

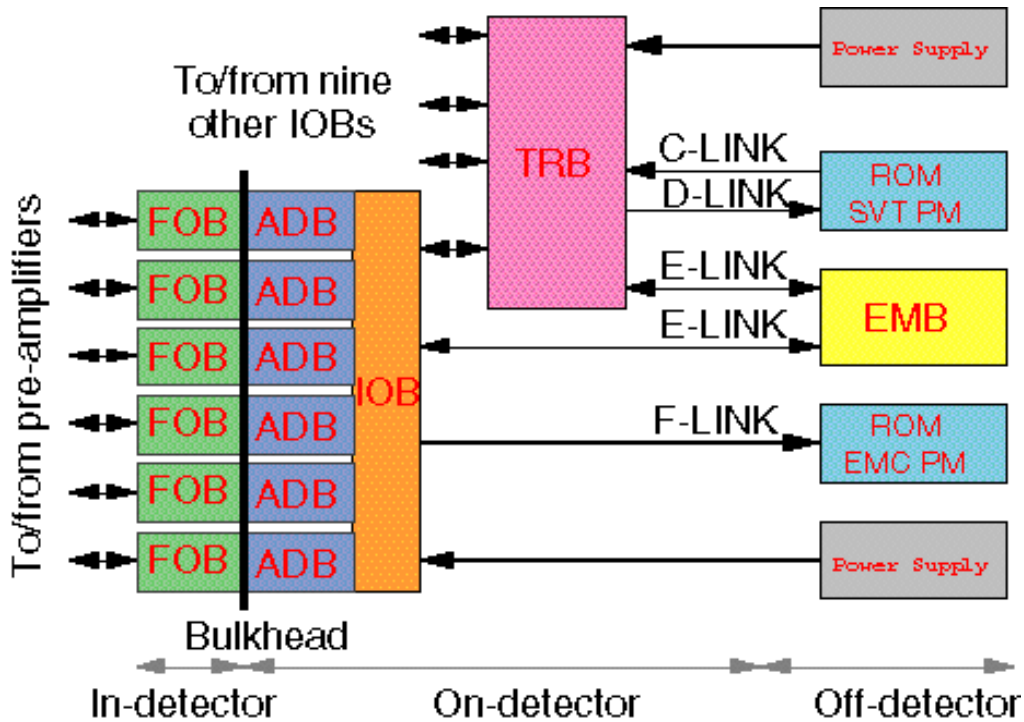
Software Tasks for EMC

We used a WBS structure just to keep track of what needs to be done and who will do it

- **1 EMC Online**
 - 1.1 Dataflow
 - 1.1.1 Configuration Databases
 - 1.1.2 Diagnostics
 - 1.1.3 Feature Extraction
 - 1.1.4 Calibration
 - 1.2 OEP
 - 1.2.1 Fast Monitoring
 - 1.2.2 Level Three
 - 1.2.3 Calibration
 - 1.3 Detector Controls
 - 1.3.1 Microcode
 - 1.3.2 Epics Displays
 - 1.3.3 Epics Databases

1.1.1 Configuration DBs

We will need a full Database Model of the interconnection of the system



- 1.1.1.1 Hardware Configuration
- 1.1.1.1.1 Preamps
- 1.1.1.1.2 ADBs
- 1.1.1.1.3 IOBs
- 1.1.1.1.4 TRBs
- 1.1.1.1.5 Fibres
- 1.1.1.1.6 PCs
- 1.1.1.1.7 ROM
- 1.1.1.1.8 ROCS/CLINKS
- 1.1.1.1.9 VME/FCTS
- 1.1.1.1.10 Interface to Trigger
- 1.1.1.1.11 Module/Crate Interconnect

Unique Descriptions

- A Channel is characterized by all of the logical/serial numbers of the components in the system.
 - Crystal Number
 - Preamp Number
 - Preamp Cable Number
 - FOB Channel, FOB Number
 - Crate Slot, Crate Number
 - ADB Channel, ADB Number
 - IOB Slot, IOB Number
 - Fibre Number
 - PC Slot, PC Number, Channel Number
 - VME Slot, VME Crate

1.1.1.2 Software Configuration

Version management, state management will be crucial to keeping the system running

- **GUIs for EMC SW use**
 - Tools for updating the configuration
 - Diagnostic Selection
 - Documentation/Help for the shift
- **Software Repository Management**
 - which version of code is current
 - temporary use of test code
- **Security**
 - who can update code
 - who polices code updates
- **Defaults/State Management**
 - Getting back to a safe known state (release)
 - Verification of the state of the system
 - Late night module replacement...

