Development of an ASIC for CCD readout at the vertex detector of the International Linear Collider

P.Murray [a], S.L.Thomas[a], K.D.Stefanov[a], T.Woolliscroft [b]

[a] Rutherford Appleton Laboratory, Didcot, U.K. OX11 0QX
[b]Department of Physics, University of Liverpool, U.K. L69 3BX

P.Murray@rl.ac.uk

Abstract

The Linear Collider Flavour Identification Collaboration is developing sensors and readout electronics suitable for the International Linear Collider vertex detector. In order to achieve high data rates the proposed detector utilises column parallel CCDs, each read out by a custom designed ASIC. The prototype chip (CPR2) has 250 channels of electronics, each with a preamplifier, 5-bit flash ADC, data sparsification logic for identification of significant data clusters, and local memory for storage of data awaiting readout. CPR2 also has hierarchical 2-level data multiplexing and intermediate data memory, enabling readout of the sparsified data via the 5-bit data output bus.

I. Requirement

A. International Linear Collider

The ILC will bring into collision e+ and e- beams at centre-of-mass energies of initially 200 to 500GeV. The products of the resulting interactions will be recorded by two detectors. An important component of each of these is the vertex detector. This is designed to measure very precisely the tracks of charged particles close to the interaction point (IP), allowing the identification of those which originate from decay vertices displaced from the IP. Hence, particles containing b and c quarks and tau leptons can be efficiently detected.

To achieve the necessary precision, the vertex detector sensors must have a resolution < 5 microns and present a minimum of material to the particles which traverse them. The power consumption of the sensors and their associated readout systems must thus be small, allowing gas cooling. These requirements are fulfilled by CCDs, on which the LCFI Collaboration has concentrated its R&D efforts [1].

The high pair production backgrounds at the ILC require that the CCD pixel columns be read out in parallel at 50MHz if the occupancy is to be kept below the 1% desirable for pattern recognition. This requires a readout chip with a channel for each column of CCD pixels. The low occupancy and the large number of pixels (the entire vertex detector contains nearly 10^9 pixels) makes on-chip data sparsification desirable.

B. CPR (column parallel readout) project

The long term aim of the CPR project is to provide complete readout of the CPCCD (column parallel CCD) detector [2]. CPCCD outputs are bump bonded onto the CPR chip inputs at a pitch of 20µm. CPR2 is the third generation prototype designed to in the project. CPR0 was a test structure for the ADC. CPR1 included the front end amplifiers and was successfully tested in bump bonded configuration. CPR2 includes sparsification and readout circuitry, with addition of timestamp and position data to enable event reconstruction. The chip is fabricated on the 0.25µm IBM process.

II. CPR2 chip

A. Overview

The CPR2 chip has 250 channels on 20µm pitch for bump bonding to the CPCCD. For test purposes, channels are divided into two types, half driven by charge preamplifiers directly coupled to the CCD outputs, the other half by voltage preamps connected to the CCD through source followers.

Because charge from a single particle may be deposited in several neighbouring pixels it is necessary to compute the sum of the digitized data for each 2x2 group of pixels, before comparing the result with a threshold in order to determine whether or not to store and read out the data. To prevent losing important information, the chip is also designed to capture and read out data in pixels surrounding those whose data sum has triggered readout.

Internal data storage is needed because the chip has only one (5-bit) output bus. The internal memories are continuously read out to the bus by a clock-driven multiplexer at 8 times the front end frequency. The chip also includes a scan register for testing the ADC and the sparsification logic. A photograph of the chip is shown in Figure 1.

B. Preamps and ADC

The CPCCD sensor has both source follower and direct output connections to the CPR2 chip. The amplifiers are driven by step voltages in the range of 0 to -3mV (voltage amplifiers) and 0 to 2000 electrons (charge amplifiers). They are designed to produce a 0 to 100mV output step to be digitized by the ADC. Larger signals can be handled but with loss of linearity.

