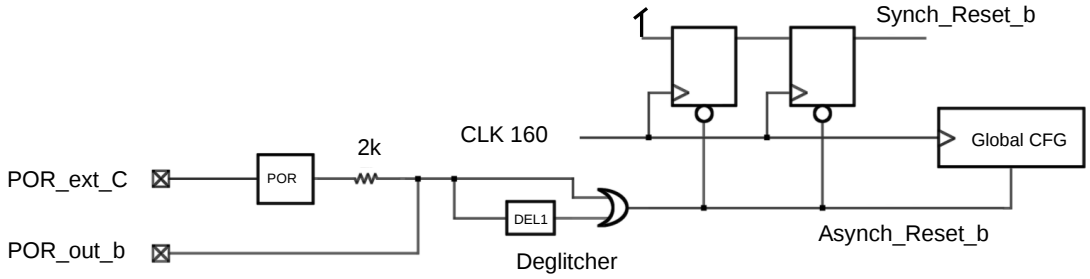


Figure 38: Power-on reset usage. Nets EXT_POR_CAP and POR_OUT_B are a available on dedicated wire bond pads.

The POR output is an asynchronous signal that directly controls the configuration memory reset. This reset loads the pre-defined default values in all registers. The pixel configuration registers are not reset. The pixel configuration registers do not have a reset function. Instead, the pixels are switched to use hard-wired default configuration instead of the values stored in their registers. The MUX switch that selects which configuration the pixels use is controlled by one of the global configuration registers (PIXDEF). Thus, there is only one global configuration, stored in registers, which resets to a default value after power up, but there are two pixel configurations, one hard-wired and one stored in registers, and a MUX to toggle between them.

In order to generate a synchronous reset to reset logic circuits, the asynchronous POR output is synchronized with the 160 MHz clock (CMD_CLK) as shown in Fig. **??**. For this to work, the 160 MHz clock must be present *before* the POR reset condition has been lifted. The CDR circuit (Fig. 35) will in fact generate a clock as soon as it is powered, even if there is no external serial bitstream provided for the PLL to lock to. The frequency of this prompt clock is the Voltage Controlled Oscillator (VCO) frequency divided by 8. While in reset, the PLL is not yet active and the VCO is controlled by the voltage on wire bond pad PLL_V_VCO (Table 17). As PLL_V_VCO

is internally pulled up to VDDD, if this pad is not connected the VCO will come up at its maximum frequency, which is about 2.5 GHz before irradiation, and this will lead to about 320 MHz logic clock instead of the desired 160 MHz. While this should still be adequate to reset the logic, an external resistor to ground on PLL_V_VCO is recommended (value to be determined by test) in order to lower the VCO reset voltage.

On the other hand, if the CDR is bypassed, requiring the user to supply the 160 MHz clock, this external 160 MHz clock should be supplied *before* power is applied for the synchronous reset to work properly.

The synchronous reset acts on the the Channel Synchronizer and the Command Decoder blocks (Fig. 34). Immediately after this reset, the Channel Synchronizer will lock to the correct phase of the incoming command stream, using Sync symbols (see Sec. 9.2) and the Command Decoder will begin processing commands. The lock signal will be asserted on wire bond pad STATUS, as mentioned earlier. The user must send an ECR command to the chip in order to reset the pixel matrix and the data path. An ECR is *not* generated automatically by the POR.

7.3 JTAG and Command Bypass

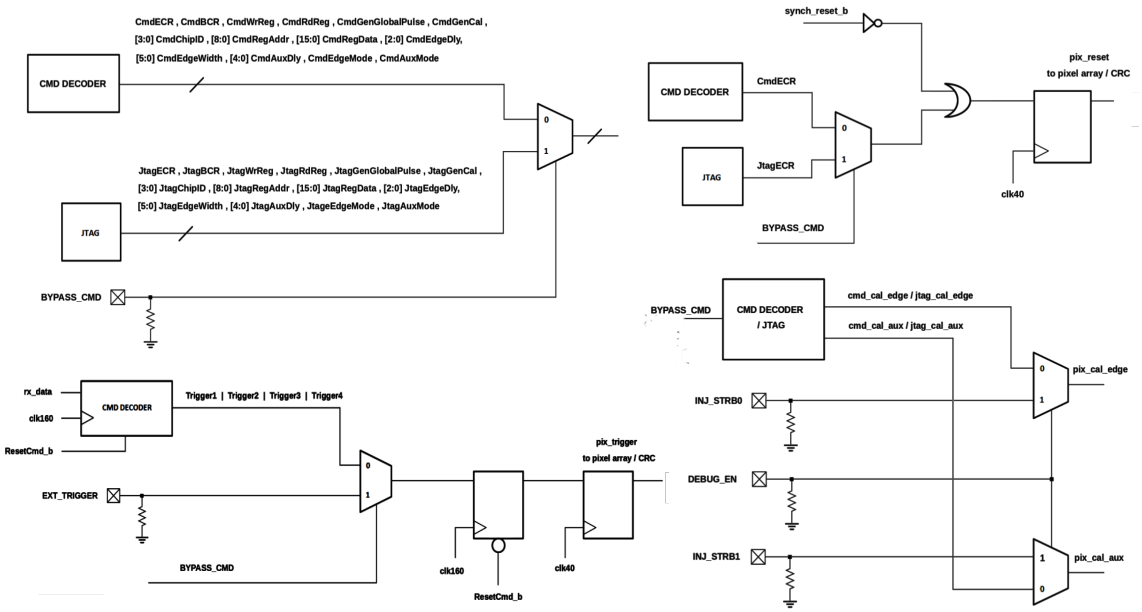


Figure 39: Schematic of command decoder bypass options.

The JTAG interface has a dedicated reset pin as defined in the JTAG standard. This pin has an internal resistor ensuring the default value is in reset, which means the JTAG interface is not used. To use the JTAG interface, this pin must be externally forced, which will take the JTAG out of reset. The JTAG interface can be used to spy on on internal registers even during normal operation and to run structural tests (scan chains). One can also fully control the chip from the JTAG interface and for this the normal command path must be bypassed. Fig. 39 shows how the command decoder

can be bypassed. All commands with no precise timing can be supplied via the JTAG interface (Sec. 7.3), while trigger and calibration pulses must be externally supplied on dedicated wire bond pads (Table 20).

7.4 Trigger

Trigger pulses are normally generated by the command decoder block (CMD) as shown in Fig. 39 bottom left. (in each 40 MHz BX_CLK cycle there can be one trigger pulse). Trigger pulses can also be supplied externally if the CMD is bypassed, or generated automatically (self trigger) as described in Sec. 10.10. The trigger book keeping uses a 32-deep Trigger Table (TT). A trigger pulse (see Sec. 9) causes the next available row in this table to be filled as depicted in Fig. 40. The pulse is also sent to the pixel matrix where it causes storage of all hits that have reached the programmed latency at that point. To identify those hits as belonging to this particular trigger pulse, a trigger ID is distributed to the matrix along with the pulse. This ID is nothing more than the 5-bit row number in the TT. The TT row is filled with a trigger tag and the bunch crossing ID (BCID) of the hits to be read out. The trigger tag is an arbitrary 5-bit code provided by the user in the trigger command, plus 2 bits for the position of the specific trigger within the 4 bunch crossings covered by the trigger command. Thus the trigger tag is 7 bits. The BCID is the value of an internal counter counting bunch crossing clocks. The user can reset this counter with the BCR command (see Sec. 9) in order to synchronize the the counter in the chip with the DAQ. In the absence of trigger tags this is essential to correctly identify the events read out, but with trigger tags the BCID counters do not need to be synchronized and may not even be read out. The TT is emptied (first in

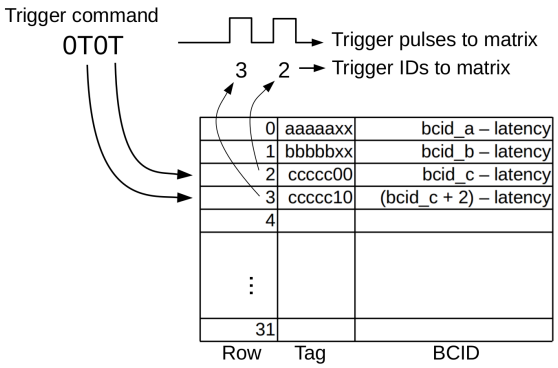


Figure 40: Trigger Table (TT) with example of filling with one trigger command encoding two triggers (first and third bunch crossings).

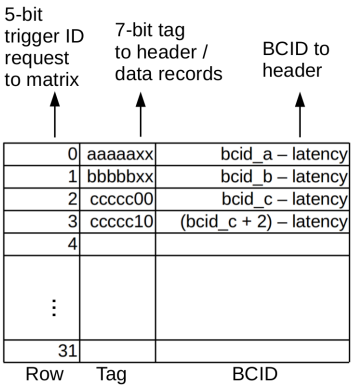


Figure 41: Illustration of how Trigger Table (TT) is emptied by data readout.

first out) by the data readout (Fig. 41). After an event has been read out, the corresponding row in the TT is cleared. Readout is data driven: readout proceeds as fast as possible until the TT is empty. The TT pointers and BCID counter are all triple redundant. The tag and BCID values stored in the TT are Hamming code protected against single bit flips.

7.5 Global Registers

As far as the user is concerned, all non-hit information within the chip consists of global registers with a 9-bit address and a 16-bit content- there is no other information format. However, the

implementation of the registers (transparent to the user) varies according to their function. All registers are synthesized in the digital bottom of chip (assignments are listed in Sec. 11.3). The different register implementations are:

Global Configuration (R/W):
These are 16-bit, SEU-hard triple redundant (TR) latches with automatic TR error correction. They use an analog macro that is placed by the synthesis tool.

Pixel Configuration Portal (R/W):
This is global register address 0. It is not really a register but an interface to the pixel pair(s) that is(are) pointed to by the row and column address registers (see Sec. 9.3).

Diagnostic Counters (R/W):
There are fifteen 16-bit Counters, synthesized with standard cells. A write register operation will reset the counter to zero. Therefore, as part of the start up procedure all these counters should be written. The clock driving each counter is any given error/warning/status signal. For example, any time a TRE mismatch is detected (and corrected) in any global configuration register, the TRE output signal will be asserted, and this signal can be counted (this should be considered a warning, not an error, since it was automatically corrected). Similarly Command Decoder errors, loss of sync events, etc., are counted this way. The counters stop when they reach the maximum value (they do not loop).

Ring Oscillator Counters (R-only):
These read-only registers each contain a 12-bit and a 4-bit counter. The clock for the 12 bit counter is provided by a ring oscillator, while the 4-bit counter counts start/stop pulses. See Sec. 10.7 for details. They are reset using the global pulse signal.

Monitoring ADC (R-only):
The monitoring ADC outputs are read out with a read register command like everything else. This register address is just a port to the ADC outputs- there is no actual register and it cannot be written.

SEU-soft Dummy Registers (R/W):
This are registers made of 16 standard cell latches that have no function. They are just random access memory. Different registers use latches from different libraries. Useful for easily measuring SEU and even for monitoring radiation dose rate, which SEU's will track.

8. Analog Chip Bottom

The analog chip bottom (ACB) block, a multi-purpose block, is located in the periphery of the chip. The task performed by the ACB are described by the block diagram in Fig. 42. The first task of the block is to provide different current references to current DACs. A 4 μA current reference can be regulated with a DAC in order to compensate for the process variations. The second task of the ACB is to monitor different signals coming from the RD53A chip 4 μA current reference, temperature sensor, radiation sensor, VCO, etc...). Such signals are digitized by a 12 bit ADC. The ACB also includes the clock data recovery, the serializer and power on reset blocks. All these circuits have been thoroughly tested, also from the standpoint of radiation tolerance before the integration in the RD53A chip. The third task of the ACB is to provide the two voltage levels for the calibration circuit.

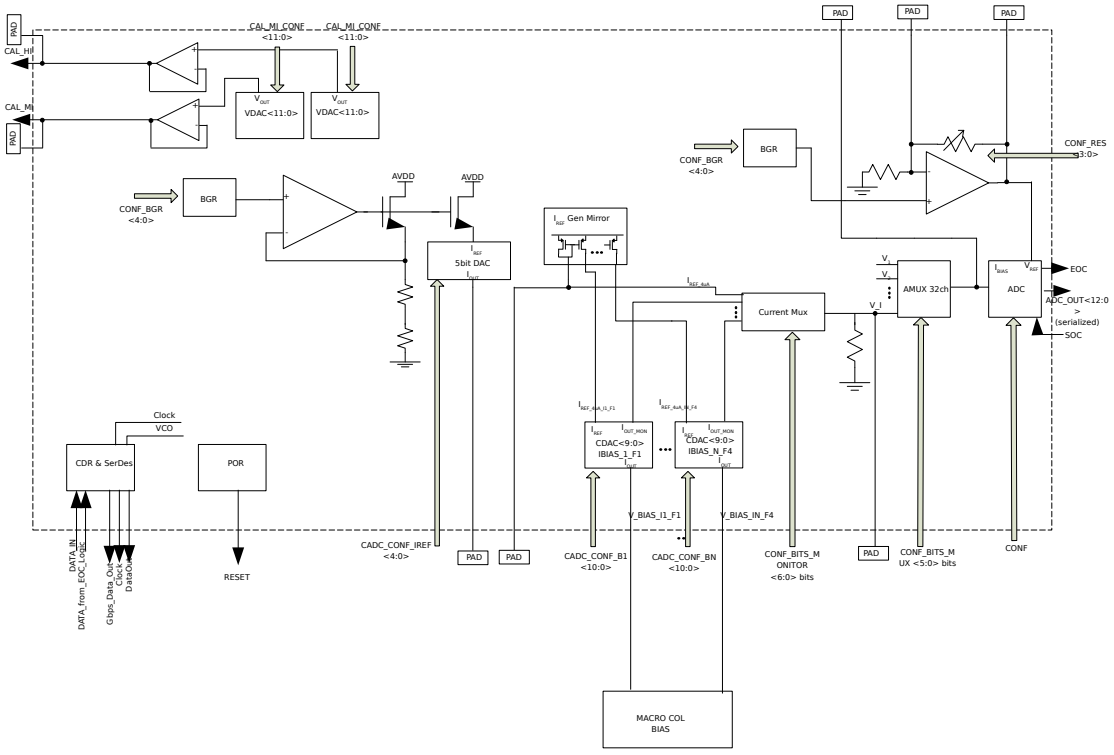


Figure 42: Analog chip bottom block diagram

8.1 Monitoring Block

The Monitoring block is designed to be used in the chip RD53A for the digitization of several important parameters in the chip, as the temperature, the total dose and voltages or currents values in different parts of the chip.

