# Pixel Support and ROD Status


ATLAS Wisconsin Group

K. Dao, D. Fasching, O. Hayes, R. Jared

ATLAS Irvine Group

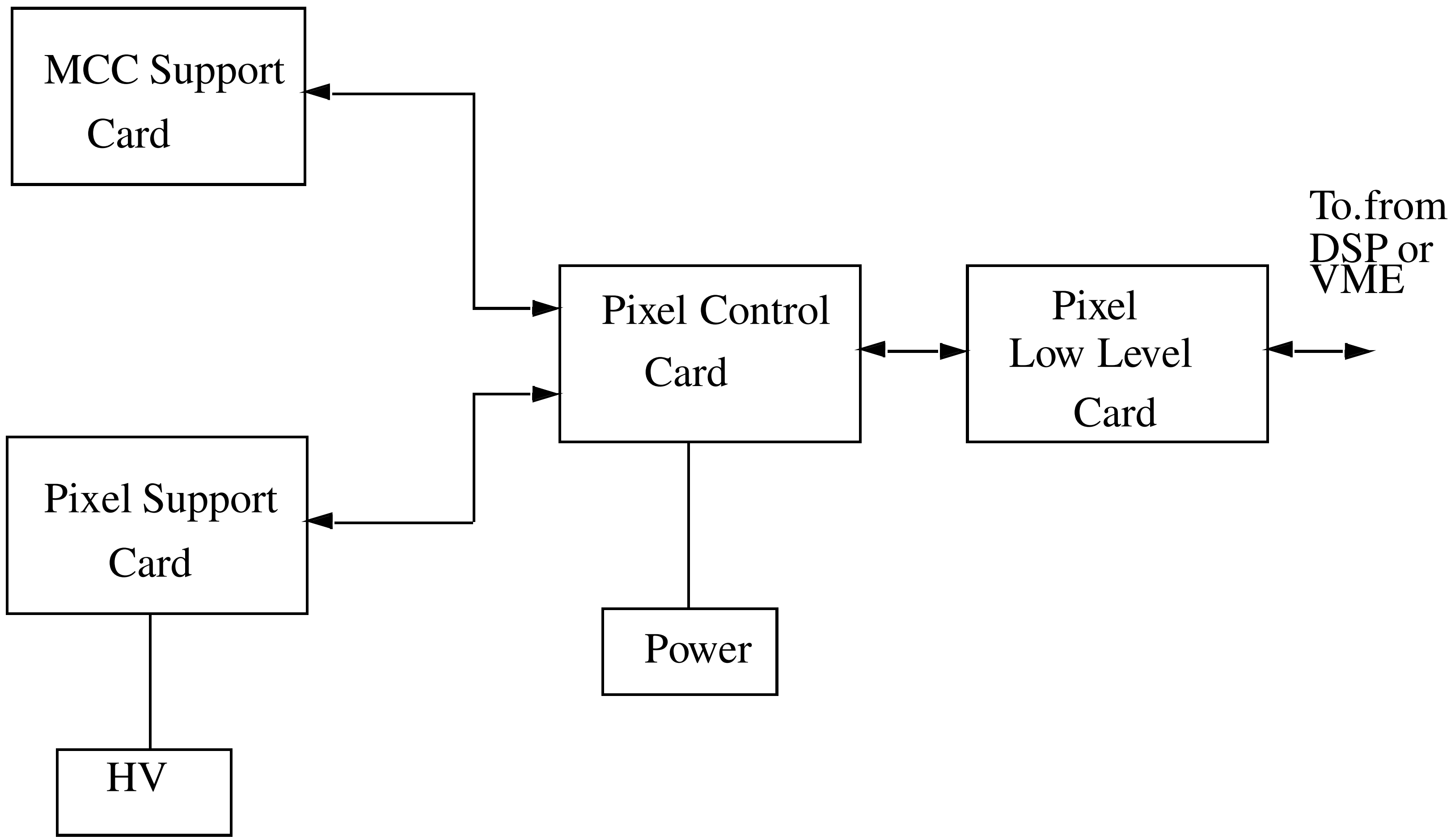
T. Fahland, A. Lankford, S. Pier, M. Schernau


March 4, 1999

# Outline

- Pixel Test Beam Support
  - System Overview
  - Pixel Low Level Card
- ROD Status
  - DSP Based
  - FPGA Based

Pixel Test Beam Support Overview



# System Overview

Pixel Low Level (PLL)
FPGA based board works on the following devices w/o reconfiguration, i.e. with only one set of code to maintain and one set of proms to keep track of and distribute.