The 5-bit flash ADCs have adjustable high and low reference voltages giving a variable dynamic range. They are normally used for the range 0 to 100mV. The flash encoding circuit produces a modified Gray coding system which minimises errors due to “bubbles” in the thermometer code produced by the comparators. The ADC output is fed to a code converter which converts it to binary format so that arithmetical operations can be performed.

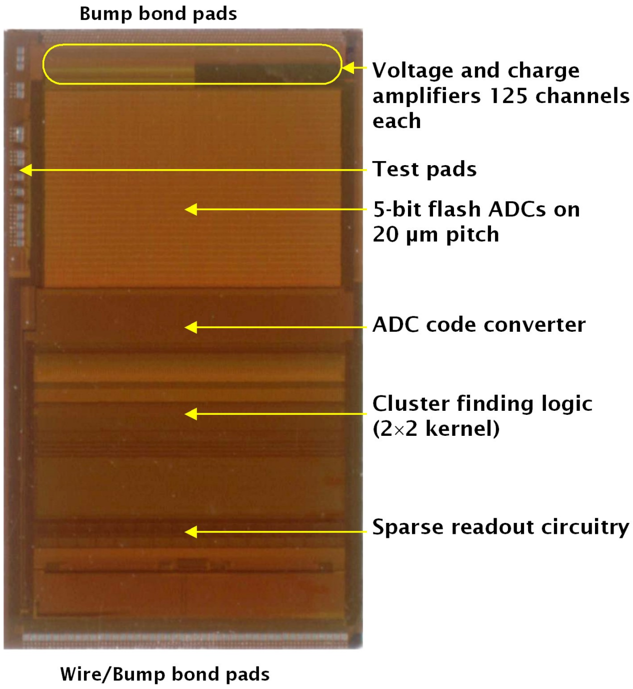

Bump bond pads
Voltage and charge amplifiers 125 channels each
Test pads
5-bit flash ADCs on 20 µm pitch
ADC code converter
Cluster finding logic (2×2 kernel)
Sparse readout circuitry
Wire/Bump bond pads

Figure 1: Photograph of CPR2 chip

C. *Cluster finding logic*

The function of the cluster finding logic is to compute the sum of all 2x2 arrays of pixels and compare with a threshold, thus triggering local data storage if appropriate. A diagram of the cluster finding logic for a single channel is shown in Figure 2. All channels have the same logic.

The 5-bit data at the output of the code converter is input to a register which functions as a 1 clock cycle delay element. The output of the register is fed to one input of a 5-bit adder. The other input of the adder comes directly from the code converter. Thus the data from the current pixel is added to that of the previous one to give a (6-bit) sum of two vertically separated pixels. This is the partial sum. This sum is input to the 6-bit adder and added to the partial sum from the neighbouring channel to the right. This forms the total sum (7-bit), which is then stored in a register.

In the next clock cycle the total sum is applied to the 7 bit logic comparator, the other input of which is the global threshold bus. If the data sum is greater than the threshold the comparator output is high. The result is fed to the "or" gate which triggers the memory controller to store raw data from the ADC in the local channel memory. The same comparator signal is applied to the "or" gates of the two channels to the right of the current one, and to that of the channel to the left of the current one.

This triggers storage of data in each of these channels also, thus saving the data of the 2x2 cluster and one channel on either side of it.

