Functional Description of the
Multi Channel Electronics

Mark Halpern 11 June 2003
Latest revision: 27 Sept. 2003

This note is meant to provide an overview of the Multi-channel Electronics (MCE) design with enough details that the major design decisions are apparent. Fuller details are available in specific descriptions of each card or subsystem, and of the firmware driving them. A separate series of analysis notes describes details of design decisions for specific topics, including output filtering, noise performance, data rates, timing jitter, backplane communications.

BASIC OPERATION:

The electronics described here borrows heavily from the designs used at NIST. Changes have been made to use some newer components, integrate and automate the functions and match to the specific *SCUBA2* configuration.

Each full *SCUBA2* bolometer array consists of four sub-arrays arranged in a two-by-two grid in the focal plane. In basic operation, one box of the MCE controls the squid amplifiers and multiplexor, and reads the signals from one 32×41 pixel sub-array. The system hardware is is completely modular at the sub-array level. Each sub-array is connected via woven cryogenic cables to a single box of MCE, mounted directly on the cryostat wall. Each box of the MCE is in turn connected by an optical fibre to a single data acquisition computer running real-time Linux. There is no mechanism to synchronize clocks between sub-arrays. Because the electronics all run in the same cryostat, they share chassis ground of course.

The MCE are built onto 6U cards which are housed in a standard width RF chassis. Each card has a serial number, and each slot has a card-readable slot ID, so full card locations are available to the user of the electronics. The electronics are cooled (or heated if necessary) by filtered, positive pressure air which flows sequentially through the four chassis of a full array. The five cryogenic cables of each sub-array are epoxyed into the cryostat wall and terminate in 100 pin MDM connectors. A custom-built captive filtered connector allows blind mating of these cables with the chassis bolted to the cryostat. These filtered connectors are manufactured as part of the instrument back-plane and are connected to it via short flexible tails. The filtered connectors and the cryostat wall form a faraday cage protecting the bolometer arrays. Details of all of these mechanical arrangements are handled in separate documents on the chassis design.

To facilitate high altitude operation in a setting which is not ideal for handling electronics, and to avoid the need for stocking customized spare parts, any spare card can be placed directly into the appropriate slot any chassis without component substitution, jumpers or switch setting. Mechanical keying prevents insertion of cards into inappropriate slots in a chassis.

In this document, we will provide an outline of the data handling on readout, MCE internal communications, startup operations.

DATA HANDLING Four **Readout Cards (RC)** read eight output lines each with a
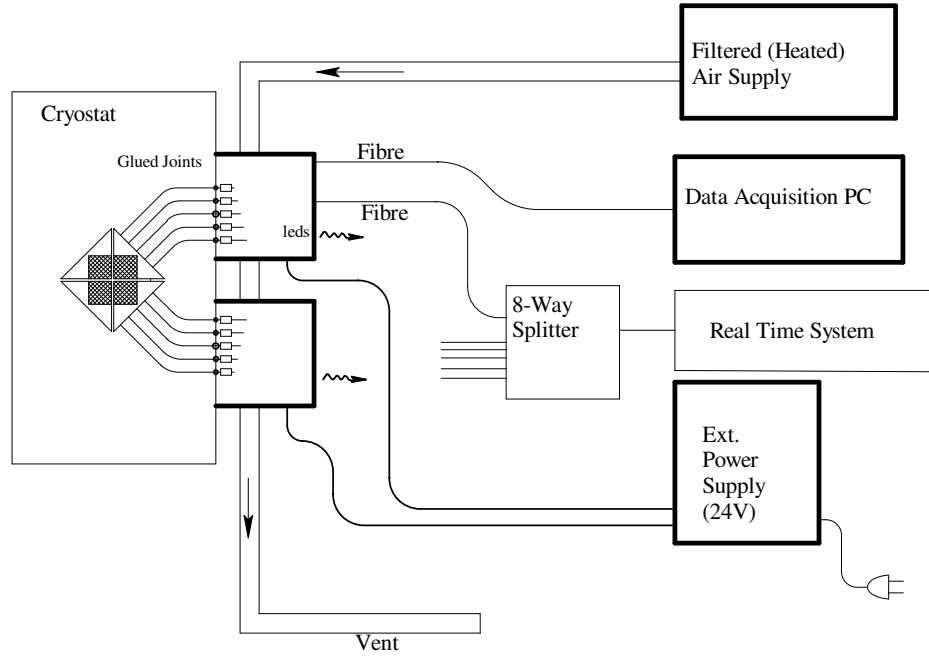
Figure 1: **Block Diagram of the MCE on SCUBA2** Each quadrant of the array at each passband is attached to a dedicated box of the Multi-Channel Electronics. Each Electronics chassis is in turn connected to a single data acquisition PC (DAC) running real-time Linux. A single 24V external power supply and a single filtered air supply feed all 8 subracks. The copper wires of the power supply are the only external electrical connection to the Electronics. All signal connections are fibre optics. The boxes in **bold** are provided by UBC. (This is not a real block diagram, since all SCUBA2 block diagrams are in VISIO^{TM}.)