The monitoring block (Fig. 43) contains 3 main sub-blocks:

- Input stage composed of a 32 to 1 analog multiplexer
- Reference voltage design based on the bandgap structure

- 12 bit ADC based on the Successive Approximation Register (SAR) architecture

The clock input frequency is the same as the main chip input clock which is typically 40 MHz. A 128:1 frequency divider is implemented in order to generate the internal clock driving the ADC at 312 kHz.

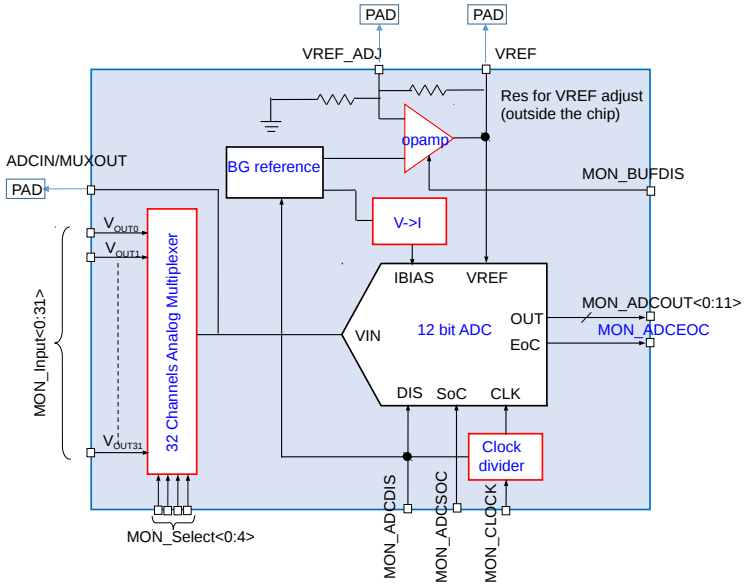


Figure 43: Monitoring block diagram

8.1.1 Analog Multiplexer

The analog multiplexer has 32 sets of switches that connect analog inputs to a common output. A 5-bit address is decoded with combinatorial logic to select only one transmission gate to be in the ON state at any given time. Each transmission gate is built with a parallel combination of NMOS and PMOS transistors driven by a complementary gate. In this case the input signal is transmitted to the output without the threshold voltage attenuation. The present design is using Standard Threshold Transistors (SVT). A High Threshold Voltage Transistors (HVT) could be used to minimize the effect of the leakage current but the TID effect have to be considered.

8.1.2 Bandgap Reference

A voltage reference with very low sensitivity to process, temperature or power supply variations is highly required for the pixel chip design. Furthermore, a such design is essential for the block designs, as voltage regulators, A/D and D/A converters. In principle, the readout chip for Atlas pixels is designed to operate at low temperature achievable with cooling. However, the chip has to operate normally in the extended range of -40°C to +60°C. Since the temperature measurement and digitization is one of the main objectives of the monitoring block, the ADC voltage reference should be stable as function as the temperature variation and should be very tolerant to the TID. In fact a voltage shift of 1 mV corresponds to an error of 5 LSB at the top of the ADC input scale. This 1 mV shift corresponds to an error of 1°C in the temperature measurement. The table 1 is

summarizing the main specifications required for the ADC voltage reference implemented in this monitoring block. Keeping the variation below 1 mV is essential in order to keep the temperature error measurement below 1°C.

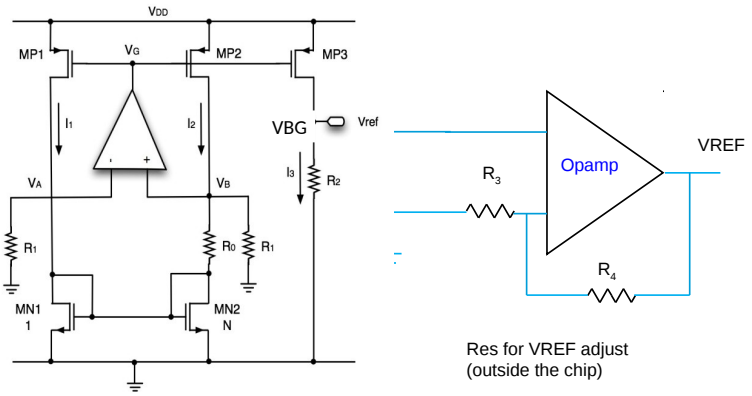


Figure 44: Bandgap reference diagram

Fig. 44 shows the principle used for the voltage reference design. The bandgap cell is based on NMOS device biased in the weak inversion region and it is compatible with the low supply voltage. It is designed by Bergamo-Pavia institute. The typical output value is 400 mV and configuration bits are used for the bandgap voltage adjustment which is done by adjusting the resistance R0. The configuration bits are set to the value allowing to minimize the variation of the voltage versus the temperature. An op-amp is used here to generate the ADC voltage reference from the BG voltage. The opamp is used in non inverting configuration and R3 and R4 adjust the gain and so the reference voltage. The typical gain value of 2.25 (1+R4/R3) is required to reach a reference voltage of 900 mV. R3 or R4 can be adjusted on chip by a configuration bits from the global register or off chip by selecting the appropriate values for R3 or R4.

8.1.3 SAR ADC

The successive-approximation ADC (Fig. 45) is the most popular architecture for data-acquisition applications, especially when multiple channels require input multiplexing. The internal DAC stage is based on a capacitance network supplied through the reference voltage to generate the 12 bit voltage scaling. The integrated capacitance area is chosen to keep the mismatch as low as possible and to achieve a very low noise allowing a high accuracy and good linearity.

For ADC tests and characterization, the input voltage and the voltage reference are provided by a commercial 16 bits DACs (LSB is 27.5 μV). The ADC reference voltage is set at 900ÂămV and the input voltage is varying from 0 to 900 mV by a step of 27.5 μV. The Differential Non Linearity (DNL) is measured with the histogram method. Fig. 46 shows that the DNL is below 1 LSB (1 LSB = 200 μV) and so the ADC doesn't show any missing code. However, we observe that the DNL increases for high ADC code values. This is because, in the present design, the input capacitance of the comparator varies with the input voltage and it is not maintained negligible. The Integral Non Linearity (INL) calculated from the DNL values is estimated to 2.5 LSB.

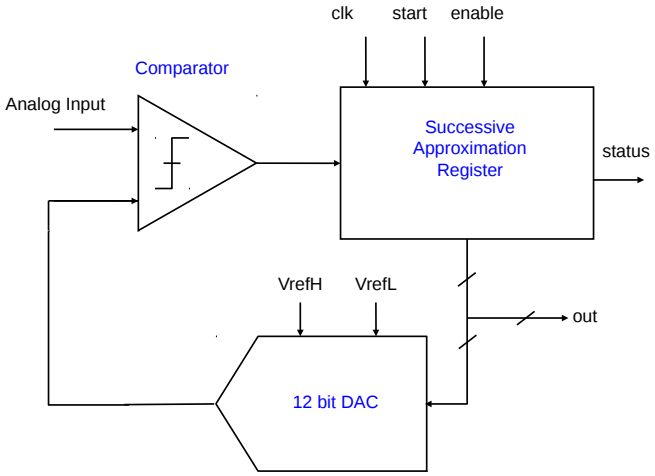


Figure 45: Successive approximation ADC

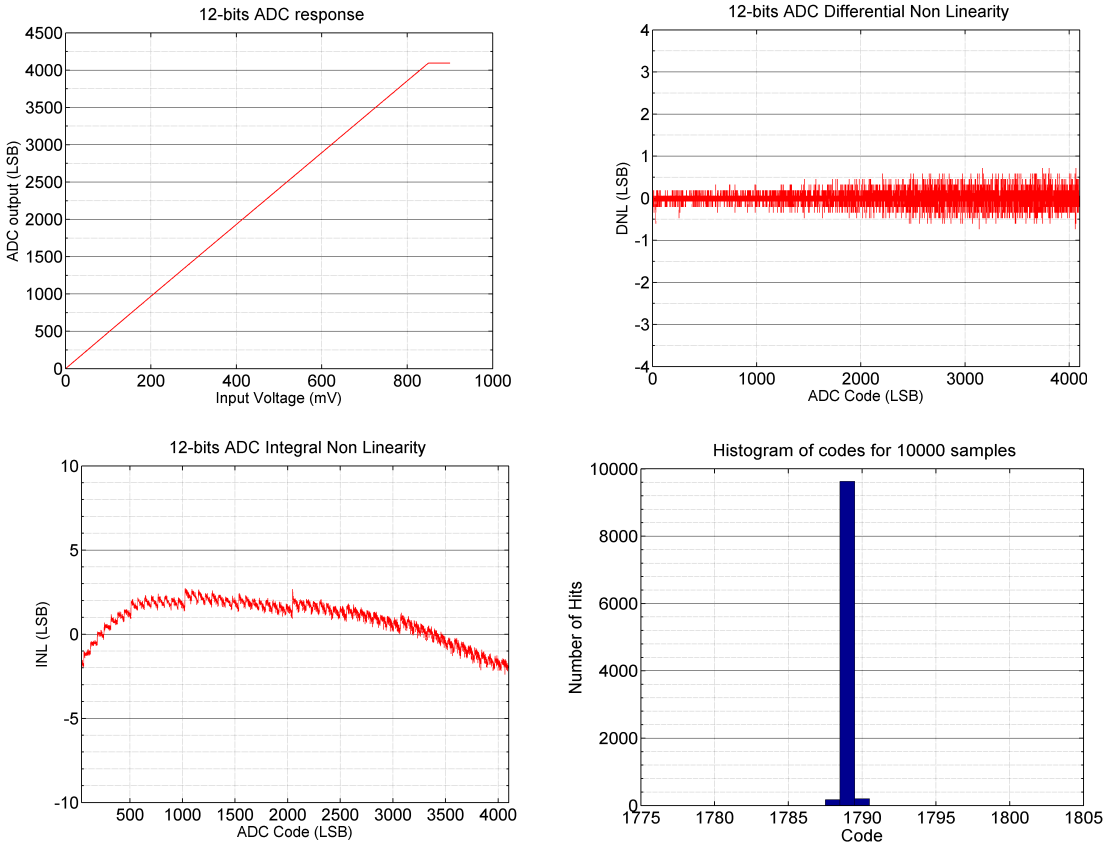


Figure 46: ADC performance plots

8.2 Calibration Injection Voltages

The main function of this calibration block is to generate and distribute 2 dc voltage levels CAL_HI and CAL_MED required by the injection circuit of each pixel in the matrix array. The block diagram of this block is shown in Fig. 47, where buffers and DAC are located in the ACB. Two resistors has been included at the output of the buffers, in order to guarantee the stability of the buffers. The two voltage levels, called CAL_HI and CAL_MED, are generated by a 12 bits voltage DAC IP and are buffered by the buffer IP. The idea is to drive the full size chip pixel matrix by using only two buffers. Simulations show that the maximum number of the pixels that can be pulsed at the same time has to be set to 400 in order to respect the timing requirements. During the calibration phase, the CAL_MED is set to the voltage value of 300 mV while the CAL_HI can be adjusted from 322 mV to 864 mV allowing to set the charge injection value in the range of 1.2 ke^- to 30 ke^-.

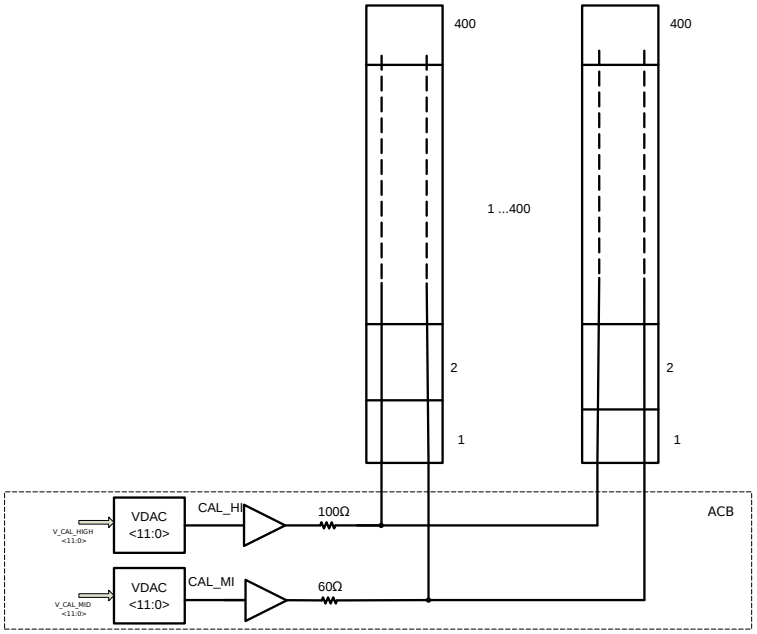


Figure 47: Charge injection voltage generation

9. Input, Output and Configuration

The RD53A chip is fully controlled with a single serial input stream with a custom encoding described in Sec. 9.2. All data, messages, and configuration read-back are output on one of two alternate high speed serial ports. Port A consists of 4 CML outputs sending encoded data at 1.28 Gbps nominal bandwidth. Port B consists of a single controlled impedance differential CMOS output sending encoded data at 5.12 Gbps nominal bandwidth. The encoding used is Aurora 64b/66b. All clocks needed for operation are derived from the 160 Mbps input using a PLL based Clock and Data Recovery circuit (CDR) described in Sec. 9.1. The CDR can also be bypassed and and external clock sources used instead.

Output data are serialized with the Aurora protocol and transmitted on a multilane 1.28 Gbps port (default) or on a single lane 5.12 Gbps port, as discussed in Sec. 9.4. All output that is not hit data looks to the user as readback of random access memory with 16-bit words and 10-bit extended address space (the 10^{th} bit distinguishes between true global register address and pixel row number for the 9 bits that follow). This includes readback of configuration data, monitoring values, errors and warnings, and test structures. All information types use the same format of 10-bit extended address plus 16-bit value, collectively known as global registers and described in Sec. 7. Most global registers can also be written to for various purposes, but only a subset of them contain chip configuration, as defined later. A few global registers are read-only. The writing and reading of pixel configuration, like everything else, takes place through a global register, described in Sec. 9.3. RD53A supports *trickle configuration*, which means that all configuration (global and pixel) can be gradually updated during operation by continually sending write commands in between triggers. This avoids the need for SEU hard configuration storage.