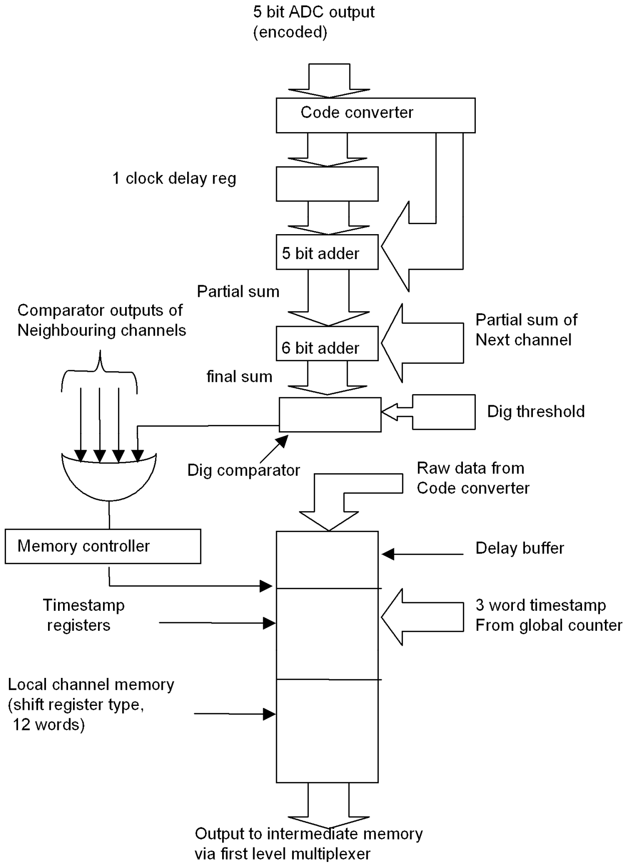
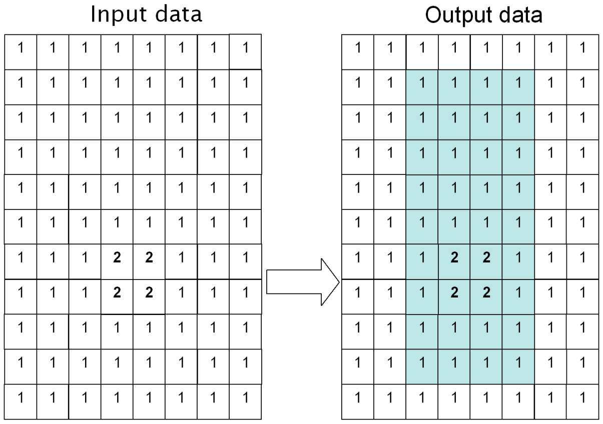

5 bit ADC output (encoded)
Code converter
1 clock delay reg
5 bit adder
Partial sum
Comparator outputs of Neighbouring channels
6 bit adder
Partial sum of Next channel
final sum
Dig threshold
Dig comparator
Raw data from Code converter
Memory controller
Delay buffer
Timestamp registers
3 word timestamp From global counter
Local channel memory (shift register type, 12 words)
Output to intermediate memory via first level multiplexer

Figure 2: CPR2 cluster finding logic

Figure 3 shows data stored for a minimum cluster in which only one channel detects a "data over threshold" condition. The shaded data is output from the chip. In this case the threshold is 7.



Input data

1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	2	2	1	1	1
1	1	1	2	2	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1

Output data

1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	2	2	1	1	1
1	1	1	2	2	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1

Figure 3: Input - Output data for a minimum cluster in CPR2

D. *Local data storage and time stamping*

As can be seen, each channel stores data for 9 pixels, including the two preceding the hit pixels in the readout sequence. This is done by saving all raw data in a delay buffer. This is overwritten on every clock cycle if the comparator output is low. If the latter is high, the data is shifted to local channel memory, preceded by the timestamp words and followed by 7 more words of current raw data just emerging from the ADC code converter. The local channel memory takes the form of a shift register which can be "frozen" by the memory controller circuit.

The timestamp buffer is downstream from the delay buffer. It consists of 3 5-bit elements for storing the 3 5-bit timestamp words. The latter are generated by a global 15-bit counter, the bit lines being distributed across the chip. The timestamp buffer loads a new timestamp every clock edge until the memory controller, triggered by the comparator, converts all registers into a shift register. This happens for 12 clock cycles, at the end of which the local memory, which lies downstream of the timestamp buffer, will contain the 3 timestamp words and the 9 data words. At this point the memory controller freezes all shifting, awaiting readout by the readout multiplexer.

