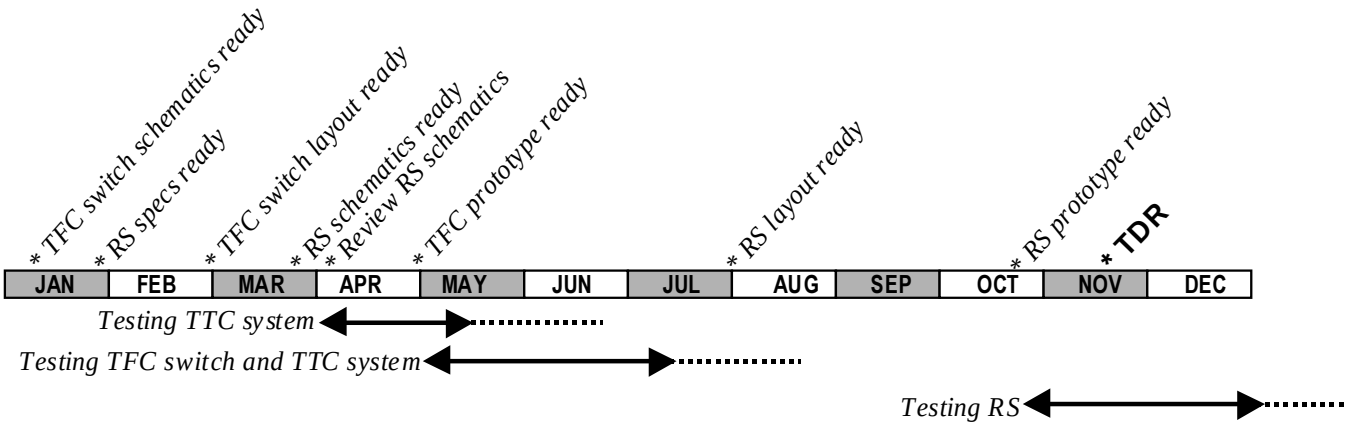
Progress and status of the TFC system

Please, refer to the documents "The LHCb TFC system", "The TFC Switch Specifications", and "Readout Supervisor Specifications" on the TFC Web page (http://lhcb-comp.web.cern.ch/lhcb-comp/daq/TFC/html/TFC_intro.html) for an overview and the details of the system.

In view of the upcoming TDR the aims for this year are to:

- Design the TFC components endemic to LHCb - The Switches and the Readout Supervisor. Specifications were finalized last year.
- Review the TFC architecture and the components endemic to LHCb.
- Layout the first prototype of the TFC Switch and the Readout Supervisor.
- Simulate the Readout Supervisor at several levels.
- Set up a test bench to make feasibility tests of the way the TFC architecture exploits the TTC transmission system.
- Produce a first prototype of the TFC Switch and the Readout Supervisor.
- Test critical points of the TFC Switch and the Readout Supervisor.

Below is an overview of the planned schedule for this year. With the exception of a delay in the area of testing the TTC system, the schedule has been maintained up to now. The delay is mostly related to difficulties using the TTC PCI/PMC receiver designed by ATLAS. The details are outlined below.



1. Switches

The Switches (TFC Switch and Throttle Switches) and to a large extent the TFC architecture was reviewed by internal and external reviewers 8 November 2000. Except for a few detailed comments and suggestions concerning the schematics, the Switch specifications and the design were felt adequate. The TFC architecture was very well received.

The first prototype was produced and mounted in April-May without problems somewhat later than planned due to late delivery of components. The main aim with the first prototype was to measure two crucial quantities:

- Time skews between the different switch paths. As a given FE may use different Readout Supervisor for testing at different times, it is of crucial importance that the internal time skew between different paths in the Switch be minimal (initially aimed at <100ps). Otherwise, the FE will suffer from timing alignment problems.
- Contribution to the jitter on the clock distributed to the FE.

As the Credit Card PC was not available, the Switch was configured and tested by downloading directly the Switch configuration to the onboard FPGA via external JTAG. Using the board with the Credit Card PC as on board controller remains to be tested.

All paths functioned properly and all measurements of the performance were carried out. Measurements were done between 20 – 100 MHz and mostly with an oscilloscope at 25 GS/s (40ps resolution). Below is a summary of the results:

- The output jitter (FW) was measured to be typically 80 – 100 ps. The input jitter from the generator was measured to be ~50 ps.
- Skew was measured from input 1 to all outputs with output 1 as reference. This measurement thus gives the skew in the output path (16:1MUX to output) only:

Output	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Skew(ps)	0	3730	-65	3150	-95	2630	-85	2060	-295	2720	-345	555	-250	345	-465	-580

- The total path delay was measured between all inputs and all outputs. From the results one can draw the conclusion that the maximum skew between all the inputs to each multiplexer is between 100-400 ps. The trend in the variations is the same from one multiplexer to another. These measurements were also consistent with the measurements above.
- The internal propagation delays of the multiplexers were measured. This varied between 400 ps – 1000 ns +/- 100ps (Specs claim maximum 850 ps with a 50 ps internal variation from path to path). This is an unavoidable source of skew.
- In order to understand better the skews in the output path and a discrepancy with the predicted total delays calculated from the line lengths (using 5ns/m or 7.87 mills/ps), the propagation delays between the output of the multiplexers and the inputs of the

NORs were measured. These path consist purely of signal lines. It was found that there is a ~40% underestimate in the propagation delays more or less consistently as if the speed of the signal is not 5 ns/m but rather 40% slower. A cross-check with the total propagation delay was done by calculating the delay based on the line length, and adding 200 ps(specs) for the receiver, the measured value for the multiplexer and 300 ps(specs) for the NOR. For the path in1-out1 this gives 4.4 ns to be compared with measured 7.7 ns (43 % slower). For in1-out2 the calculated is 6.1 ns compared to 11.4 ns measured (46 % slower).

