Available online at www.sciencedirect.com

ScienceDirect

ELSEVIER

Physics Procedia

Physics Procedia 00 (2011) 1–8

The EDRO board connected to the Associative Memory: a “Baby” FastTracKer processor for the ATLAS experiment

Annovi A.[a], M. Beretta[a], V. Bevacqua[b,h], F. Cervigni[b,h], F. Crescioli[b,h], L. Fabbri[c,g], P. Giannetti[b], F. Giorgi[c], D. Magalotti[e], A. Negri[d], M. Piendibene[b], C. Roda[b], C. Sbarra[c], M. Villa[c,g], R.A. Vitillo[b], G. Volpi[1]

[a]INFN - LNF Via E. Fermi, 40, 00044 Frascati (Roma), Italy
[b]INFN - Pisa, Largo B. Pontecorvo, 3, 56100 Pisa, Italy
[c]INFN - Bologna, Viale B. Pichat, 6/2, 40127 Bologna, Italy
[d]INFN - Pavia, Via Agostino Bassi, 6, 27100 Pavia, Italy
[e]Università degli Studi di Perugia, P.zza Università, 1, 06100 Perugia, Italy
[f]University of Chicago, Department of Physics, 5720 S. Ellis Ave, Chicago, IL 60637, USA
[g]Università di Bologna, Via Zamboni, 33, 40126 Bologna, Italy
[h]Università di Pisa, Lungarno Pacinotti, 43, 56126 Pisa, Italy

Abstract

The FastTracKer (FTK), a hardware dedicated processor, performs fast and precise online full track reconstruction at the ATLAS experiment, within an average latency of few dozens of microseconds.
Before production of the final system for tracking in high-occupancy conditions with the best of available technology, we plan to use existing prototypes of the FTK hardware to exercise its functions in the ATLAS environment.
We describe the “baby FTK”, consisting of a few hardware elements implementing the first stages of the system, and discuss our plans to grow the system into a full-functionality FTK “vertical slice” covering a small projective wedge of the detector. We report on the performances and structure of the “baby FTK”, including the pixel/strip hit clustering (clustering mezzanine), hit organization and distribution (EDRO) and the Associative Memory pattern recognition function. We describe briefly also the possible future evolution including the addition of the Track Fitter.

Keywords: Tracking, Trigger concept and systems, Digital electronics, Associative Memory, FPGA
PACS: 29.40.Gx

1. FastTracKer & the Vertical Slice

The FastTracKer (FTK) [1] has been approved recently by ATLAS as a trigger upgrade for the first phase of the LHC luminosity upgrade (Phase I, up to $\mathcal{L} = 2\times10^{34}$ cm^{-2}s^{-1} and 40 pileup events). It is a very challenging design that will require the next 4 years to be completed for a first data taking in 2015.
The FTK algorithm is conceptually made of two pipelined processors, the Associative Memory finding low precision tracks, and the Track Fitter refining the track quality with high precision fits. The Associative Memory has a bank of possible low precision tracks (patterns) and it is able to find in real time which tracks are present in the event (roads). The Track Fitter performs a linear fit of all combinations of full-resolution hits inside each road to reconstruct offline-quality tracks. The final FTK system will be made by 128 core pipelines, performing the described algorithm in parallel on different sectors of the detector covering the full volume. The core pipeline will use 8 out of 11 silicon layers available. After this first stage all tracks are collected by 32 final processors that performs a second linear fit using 11 layers (the candidate 8 layers track plus 3 extrapolated hits from the missing layers).

