

2/13/97
B.Baldin
S.Hansen
M.Kozlovsky

Proposal for muon local GS synchronization

The synchronization of several muon crates in local data taking and testing modes was always an issue during Run I at D0. There are big advantages to have such a feature implemented such as following:

- ability to readout and combine information from different detectors in 'VAX local' mode, thus allowing small scale physics analysis of the data
- ability to run any part or whole muon system locally with beam or using cosmic triggers and pulsers without involving the TFW and D0 DAQ
- ability to mimic almost all TFW activities without having the TFW running allows to debug the muon system independently by using simple tools like scope and logic analyzer

In order to implement such a synchronization one needs to understand what 'VAX local' means. In Run I there were several programs developed to allow user to run a VAX based program which will allow to collect data from one muon crate (module), transfer it via Vertical Interconnect (VI) to the VAX and process it. Such programs allowed independent tests to be done in parallel with different crates and were found very useful tools. The attempts to do this with several crates simultaneously within one program have been not very successful due to the asynchronous nature of the VI interface.

In Run II configuration there is an option allowing to solve this problem. One can imagine to have a VME module (Trigger Fanout Card) that will distribute major timing signals and at least L1 decisions to all muon readout crates (GSs). The data readout can be performed by usual way via VIs, but trigger control will be similar to the standard TFW control with few minor differences. The Trigger Fanout Card (TFC) has to have the following features:

- four maskable inputs for trigger signals received from the L1 system or external source (NIM)
- one serial input for timing and control signals received from the external source (MFC)
- sixteen serial outputs to provide timing and control signals for the MFCs
- internal 53.1047 MHZ quartz oscillator
- internal "ready" latch set up by the L1 signals and reset by the VME command
- internal programmable crossing counter (8 bit)
- internal programmable turn counter
- internal FIFO memory for delayed L2 decisions
- VME slave interface

The serial links for this implementation can be AMCC S2043/S2043 chipset used by Arizona University in their L1 design. To reduce the bandwidth, one can use 10 bit mode and 531 MHZ bit frequency in this case. This will allow for seven 8 bit byte transfer every 132 ns. The clock frequency for the AMCC chips has to be 53.1047 MHZ. The set of signals for serial transfers can be arranged similar to the SCL spec like the following:

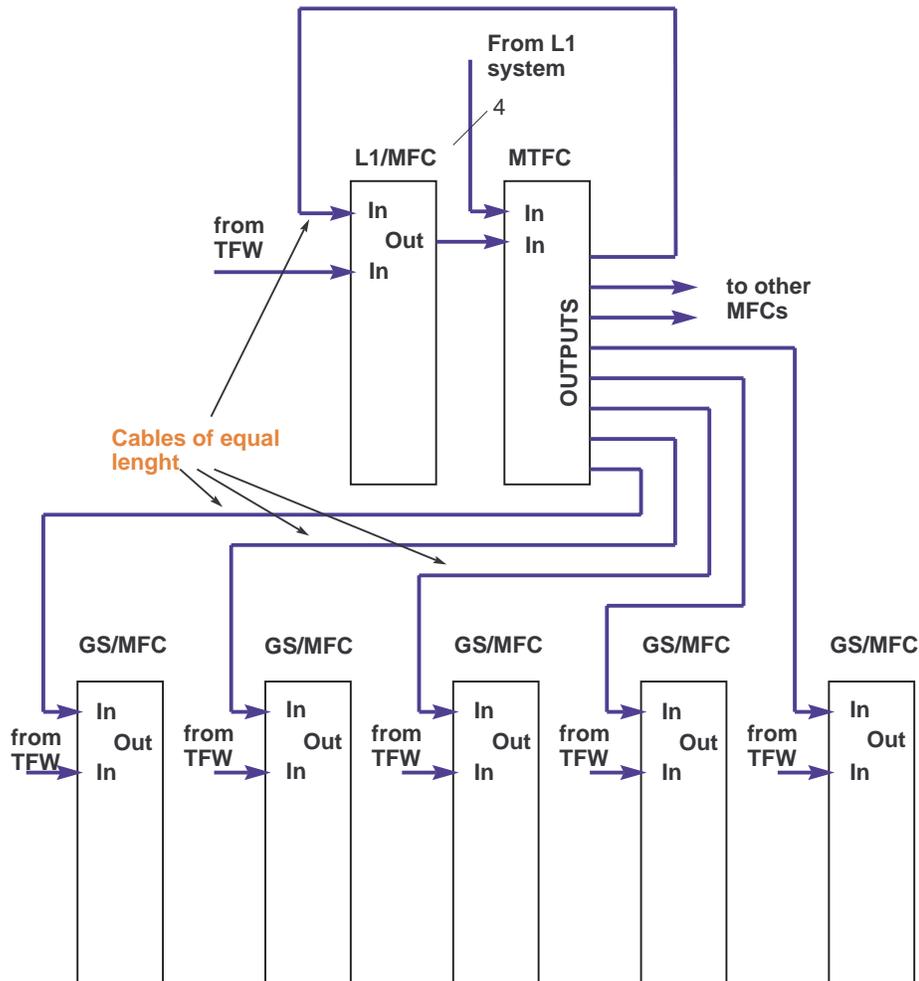
- INIT
- 53.1047 MHZ (recovered clock from the AMCC RCLK pin)
- First Crossing
- GAP
- SYNC GAP
- L1 Accept
- L2 Accept

- Crossing Number (8 bit)
- Turn Number (16 bit)

The additional feature necessary to implement to the MFCs are the following:

- serial transmitter re-transmitting the SCL signals with the output at the front panel connector
- serial receiver to accommodate timing and control signals distributed by the TFC

It is necessary to mention that the cable connection between TFC and MFCs seems to be very short (no longer than 15..20'). This allows to provide equal cable lengths for all MFC connections and eliminate unwanted re-adjustment of the timing delay within front-ends. Though due to the additional propagation delay for the timing signals coming from the TFW it may be necessary to make common change in all front-ends simultaneously when collecting data from the beam in the local mode.



Additional Muon Fanout Card features:

- one serial input for the MFC synchronization
- one output with serial data encoded timing signals

Muon Trigger Fanout features:

- one serial input for the MFC synchronization
- four maskable inputs from L1 system (NIM)
- 16 outputs with serial data encoded timing signals
- internal 53.1047 MHz quartz oscillator
- internal "READY" latch controlled by VME
- internal programmable crossing counter (8 bit)
- VME slave interface