1.1.2 Diagnostics

Standalone Hardware Diagnostics

1.1.2	Diagnostics
1.1.2.1	Hardware Diagnostics (Standalone)
1.1.2.1.1	Preamps
1.1.2.1.2	ADBs
1.1.2.1.3	IOBs
1.1.2.1.4	TRBs
1.1.2.1.5	Fibres
1.1.2.1.6	PCs
1.1.2.1.7	ROM
1.1.2.1.8	ROCS/CLINKS
1.1.2.1.9	VME/FCTS
1.1.2.1.10	Interface to Trigger
1.1.2.1.11	Module/Crate Interconnect

- **Each Card will have a diagnostic suite**
- **both FCTS and VME commands will be used for diagnostics**
- **What is the interface for running diagnostics?**
- **This code will likely be the first developed, what do we need to do so that we can reuse it**

1.1.2.2	Hardware Diagnostics (Realtime)
1.1.2.2.1	Error Detection (ROM)
1.1.2.2.2	Error Flagging (ROM)
1.1.2.2.3	Error Recovery (ROM)

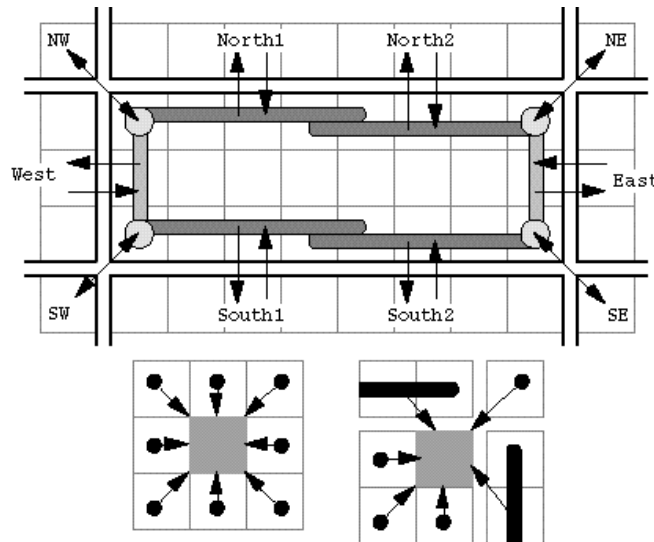
Interface to Trigger

The EMC Personality Cards form the EMC trigger towers

- **Each FLINK generates a trigger sum by adding together the 24 crystals on the fibre**
 - Crystals below a threshold are suppressed
- **The tower energy sum is sent to the EMC trigger processor via a front panel cable on the ROM**
- **The UPC can be put in a test mode which sends the module serial number over the trigger link**
 - This serial number is sent in response to a private EMC FCTS command (Start Spy)
 - The test packet also sends a bit pattern for checking bit/link integrity
- **This diagnostic will require both the EMC and the trigger in the same partition**