preamp followed by a 14-bit 50 MHz A/D converter. Meanwhile an **Address Card (AC)** asserts one after another of 41 addresses at a frequency, $f_{address}$, not to exceed 850 kHz. The address frequency will be a sub-harmonic of the 50 MHz A/D clock, ie $f_{address} = 50\text{MHz}/n_{ad}$ where $n_{ad} \approx 60$. At each new address, the first m_{blank} readings are discarded to allow for the settling of the array and perhaps timing jitter, and the next $n_{ad} - m_{blank}$ readings are co-added. A complete array image is collected at

$$f_{frame} = f_{address}/41 \approx 20\text{kHz}.$$

We call the 41 addresses columns, and the 32 lines going to the four **RC** are rows. Just a reminder, the 41st row is all "dark" pixels, not exposed to light from the telescope.

Card synchronization: A single **Clock Card (CC)** acts as the master card in each box. The clock card is the only card which communicates with the outside world. The CC generates a 25 MHz clock derived from a crystal, which is piped to each of the other cards which synchronize to it by running a phase-locked-loop at $16 \times 25 = 400\text{MHz}$. A 25 MHz clock is re-generated on each board synchronous with the CC clock *as received.* this clock is used to decode **Received** data and commands.

Once the CC issues a command to begin data collection, all cards set internal counters to zero and begin the reading cycle. There are $n_{image} = 41 \times n_{ad} \approx 2500$ readings in one array image collection. Upon the 25 MHz clock edge corresponding to a complete address cycle the CC queries each other card for the difference between its counter and n_{image}. A non-zero value corresponds to a timing error. If timing errors exceed a yet-to-be determined level, the CC generates a reset at the next address cycle completion.

In this way, the cards in a single box run 'open loop' until timing errors occur. No attempt is made to synchronize one box with another or any box with the outside world. To be explicit, the precise timing of the address cycle is not triggered from outside of any MCE box, nor is it synchronized between sub-arrays.

The operations described above run continuously in all of the following modes.

Operating Modes: There are several specific operating modes which differ in data filtering and other details.

Normal Science Mode We expect a **Data Valid Pulse (DVP)** pulse from the **Real Time System (RTS)** to arrive at the CC nearly regularly with nominal frequency f_{dvp} not to exceed 400 Hz. (Probably $f_{dvp} = 200$ Hz.) This is quite different from our previous practice with flight instruments, in which the telemetry stream runs on a regular clock. The purpose is to synchronize the data stream with mirror motion, which is controlled separately. The MCE will filter data from every pixel with a bandwidth, ΔB chosen to match the sample frequency. Science Mode does not tax any of the resources of the system, so we imagine that if there is a need one could implement a faster rate by altering the firmware. This might be useful in polarimetry, where performing sky subtraction more rapidly than the mirrors can move is advantageous.

The steps in Science Mode are as follows.