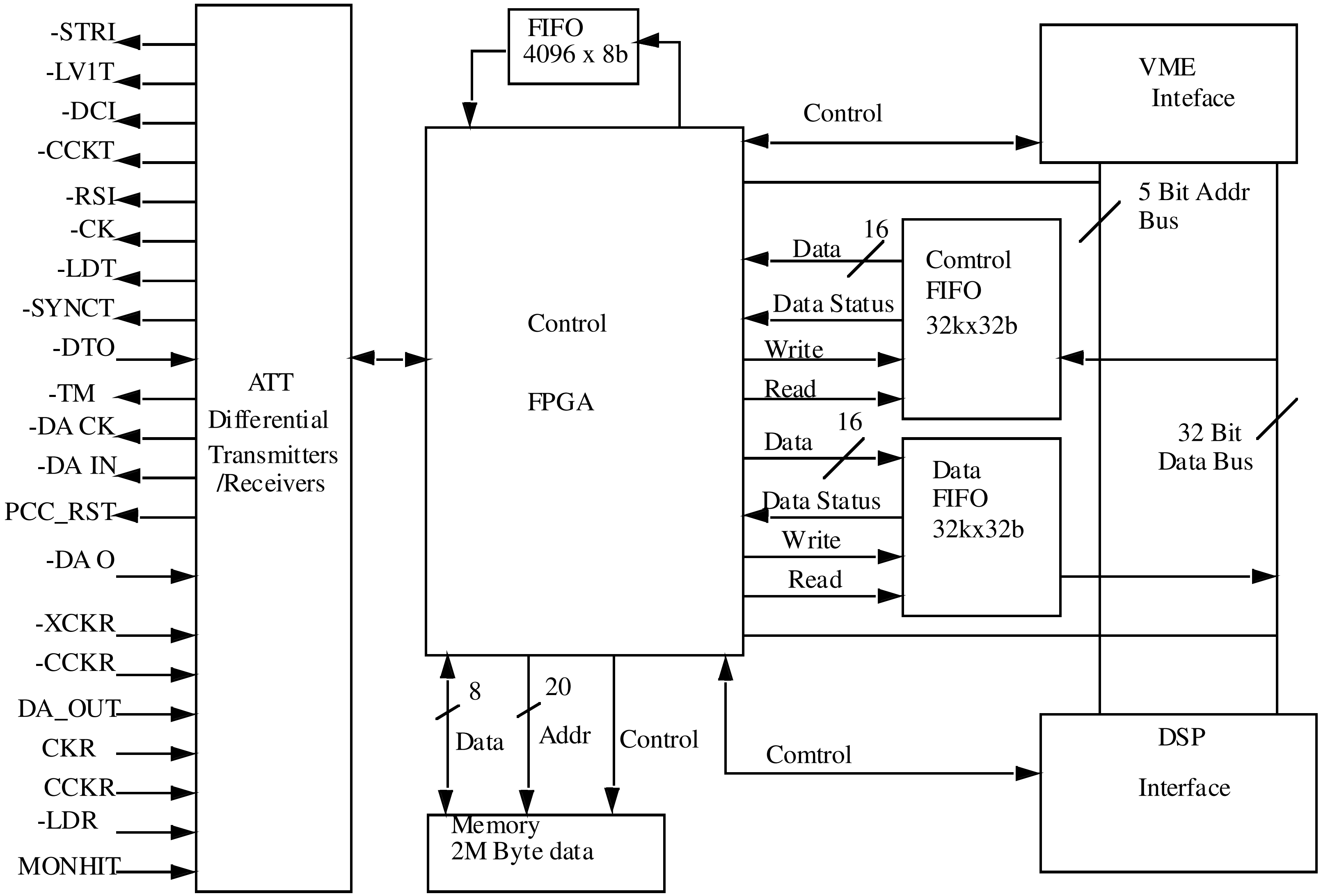
Works with the following ICs

- FE-A (single chip)
- FE-B (single chip)
- MCC "replacement" chip (basically a programmable MUX by which one can configure and readout a single chip on a module)
- MCC (16 chip modules)

Works with following Pixel Control Cards (PCC)

- Seigen PCC
- LBNL PCC

# Pixel Low Level Cards



# Pixel Low Level Card

User can communicate via DSP based VME board (UCI) or through a direct VME interface (Cyprus). A VHDL "bus-interface" makes all communication with the board appear to the user as simple register reads/writes. This is a very simple interface which in turn makes user code simpler and therefore more robust.

# Pixel Low Level Card

Philosophy: list processor

Commands stacked in a 32 bit wide 32k deep "command" FIFO.

A "go" signal starts the FPGA processing the commands and Data in the FIFO.

The commands are parsed by the FPGA which then executes the required operation.

Some commands (e.g. configuration commands) have associated data. This data is stacked in the command FIFO after the command and is unformatted apart from a special last_data_word bit.