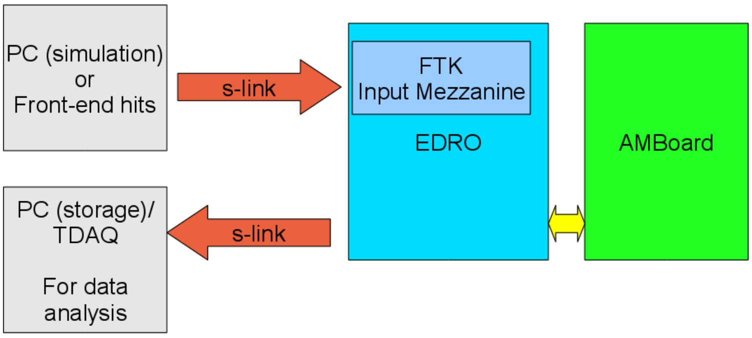
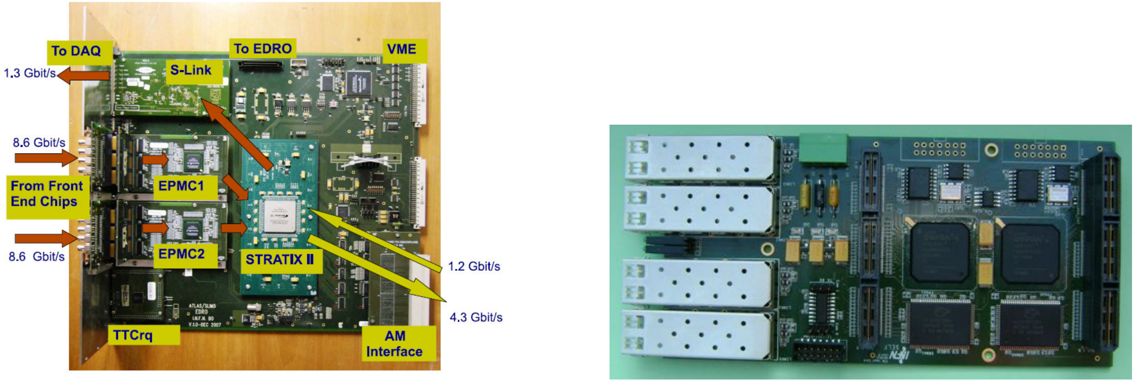


Figure 1: **EDRO-AM Standard setup**

We need to gain experience on the ATLAS TDAQ system and test the integration of the FTK functions in the experiment during this complex design effort, well before production. To understand the system issues and to develop the needed control software, we plan early parasitic commissioning of a small proto-FTK, based on existing prototypes and able to reconstruct tracks inside a narrow azimuthal slice (tower) of the detector. Parasitic commissioning means that there will be no impact on normal ATLAS data taking, thanks to a duplicated additional output fiber provided for FTK by the tracking front-end. The data flow check can be disabled on the FTK channels allowing ATLAS to take data regardless of FTK status. The FTK output can be written to the calibration stream for off-line studies.
We call this proto-FTK a “vertical slice” because it will be small (operating on a slice of the detector) but functionally complete from the detector inputs up to the track output available for the L2 CPUs.
The first elements tested together, in a standalone configuration, were prototypes developed for the SLIM5 collaboration [2]: the EDRO board (Event Dispatch and Read-Out) and the AMBslim board [3]. Recently we have added the input mezzanine [4] (FTK Input Mezzanine, FTK_IM) to receive data from the detector and to perform hit clustering. The EDRO board is now able to receive on the FTK_IM clustering mezzanine detector raw data on S-links. The FTK_IM can calculate the pixel and SCT cluster centroids. In the laboratory, initially, the detector data will be produced by a “pseudo front-end” (a CPU). The clusters will be transferred to the AM board that finds low resolution candidate tracks called roads, to be provided back to the EDRO. After tests in the laboratory, the vertical slice will be moved to the experiment and will spy real data during normal data taking. Its development is divided into two stages. During the first stage (the ”baseline vertical slice”) the EDRO will deliver the found roads from the AM board to the CPU using the S-link connection. The roads will be received by the ATLAS DAQ by dedicated L2 CPUs and written in the calibration data base. The remaining event processing will be completed by running the FTK track fitting simulation on the collected data. This setup (see figure 1) is very similar to what already exists and should be enough to develop the CPU software and the FPGA firmwares, to develop and test the integration in ATLAS, to discover early possible technical problems and to provide a test stand where prototypes of the missing elements of the system can be tested on real data. The size of the detector tower in the baseline configuration will be small, since we will use only a single AM Board, with a limited associative memory bank size (see section 2).