A few mistakes were found in the layout. They can account for the grater part of the skews but only assuming the overall 40 % slower signal speed. If the reason is the dielectric characteristics of the board it will be different for the surface layer and internal layer, thus leading to inevitable skews even if the lines would be exactly of the same length. In addition, the variation in the internal propagation delays from chip to chip will contribute with several hundred ps to the skews.

From these studies it is evident that it will be practically impossible to attain a max skew of 100ps from path to path. A solution would be adding delay chips at the outputs. This is under investigation. However, an important issue is the temperature dependence of delay chips.

Summary

The performance of the first Switch prototype has been carefully measured. The jitter on the clock at the output of the Switch is at an acceptable level. Some errors were found in routing the lines to equalize the path lengths. These will be corrected.

However, the measurements show that, even after correcting the path lengths, the current design does not have the level of performance required with respect to the time skews. The solution is to route all lines on board layers with the same dielectric characteristics, and add appropriate delay lines at the outputs to compensate for the intrinsic skew of the components. It may also be of interest to choose input and output coupling capacitors with less tolerance to improve the signal shape.

The Switch has not been tested yet with the Credit Card PC due to unavailability and nor with a TTC receiver at the output due to the problems with the TTCpr.

The first prototype will also be sufficient for the initial tests of the first prototype of the Readout Supervisor.

2. Readout Supervisor

The Readout Supervisor is almost entirely based on FPGAs. The logical design of the PLDs together with a first draft of the schematics was ready by the end of March. The specifications and the logical design were put through a review with both internal and external reviewers 4 April 2001. A few detailed functional suggestions were made. Otherwise, the specifications, the design, and the flexibility were very appreciated. The importance of simulation was emphasized.

The specifications of the Readout Supervisor has been simulated in a high level behavioral model together with a behavioral model of the LHC machine, the trigger decision units and the FE by Jorgen Christiansen, Ivan Garcia Alfonso, and Pablo Vasquez.using VisualHDL.

The FPGA designs have been simulated using MaxPlus all along the design phase. In order to check the designs and crosscheck the MaxPlus simulations, some of the blocks have been simulated at gate level using LeapFrog.

The behavioral model of the LHC machine, the trigger decision units, and the FE has also been refined in order to support a VisualHDL simulation of the real RS design. The behavioral model of the RS is currently being replaced block by block by the FPGA implementations at gate level including delays in order to simulate the logical blocks together. The L0 trigger path in the RS has already been simulated and shows that the current design, using three or four clock, cycles works. The work to implement the entire RS in the simulation is continuing.

The RS interfaces to the L0 and the L1 Trigger Decision Units have been agreed on with the people involved.

A minimal version of the Readout Supervisor is currently in production. The layout was ready at the end of August. The board is now being produced. The minimal version will have all crucial paths implemented and almost all the functionality. What has been removed is essentially the internal FE of the RS (analogous to a normal FE sampling the RS event information) and some of the optional state machines to send triggers and commands. There are also fewer counters. For more details see the last chapter in the “Readout Supervisor Specifications” document.

The aim with the first prototype of the Readout Supervisor is to verify that the FPGAs are sufficiently fast with safe margin for the Readout Supervisor functions requiring synchronous operation. This may force use of smaller FPGAs in which case the problem of space and routing has to be assessed. The aim is also to measure the performance and to check the concurrent operation of the many functions.

The first prototype of the Readout Supervisor will also replace the use of the TTCvi and the TTCvx encoder currently being used in the TTC test bench.

3. TTC system

As the distribution of L1 triggers and commands to the Front-End requires transmitting ~1.1 MHz of short broadcasts over the TTC system, it is a crucial point to verify. Lacking a Readout Supervisor, a test bench was devised using existing equipment:

ALEPH FIC → TTCvi → TTCvx → TTCtx → (optical tree-coupler) → TTCpr

The first tests without a TTCpr (it was not available) but using a scope showed no problems broadcasting short channel-B commands at 1.1 MHz. Approximately 1.6 MHz was measured. Of course this does not prove the integrity of the broadcasts.

As the Readout Supervisor will use the same encoder circuit as the TTCvx and we are planning to use the TTCtx, the test bench has also been useful to gain experience with the equipment and study the performance.

The TTCpr is a PCI (or PMC) receiver that carries a TTCrx. The incoming broadcasts on channel B are channeled into a FIFO. An FPGA handles shipping the broadcasts over to the host PC via PCI using DMA. The original version of the TTCpr for ATLAS is designed to receive L1A triggers, that is, upon receiving a trigger on channel A it expects a long broadcast and stores it in the FIFO together with the Event ID.

With help from ATLAS, the FPGA code was adapted to handle short broadcast and ignoring the activity on channel A.

Currently, two problems are under investigation. The TTCpr with the new FPGA code seems to receive the short broadcasts but the transfer over to the host PC does not work.

In order to still verify the 1.1MHz broadcast rate or rather the same throughput, we attempted sending L1A, and hence long broadcasts, at a rate which would give the same throughput (1.1 HMz * 16 bits / 42 bits = 420 kHz). However, it seems broadcasts are lost because the EventIDs in the buffer in the host PC show jumps of between one to three broadcasts. It is not yet understood whether the problem is in the TTCvi or in the transfer between the TTCpr FIFO and the host PC. At a low rate (tens of kHz) the broadcasts arrives correctly.