# Pixel Low Level Card

Three classes of operation:

- Configuration of front end chips and MCC
- Data taking
- Write MCC receiver and receive event data back

# Pixel Low Level Card

## DATA TAKING:

Trigger options:

1) "hitbus" trigger
2) external NIM trigger
3) PLL internally generated trigger

Strobe generation options:

1) external NIM strobe
2) PLL internally generated strobe (used with 3 above)

# Pixel Low Level Card

Data decoding modes:

1) DFIFO modes
    a) raw data
    b) word aligned data
2) SRAM modes
    a) raw data
    b) word aligned data
    c) histogram
        i) by TOT
        ii) by index which is communicated via a command word

note: The DFIFO and SRAM modes are set independent of one another, e.g. you can write raw data to the DFIFO while using the SRAM in histogram mode.

# Pixel Low Level Card

Write MCC receiver and receive event data back :

This is a special mode available to take advantage of a debug capability of the MCC.

To the PLL it looks like a write MCC configuration type command except that there is return data which is treated as real event data. In this mode, the MCC receiver logic, which normally receives data from the front end chips, is written to directly. The MCC then decodes the data, formats it and transmits it upstream as if it were event data.

# Pixel Low Level Card

Pending requests:

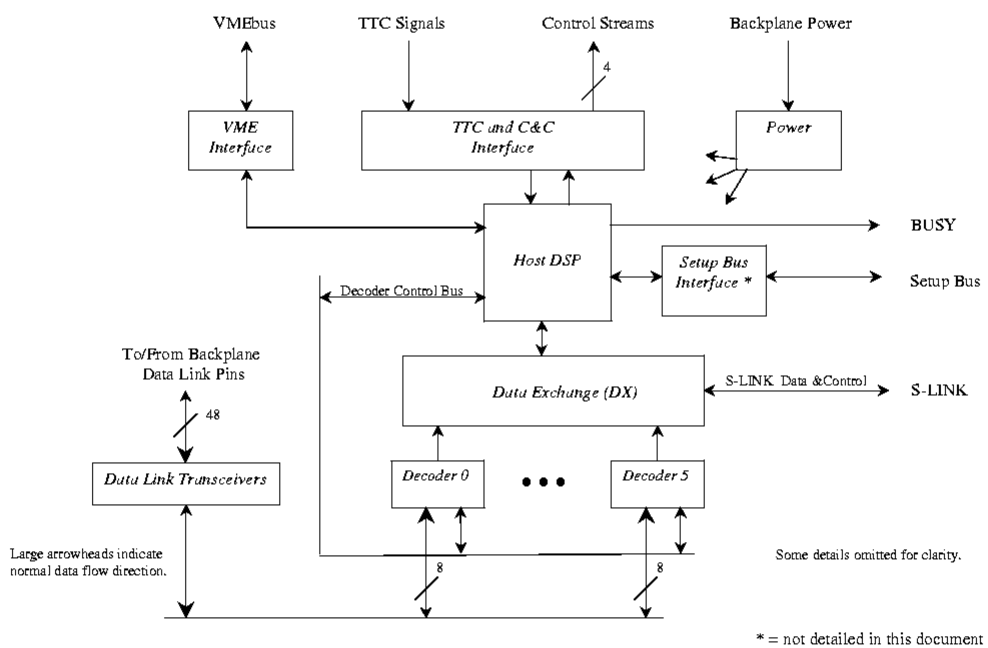
- Maximum delay between events in PLL trigger mode is 4096 clocks, limited by the depth of the TFIFO.  May double this by adding a delay (counter).  After the TFIFO reached the end of the pattern, rather than restarting immediately, playback would restart after the delay.  The delay counter would be enabled with a switch in the data taking command word.

Note:  TFIFO is currently in debug.  It appears basically to work with the caveat that in TFIFO mode there are twice as many events generated as were requested.

- In hitbus mode, it has been requested that a deadtime be available between events.

# ROD DSP Based

ROD99 Subsystems--DSP Implementation




ROD99BlockDiagram3.doc, S.Pier, UCI, 09/03/98 p. 2 of 16

# ROD DSP Based

## Strengths of DSP ROD:

•DMA is used to read input and write output data

•Input DMA data is placed in 1K blocks of effective FIFO memory in the DSP automatically