Module Interconnect

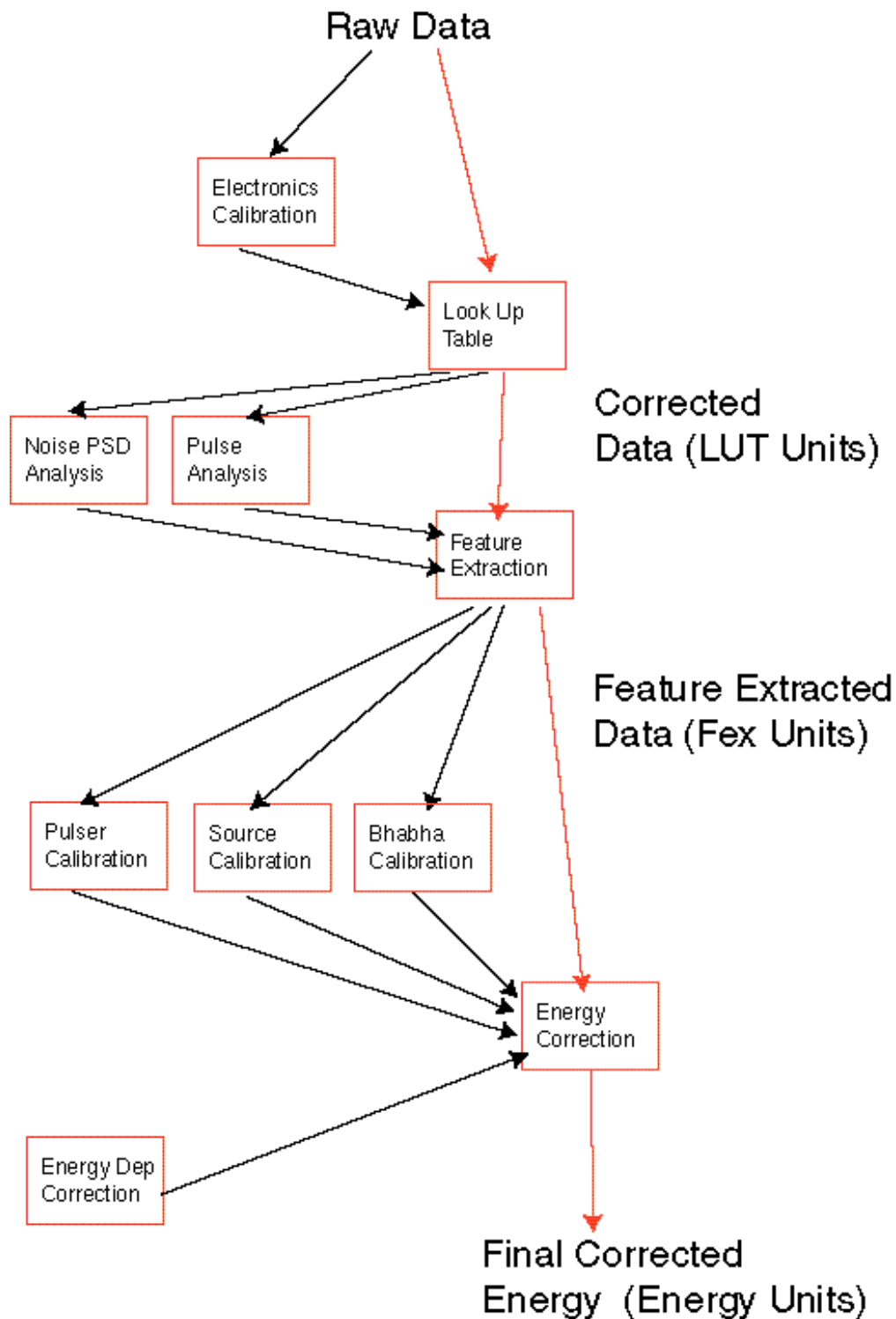
In order to reduce Computation, selective Feature extraction is performed



- **Each Fibre must determine which of its 24 crystals to Feature Extract using**
 - The FeX bits for the 24 channels on the Fibre
 - The 10 FeX bits from neighboring edges
- **Each Fibre sends 10 bits out across the backplane to neighboring fibres**
 - 60 Signals for each ROM
 - 3 Fibres
 - 10 out / 10 in for each fibre
 - Some neighbors are in adjacent crates
- **LUTs can be used to selectively set FeX bits and check the neighbor interconnect**