Each RC performs an IIR numerical filter on the data designed to produce the desired audio bandwidth. The IIR is updated at the address cycle rate, $F_{address} = 20$ kHz for each pixel,
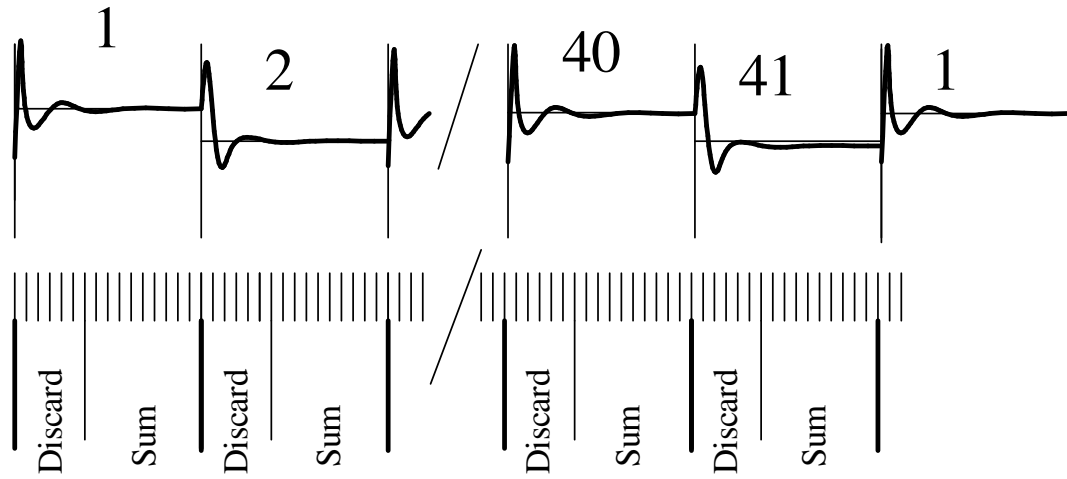
Figure 2: **Schematic Timing Diagram of the Readout Card.** The AD converters are trigerred regularly at 50 MHz, and the address card switches from one address to the next on a clock edge. The address rate will be an integral sub-harmonic of 50 MHz. At each address, an initial period of data are discarded to allow settling. The next group of samples are co-added, stored and fed into the IIR filter. Each readout card counts the number of ADC clock cycles in a complete cycle of addresses and reports this to the CC if queried. Note that the 41st row of pixels is 'dark', *ie* not exposed to radiation from the telescope.

and at all times there is a filtered image of the full sub-array available. (Optionally, an FIR filter could be implemented on the planned hardware, but we do not prefer it. See the analysis note on filtering below.) Whether IIR or FIR filtering is used, the filtering algorithm itself does not sub-sample the data stream. A 20 kHz filtered data stream is available at all times which has very little power above $f_{nyq} \approx 100$ Hz.

Upon receipt of a Data Valid Pulse, the CC waits to the next completion of an address cycle and then:
 -Requests the filtered partial-array image from each RC. This is the sub-sampling of the data;
 -Requests other houskeeping data from all card (temperatures, error rates, frame counts, slot ID,...);
 -Packages a sub-array of data;
 -Sends data package to the **Data Acquisition Computer (DAC)**.

This approach guarantees that the amount of data from each bolometer, and the relative phase of each bolometer signal compared to the DVP are identical in every reported sample. This is crucial in subsequent analysis.

 Each RC reports back to the CC on a point-to-point LVDS parallel line, that is a separate line per card with no collisions possible.

 The necessary data precision is

$$N = \text{14bits per sample} + ln_2(\text{64 samples at 50MHz}) + ln_2(\text{200 samples at 20KHz})$$
$$= \text{28 bits per data point.}$$
(1)

so we will use 32 bit words. The data rates in Science Mode are all well below system capacity. (In fact, if we were cramped for data size, 28 bits is an overestimate of the required precision. The noise only improves as the square root of the number of data points co-added. Thus, even at 24 bits *SCUBA2* will not be digitization-noise limited.)

 The timing jitter introduced by waiting until the next complete address cycle is $14\mu s$ rms, and is acceptable. See `timing-jitter.pdf`REF

 <u>Engineering Mode</u> There are several modes which are all called **Engineering Mode (EM)** which have no audio filtering at the 20 kHz level, and therefore much higher data rates. The precise data rates and hardware data rate limits are documented in **Bryces note** and `data-rates.pdf`.REF The main limit one encounters in EM will be the 25 Mbits per second limit on transfer from any RC to the CC.