•A processor (DSP) makes the ROD flexible

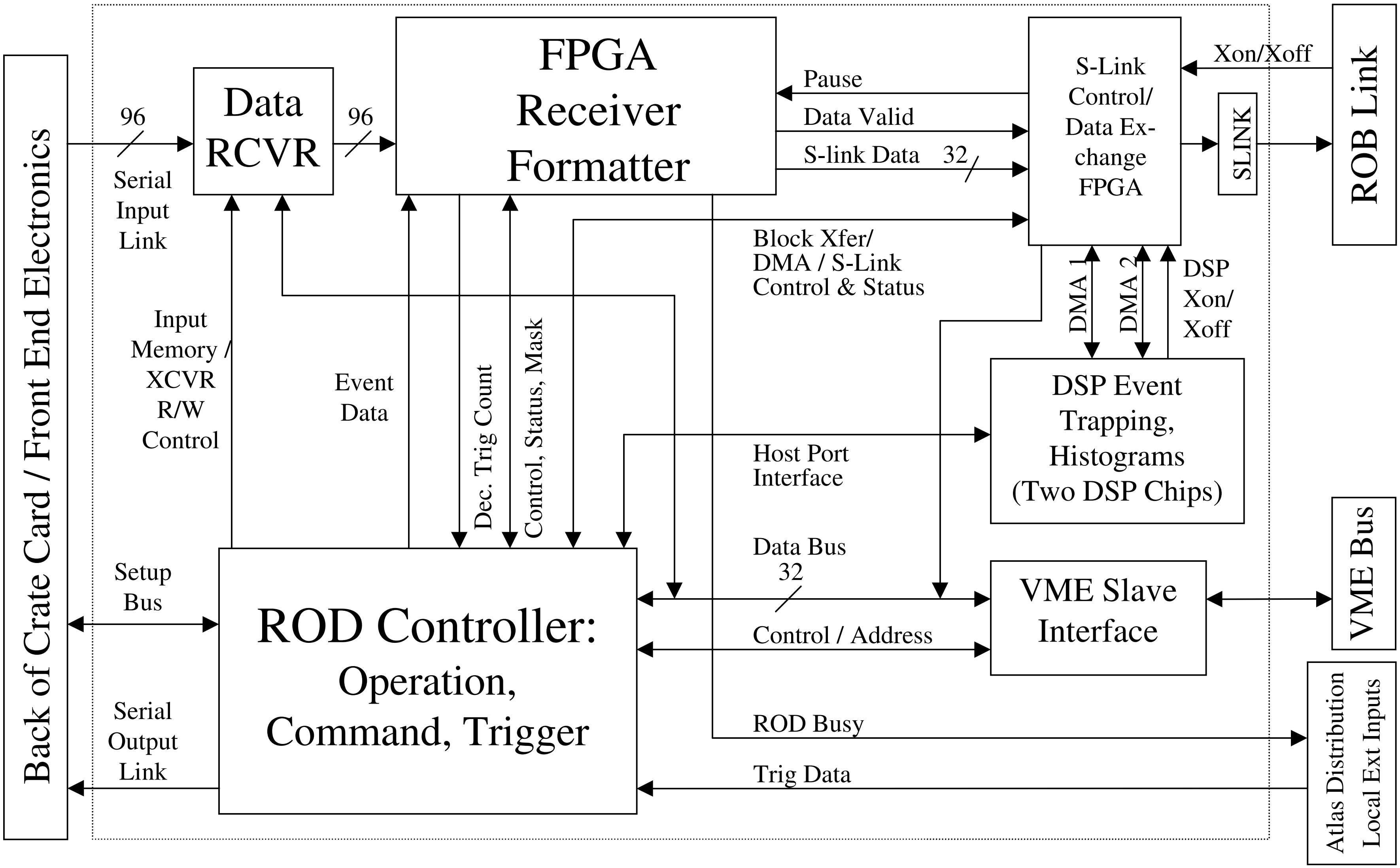
•Can easily respond to future undefined needs

•Emulator on PC can be used with JTAG, in circuit, to debug DSPs on board

•Code is developed/tested on commercial PCI TMS320 module

•Event fragment are thrown away by the software to prevent buffer overflow

# ROD FPGA Based



# ROD FPGA Based

## Strength of FPGA ROD

- The operation of the data path is deterministic
- Concurrent operations in FPGA
- Will keep up with full occupancy
- Design can be fully simulated at the board level with behavioral model of DSP I/O
- VHDL code allows for future upgrades
- DSP Histograming allows for upgrades to reduce data volume
- DSP controller allows for complex ROD calibration.

# Test Beam Support and ROD Current Status

•17 PLL cards have been fabricated

•Minor software updates are in progress for the PLL.

•A philosophical differences exist between UCI and Wisc. Wisconsin wants a deterministic design (FPGA based). UCI wants a flexible design (DSP/processor based).

•The last several month have been spent to technically resolve the differences in philosophy between UCI and Wisconsin. A hybrid model was investigated that used FPGA for decoding and DSP for Gathering the event data. This effort technically failed.

•The DSP approach is reasonably advanced.

•The FPGA approach is reasonably advanced.

•A review will be held at the end of March to pick the approach.