9.1 1.28 Gbps Output Serializer

The 1.28 Gbps output serializer block (Fig. 48) is reproduced 4 times, once for each of the Aurora lanes as discussed in Sec. 9.4. This is the only serializer type in RD53A (type 0) even though the configuration allows for a second type. It uses the 640 MHz clock from the CDR to serialize with double data rate the encoded output data. This can the chip data as described in Sec. 9.4 or a test pattern selected within the serializer itself. The test pattern options are PRBS (Pseudo Random Bit Sequence) indicated by LFSR (Linear Feedback Serial Register) in Fig. 48, the 640 MHz clock, or all zeros. The serializer can also be completely disabled. The speed can be reduced by a factor of 2 to 8. The output driver type is CML, with two programmable, synchronous pre-emphasis taps.

In addition to being sent to the serializer, the 640 MHz clock is used to digitally delay the CMD_DATA and CMD_CLK CDR outputs shown in Fig. 35. The delay step size is therefore 1.5625 ns. The total number of delay steps is user configurable. This allows to synchronize all chips in a system to a common bunch crossing clock (to within the step size). The same delay method is further applied to the CAL_edge internal signal, in order to implement a fine delay of the calibration injection relative to the local (after delay) bunch crossing clock.

9.2 Command Protocol

Control of the chip over a single differential serial input has been implemented using a simple custom protocol. The protocol provides encoded clock and commands on a single link, is DC-balanced

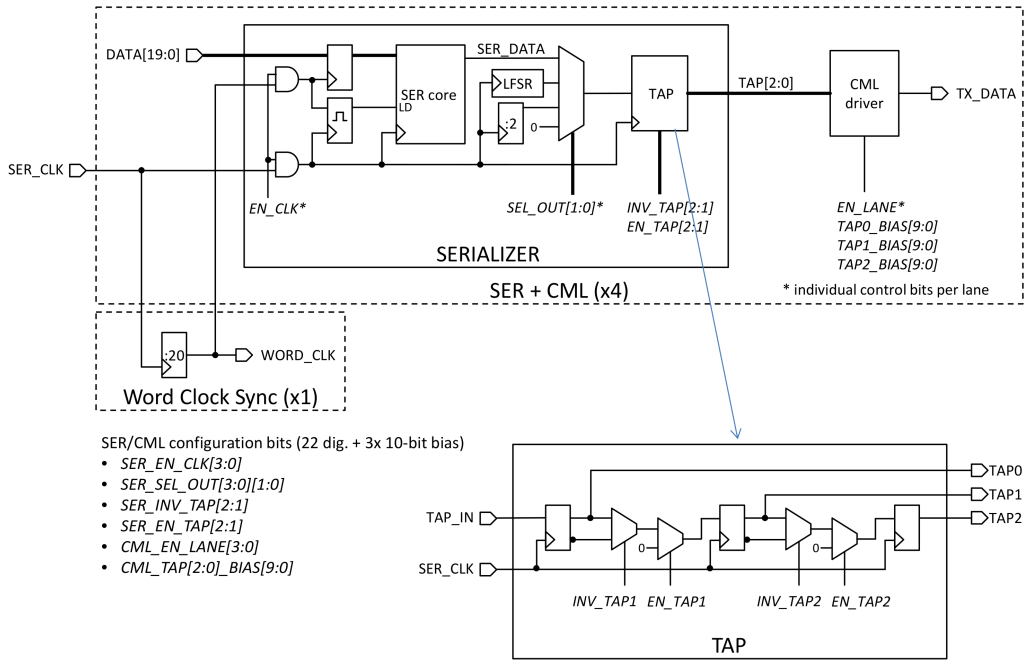


Figure 48: Diagram of the 1.28 Gbps serializer block operation options

with short run length for A/C coupling and reliable transmission, and has built in framing and error detection. These are all the properties that could have been obtained with 8b/10b encoding, but the 10-bit frame length of 8b/10b would have required 200 Mbps link speed in order to maintain an integer number of bunch crossings per frame, as needed for synchronous trigger operation. In contrast, the custom protocol developed runs at 160 Mbps, which makes for better transmission on low mass cables and can be directly driven from GBT e-links.

The protocol consists of a continuous stream of 16-bit frames. Each frame is exactly DC balanced. There is a unique sync frame (Table 4), plus 3 kinds of TTC frames: trigger, command, or data. All these frames contain two 8-bit symbols which are themselves DC-balanced. Furthermore, symbols that begin or end with three or more 1's or 0's are not used, resulting in a maximum run length of 4, except for the sync/idle frame which has a run length of 6. The full list of valid symbols is given in Tables 3, 4, and 7. There are 15 trigger symbols (Table 3) allowing the encoding of 15 trigger patters, 32 data symbols (Table 7) allowing the encoding of 10 bits of data per data frame, and 7 command symbols (Table 4). A single bit flip always results in an invalid symbol- formally all symbols are separated by a Hamming distance of 2.

Because one 16-bit frame spans 4 LHC bunch crossings, the trigger commands must specify a 4-bit map indicating which of the 4 bunch crossings are actually triggered; hence 15 trigger patterns. The triggering is synchronous, and therefore trigger frames must be sent at specific times. All other frames "fill the spaces" between trigger frames. It is recommended that at lest one sync frame be inserted at least every 32 frames. The trigger frame also contains a data symbol, which the chip interprets as a 5-bit tag to identify the trigger (see Sec. 7). This tag will be returned with the data corresponding to that trigger (See Sec. 9.4).

RD53A interprets the protocol in three phases (which will be transparent to the user): Initialization 9.2.1, Data Transmission 9.2.2 and Decoding 9.2.3. The decoding timing and exception handling are covered in Sec. 9.2.4

Symbol Name	Encoding	Trigger Pattern	Symbol Name	Encoding	Trigger Pattern
			Trigger_08	0011_1010	T000
Trigger_01	0010_1011	000T	Trigger_09	0011_1100	T00T
Trigger_02	0010_1101	00T0	Trigger_10	0100_1011	T0T0
Trigger_03	0010_1110	00TT	Trigger_11	0100_1101	T0TT
Trigger_04	0011_0011	0T00	Trigger_12	0100_1110	TT00
Trigger_05	0011_0101	0T0T	Trigger_13	0101_0011	TT0T
Trigger_06	0011_0110	0TT0	Trigger_14	0101_0101	TTT0
Trigger_07	0011_1001	0TTT	Trigger_15	0101_0110	TTTT

Table 3: List of trigger command and symbols used to encode the 15 possible trigger patterns spanning four bunch crossings. Note there is no 0000 pattern as that is the absence of an trigger.

Command	Encoding	ID/(A)ddress/(D)ata 5-bit Fields					
ECR	2× 0101_1010						
BCR	2× 0101_1001						
Glob. Pulse	2× 0101_1100	ID<3:0>,0	D<4:0>				
Cal	2× 0110_0011	ID<3:0>,D15	D<14:10>	D<9:5>	D<4:0>		
WrReg	2× 0110_0110	ID<3:0>,0	A<8:4>	A<3:0>,D<15>	D<14:10>	D<9:5>	D<4:0>
WrReg	2× 0110_0110	ID<3:0>,1	A<8:4>	A<3:0>,D<15>	D<14:10>	9×(D<9:5>	D<4:0>)
RdReg	2× 0110_0101	ID<3:0>,0	A<8:4>	A<3:0>,0	00000		
Noop	2× 0110_1001						
Sync	1000_0001_0111_1110						

Table 4: List of non-trigger commands and address or data fields associated with each. In some fields a padding bit is added (indicated by "0") in order to make up 5 bits per field and so have an integer number of 8-bit symbols after encoding, as well as an integer number of 16-bit frames (double vertical lines denote frame boundaries).

The non trigger commands are defined in Table 4 along with the command arguments. Each command frame contains one pair of identical command symbols, which allows for error correction of commands. The command is only executed once per frame. The text descriptions below provide further information to supplement the table. As one single command line might drive multiple chips there is a mechanism built into the protocol that allows each chip to know if the command has to be processed or skipped. The two commands without data fields (ECR and BCR) are always processed. For all commands that have some associated data, the first field always contains the 3-bit Chip Id plus a Broadcast bit. This 3-bit chip ID allows controlling up to 8 chips with a single link. If the Broadcast bit is set, Address is ignored and the command is executed. Otherwise the command is only executed if Address matches the wire-bonded Chip ID value. The commands are always fully decoded including the associated data: the Chip ID only controls whether the

command is executed (technically whether the command decoder block outputs to the rest of the chip are enabled).

ECR (Event Counter Reset):
The name is historical as the functionality is much more than resetting a counter. This flushes the hit data path inside the chip, clearing all prior pending triggers and hits. Configuration data and pending errors/warnings are unaffected. this command should be issued prior to sending triggers for the first time since power-up.

BCR (Bunch Counter Reset):
Resets the internal counter of bunch crossing clock cycles. This is important in the case that thus counter needs to be synchronized with with the DAQ system. When using trigger tags that is not necessary.

Global Pulse:
The global pulse command sends a single pulse with a duration of 1, 2, 4,... 512 bunch crossings depending on the value of the data provided. Values 0-9 result in durations of 2^{value}, while values greater than 9 result in 512. The global pulse is routed as explained in Sec. 7.

Cal (Calibration Injection):
The same command is used for both analog and digital injection. Whether injection will be analog or digital is decided by a global configuration setting, but the Cal command produces the same output regardless. To understand the Cal command it is necessary to first read Sec. 5.4. The command controls the generation of the two internal signals CAL_edge and CAL_aux. For digital injection only the CAL_edge signal is relevant. The CAL_edge signal to be generated is specified by the first 10 data bits of the Cal command: mode (1 bit), delay in bunch crossing clocks (3 bits), and duration in 160 MHz clocks(6 bits). The CAL_aux signal is specified by last 6 data bits: value (1 bit) and delay in 160 MHz clocks (5 bits).

The CAL_edge mode selects between step (mode=0) or pulse (mode=1). For both modes a rising edge is generated after the specified delay unless the duration is zero. If duration is zero then no action is taken; also if the previous state of CAL_edge was high then nothing will change (signal is already high, so rising edge is not possible). A falling edge is then generated after delay plus duration *only* in pulse mode. The CAL_edge delay value is in units of bunch crossings, but the duration is in cycles of the 160 MHz clock, allowing for tests of fast time over threshold counting. Up to this point the generated edges are synchronous with the 160 MHz clock, but before CAL_edge is distributed to the pixel matrix, a programmable fine delay is applied. This fine delay is pre-programmed in a global register and is in units of 640 MHz clock cycles. This allows precision scanning of the pixel timing.

The CAL_aux signal is just one edge (no pulse mode) and does not have fine delay applied. CAL_aux is simply switched to the specified value after the specified delay, in units of 160 MHz clocks. This fine scale for the delay allows changing the CAL_aux value in the middle of a bunch crossing cycle, as is needed to inject charge in two consecutive crossings. A typical single pulse analog injection sequence is specified in Table 5. If instead one

wishes to inject two consecutive pulses separated by a 5 bunch crossings (for example), the command sequence of Table 6 would be used. Timing diagrams of step and pulse modes are shown in Figs. 49 and Figs. 50.

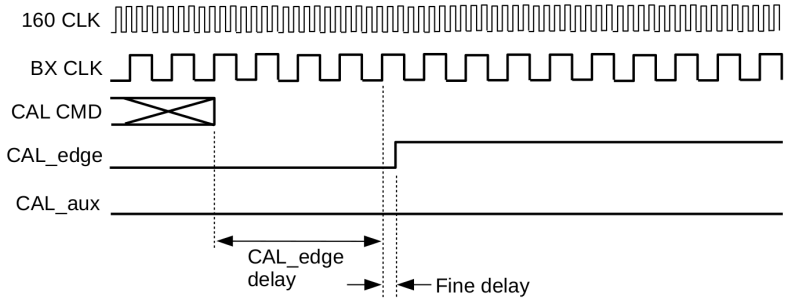


Figure 49: Timing diagram for calibration step mode (mode bit = 0)

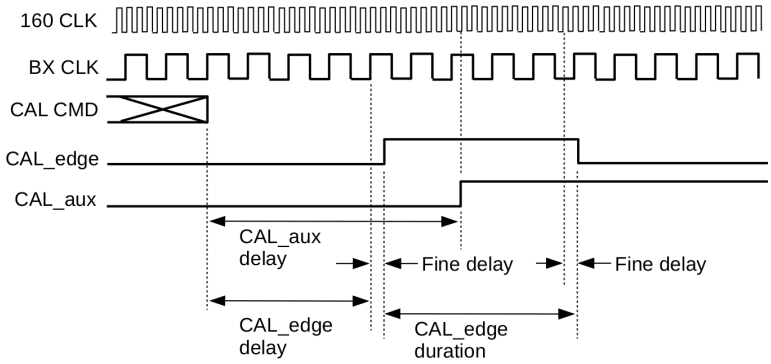


Figure 50: Timing diagram for calibration pulse mode (mode bit = 1)

Command	CAL_edge	CAL_aux	Comment
Cal	mode=1, delay=0, duration=4	value=0, delay=0	arms both signals to low
none	n/a	n/a	wait for settling
Cal	mode=0, delay=0, duration=1	value=0, delay=0	single Cal edge; no delay
none	n/a	n/a	wait for latency
Trig	n/a	n/a	reads out the injection

Table 5: Sequence of commands for typical single pulse analog injection. The sequence if repeated for the next injection. Up to the user to decide (or study) how long to wait for settling. How long to wait for latency depends on the programmed latency value.

WrReg (Write Register):

In addition to chip ID the command has 9 bits of Address and either 16 or 96 bits of Data. The amount of data is determined by the bit immediately following the chip ID (the WrReg command appears twice in Table 4, once for each value of this bit). This command writes the 16 (96) bits of Data into the addressed global memory register. Addr<8:0> is used to address

Command	CAL_edge	CAL_aux	Comment
Cal	mode=1, delay=0, duration=0	value=0, delay=0	arm CAL_aux to low
none	n/a	n/a	wait for settling
Cal	mode=1, delay=0, duration=20	value=1, delay=10	double pulse injection
none	n/a	n/a	wait for latency
Trig	n/a	n/a	reads out first injection
Trig	n/a	n/a	reads out second injection

Table 6: Sequence of commands for example double pulse analog injection. The sequence if repeated for the next injection pair. Two trigger commands are needed for readout because the injections are separated by 5 bunch crossings, but for injections within 4 bunch crossings one trigger command may suffice. Up to the user to decide (or study) how long to wait for settling. How long to wait for latency depends on the programmed latency value.

the register to write to. In case 96 bits of data are supplied instead of 16, the command writes the same register 6 times in succession. This is not useful for a normal register, since only the last 16 bits would be relevant, but it is useful for writing register 0 in auto-increment mode (see Sec. 9.3), resulting in a factor of 2 speedup of pixel configuration. Register 0 is special: it is a virtual register used to write and read pixel configuration. This command does not produce any output from the chip. Up to 512 16-bit wide registers can be addressed, but not all 512 possible register addresses are used. If an attempt is made to write to an unused address, no register will be written and warning code will be generated. The register memory map is given in Tables 22, 23.