 These modes are primarily intended for the use of people testing the electronics or the array, and not for astronomical data collection.

 As a general rule the DV pulse is not needed in these modes.

 A1 In EM A the output of one A/D converter per RC is sent to the CC at the address rate of ≈ 0.85MHz. This data stream is

$$(\text{24 bits/word }) \times 0.85\text{MHz} = 21 Mbps$$

which fits on, and nearly fills, the dedicated LVDS line from each RC.

In EM A1 the commands are (Select an AD Channel), (Start), and (Stop).

A variant of A1 could be implemented in which any $n \times m$ array of pixels is reported at 20 kHz as long as $n \times m \leq 41$. Let's call this **EM A2**.

EMA1 and EMA2 allow long continuous data sets useful for fourier analysis.

B The full array produces too much data to communicate it from any RC to the CC within the available bandwidth. However, in EM B we store the full array images on the RC for a period and then report the data out to the CC. The amount of data stored per frame is

$$24 \text{ bits } \times 8 \text{ channels } \times 41 \text{ addresses } = 960 \text{ bytes.}$$

We will allocate 200 kbytes storage on each RC and store $m \leq 200$ full array images, or < 10 ms of data. Upon query from the CC, each RC will report the stored data.

These bursts of data will help test timing etc in the MCE.

Commands from the CC are (Start collection of m images), (Report Images).

The precision per word required in EM A and EM B is 14 bits plus ln(64 readings per address) = 24 bits.

C In EM C, no co-addition of the 50 MHz data is performed. Obviously this will produce an enormous amount of data so only one A/D converter per RC is read out, and that is only read out for a short period. Since no co-addition of data takes place, 14 bits per word is sufficient (well, 16 bits), and the maximum sample time for one A/D in the same 200 kByte block is 100 k samples, or 20 ms of data. Obviously, in this mode there is no blanking period associated with the address clock.

The commands in EM C are (Start C for a specific AD for n samples), (report Data Stream).

This mode will be useful in looking at address settling times, A/D noise, etc.

COMMUNICATIONS All cards send data to the CC by parallel point-to-point LVDS.

Cards only send data to the CC upon request.

All external commands and timing signals go to the CC only.

The CC issues appropriate commands to each slave card.

The CC only sends data to the DA PC upon request.

The CC can initiate shut-down without an external command if a dangerous condition is sensed. Because unsolicited messages to the PC are not allowed, if auto-shutdown is imminent, an error message will be stored in non-volatile storage on the CC.

The cards communicate to the Clock Card via asynchronous transfer. The details are described in the design analysis note `Backplane Asynchronous Communication` .