In a second stage the vertical slice could grow again to cover a larger detector portion, to include a hardware track fitter and to be used for real triggering in the experiment. This can happen only if the run before the year-long shutdown to prepare the LHC for higher collision energy will be long enough to allow the full development of the first stage and the extension to the next stage. The vertical slice will continue to exist as a test stand for the new prototypes while a new extended version, the FTK demonstrator, will take data. The demonstrator will use the GigaFitter board from the CDF experiment [5] to perform track fitting. It will use a new EDRO firmware with the capability to associate full-resolution silicon cluster data with the roads found by the AM to provide data to the GigaFitter.
The physics case for a possible early small demonstrator is under study. The most simple application would be the search of the primary vertices in each event and the calculation of the beam spot. A second possible application would be the identification of slow massive particles at level-2. They would appear in the silicon detector as highly ionizing, isolated, “muon-like” tracks and in the barrel tile calorimeter as late (slow) clusters. All these criteria can be applied at level 2 to select muon-like particles of 15 - 20 GeV transverse momentum.



(a) (b)

Figure 2: **(a) EDRO Board & (b) FTK Input Mezzanine**

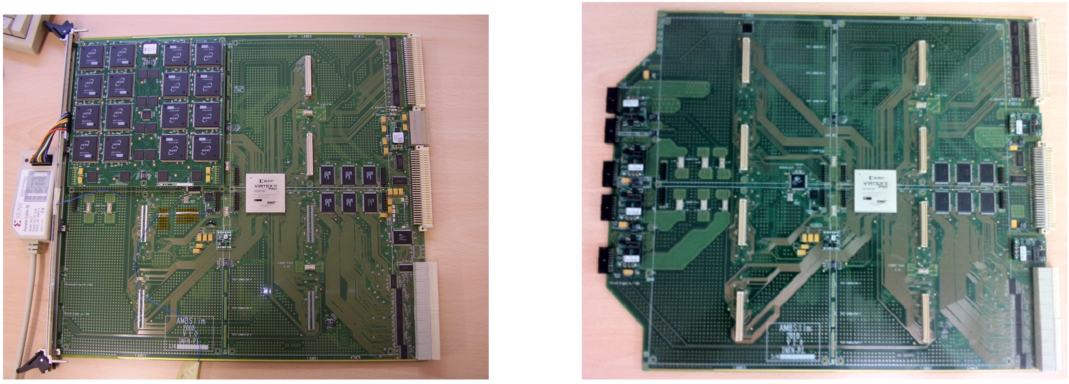
In the remainder of these proceedings, we report on the performances and structure of the ”baby FTK”, the first nucleus of the vertical slice, including the pixel/strip hit clustering (clustering mezzanine), hit organization and distribution (EDRO) and the Associative Memory road funding function. We describe the pieces of the test stand in section 2, the software infrastructure in section 3, the performed tests in section 4. In section 5 we briefly sketch the possible future developments.

2. The existing baby FTK

The setup of the existing baby FTK is described in figure 1: the EDRO is equipped with the FTK Input Mezzanine and connected with an AM board through the P3 connector. The system can receive hits from a PC or from the silicon RODs through an S-link, but until now we have not used the S-link and we simply generate hits inside the FTK_IM or the EDRO board or the AM board. Hits can be read from a file and can be written by VME to an input FIFO that is available inside each board. The FIFO output is disabled until the events have been completely loaded. When the FIFO output is enabled the events are sent downstream at 40 MHz, the clock frequency of our system.
The EDRO also has the capability to generate loops of events with a small number of straight tracks.