RdReg (Read Register):
This command has 9 bits of address following the Chip ID. It initiates the readout of the addressed register. Address 0 is special: it is the the pixel register as described in Sec. 9.3. The 16-bit register value is returned in the data stream as described in Sec. 9.4. Not all 512 possible register addresses are used. If readback of an unused address is requested, the value returned will be 0 and warning code will be generated. The register assignment list is given in Tables 22, 23.

9.2.1 Command Protocol Initialization

The sync pattern (Table 4) can not be produced through any combination of TTC frames and therefore can be searched for to lock the correct frame boundaries. Sync frames must be sent at the start of operation so that the framing can be locked, and they are also sent whenever there is no information to send (no trigger, command, or data pending), or as needed to ensure that there is at least one sync frame in every 32 frames (this is a recommendation, not a strict requirement). Typically at the start of operation (power up) there are no commands or triggers to immediately send, and so by default transmission always begins with sync frames. The transitions in the string of sync frames will allow the clock recovery circuit to lock to the correct 160 MHz frequency and phase, and using this clock the the frame alignment circuit will search for sync and count each valid

Symbol Name	Encoding	Data Value	Symbol Name	Encoding	Data Value
Data_00	0110_1010	5'b00000	Data_16	1010_0110	5'b10000
Data_01	0110_1100	5'b00001	Data_17	1010_1001	5'b10001
Data_02	0111_0001	5'b00010	Data_18	1010_1010	5'b10010
Data_03	0111_0010	5'b00011	Data_19	1010_1100	5'b10011
Data_04	0111_0100	5'b00100	Data_20	1011_0001	5'b10100
Data_05	1000_1011	5'b00101	Data_21	1011_0010	5'b10101
Data_06	1000_1101	5'b00110	Data_22	1011_0100	5'b10110
Data_07	1000_1110	5'b00111	Data_23	1100_0011	5'b10111
Data_08	1001_0011	5'b01000	Data_24	1100_0101	5'b11000
Data_09	1001_0101	5'b01001	Data_25	1100_0110	5'b11001
Data_10	1001_0110	5'b01010	Data_26	1100_1001	5'b11010
Data_11	1001_1001	5'b01011	Data_27	1100_1010	5'b11011
Data_12	1001_1010	5'b01100	Data_28	1100_1100	5'b11100
Data_13	1001_1100	5'b01101	Data_29	1101_0001	5'b11101
Data_14	1010_0011	5'b01110	Data_30	1101_0010	5'b11110
Data_15	1010_0101	5'b01111	Data_31	1101_0100	5'b11111

Table 7: List of command symbols used to encode data values

appearance of this pattern in 16 separate channels (one channel for each possible frame alignment). When the count for one of the channels, i, reaches a threshold N_{lock}, *sync* is declared as acquired, channel i is adopted as the correct channel, and the count of the remaining 15 channels is reset. The number of sync symbols needed at the start of transmission to guarantee a lock is $2N_{lock}$. The 40 MHz bunch crossing clock is generated as the bit pattern 1100110011001100 aligned to channel i. Thus there are 4 bunch crossings with a fixed phase relationship to the sync frame, which can be labeled $BXa..BXd$. The counting of sync sequences continues in all the channels, but every new sync sequence detected on the lock channel i resets the count for all the other channels. If ever the count for a channel that is not the lock channel reaches a second threshold N_{unlock}, sync is declared lost, and a new sync lock is acquired on the first channel that reaches the locking threshold N_{lock}. This allows for continuous channel monitoring and automatic sync lock as long as enough sync symbols are transmitted on the input line. N_{lock} and N_{unlock} should be user configurable, but must have nonzero default values.

9.2.2 Command Protocol Transmission

During transmission the necessary sequence of frames is sent to control the chip. Trigger frames are sent at specific times, and the "space between trigger frames" is filled with commands, data and sync frames. The data frames following a command frame are interpreted as belonging to that command regardless of intervening trigger, BCR, ECR, or sync frames.

9.2.3 Command Protocol Decoding

After link initialization and data being encoded and transmitted on the transmitting end, inside the chip the 160 MHz clock is recovered using a PLL, and incoming data are frame-aligned and

decoded. The frame alignment of the incoming bits to the recovered 160 MHz clock has been performed as described under initialization. The locked channel from frame alignment (and only this channel) is fed to the Command Decoder logic block. In the absence of a lock, nothing is fed to the command decoder, so until a lock happens no commands can be interpreted. The locked condition guarantees that the bits fed to the command decoder are correctly aligned with the clock. Protocol consistency is checked by requiring that there is always a valid symbol pair or a sync in each 16-bit frame. The 16 bits are fed to the command decoder with a parallel bus. In case of correct detection the indicated action is performed according to the command type or Chip ID. All symbols are always checked and decoded, even if they follow a Chip ID that does not match the wire bonded ID (if not wire bonded the default is zero). However, the Command Decoder will produce an output only if the command is not addressed (Trigger, ECR, or BCR), if decoded Chip ID matches the wire bonded ID, or if the decoded broadcast bit is 1. The detection of an invalid symbol is handled differently depending on the frame and expectation (current state). The handling of such exceptions is shown in Table 8.

Frame received	**Frame Expected**	**Error/Action**
invalid, data	data	Aborted command
data, invalid	data	Aborted command
invalid, invalid	data	Aborted command
invalid, data	not data	Lost trigger
invalid, invalid	not data	Corrupted frame
invalid, sync	any	Corrupted sync
sync, invalid	any	Corrupted sync
invalid, command	any	Execute with warning
command, invalid	any	Execute with warning
trigger, invalid	any	Lost Tag (*)
command, command	data	Ignored command

Table 8: Command Decoder response to invalid or unexpected symbols. (*) Tag value 0 is substituted. User is free to never send tag value 0, in which case there will be no confusion.

9.2.4 Command Protocol Timing

A timing diagram for generating commands from the machine clock and a trigger pattern is shown in Fig. 51. The decoded commands are executed 25 ns after the end of the last frame of the command data. “Executed” means that the outputs of the Command Decoder block in Fig. 34 change state, which happens of a rising edge of the beam clock. In many cases the execution is instantaneous (outputs change state and that’s it), but the Trigger, Cal and Global Pulse commands have a delay and duration. The trigger command sends 1 to 4 pulses in 4 consecutive beam clock cycles, and thus is completely finished before a new command can be completely received (since 1 frame is 4 beam clock cycles). The Cal and Global pulse commands can occupy their respective output lines (CAL_edge, CAL_aux, and Global_pulse) for many clock cycles. A new trigger or global pulse command should not be sent before the first such command is complete (up to the DAQ to ensure this), but any other command can be sent and will be executed normally.

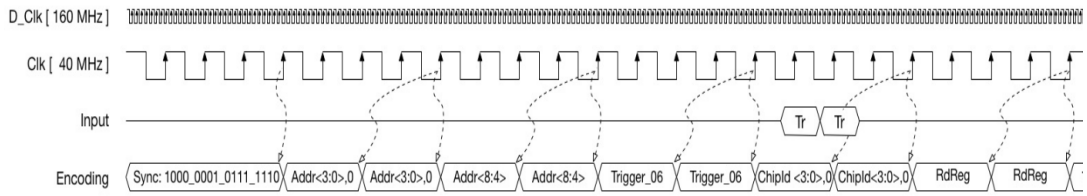


Figure 51: Timing diagram showing command encoding with duplicated symbols. Time progresses from right to left. The triggers are encoded in commands synchronous with the bunch crossing clock edges as shown. Each 16-bit frame (at 160 Mbps) is aligned with every forth beam crossing clock edge.

9.3 Pixel Configuration

Each pixel has 8 or fewer bits of local configuration. The exact meaning of the bits varies depending on the front end type according to Table 30. However, from the point of view of writing or reading back the pixel configuration, each pixel is seen as one half of a 16-bit register, regardless of whether all 8 bits are used or not by a given front end flavor.

Internally, the writing and reading of configuration values from the pixels uses an addressed bus to every 4x1 pixel region for DBA cores, or 4x1 pixel group in the case of CBA cores (so can think of of full chip as made of 4x1 regions, regardless of readout architecture). All reading and writing is done two pixels at a time in a given column of 4-pixel regions. However, multiple quad columns can write in parallel, but readback can only take place from one quad column at a time. There are thus two write modes, single region and broadcast, while read is always single region. Broadcast mode uses a 3-bit mask to define which Front End flavors to broadcast to. The write and read operations are controlled by three global registers, the core column, the core row, and the pixel mode configuration registers. The core column and core row registers have a special feature called auto increment, which reduces the number of commands needed to fully configure the chip. The pixel data is written into or retrieved form global register 0 with normal write and read register commands. This is a virtual register acting as a portal to whatever region(s) is(are) pointed to be column and row config registers. The typical pixel matrix configuration write sequence is given in Table 9. Note that this takes 39200 commands to accomplish; at 40 bits per Write Register command (before encoding) and effective command bandwidth of 60 Mbps during 1 MHz triggered operation, this will take 26 ms (double for a full size chip). Writing a uniform (all pixels the same) configuration is 200 times faster because each Write Pixel command can write a whole row Table 10. Writing a single row with arbitrary data can be done by selecting auto column mode instead of auto row mode. The readback of the pixel configuration for the whole matrix can proceed exactly as shown in Table 9, substituting the Read Register command instead of Write Register. Writing or reading an individual, arbitrary pixel pair follows steps 1-3 of Table 9, setting the desired columns and rows instead of 0,0. The fastest possible readback could be achieved by sending a continuous stream Read Register commands. As this command consists of 40 bits after encoding, and results in 16 bits of configuration bits coming out, the maximum possible readback bandwidth is $160 \times 16/40 = 64$ Mbps.

Step	Command	Address	Explanation
1	Write_Register	column and mode config	set columns 0-1 and auto row mode
2	Write_Register	row config	set row 0
3	Write_Register	0	config first 2 pixels
4	Write_Register	0	config for next row 2 pixels
196	Write_Register	0	config for last row 2 pixels
197	Write_Register	column and mode config	set columns 2-3 and auto row mode
198	Write_Register	row config	set row 0
199	Write_Register	0	config for next row 2 pixels
39200	Write_Register	0	config last 2 pixels in chip

Table 9: Sequence to write an arbitrary pixel configuration. For readback replace Write Pixel commands with Read Pixel commands.

Step	Command	Address	Explanation
1	Write_Register	column and mode config	set broadcast and auto row mode
2	Write_Register	row config	set row 0
3	Write_Register	0	config all pixels, first row
4	Write_Register	0	config all pixels, second row
196	Write_Register	0	config all pixels, last row

Table 10: Sequence to write a default (all pixels the same) configuration

9.4 Data Output Protocol

The output is encoded with Aurora [2] 64b/66b on 1 to 4 parallel lanes (programmable). Each lane is nominally 1.28 Gbps, but can be divided down by 2, 4, or 8, or driven by an external clock. The multi-lane Aurora protocol uses *strict alignment*, which means that all lanes send the same type of Aurora frame at any given time. Aurora frames consist of 66-bits (Fig. 52). Each frame has a 2-bit sync header that can be 01 or 10; thus there are two types of Aurora frames, called data and command, respectively. RD53A implements two independent output "channels", data and register (or service), which are time-multiplex onto a single bitstream. The output stream basic unit consists of N_D data frames plus one register/service frame. This N_D:1 ratio is programmable so the user can decide what fraction of bandwidth to reserve for data. Data frames always have a 01 header (Aurora data frames) while register frames always have a 10 header (Aurora command frames). However, when there is no data to send out, idle frames will be substituted for data frames, and the idle frame has a 10 just like register frames. Register frames will not be sent except in their allocated turn every N data or idle frames. The interval N is used on every lane regardless of how many lanes are active. For example, with N_D=48, 2% of the output bandwidth is permanently unavailable for hit data (in addition to the 3% consumed by the 2-bit 64b/66b header). At 4×1.28 Gbps output bandwidth this 2% is sufficient for the maximum possible register readback of 64 Mbps, since 2% of 5 Gbps = 100 Mbps (See Sec. 9.3). In the register frames, an 8-bit code follows the sync header, as specified by the 64b/66b protocol, leaving 56 bits available for user information, which are allocated as two registers (10-bit extended address plus 16-bit value = 26 bits) plus 4 status bits:

2x([e-address (10 bits)] [value (16 bits)]) [status (4 bits)]

The 10-bit extended addresses (e-address) in the register frame are: MSB=0, followed by the 9-bit global register address, or MSB=1, followed by the 9-bit pixel row address in case of reading global register 0 (the pixel configuration portal register). The separation of the output into two time-multiplexed channels guarantees a certain bandwidth for both data and register information without the need for a complex priority arbitration containing safeguards against all possible pathologies. The format is depicted schematically in Fig. 52.

The periodic register frame coming out every N_D data frames is filled automatically, even without there having been a read register command. The two 16-bit registers are denoted Ai and Bi, where i is the lane number (0 to 3). The automatic filling of the Ai and Bi registers is controlled by eight configuration registers Auto-Ai and Auto-Bi, which have default values, but which the user is free to change. The auto-fill register addresses are specific to each lane. Thus if only lane 0 is used then only Auto-A0 and Auto-B0 are functional. RdReg commands will queue the registers specified by the command for output on lane 0 only, with priority over auto fill. Lanes 1 to 3 are unaffected by the RdReg command and only output their assigned auto-fill registers. If only one RdReg command has been received, then the A0 register will be auto-filled while the B0 register will contain the requested register. If more than one was received then both registers will be requested registers and auto-fill will wait. It may instead be desirable to monitor something of interest with the Auto-A/B registers and it is up to the user to decide what that is. The 4 status code has a fixed meaning as specified in Table 11.