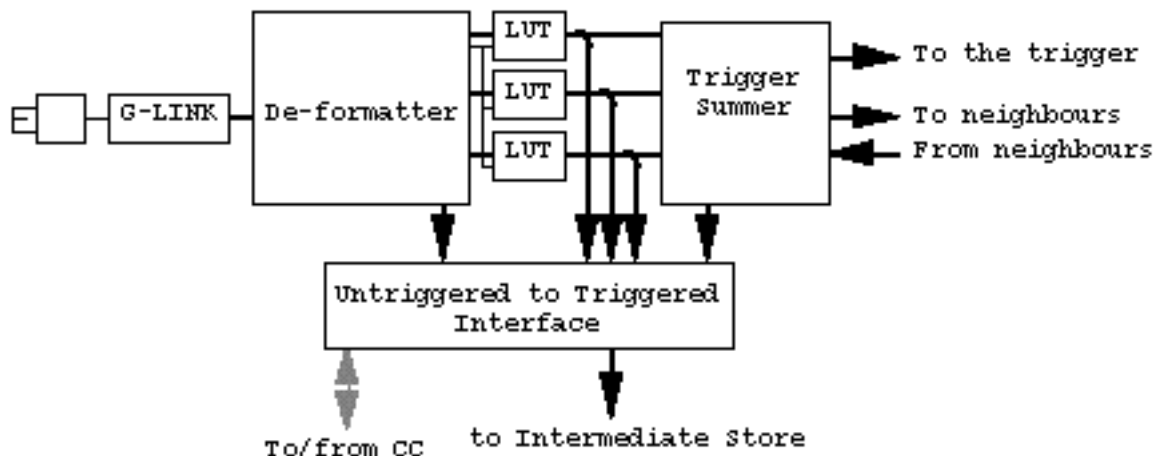
1.1.4 EMC Calibrations

There are 4 types of Calibrations in the EMC



Look Up Tables

The EMC LUT is used both for data correction and threshold cuts



- **12 Input Bits**
- **Output Bits (18)**
 - Corrected Data(16 bits)
 - Add Bit
 - Used by trigger summer
 - FeX Bit
 - Used for Selective Feature Extraction
- **LUTs change for source calibration**
 - requires a more sensitive LSB
- **Programming the LUTs will require**
 - Electronic Calibration Results
 - “Offline” Calibration Results

1.1.4.1 Electronic Calibration

The Electronic calibration could be handled entirely in the ROM

1.1.4.1	Electronics
1.1.4.1.1	High Precision Calibration
1.1.4.1.1.1	Pulsing Range Forced No LUT
1.1.4.1.1.2	Pulsing AutoRange No LUT
1.1.4.1.1.3	Piecewise Linear Fitting
1.1.4.1.1.4	LUT Generation
1.1.4.1.2	Rapid Constants Verification
1.1.4.1.2.1	Pulsing using LUT
1.1.4.1.2.2	Constants Validation
1.1.4.1.3	Constants Management
1.1.4.1.4	Failure Reporting
1.1.4.1.5	Realtime Pedestal Monitoring
1.1.4.1.6	LUT Management

- **The result of a full electronics calibration is enough information to construct the LUT**
 - Should be around 48 words/channel
 - Needs to be stored persistently
 - data taken with unit LUT
- **A normal calibration will verify the LUT**
 - uses the calibrated LUT
 - checks that the calibration is still valid.
 - No need to change LUT if the calibration is OK
- **Cal system commands used**
 - Set CAL/CARE control
 - Write CAL Dac

1.1.4.2 Pulser Calibration

The Pulser calibration is a low rate calibration that may run asynchronously

1.1.4.2	Light Pulser
1.1.4.2.1	Pulser event handling
1.1.4.2.2	Reference system readout
1.1.4.2.3	Local histogramming
1.1.4.2.4	Local Analysis
1.1.4.2.5	Constants Management

-
- **runs asynchronously during data taking**
 - Normal Feature Extraction
 - Histogramming selected by trigger bit
 - A trigger bit is set
 - One VME resident ADC will need to be read
 - Max rate is about 10 hz
 - **Dedicated pulser runs**
 - may require some VME writes to set DACs
 - **Reference system**
 - will need to be read periodically
 - again this is a VME resident ADC
 - will live in a EMC VME crate

1.1.4.3 Source Calibration

Data collection for the source calibration is completely non-standard

1.1.4.3	Source
1.1.4.3.1	Self Trigger
1.1.4.3.1. 1	Source LUT Generation
1.1.4.3.2	Feature Extraction
1.1.4.3.3	Local Histogramming
1.1.4.3.4	Local Analysis
1.1.4.3.5	Constants Management

• **Hardware Help for Self Trigger**

- UPC will check all FeX bits in an “event” and give this bit to the DMA supervisor
- The DMA supervisor can suppress the data for events where no FeX bits were set in the event

• **all ROMs run asynchronously, flat out**

• **Event Buffer**

- Set to allow one maximum size event
- One L1 Accept needs to be generated

• **Data Analysis**

- Fex with special LUT (higher sensitivity)
- Local Histograms kept