2.1. EDRO

The EDRO (Event Dispatch and Read-Out, in figure 2a) is a fully programmable 9U VME board. The EDRO board is designed for high-performance and has been used in a variety of DAQ environments, including the LUCID forward luminosity monitor of ATLAS and in the SLIM5 R&D experiment. It is based on 5 mezzanine slots: one main mezzanine hosting a Stratix IV FPGA that is the brain of the board, one mezzanine to receive a 40 MHz external clock (TTCrq), one mezzanine for S-Link output and two slots for programmable input mezzanines (EPMC). The EPMC slot is compatible with the FTK Input Mezzanine. It has a dedicated P3 connector for communication with the Associative Memory board. It is able to sustain a combined 17.2 GBit/s input rate on the EPMC slots, an input/output rate of 40 MHz with the Associative Memory board, and a 1.3 GBit/s output rate through the S-Link cable.
All central functions of the EDRO board in the baby-FTK system have been already tested. It has successfully received hits from the FTK Input Mezzanine and sent them to the Associative Memory. It has received from the Associative Memory the patterns found in each event (roads) and it has sent the triggered events (all hits + all roads) to external storage via the S-Link output (see section 4).



(a) (b)

Figure 3: **AM Board** - (a) standard VME VIPA version (b) custom 48V version

2.2. FTK Input Mezzanine

This mezzanine (shown in figure 2b) will receive the hits from the Pixel & SCT RODs and perform the clustering algorithm for 2D pixel hits [4] and 1D association of SCT strips clusters.

This is the very beginning of the FTK pipeline so it was useful to have this hardware early to connect it to the detector. It has been developed to be compatible with both the EDRO for Vertical Slice use and the final FTK Data Formatter board.

The mezzanine is based on two Spartan VI FPGA from Xilinix, each receiving two S-Link optical channels from the silicon detectors RODs. The two FPGAs have a SRAM each to store data for the clustering algorithm. The mezzanine is able to sustain 40 MHz input data rate over the S-Link channels and delivers clusters in real time to the hosting motherboard. This step is very important in the FTK chain: optimal cluster finding will reduce the amount of data transferred to the associative memory and it will decrease the amount of duplicate found roads, which might increase the track fitting time if not correctly suppressed. The track fits are more precise if they use better cluster coordinates. Different clustering algorithms are under study in the simulation for the pixel detectors. Up to now we did not use the clustering in the baby FTK, but we simply used the FTK_IM to inject cluster centroids in the system reading them from disk and loading them by VME as described above.

2.3. Associative Memory Board

The Associative Memory board (shown in figure 3) that will be used in the Vertical Slice is an intermediate prototype between the board used at SLIM5 and CDF [6] and the final FTK board, currently under design. The intermediate prototype is still based on CDF associative memory chip (AMchip03 [7]) and it is compatible with the LAMB mezzanines from CDF that can handle 16 or 32 AM chips. It has a maximum capacity of 640k patterns using 4 LAMBs with 32 associative memory chips each. The final associative memory board of FTK will require a chip with much larger pattern capacity [8].

Like the final FTK board, the prototype AM board receives a separate bus for each input layer, to load hits from different layers in parallel. The prototype has 6 buses since the AMchip03 supports only 6 layers, while the final FTK board and final AMchip will have 8 buses. It is able to sustain a 40 MHz input rate. Also the road output rate is 40 MHz. It can work in pipeline with other Associative Memory Board: it has an input bus for roads found by the previous AM board and 6 output buses to send hits to the next AM board.

The prototype board has been developed in two versions. An older version (figure 3a) works on crates with a standard VME VIPA power supply and backplane but can support power only for 64 AM chips on the board. The latest version (figure 3b) needs a custom VME VIPA backplane with additional 48V power source (4 pins) to provide the needed power for 128 AM chips per board. An extension of the board in the front has been necessary to allocate large DC-DC converters from 48 Volts down to 1.8 V, the core AMchip voltage. The board has 6 DC-DC converters, each one providing a maximum of 25 A at 1.8 Volts, for a total of 150 A and a maximum power of 270 W. These AM boards