The data frames contain fixed length records in RD53A. There are two modes: header mode and tag mode. In both modes every 64-bit data frame contains hit information from two groups

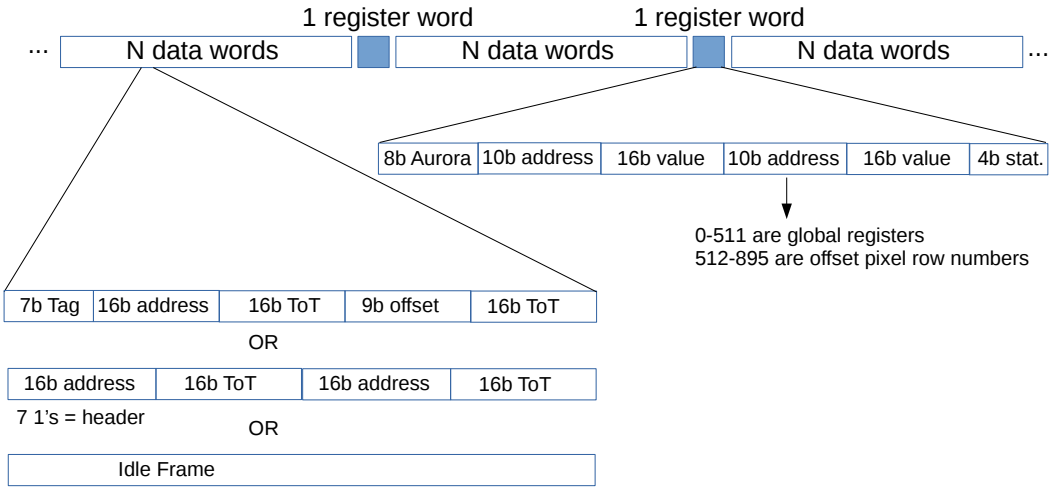


Figure 52: Schematic diagram of output data format, consisting of *N* data or idle frames followed by one register frame. Data format can be programmed to tag or header, but not both. Idle frames are substituted when there is no data.

Status Code (decimal)	**M**eaning
0	Ready
1	There has been an error since the last register frame
2	There has been a warning since the last register frame
3	Both 1 and 2
4	Ready
5	Trigger queue is full (don't send any more triggers)
6	No input channel lock
7-15	spare

Table 11: Meaning of 4-bit status code

of 4 pixels each. The content descriptions below are before scrambling (or after descrambling). The 4 pixels are always in the same chip row with increasing column numbers. Thus the address of each group is given by a 4-column and a row. In header mode there are event header frames in addition to data frames. These header frames can be Aurora data frames (start with 01 64b/66b header) rather than Aurora command frames, which is convenient when using mutilane output, and is the scheme described below. However, it may be necessary to use Aurora command frames for example when including data compression. The reason that using only Aurora data frames is convenient for multilane output is that, otherwise, because of strict alignment, whenever an Aurora command frame is sent it must be sent in all lanes, thereby quadrupling the event header overhead in addition to requiring padding of events for all lanes to "catch up". No such added overhead or padding are needed if the event header is just another data frame as far as Aurora is concerned.

Header Mode All records are exactly 32 bits each. A header begins with 1111111. The remaining 25 bits contain the 5-bit trigger tag that was supplied by the user, the 5-bit trigger ID from the pending triggers queue inside the chip, and the bunch counter value:

[1111111] [trigger tag (5 bits)] [trigger ID (5 bits)] [BCID counter value (15 bits)]

Any other value of the first 7 bits indicates hit data for 4 pixels, which consists of:

[4-column (7 bits)] [row (9 bits)] 4× [ADC (4-bits)]

There is a 4-bit ADC value for each pixel. Note that the 4-column valid addresses range from 0 to 99 decimal. Similarly the valid row addresses range from 0 to 383 decimal. Special row addresses could in the future be used to identify other types of information, such as high precision charge values from bottom of column measurement.

Tag Mode There is only one record type, 64 bits long. Each record contains a 7 bit trigger tag, and two hit subrecords, the first is 32 bits long and the second 25 bits long.
The 32-bit subrecord consists of:

[4-column (7 bits)] [row (9 bits)] 4× [ADC (4-bits)]

The 25-bit subrecord consists of:

[4-column offset (4 bits)] [row offset (5 bits)] 4× [ADC (4-bits)]

Where the 4-column and row offsets give the address relative to the first subrecord. In case the first subrecord contains the final hits for the given trigger tag, then all 25 bits of the second subrecord will be zero. Also if there is no hit within range of the relative addresses possible, then all 25 bits of the second subrecord can also be zero.

10. Test Features and Miscellaneous Functions

10.1 Heartbeat and Fixed Output Patterns

The heartbeat function allows the output data ports to be active after power is applied, even without any clock and command provided. The CDR block will generate an serializer clock of approximately 1.28 Gbps even is no signal to lock to is supplied to the PLL. The serializer will then produce AURORA idle frames as well as service frames as described in Sec. 9. This constitutes a heartbeat that can be detected by the DAQ system.

Special output sequences can also be selected instead the AURORA output protocol. These include a simple clock (the pattern 10101010...) as well as the pseudo-random bit sequences PRBS-7 and PRBS-31. These are selected through a configuration register.

10.2 Alternate Readout

Output data can be read through the JTAG interface instead of the AURORA encoded high speed outputs. This is of course much slower and can therefore only be used with very low hit rate and trigger rate, but can be useful for debugging purposes. The data are retrieved before any AURORA encoding.

10.3 Boundary Scan

A limited boundary scan chain is implemented n RD53A. This allows structural testing of the global registers. Since the registers are triple redundant it is not possible to detect single failures without a boundary scan.

10.4 General Purpose LVDS Outputs and Hit-OR

RD53A has four LVDS outputs that can be used to monitor or extract internal switching signals. Which signals are connected to these outputs is selected via a configuration register as shown in Table 25. The default assignment is the four Hit-OR nets.

10.5 Analog Muxes

Two analog multiplexers, one current and one voltage, are included to measure DC levels and current of interest. Each MUX has its output available on a wire bond pad for use with external instruments, and is also routed to the generic ADC for monitoring. The current MUX output has an internal resistor to ground

10.6 ADC and Temperature/Radiation Sensors

10.7 Ring Oscillators

Eight ring oscillators are included for radiation testing. Each one is built with using a different logic cell (Table 28). The number of cells in each ring was chosen to have a simulated frequency close to 1 GHz in the typical process corner and at 1.2 V VDDD. Each oscillator drives a 12-bit counter as shown in Fig. 53, while a 4-bit counter is used to count the number of start/stop pulses received. The counters will count while start/stop is high. The start/stop pulses are supplied using the Global Pulse command. The use of Hit OR as start/stop to obtain a precision ToT measurement

is not implemented in RD53A. Nor is the use of Hit OR to start and Aurora command frame to stop in order to obtain a precision leading edge time measurement. The counters are reset by writing to the corresponding register address. Because the ring oscillator frequency depends on voltage and temperature differently from radiation for each logic cell used, with suitable calibration the ring oscillators could be used to measure temperature and VDDD voltage as well as radiation.

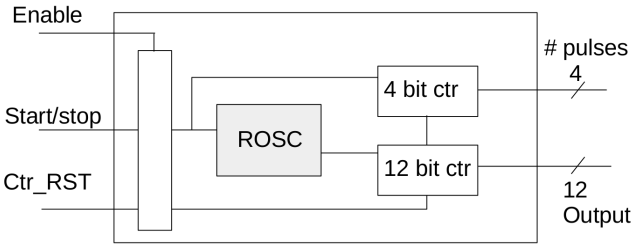


Figure 53: Diagram of a ring oscillator block. There are 8 such blocks differing only in the number and type of the ring oscillator logic cells.

10.8 Top Pads and Pixel Outputs

The pads at the top of the chip (Table 21) provide direct access to test outputs as well as Cap Measure circuits (Sec. 10.9). The charge sensitive amplifier (CSA) output for the Synchronous and the Linear analog front-ends is available for 3 pairs of adjacent top row pixels in each front-end. The preamplifier output for these pixels is connected to a PMOS source follower by means of an enabling switch controlled by a dedicated pad, as shown in fig. 54. In the example shown in the figure, the source follower outputs are connected to pad 34 and 35 (*Lin_SF_OUT6*, *Lin_SF_OUT5* respectively) enabled by the signal provided to the pad 36 (*Lin_OE56*). An external pull-up resistor, R_{sf} (to 1.2V) has to be connected to each source follower output pad. The value of this resistor for the different front-ends is reported in table 21. The source follower is able to properly drive a maximum load, C_p, of 10 pF. 9 analog outputs are available for the Differential analog front-end.

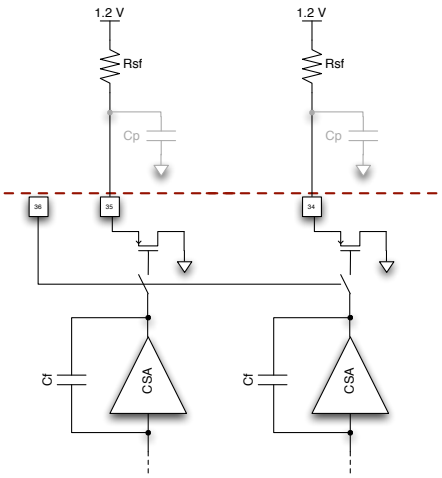


Figure 54: Source follower connected at the preamplifier output of the analog front-end.

The top pad frame includes a number of pads conceived for voltage drop monitoring on power supply lines. They are internally connected to the vertical power distribution bars at the top of the matrix. These pads are available for the three front-end integrated in RD53A both for the analog and the digital power supplies. Monitoring pads labeled as *_WC_* refer to those pads connected in regions of the matrix where we expect the worst case conditions in terms of voltage drop. For example, *VDDA_LBL_MON* can be exploited to monitor the VDDA power supply for the Differential analog front-end, while *WC_VDDA_LBL_MON* provides an estimate of the maximum drop for the VDDA voltage for the same front-end.

10.9 Capacitance Measurement Circuits

The capacitance measurement circuits allow determination of the as-built front-end injection capacitor values. They are integrated in the right and left sides of the top pad frame, and are labeled as *_DX* and *_SX*. The circuits are shown in fig 55, called *cap measure* and *parasitic cap measure*. The cap measure circuit is connected to a parallel array of 100 MoM capacitors (C_{test}), each identical to the injection capacitors in the analog front-ends, whose nominal value is 8.5 fF. The parasitic cap measure circuit embodies a replica of the previous circuit without the array, and allows evaluation and subtraction the parasitic capacitance associated with the measurement circuit itself (C_p). These circuits include a charge pump with PMOS and NMOS transistors controlled by separate externally supplied clocks. These clock signals must be non-overlapping. The circuit has been simulated up to a clock frequency f_{CLK}=10 MHz. In simulation, the average current flowing in VDD_CAP of the cap measure circuit is $I_{avg,cap}$= 11 μA for f_{CLK}=10 MHz, while the average current flowing in VDD_CAP of the parasitic cap measure is $I_{avg,pcap}$= 1 μA. The average value for one single capacitor emulating the analog front-end injection capacitance can be obtained by means of:

$$C_{inj} = \frac{1}{100}\frac{I_{avg,cap} - I_{avg,pcap}}{f_{CLK}\text{VDD\_CAP}}, \tag{10.1}$$

where VDD_CAP is supplied with 1.2 V.

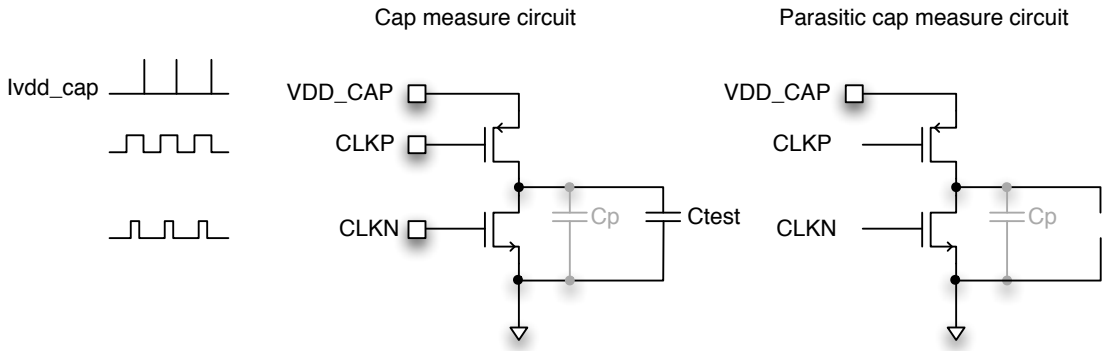


Figure 55: Capacitance measurement circuits.

An R=10 Ω and C=10 μF external RC filter can be used to measure of the average current flowing in the *VDD_CAP* pad.

10.10 Self-Trigger

A simple self-trigger logic is implemented identically for each of the four Hit-OR nets and the outputs selected and merged as shown in Fig 56. The delay chain ensures that a trigger is not sent to the pixel matrix before the maximum possible ToT count has been reached. Therefore the self trigger has a fixed latency of 16 bunch crossings. The synchronous delay chain also ensures that runt pulses in the Hit-OR do not produce a trigger. One trigger pulse will be generated per Hit-OR rising edge as the edge propagates through the final flip-flop.

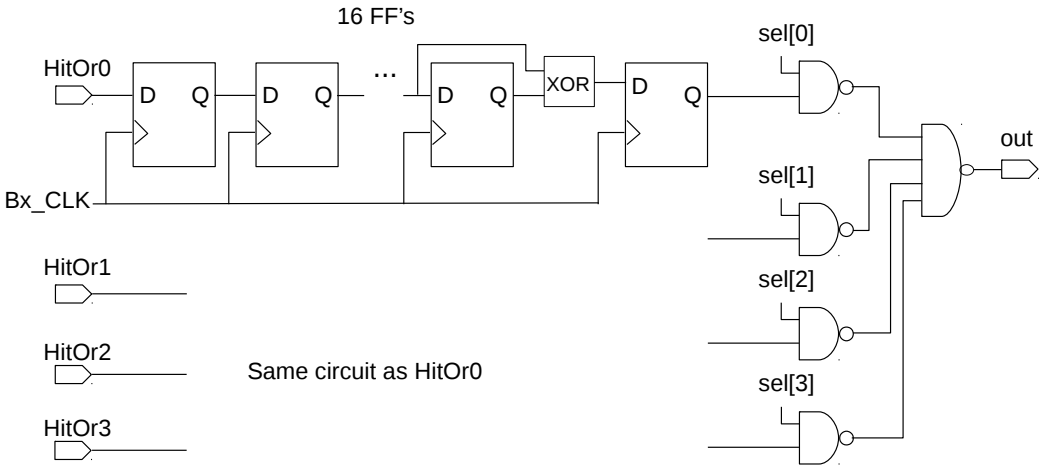


Figure 56: Schematic of the self trigger logic.