E. Readout system

The main elements of the readout system are shown in Figure 4. The 256 channels are grouped into 16 groups of 16 channels. Each channel group is served by a multiplexer which transfers channel pixel data into an "intermediate memory". The latter only has room for 12 5-bit words, ie the contents of one channel memory. It is in the form of a freezable shift register, like the channel memory. The multiplexer is based on a cyclic shift register which interrogates each channel in a group in turn until one is found with data to read out. It then shifts the data into the intermediate memory, prepending an extra word which identifies which of the 16 channels the data came from.

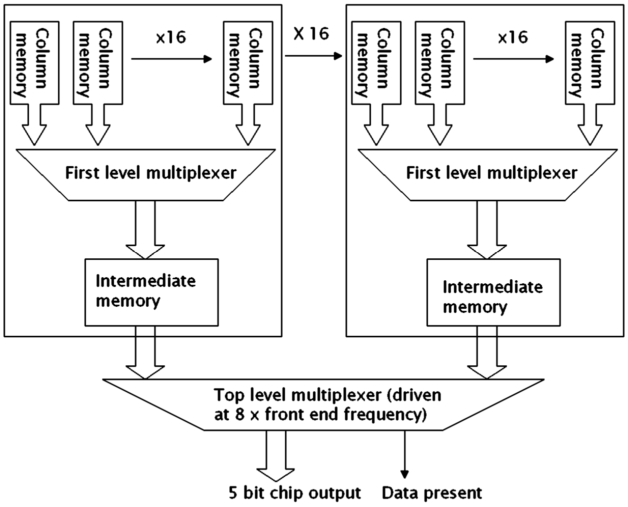


Figure 4: CPR2 readout system

The multiplexer pointer is frozen until the intermediate memory is emptied. It then begins searching again from this point, rather than resetting to a particular position. This system ensures that pointer positions become randomized. If this were not the case, channels closer to the multiplexer pointer reset position would have a greater chance of being read out. This would introduce a bias into the data.

In the final readout operation, the top level multiplexer, which works like the first level multiplexer, locates an intermediate memory containing valid data, and shifts this directly to the chip output. It prepends an extra word indicating which channel group the data has come from. The top level multiplexer controller also outputs a "data present" signal onto an output pad. This enables the external system to know when data is being output. Thus with the low and high order position words and the three timestamp words the external system can reconstruct the positions of the pixels corresponding to the 9 pixel data words. Note that the top level multiplexer is driven in the simulations at 8 times the frequency of the rest of the chip.

F. Test system

The chip is equipped with a scan register and 5–bit output bus inserted between the code converter outputs and the sparsification logic. In one mode of operation, digital data can be shifted in, then applied to the sparsification logic (serial in, parallel out operation). The latter is then clocked and the procedure repeated. Thus the sparsification and readout system can be verified. In another mode ADC outputs can be loaded into the scan register and then shifted to the output (parallel in, serial out operation). Thus preamp and ADC operation can be tested. In the final test mode the scan register can enable the output of a selected ADC onto a 5-bit bus which runs across the chip, the other outputs being set to high impedance. This enables selected ADC outputs to be tested in real time.

G. CPR2 tests

The chip has been tested both bump bonded to a CPCCD, and in stand-alone mode. For CCD testing, a resolution of 140 electrons rms was achieved for 5.9keV X-rays from an ^{55}Fe source. The base-line electronic noise was 44 electrons rms (Figure 5). The clock frequency was 2MHz, with the ADC set on 0-300mV range. The plot was adjusted to set the zero for energy at the centre of the noise peak.