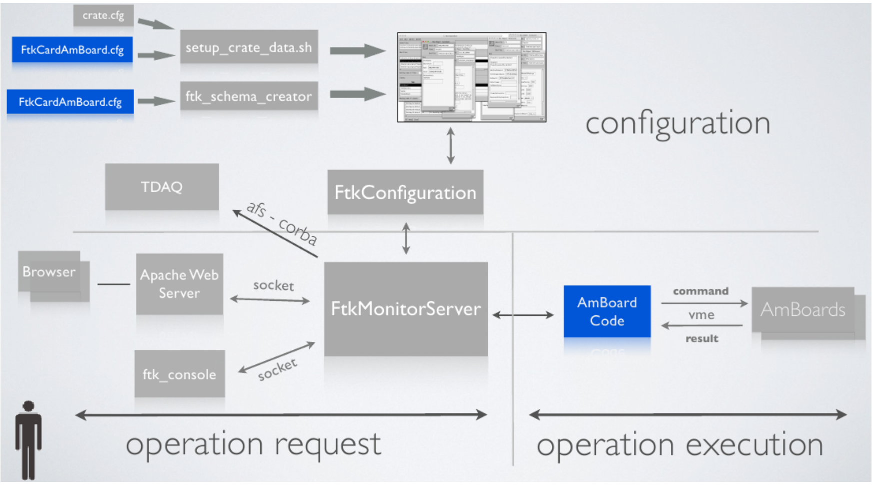


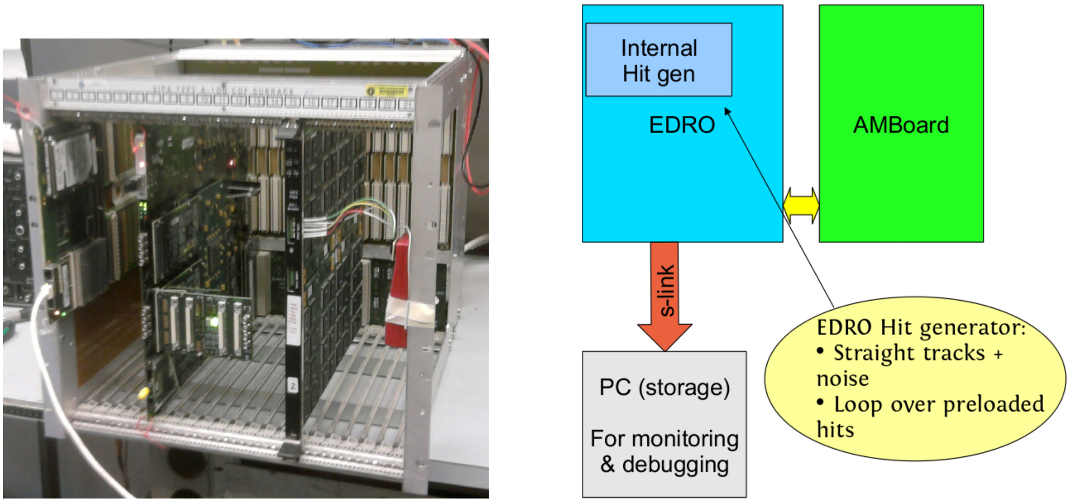
Figure 4: **FTK Vertical Slice software infrastructure** - a view of the tools for the FTK Vertical Slice infrastructure. In blue are highlighted the only parts specific for each HW component. A detailed description is in section 3.

will be fundamental also for cooling tests in the FTK custom crates.

3. Software Infrastructure

Several software tools were developed for the Vertical Slice, most of them will be useful also for the final FTK project.
The ATLAS experiment has a complex and extensive software infrastructure to configure, control and monitor its hardware components. We needed to develop the code specific for our hardware within this infrastructure. On the other hand we needed also a stand-alone set of tools for rapid and effective development of the hardware in-house.