11. Reference Tables

This section collects the tables of pin and register assignments that will be needed as quick references during debugging and operation, along with important notes. By placing them all together it will make them easier to find/use this way without having to scroll through the whole document each time.

11.1 Bottom Pinout

An overview of the pin organization along with special pin assignments are given in Table 12, while individual pins each of the main blocks are listed in the subsequent tables, all referenced from Table 12.

Pin Numbers	Count	Group or Pin Description	Comments
1	1	sensor guard ring bumps	isolated from silicon, no ESD prot.
2-13	13	analog I/O	see table 19
14	1	substrate contact	also ESD bus low
15-28	14	analog power port	see table 13
29-42	14	digital power port	see table 14
43	1	external POR capacitor	5 nF suggested
44	1	POR out/in	low = reset
45-54	10	CMOS I/O (1/2)	see table 20
55-59	5	analog ShuLDO control	see table 15
60-73	14	analog power port	see table 13
74-87	14	digital power port	see table 14
88	1	substrate contact	ESD?
89-111	23	I/O block	see table 17
112-125	14	analog power port	see table 13
136-139	14	digital power port	see table 14
140-144	5	digital ShuLDO control	see table 16
145-146	2	Additional LVDS outout (p,n)	hard wired to logic 0
147	1	backup POR input for bandgap references	PD, use only in case of problems
148	1	disable ShuLDO compensation	PD, expert use
149-150	2	part of I/O block	see table 17
151-155	7	spares	reserved for 5 Gbps output
156-169	14	analog power port	see table 13
170-183	14	digital power port	see table 14
184	1	substrate contact	also ESD bus low
185-192	8	auxiliary I/O	see table 18
193-197	5	CMOS I/O (2/2)	see table 20
198	1	sensor guard ring bumps	isolated from silicon, no ESD prot.

Table 12: Bottom wire bonding pad pinout overview. See referenced tables for pins within a group. PD means internal pull down.

Pin	Net	Description
0	GNDA	internal analog ground
1	GNDA	internal analog ground
2	GNDA	internal analog ground
3	GNDA	internal analog ground
4	GNDA	internal analog ground
5	VDDA	internal analog power rail
6	VDDA	internal analog power rail
7	VDDA	internal analog power rail
8	VDDA	internal analog power rail
9	VDDA	internal analog power rail
10	VINA	constant current supply input
11	VINA	constant current supply input
12	VINA	constant current supply input
13	VINA	constant current supply input

Table 13: Pin assignments of analog power port. Pin numbers are relative to first (leftmost) pin.

Pin	Net	Description
0	VIND	constant current supply input
1	VIND	constant current supply input
2	VIND	constant current supply input
3	VIND	constant current supply input
4	VDDD	internal digital power rail
5	VDDD	internal digital power rail
6	VDDD	internal digital power rail
7	VDDD	internal digital power rail
8	VDDD	internal digital power rail
9	GNDD	internal digital ground
10	GNDD	internal digital ground
11	GNDD	internal digital ground
12	GNDD	internal digital ground
13	GNDD	internal digital ground

Table 14: Pin assignments of digital power port. Pin numbers are relative to first (leftmost) pin.

Pin	Net	connect to	Description
55	SLDO_IOFFSETA	resistor and cap to GND	offset voltage set and decouple, or override
56	SLDO_VDDSHUNTA	VINA	power and enable the shunt function
57	SLDO_REXTA	res. to VINA	to use external current setting resistor
58	SLDO_RINA	VINA	to use internal current setting resistor
59	SLDO_VREFA	cap. to GNDA	V-ref. decouple or override

Table 15: Pin assignments of analog ShuLDO regulator control.

Pin	Net	connect to	Description
140	SLDO_VREFD	cap. to GNDD	V-ref. decouple or override
141	SLDO_RIND	VIND	to use internal current setting resistor
142	SLDO_REXTD	res. to VIND	to use external current setting resistor
143	SLDO_VDDSHUNTD	VIND	power and enable the shunt function
144	SLDO_IOFFSETD	resistor and cap to GND	offset voltage set and decouple, or override

Table 16: Pin assignments of digital ShuLDO regulator control.

Pin	Net	connect to	Description
89-90	EXT_CMD_CLK	option	LVDS command clock in for CDR bypass mode
91-92	CMD	DAQ	LVDS serial command stream input
93	VDD_PLL	regulated VDD	PLL power, try VDDD, VDDA, or ext. supply
94	GND_PLL	ground	PLL ground
95-96	EXT_SER_CLK	option	LVDS serializer clock in
97	VDD_CML	regulated VDD	CML driver pwr, try VDDD, VDDA, or ext. supply
98	GND_CML	ground	CML driver ground
99-100	GTX3	DAQ	CML AURORA data lane 3 out
101	GND_CML	ground	CML driver ground
102-103	GTX2	DAQ	CML AURORA data lane 2 out
104	GND_CML	ground	CML driver ground
105-106	GTX1	DAQ	CML AURORA data lane 1 out
107	GND_CML	ground	CML driver ground
108-109	GTX0	DAQ	CML AURORA data lane 0 out
110	GND_CML	ground	CML driver ground
111	VDD_CML	regulated VDD	CML driver pwr, try VDDD, VDDA, or ext. supply
149	PLL_RST_B	POR	Active low reset of PLL
150	PLL_V_VCO	Voltage/nc (PU)	Oscillator ctrl. V while in reset state

Table 17: Pin assignments of main I/O pad group. All differential pairs are ordered non inverting first (lower pin number) and inverting second (higher pin number). (PU) means the pad has an internal pull-up resistor and "nc" beans no connection.

Pin	Net	connect to	Description
185-186	AUX_LVDS_0 (p,n)	test DAQ / scope	LVDS output 0
187-188	AUX_LVDS_1 (p,n)	test DAQ / scope	LVDS output 1
189-190	AUX_LVDS_2 (p,n)	test DAQ / scope	LVDS output 2
191-192	AUX_LVDS_3 (p,n)	test DAQ / scope	LVDS output 3

Table 18: Pin assignments of auxiliary I/O pad group.

Pin	Net	connect to	Description
2	VREF_ADC_IN	VREF_ADC_OUT	ADC reference input
3	VREF_ADC_OUT	VREF_ADC_IN	ADC reference output
4	IMUX_OUT	ammeter	output from current multiplexer
5	VMUX_OUT	voltmeter	output from voltage multiplexer
6	VINJ_HI	option	external high level injection voltage
7	VINJ_MID	option	external mid level injection voltage
8	IREF_IN	IREF_OUT	main ref. current output- mandatory to connect
9	IREF_OUT	IREF_IN	main ref. current input- mandatory to connect
10	IREF_TRIM0	VDDA if needed	trim b0 of the main current ref.
11	IREF_TRIM1	VDDA if needed	trim b1 of the main current ref.
12	IREF_TRIM2	VDDA if needed	trim b2 of the main current ref.
13	IREF_TRIM3	VDDA if needed	trim b3 of the main current ref.

Table 19: Pin assignments of analog I/O pad group

Pin	Net	connect to	Description
45	INJ_STRB0	DAQ/nc	direct control of CAL_edge signal
46	INJ_STRB1	DAQ/nc	direct control of CAL_aux signal
47	TDI	JTAG/nc	data input
48	TCK	JTAG/nc	clock input
49	TMS	JTAG/nc	TMS input
50	TDO	JTAG/nc	data output
51	TRST_B	JTAG/nc (PD)	reset input, active low
52	CHIP_ID0	VDDD/nc (PD)	chip address bit 0
53	CHIP_ID1	VDDD/nc (PD)	chip address bit 1
54	CHIP_ID2	VDDD/nc (PD)	chip address bit 2
193	DEBUG_EN	VDDD/nc (PD)	Enable JTAG interface
194	BYPASS_CMD	VDDD/nc (PD)	bypass the command decoder
195	BYPASS_CDR	VDDD/nc (PD)	bypass clock and data recovery
196	TRIG_IN	DAQ/nc (PD)	direct trigger signal input
197	STATUS	DAQ/nc	Status output flag

Table 20: Pin assignments of CMOS I/O pad group. Note this consists of two physically separated subgroups. (PD) means the pad has an internal pull-down resistor and "nc" beans no connection. Except for the chip ID, none of these pads would be connected in a module.

11.2 Top Pinout

Pin	PAD Name	Purpose	Notes
T1(R)	VDD_TOP	VDD power pad for ESD / output buffers	
T2	GND_TOP	GND power pad for ESD / output buffers	
T3	VDD_CAP_DX	VDD for the cap measure	I=11 μA nom. at 10 MHz
T4	VDD_PCAP_DX	VDD for the parasitic cap measure	I=1 μA nom. at 10 MHz
T5	CLKN_DX	Clock for NMOS in the cap measure	non overlapping 10 MHz max.
T6	CLKP_DX	Clock for PMOS in the cap measure	non overlapping 10 MHz max.
T7-T8	NC		
T9	GNDD_Diff_MON	GNDD monitor for Diff	
T10	NC		

T11	VDDD_Diff_MON	VDDD monitor for Diff	
T12	Diff_OUT2_30	Diff analog output 2_30	
T13	Diff_OUT2B_30	Diff analog output 2B_30	
T14	Diff_OUT1_30	Diff analog output 1_30	
T15-T17	NC		
T18	VDDA_Diff_MON	VDDA monitor for Diff	
T19	NC		
T20	GNDD_WC_Diff_MON	Worst case GNDD monitor for Diff	
T21	GNDA_Diff_MON	GNDA monitor for Diff	
T22	Diff_OUT2_20	Diff analog output 2_20	
T23	Diff_OUT2B_20	Diff analog output 2B_20	
T24	Diff_OUT1_20	Diff analog output 1_20	
T25	VDDD_WC_Diff_MON	Worst case VDDD monitor for Diff	
T26-T29	NC		
T29	VDDA_WC_Diff_MON	Worst case VDDA monitor for Diff	
T30	GNDA_WC_Diff_MON	Worst case GNDA monitor for Diff	
T31	Diff_OUT1_10	Diff analog output 1_10	
T32	Diff_OUT2B_10	Diff analog output 2B_10	
T33	Diff_OUT2_10	Diff analog output 2_10	
T34	Lin_SF_OUT6	Lin source follower out 6	To 1 kΩ to 1.2 V. 10 pF max. load
T35	Lin_SF_OUT5	Lin source follower out 5	To 1 kΩ to 1.2 V. 10 pF max. load
T36	Lin_OE56	Lin source follower 5/6 output enable	Int. pull-down
T37-T39	NC		
T40	VDDA_Lin_MON	VDDA monitor for Lin	
T41	NC		
T42	GNDA_Lin_MON	GNDA monitor for Lin	
T43	NC		
T44	GNDD_WC_Lin_MON	Worst case GNDD monitor for Lin	
T45	NC		
T46	VDDD_WC_Lin_MON	Worst case VDDD monitor for Lin	
T47	Lin_OE34	Lin source follower 3/4 output enable	Active H, Defualt L
T48	Lin_SF_OUT4	Lin source follower output4	To 1 kΩ to 1.2 V. 10 pF max. load
T49	Lin_SF_OUT3	Lin source follower output3	To 1 kΩ to 1.2 V. 10 pF max. load
T50	VDD_TOP	VDD power pad for ESD / output buffers	
T51	GND_TOP	GND power pad for ESD / output buffers	
T52	NC		
T53	VDDA_WC_Lin_MON	Worst case VDDA monitor for Lin	
T54	NC		
T55	GNDA_WC_Lin_MON	Worst case GNDA monitor for Lin	
T56	NC		
T57	GNDD_Lin_MON	GNDD monitor for Lin	
T58-T59	NC		
T60	VDDD_Lin_MON	VDDD monitor for Lin	
T61-T62	NC		
T63	Lin_SF_OUT	Lin test source follower output	To 1kΩ to 1.2 V. 10 pF max. load
T64	Lin_SF_IN	Lin test source follower input	used to measure gain
T65	Lin_OE12	Lin source follower 1/2 output enable	Active H, Defualt L
T66	Lin_SF_OUT2	Lin source follower output2	To 1 kΩ to 1.2 V. 10 pF max. load
T67	Lin_SF_OUT1	Lin source follower output1	To 1 kΩ to 1.2 V. 10 pF max. load
T68	NC		
T69	GNDD_WC_Sync_MON	Worst case GNDD monitor for Lin	
T70	NC		
T71	VDDD_WC_Sync_MON	Worst case VDDD monitor for Sync	
T72	Sync_SF_OUT6	Sync source follower output6	To 2.5 kΩ to 1.2V. 10 pF max. load
T73	Sync_SF_OUT5	Sync source follower output5	To 2.5 kΩ to 1.2V. 10 pF max. load
T74	Sync_OE56	Sync source follower 5/6 output enable	Active H, Defualt L
T75	Sync_SF_OUT	Sync test source follower output	To 2.5 kΩ to 1.2V. 10 pF max. load

T76	Sync_SF_IN	Sync test source follower input	used to measure gain
T77	NC		
T78	VDDA_WC_Sync_MON	Worst case VDDA monitor for Sync	
T79	GNDD_Sync_MON	GNDD monitor for Sync	
T80	GNDA_WC_Sync_MON	Worst case GNDA monitor for Sync	
T81	NC		
T82	VDDD_Sync_MON	VDDD monitor for Sync	
T83	Sync_OE34	Sync source follower 3/4 output enable	Active H, Defualt L
T84	Sync_SF_OUT4	Sync source follower output4	To 2.5 kΩ to 1.2V. 10 pF max. load
T85	Sync_SF_OUT3	Sync source follower output3	To 2.5 kΩ to 1.2V. 10 pF max. load
T86	NC		
T87	VDDA_Sync_MON	VDDA monitor for Sync	
T88	Sync_OE12	Sync source follower 1/2 output enable	Active H, Defualt L
T89	Sync_SF_OUT2	Sync source follower output2	To 2.5 kΩ to 1.2V. 10 pF max. load
T90	Sync_SF_OUT1	Sync source follower output1	To 2.5 kΩ to 1.2V. 10 pF max. load
T91	GNDA_Sync_MON	GNDA monitor for Sync	
T92	CLKP_SX	Clock for PMOS in the cap measure	non overlapping 10 MHz max.
T93	CLKN_SX	Clock for NMOS in the cap measure	non overlapping 10 MHz max.
T94	VDD_PCAP_SX	VDD for the parasitic cap measure	I=1 μA nom. at 10 MHz
T95	VDD_CAP_SX	VDD for the cap measure	I=11 μA nom. at 10 MHz
T96	VDD_TOP	VDD power pad for ESD / output buffers	
T97(L)	GND_TOP	GND power pad for ESD / output buffers	

Table 21: RD53A Top PAD Frame pinout

11.3 Global Register Assignments

The full list of global register assignments is given in Tables 22, 23, while the further notes on specific registers as needed are given in Sec. 11.4. Recall that not all global registers contain configuration. Some global registers counter values, monitoring outputs such as ADC values, etc. (see Sec. 7.5). All registers are 16 bits wide, but in most cases fewer than 16 bits are actually used- the used bits are indicated by the bit maps in the tables. Unused bits are still implemented as memory and can be written and read, but serve no function within the chip.