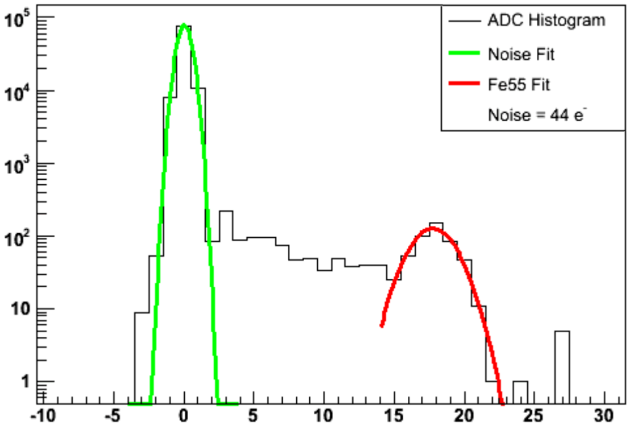


Figure 5: ^{55}Fe X-ray measurement

In stand-alone mode, the ADCs and both types of amplifier have been verified using the scan register. Also using the scan register the sparsifying logic and the readout system have been verified. This involves loading the register with sequences representing possible physics data, with some 2x2 clusters being above threshold. The chip was shown to read out clusters correctly at low levels of occupancy.

Problems arise, however, when one cluster is followed by another in the same channels. In such a case it can happen that the relevant channel memories have not been read out before the new cluster data arrives. In such cases the data cannot be stored and is thus lost. The problem occurs because the channel memories have space for only one minimum sized cluster (9 5-bit words) and this must be cleared before more data is stored.

The time separation of clusters needed for correct data readout depends on the number of channels to be read out at any one time. Since the readout bus is only 5 bits wide a large occupancy lengthens the average time needed to clear a memory, thus making it more likely that data will be lost. Minimum separation is in the range of 50 to 90 clock cycles, depending on occupancy. This "dead time" problem is the main cause of error in the CPR2. Figure 6 shows a situation where data will be lost by the CPR2. Only the shaded data will be read out.

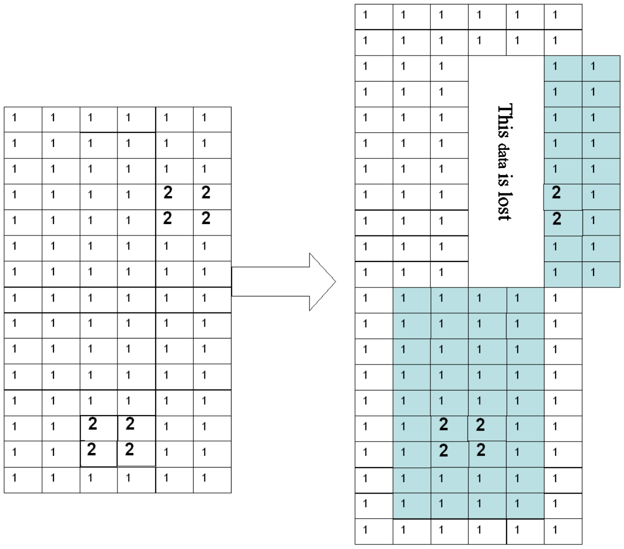


Figure 6: Data loss in CPR2 chip

III. CPR2A CHIP

A. *Overview*

In order to reduce the "dead time" problems associated with CPR2 and to correct some other problems, a new prototype, the CPR2A is being designed. The main difference with CPR2 is that the new chip will have more local channel memory. The new memory will be able to store 38 words instead of 12 in the same space. This will enable each channel to handle several clusters in succession, although inevitably the memory will fill up in some cases. In such a case the memory controller has been designed to overwrite pre-existing data rather than to abort writing. This means that the CPR2A failure mode will involve preserving later data and losing earlier data (opposite to CPR2).

The local memory controller in the CPR2A will also facilitate the storage of variable length clusters. Thus the timestamp can be followed by 6 words of pixel data, 6+8=14 words, or 6+8+8=22 words, depending on the length of the cluster. This feature is useful when dealing with long clusters or ones which are very close together, since it reduces the number of time-stamp words needed.