START UP SEQUENCE The normal power up sequence is as follows
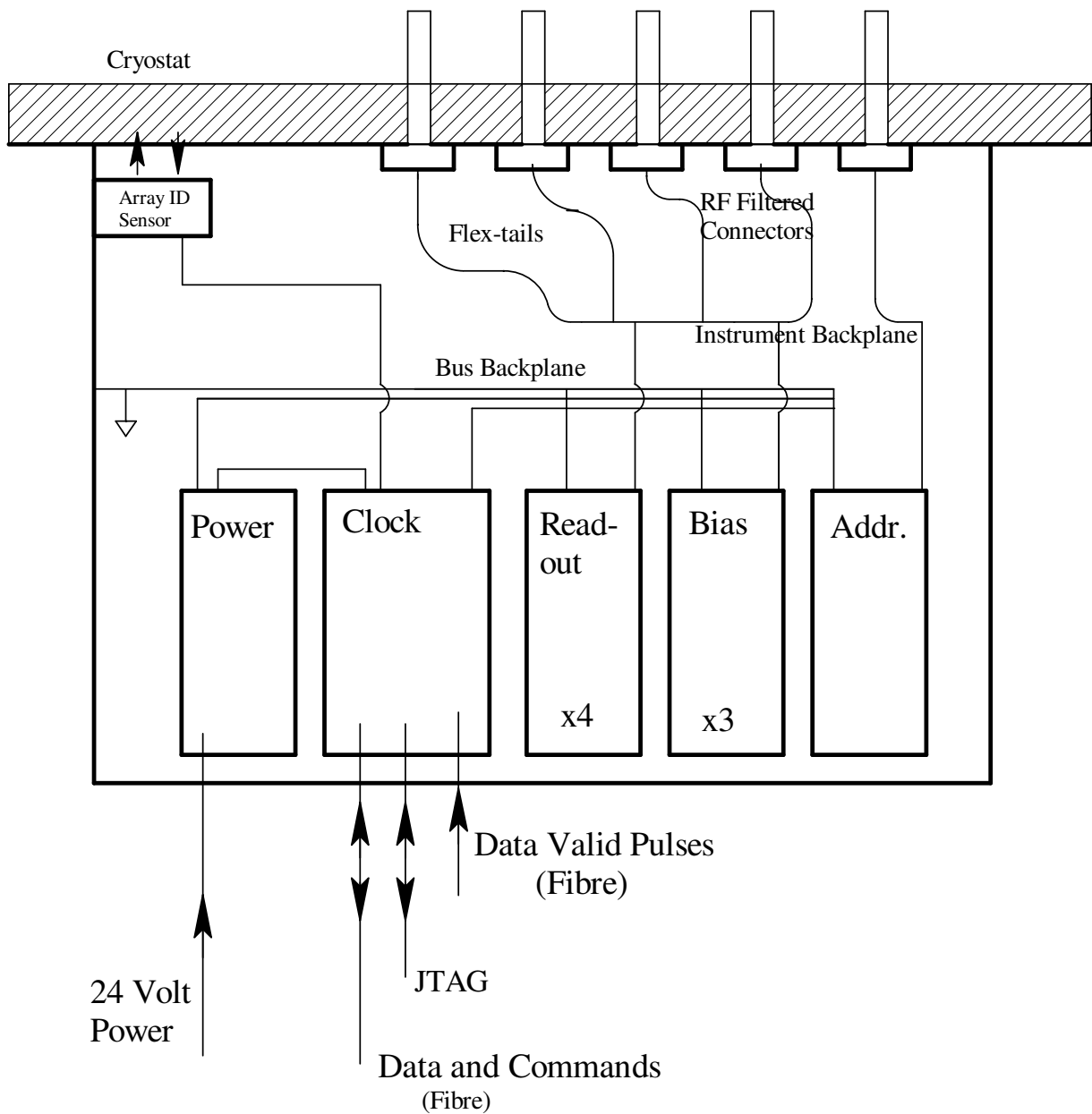-Reset Line is asserrted.
-Logic Voltage is turned on.

Figure 3: **Diagram of a single Chassis** The cryostat cables are wired point-to-point to the bias, readout, and address card via the mixed rigid/flexible Instrument Backplane. One of the filtered connectors contains only address lines and is connected only to the address card. The others contain mixed signals and wires fro each go to both the bias and readout cards, as appropriate. The Bus Backplane contains power, which is common to all cards, and also LVDS lines, which are point-to-point. Each Chassis reads its location optically from the cryostat, and each card reads its slot ID via connection to the Bus. The JTAG cable is normally not connected. In normal operation the only connections to the electronics are fibre connections to the clock card, and 24 Volts power to the power supply card.

-Analog Voltage is turned on.

-Reset Line is released.

-Each FPGA **except CC** is configured to 'Application Mode" from on-board flash memory.

-The CC FPGA is configured to a 'Factory Mode' which

Runs NIOS

Runs LVDS to other cards

Runs Fibre to DA PC

Acknowledges Commands from PC and

Can auto-start its own 'Application Mode'.

-Lock each PLL to 25 MHz

-Each Card reads its own status

-Reset **all** DACs to 0 Volts

-CC Polls all cards for status

-Wait N seconds.

-Then, if status is OK and no STOP command received from the PC, the CC goes to Application Mode, capable of running the data acquisition tasks described above. At this point, the electronics will do nothing unless a request is received from the Data acquisition Computer. Presumably the first request will be associated with setting up and biasing the array of amplifiers.