Address	Name	Meaning	Bit Map	Default	Sec.
		Pixel Section			
0	PIX_PORTAL	Virtual register to access pixel matrix	[F:8,7:0]	n/a	9.3
1	REGION_COL	Set value for which pixel 2-column to access	[7:0]	0	9.3
2	REGION_ROW	Set value for which pixel row to access	[8:0]	0	9.3
3	PIX_MODE, B_MASK	mode bits, front end flavor mask	[5:3,2:0]	0	9.3
4	PIX_DEFAULT_CONFIG	hex 9ce2 selects pixel default configuration	[F:0]	hex 9ce2	7.2
		Synchronous Front End			
5	IBIASP1_SYNC	Cascode main branch bias current	[8:0]	100	5.3
6	IBIASP2_SYNC	Input device main bias current	[8:0]	150	5.3
7	IBIAS_SF_SYNC	Follower bias current	[8:0]	100	5.3
8	IBIAS_KRUM_SYNC	Krummenacher feedback bias current	[8:0]	140	5.3
9	IBIAS_DISC_SYNC	Comparator bias current	[8:0]	200	5.3
10	ICTRL_SYNCT_SYNC	Oscillator bias current	[9:0]	100	5.3
11	VBL_SYNC	Baseline voltage for offset compensation	[9:0]	450	5.3
12	VTH_SYNC	Discriminator threshold voltage	[9:0]	300	5.3
13	VREF_KRUM_SYNC	Krummenacher voltage reference	[9:0]	490	5.3
*** 30	CONF_FE_SYNC	Auto-zero, SelC2F, SelC4F, Fast ToT	[3:0]	0,1,0,0	5.3
		Linear Front End			
14	PA_IN_BIAS_LIN	Preamp input branch bias current	[8:0]	300	5.2
15	FC_BIAS_LIN	Folded cascode branch current	[7:0]	20	5.2
16	KRUM_CURR_LIN	Krummenacher feedback bias current	[8:0]	50	5.2
17	LDAC_LIN	Fine threshold voltage	[9:0]	80	5.2
18	COMP_LIN	Comparator bias current	[8:0]	110	5.2
19	REF_KRUM_LIN	Krummenacher voltage reference	[9:0]	300	5.2
20	Vthreshold_LIN	Global threshold voltage	[9:0]	408	5.2
		Differential Front End			
21	PRMP_DIFF	Preamp input stage bias current	[9:0]	533	5.1
22	FOL_DIFF	Preamp follower bias current	[9:0]	542	5.1
23	PRECOMP_DIFF	Precomparator tail current	[9:0]	551	5.1
24	COMP_DIFF	Comparator bias current	[9:0]	528	5.1
25	VFF_DIFF	Preamp feedback (discharge) current	[9:0]	164	5.1
26	VTH1_DIFF	Negative branch threshold offset V (vth1)	[9:0]	1023	5.1
27	VTH2_DIFF	Positive branch threshold offset V (vth2)	[9:0]	0	5.1
28	LCC_DIFF	Leakage current compensation (LCC) bias	[9:0]	20	5.1
29	CONF_FE_DIFF	LCC enable, add feedback cap	[1:0]	1,0	5.1
		Power Supply Rails			
31	SLDO_ANALOG_TRIM	Analog regulator output voltage trim	[9:5]	16	3
	SLDO_DIGITAL_TRIM	Digital regulator output voltage trim	[4:0]	16	3
		Digital Matrix			
32	EN_CORE_COL_SYNC	Enable columns of cores w/ Sync. front end	[F:0]	hex FFFF	6
33	EN_CORE_COL_LIN_1	Enable columns of cores w/ Linear front end	[F:0]	hex FFFF	6
34	EN_CORE_COL_LIN_2	Enable last column of cores w/ Linear front end	[0]	1	6
35	EN_CORE_COL_DIFF_1	Enable columns of cores w/ Diff. front end	[F:0]	hex FF	6
36	EN_CORE_COL_DIFF_2	Enable last column of cores w/ Diff. front end	[0]	1	6
37	LATENCY_CONFIG	Trigger latency value	[8:0]	500	7.4
38	WR_SYNC_DELAY_SYNC	Write synchronization delay for sync. front end	[4:0]	16	5.3
		Injection			
39	INJ_MODE_DEL	enable digital, analog mode, fine delay	[5,4,3:0]	0,1,0	5.4
41	VCAL_HIGH	High injection voltage	[B:0]	500	5.4
42	VCAL_MED	Medium injection voltage	[B:0]	300	5.4
46-49	CAL_COLPR_SYNC	Enable CAL for 64 Sync. col. pairs	4x[F:0]	4x hex FFFF	
50-53	CAL_COLPR_LIN	Enable CAL for 64 Linear col. pairs	4x[F:0]	4x hex FFFF	
54	CAL_COLPR_LIN5	Enable CAL for last 4 Linear col. pairs	[3:0]	hex F	
55-58	CAL_COLPR_DIFF	Enable CAL for 64 Diff. col. pairs	4x[F:0]	4x hex FFFF	
59	CAL_COLPR_DIFF5	Enable CAL for last 4 Diff. col. pairs	[3:0]	hex F	

Table 22: Global register assignment table 1 of 2. Address are decimal, bit maps are hex, defaults are decimal unless otherwise specified.*** Indicates address is listed out of numerical order.

Address	Name	Meaning	Bit Map	Default	Sec.
Digital Functions					
40	CLK_DATA_DELAY	2INV delay sel., delays for clock, command	[8,7:4,3:0]	0,0,0	7.1
43	CH_SYNC_CONF	Channel synchronizer phase, lock, unlock thresh.	[9:8,7:4,3:0]	0,16,8	9.2.1
44	GLOBAL_PULSE_RT	Selects routing of global pulse signal	[F:0]	0	11.4.1
I/O					
60	DEBUG_CONFIG	EnableExtCal, EnablePRBS	[1:0]	0,0	
61	OUTPUT_CONFIG	Data delay, Ser. type, Active lanes, Output fmt	[8:7,6,5:2,1:0]	0,0,1,0	11.4.2
62	OUT_PAD_CONFIG	Enable/configure various outputs including JTAG	[D:0]	11.4.3	11.4.3
63	GP_LVDS_ROUTE	Select signals connected to LVDS outputs	[2:0]	0	11.4.4
64	CDR_CONFIG	Various configuration values for CDR	[D:0]	11.4.5	11.4.5
65	VCO_BUFF_BIAS	Bias current for VCO buffer of CDR	[9:0]	400	
66	CDR_CP_IBIAS	Bias current for CP of CDR	[9:0]	50	
67	VCO_IBIAS	Bias current for VCO	[9:0]	500	
68	SER_SEL_OUT	Sel. serializer for output 3,2,1,0	[7:6,5:4,3:2,1:0]	1,1,1,1	9.1
69	CML_CONFIG	SER_INV_TAP (2 bits), SER_EN_TAP (2 bits) enable CMLs (4 bits)	[7:6,5:4] [3:0]	0,3, hex F	9.1
70-72	CML_TAP_BIAS	CML driver pre-emphasis for taps 0-2	3×[9:0]	500,0,0	9.1
73	AURORA_CC_CFG	Aurora values for CCwait and CCsend	[7:2,1:0]	25,3	
74	AURORA_CB_CFG0	Aurora values for CBwait bits 3:0 and CBsend	[7:4,3:0]	15,0	
75	AURORA_CB_CFG1	Aurora values for CBwait bits 19:4	[F:0]	15	
76	AURORA_INIT_WAIT	Time to wait for channel bonding	[A:0]	32	
***45	MON_FRAME_SKIP	Interval between register/service/mon. frames	[7:0]	50	9.4
***101-102	AUTO_READ_A0,B0	Addresses of lane 0 registers for auto-read	2x[8:0]	136,130	9.4
***103-104	AUTO_READ_A1,B1	Addresses of lane 1 registers for auto-read	2x[8:0]	118,119	9.4
***105-106	AUTO_READ_A2,B2	Addresses of lane 2 registers for auto-read	2x[8:0]	120,121	9.4
***107-108	AUTO_READ_A3,B3	Addresses of lane 3 registers for auto-read	2x[8:0]	122,123	9.4
Test and Monitoring Functions					
77	MONITOR_MUX	Monitor enable, I_Mon mux, V_Mon mux	[D,C:7,6:0]	0,63,127	8
78-81	HITOR_MASK_SYNC	Disable 4 Hit-ORs for the 16 Sync. FE cores	4×[F:0]	4×0	6.3
82,84,86,88	HITOR_MASK_LIN1	Disable 4 Hit-ORs for 16 Linear FE cores	4×[F:0]	4×0	6.3
83,85,87,89	HITOR_MASK_LIN2	Disable 4 Hit-ORs for 17th Linear FE core	4×[0]	4×0	6.3
90,92,94,96	HITOR_MASK_DIFF1	Disable 4 Hit-ORs for 16 Diff. FE cores	4×[F:0]	4×0	6.3
91,93,95,97	HITOR_MASK_DIFF2	Disable 4 Hit-ORs for 17th Diff. FE core	4×[0]	4×0	6.3
98	ADC_CONFIG	Bandgap ref. trim, ADC trim	[A:6,0:5]	0,0	
99-100	SENSOR_CONFIG	Temperature sensor configuration (2 registers)	2×[B:0]	11.4.7	11.4.7
109	RING_OSC_ENABLE	Enable bits for 8 ring oscillators	[7:0]	0	10.7
110-117	RING_OSC	Read-only (write resets) counters for ring osc.	8×[F:0]	n/a	11.4.8
118	BC_CTR	Bunch crossing counter. Rd only (write resets)	[F:0]	n/a	
119	TRIG_CTR	Trigger counter. Rd only (write resets)	[F:0]	n/a	
120	LCK_LOSS_CTR	Loss of lock ctr. on input stream (write resets)	[F:0]	n/a	
121	BFLIP_WARN_CTR	Command dec. bit flip warning ctr. (write resets)	[F:0]	n/a	
122	BFLIP_ERR_CTR	Command dec. bit flip error ctr. (write resets)	[F:0]	n/a	
123	CMD_ERR_CTR	Command dec. other error ctr. (write resets)	[F:0]	n/a	
124-127	FIFO_FULL_CTR	8-bit counters of FIFO full conditions	4×[F:8,7:0]	n/a	11.4.9
128	AI_PIX_COL	Rd. only value of auto-incremented pixel colpr	[7:0]	n/a	9.3
129	AI_PIX_ROW	Rd. only value of auto-incremented pixel row	[8:0]	n/a	9.3
130-133	HitOr_Cnt	16-bit rd. only counters of Hit ORs (write resets)	4×[F:0]	n/a	
134	SKP_TRIG_CNT	16-bit rd. only cntr. of skipped trigs (wrt. resets)	[F:0]	n/a	
135	ERR_MASK	Disable selected error/warning messages	[D:0]	0	
136	ADC_READ	read only ADC value. Must first run ADC	[B:0]	n/a	
137	SELF_TRIG_EN	Selects which Hit OR's generate a trigger	[3:0]	0	10.10

Table 23: Global register assignment table 2 of 2. Address are decimal, bit maps are hex, defaults are decimal unless otherwise specified. *** Indicates address is listed out of numerical order.

11.4 Notes on Specific Registers

11.4.1 GLOBAL_PULSE_RT

Reg. Bit	Pulse destination	Comment
[0]	Reset Channel Synchronizer	
[1]	Reset Command Decoder	
[2]	Reset Global Configuration	reverts to defaults
[3]	Clear Monitor Data	
[4]	Reset Aurora	triggers channel bonding seq.
[5]	Reset Serializer	
[6]	Reset ADC	
[7]	unused	
[8]	Start Monitoring	needed in RD53A
[9]	unused	
[A]	unused	
[B]	unused	
[C]	Trigger ADC to start conversion	
[D]	Activate Ring Oscillators	they run while high, stop while low
[E]	Reset Auto Zeroing of Sync. FE's	
[F]	Acquire Zero level in Sync. FE's	

Table 24: Routing of the Global Pulse signal. All bits are zero in default configuration.

11.4.2 OUTPUT_CONFIG

DataReadDelay[8:7]: Number of 40 MHz clocks +2 for data transfer out of pixel matrix. Default 0 means 2 clocks. May need higher value in case of large propagation delays, for example at low VDDD voltage after irradiation.

SelSerializerType[6]: Default 0 for RD53A serializer (9.1). None other implemented so value 1 is meaningless.

ActiveLanes[5:2]: Aurora lanes. Default 0001 means single lane mode on lane 0.

OutputFormat[1:0]: Default 0 is header mode. 1 is tag mode. 2 is both header and tag. No other modes implemented (9.1)

11.4.3 OUT_PAD_CONFIG

JTAG_TDO[D]: (default 0 = off)

STATUS_EN[C]: (default 1 = on)

STATUS_DS[B]: (decault 0)

$\overline{\text{LANE0\_LVDS}}$**[A]:** (default 1 = off)

LANE0_LVDS_BIAS[9:7]: (default 0)

$\overline{\text{GP\_LVDS\_EN}}$**[6:3]:** (default 0000 = all on)

GP_LVDS_BIAS[2:0]: (default 4)

11.4.4 GP_LVDS_ROUTE

3-bit code	LVDS_0	LVDS_1	LVDS_2	LVDS_3
001	CMD data	160 MHz clock	40 MHz clock	Matrix reset
010	Cal. edge	Cal. aux	Trig. to matrix	Hit-OR_3
011	1.28 GHz clock	640 MHz clock	160 MHz bef. del.	160 MHz delayed
111	0001 pattern	0001 pattern	0001 pattern	0001 pattern
other	Hit-OR_0	Hit-OR_1	Hit-OR_2	Hit-OR_3

Table 25: Selection of internal signals routed to LVDS outputs. The default selection 3-bit code after power up is 000 (which selects the last row).