With cluster separations greater than 9 words it becomes more efficient to store the cluster data with separate time-stamps than to store all the data in between. Figure 7 shows the output of the chip for different input cluster separations.

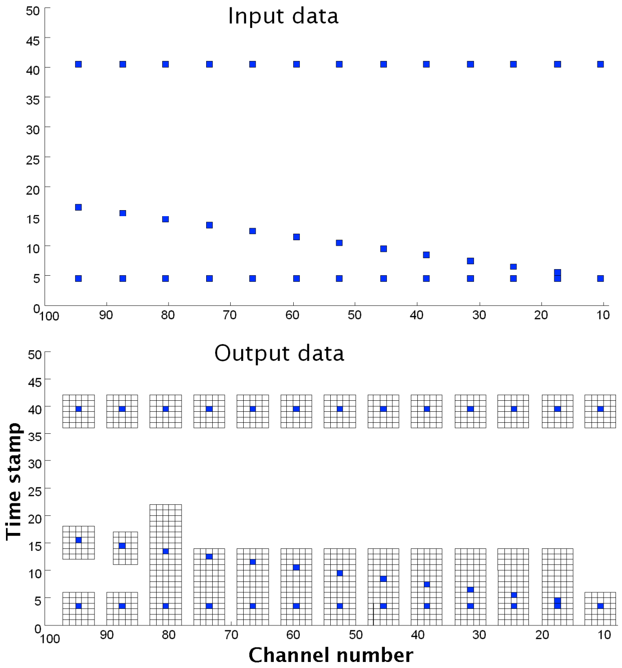


Figure 7: CPR2A cluster reconstruction

B. *Physics simulations*

The digital logic of CPR2A has already been designed and laid out. It has been extensively simulated using Verilog. Simulated physics data has been generated which can be used as an input to the simulation. A MATLAB program converts the data into simulation vectors and also into a graphical display showing the pixels and their data values using colour coding. Another MATLAB program takes the outputs of the Verilog simulation and converts these to a graphical display. By comparing the two pictures it is easy to detect any errors in the form of missing data. Figure 8 shows a detail from such a plot.

It is clear that most of the inputs shown in Figure 8 would not have been correctly read out by CPR2 because of the dead time problem. CPR2A has been more successful. Data has, however, been lost because of limited memory capacity.

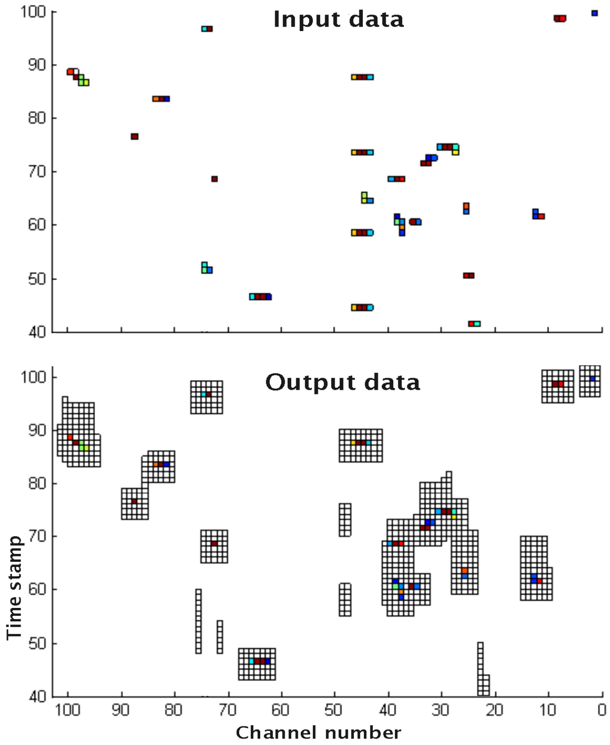


Figure 8: CPR2A physics simulation

Note in particular that the lost data is that which has an earlier timestamp in the CCD. This is because later arriving data sometimes overwrites earlier arriving data. The spatially periodic signal down the centre of the input field (due to a spiral track) is completely omitted except at the top. This is a recurrence of the dead time problem. Only a chip with a separate output for each channel could overcome this since it would require no internal memory and would thus have no memory overflow problems.