11.4.5 CDR_CONFIG

CDR_SEL_DEL_CLK[D]: Clock for delays (default 0 = 640 MHz from PLL) Figs. 35, 36
CDR_PD_SEL[C:B]: (default 0) see Fig. 35
CDR_PD_DEL[A:7]: (default 8) see Fig. 35
CDR_EN_GCK2[6]: (default 0) see Fig. 35
CDR_VCO_GAIN[5:4]: (default 3)see Fig. 35
CDR_SEL_SER_CLK[2:0]: (default 0 = 1.28 GHz from PLL)see Fig. 35

11.4.6 MONITOR_MUX

Value	Selection	Value	Selection	Value	Selection
0	Iref (nominal 4 uA)	9	KRUM_CURR_LIN	18	VTH1_DIFF
1	IBIASP1_SYNC	10	LDAC_LIN	19	VTH2_DIFF
2	IBIASP2_SYNC	11	PA_IN_BIAS_LIN	20	CDR_CP_IBIAS
3	IBIAS_DISC_SYNC	12	COMP_DIFF	21	VCO_BUFF_BIAS
4	IBIAS_SF_SYNC	13	PRECOMP_DIFF	22	VCO_IBIAS
5	ICTRL_SYNCT_SYNC	14	FOL_DIFF	23	CML_TAP_BIAS0
6	IBIAS_KRUM_SYNC	15	PRMP_DIFF	24	CML_TAP_BIAS1
7	COMP_LIN	16	LCC_DIFF	25	CML_TAP_BIAS2
8	FC_BIAS_LIN	17	VFF_DIFF	26-31	not used

Table 26: Analog current multiplexer assignments.

Value	Selection	Value	Selection	Value	Selection
0	ADC bandgap	12	CAL_MED right	24	VOUT Ana. ShuLDO
1	CAL_MED left	13	CAL_HI right	25	VREF Ana. ShuLDO
2	CAL_HI left	14	RADSENS_3	26	VOFF Ana. ShuLDO
3	TEMPSENS_1	15	TEMPSENS_3	27	grounded
4	RADSENS_1	16	REF_KRUM_LIN	28	grounded
5	TEMPSENS_2	17	Vthreshold_LIN	29	VIN Dig. ShuLDO
6	RADSENS_2	18	VTH_SYNC	30	VOUT Dig. ShuLDO
7	TEMPSENS_4	19	VBL_SYNC	31	VREF Dig. ShuLDO
8	RADSENS_4	20	VREF_KRUM_SYNC	32	VOFF Dig. ShuLDO
9	VREF_VDAC	21	VTH_HI_DIFF	33	grounded
10	VOUT_BG	22	VTH_LO_DIFF	34	grounded
11	IMUX output	23	VIN Ana. ShuLDO	35-63	grounded

Table 27: Analog current multiplexer assignments.

11.4.7 SENSOR_CONFIG

99:SENS_ENABLE1[B]: (default 0 = disabled) Enable temp/rad sensors
99:SENS_DEM1[A:7]: (default 0) Dynamic element matching bits
99:SENS_SEL_BIAS1[6]: (default 0) Current bias select
99:SENS_ENABLE0[5]: (default 0 = disabled) Enable temp/rad sensors
99:SENS_DEM0[4:1]: (default 0) Dynamic element matching bits
99:SENS_SEL_BIAS0[0]: (default 0) Current bias select
100 Register 100 is the same as 99 except for sensor 3 instead of 1, and 2 instead of 0.

11.4.8 RING_OSC

Address	Ring oscillator flavor	Len.	Address	Ring oscillator flavor	Len.
110	strength 0 inv. clock driver	55	114	strength 0 4-input NAND	19
111	strength 4 inv. clock driver	51	115	strength 4 4-input NAND	19
112	strength 0 inverter	55	116	strength 0 4-input NOR	19
113	strength 4 inverter	51	117	strength 4 4-input NOR	19

Table 28: Ring oscillator flavors and lengths (in number of gates). Each 16-bit register consists of a 4-bit counter [F:C] which counts how many start-stop cycles took place and a 12-bit counter [B:0] which counts ring oscillator clock cycles. the ring length in number of cells results in a typical frequency of 1 GHz before irradiation.

11.4.9 FIFO Full Counters

Address	Field [F:8]	Field [7:0]	Item Counted
124	Data FIFO 1	Data FIFO 0	Write attempts while FIFO is full
125	Data FIFO 3	Data FIFO 2	Write attempts while FIFO is full
126	Data FIFO 5	Data FIFO 4	Write attempts while FIFO is full
127	Data FIFO 7	Data FIFO 6	Write attempts while FIFO is full

Table 29: Data FIFO full counters

11.5 Pixel Register Assignments

	Front End Flavor		
Bit	**Diff.**	**Lin.**	**Sync.**
0	Pixel Enable	Pixel Power	
1	Injection Enable		
2	Hit-bus Enable		
3	TDAC Sign		0
4	TDAC b0		0
5	TDAC b1		0
6	TDAC b2		0
7	TDAC b3		0

Table 30: Pixel Configuration Map. The Pixel Enable bit enables the digitized output, leaving the analog part unaffected, while the Pixel Power bit, if zero, shuts down power as well as digital output.

A. Radiation Tolerance

RD53A is designed to meet specifications after 500 Mrad total dose in HL-LHC conditions. Operation should still be possible beyond 500 Mrad, but the amount of degradation will have be measured, and will depend on operating conditions such as temperature and annealing. Mitigation of Single Event Upsets (SEU) uses a two pronged approach of conventional hardening plus continuous external refreshing known as *trickle configuration*, allowing both schemes to be evaluated.

A.1 Total Ionizing Dose

A transistor simulation model has been developed that captures the behavior after 500 Mrad dose based on fits to single transistor data. The model uses worst cases bias conditions, but not high temperature annealing.

Analog circuits have been designed following the guidelines shown in Table 31 and then simulated with the 500 Mrad model. In addition all analog circuits used in RD53A have already been prototyped and radiation tested in silicon.

Device	W	L
NMOS	any	≥ 120 nm
PMOS	≥ 300 nm	≥ 120 nm

Table 31: Radiation tolerance guidelines for analog circuits.

A number of digital libraries have been simulated with the 500 Mrad model and a radiation corner model developed that is used in synthesis. In addition, sample cells from these libraries have been tested in silicon and the irradiation results compared to the models. The model is found to be pessimistic, which is expected because it is for worst case bias conditions, while digital circuits roughly spend half of the time under worst case bias. Using this pessimistic model provides margin for digital circuits. In broad terms, 7-track libraries re not used anywhere in RD53A; 9-track libraries are used in most of the pixel matrix, and 12 and 18-track libraries are used in the periphery.

A.2 Single Event Upsets

Mitigation of SEU involves protection of memory and protection of control signals and state machines. Memory in the data path is not protected, because the amount of corruption expected during the time that hit data spends in chip memory (less than 20 μs) is much less than the specified 1% hit loss. With a single bit upset cross section of 5×10^{-14} cm^2 (for a standard flip-flop) and an estimated rate of particles above upset threshold of 500 MHz/cm^2, even taking the extreme case that every single upset corrupts an entire 64-bit Aurora frame, the calculated fraction of lost hits in 20 μs is 3×10^{-8}. Protection is therefore not needed.

Table 32 compares different memory structures avaialble for configuration storage. Global configuration memory is protected with Triple Redundant Memory (TRM) latches withn error correction. Pixel configuration is stored in standard latches. Potential use of DICE latches in the pixels would provide only an order of magnitude smaller upset cross section than standard latches, which

would still lead to a single bit upset rate of 2×10^{-4} Hz. This means that 1The best solution of long-term protection of pixel configuration is therefore trickle configuration, where the chip is no longer required to hold configuration for more than a fraction of a second at a time. The same can of course be applied to global configuration.

Control paths are protected with triple redundancy of circuits such as counters, and/or deglitching of controls signals.

Device	SEU cross section	Estimated MTBU	Circuit area
Standard Latch	2.8×10^{-14} cm^2	50 ms	5 μm^2
DICE latch	2.8×10^{-15} cm^2	500 ms	11.6 μm^2
Synthesiezed TMR	5×10^{-18} cm^2	270 s	700 μm^2

Table 32: Comparison of SEU cross section measured in prototypes and the estimated mean time between upsets (MTBU) at the HL-LHC inner layer for different memory strcutrues.

B. Methodology and Design Flow

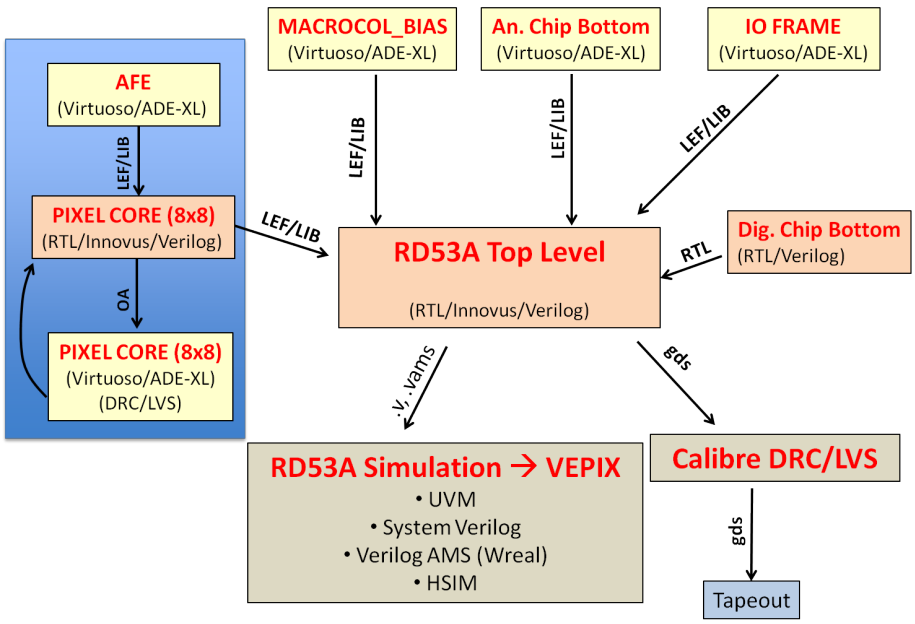


Figure 57: Top level design flow

The digital synsthesis flow uses a semi-costom approach that includes Triple Memory Redun-
1375 dancy (TMR). add more text here

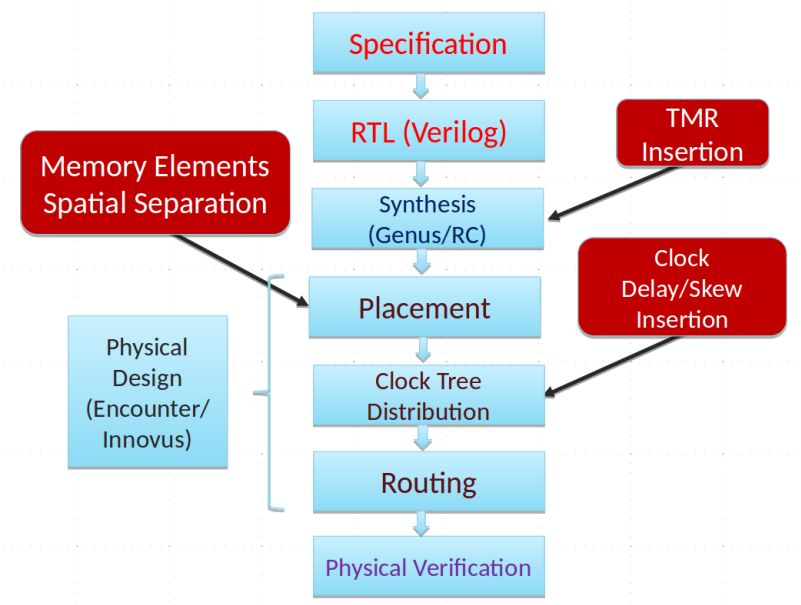


Figure 58: Digital synthesis flow including TMR.

C. Peculiarities ("Quirks") of the RD53A Design

This appendix collects known features of the design that would not be obvious and may be visible as systematic effects in testing. Most of these are due to conscious choices that were made where design compromises were needed. They will likely need to be addressed in the production chip versions.

C.1 Current Reference Trim Bits

The 4 current reference trim bits must be set by wire bonds (Table 19). They do not have internal pull-up or pull-down resistors.

C.2 Distribution of Calibration Edge

There is a significant timing spread in the distribution of the CAL_edge signal in the bottom of chip. While the propagation of the signal up a column is well synchronized for all pixels in the column, there is a different time offset for each column of cores. Table 33 shows the expected offsets from simulation.

Core columns	**Offsets (ns)**																
0-15 (sync.)	2.3	2.0	1.6	1.4	1.2	1.0	0.8	0.8	0.4	0.4	0.0	0.0	0.2	0.6	0.9	1.2	
16-32 (lin.)	0.6	1.0	1.2	1.6	1.8	2.1	2.1	2.3	2.6	2.7	3.2	3.4	3.5	3.8	4.3	4.5	4.5
33-49 (diff.)	4.8	5.2	6.0	6.1	6.3	6.4	6.6	6.9	7.1	7.6	8.0	8.2	8.5	8.7	8.8	9.1	9.2

Table 33: CAL_edge signal timing offset for each core column from simulation.

C.3 Charge Injection from Priming

Calibration charge injection takes place when the switching from a higher to a lower voltage in the pixel as explained in Sec. 5.4. The priming of the injection mechanism entails switching from low to high, which is not done with a slow ramp but also with a step. Depending on the threshold setting and the size of this wrong polarity step, a front end can also fire during priming as well as injection. This has been observed in analog simulations of the differential front end core. It should normally not present a problem since the user is free to place the priming operation at a time where firing of front ends does not matter, but the user must take care to do that.

C.4 Empty Packets in Single Lane AURORA with Saturated Readout

An anomaly in the AURORA encoding occurs in single lane mode when a large number of triggers are sent (so that some are skipped) and there are enough hits in the chip so that all triggers contain data. In this case empty packets occasionally come out, having an arbitrary trigger tag. Other packets are not affected. This behavior has not been fully mapped yet.

References

[1] RD53 Collaboration, "RD53A Integrated Circuit Specifications," CERN-RD53-PUB-15-001 (2015).

[2] Xilinx, "Aurora 64B/66B Protocol Specification," SP011 (v1.3) October 1, 2014.

[3] M. Karagouins *et. al,* "An Integrated Shunt-LDO Regulator for Serial Powered Systems," in Proc. of IEEE ESSCIRC '09, (2009).