C. Output data formatting

The formatting of output data is necessarily more complicated in CPR2A than in CPR2. This is because the column memories may contain more than one cluster, with different timestamps, and these clusters may be of different lengths. The external system must be able to distinguish these cases and it therefore needs extra information.

In addition to the 5 pixel data bits, the CPR2A chip outputs an extra bit, the “header”. This is used internally to keep track of data in the memories and ensure efficient storage of data in the intermediate memory. At the chip output, the header bit enables the external system to determine when valid data is being output. This is necessary because, with variable length or multiple clusters, the intermediate memories which feed the output are not guaranteed to be filled with valid data as in CPR2. In general these contain “null” data which must also be shifted to the output and the external system must know when this is happening so as to ignore the output. The “data_shift” signal is an output which tells the external system that the contents of an intermediate memory are being shifted to the chip output. One clock after the signal becomes high, the high order address of the data is available at the output. The system then waits for the header to go high. The data present on the outputs at this point will be the low order address. The next clock edge produces an “all zeroes” state which is used as a data separator when there is more than one set of cluster data present. This is followed by the three timestamp words. Next comes the cluster data itself which is finished when the header signal goes low.

Figure 9 shows a case in which the memory being read out contains two clusters not separated by any null data, so the header remains high. The all zeroes state intervenes between the 2 sets of cluster data.

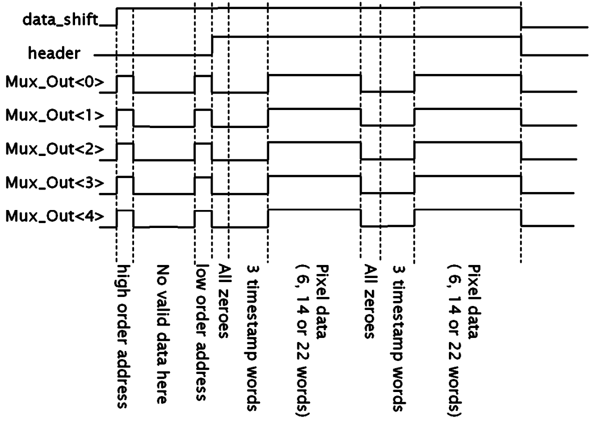


Figure 9: CPR2A output data format with 2 clusters

IV. Conclusions

The CPR2 readout chip for the CPCCD at the vertex detector of the ILC has been fabricated and tested in stand-alone and bump-bonded mode. This confirmed basic functionality but revealed problems due to dead time when more realistic physics data was input to the sparsification and readout logic.

The next iteration, CPR2A, is currently being designed, with increased memory and embodying a more complicated sparsification algorithm. It also facilitates the loading of a separate digital threshold for each channel. The logic of this chip has been simulated using realistic physics data and the dead time problem is shown to be much reduced. The design is well advanced, with projected submission in October 2007.

For the future, it is hoped to realise further versions of the readout chip using 0.13μm technology. This will facilitate the inclusion of more memory and more sophisticated sparsification algorithms than is possible with the current 0.25μm technology.

V. Acknowledgements

The authors wish to acknowledge the financial support of the Science and Technology Facilities Council and the contribution of the LCFI Collaboration.

VI. References

[1] C.J.S. Damerell, Nucl. Instr. and Meth. A 541 (2005) 178.

[2] K.D. Stefanov, Nucl. Instr. and Meth. A 569 (2006) 48.