For this reason we developed our tools in a modern client/server architecture with plug-ins.
With our architecture we are able to write once the low level control and monitor routines and automatically export their functionalities to high level programs, both stand-alone and within the ATLAS infrastructure.
This software model allows us to add new hardware (ie. a new prototype of AM board, with different registers or different AM chips) or new functionality (ie. a new quantity to monitor, a new histogram) with a minimal amount of new code to write.
This infrastructure is shown in figure 4. There are three main areas: “operation execution”, “operation request” and “configuration”. The “operation execution” area is where the actual low level code interacts with the hardware. The blue box in the figure is hardware specific and we have to add new code (a new blue box) for new hardware.
The “operation request” area imports all the libraries for hardware interaction and exposes all functions to the user or other software tools. This area is implemented in a client-server model: the FtkMonitorServer implements a simple programming language to access the low level libraries. Various clients send requests to the server in this language and translate the result to the user/software. We have written three clients: (a) command-line console, for fast scripting and tests during the development phase, (b) web application with AJAX interface, for high level monitoring and interaction, but independent of the ATLAS TDAQ infrastructure, (c) TDAQ segment to interact with the ATLAS TDAQ infrastructure and control our hardware from the standard ATLAS TDAQ software. The clients can interact with the server at the same time without conflicts. Simple authentication and authorization is possible and under development.
All the “operation request” area is hardware independent and no new specific code must be written to add new hardware if the new hardware implements the abstract objects supported (VME registers, memories, spy buffers, etc.)



(a)

(b)

Figure 5: **EDRO-AM Test crate & setup** - (a) from left to right: VME Linux SBC, EDRO, AM Board with 4 LAMBs (320k patterns) (b) schematic interconnection setup

The last area is the "configuration" area. The ATLAS TDAQ infrastructure needs many configuration files with detailed informations on every individual piece of hardware and software. In order to ease the development we have written several generators for these configuration files. The generators take as input very simple configuration files, one for each specific hardware type (ie. register addresses and names), highlighted in blue in the figure.

4. Tests results

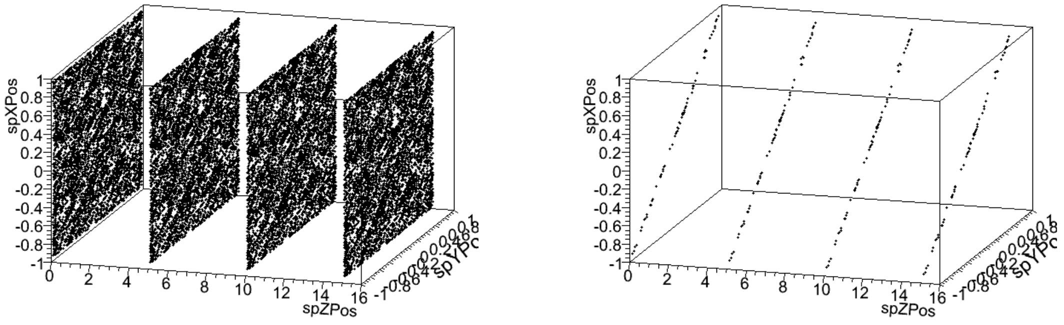
The Vertical Slice is made of flexible and modular components, so there are various possible test setup configurations to verify the correct behavior of the hardware and software in stand-alone tests and in the integrated environment. The baby FTK is a simple setup described in figure 5b: the EDRO and the AM board are in the same crate, connected by the P3 backplane, the EDRO board outputs data to an external PC. Figure 5a shows a photo of the crate and boards we used for the first tests.

The EDRO board has the ability to generate very simple and repetitive events or to send a preloaded list of hits, which has been described previously. To debug events that fail, the two boards provide circular memories called spybuffers where a snapshot of the inputs and outputs is copied. We can freeze the spybuffers and read them through VME to understand where the error happens.

This configuration has been used to test the connection between the two boards. We preloaded hits to scan every bit pattern and then looked at the spybuffers of the two boards to check that every bit is transmitted correctly.

We have tested basic operations at full speed using the internal generator of the EDRO to generate hits for single straight-track events uniformly hitting a 4-plane 2D silicon telescope. Figure 6a shows all the generated hits distributed among the four planes. We used a pattern bank to select only the tracks crossing the diagonal. Figure 6b shows the tracks that were selected by the AM board and written to disk by the EDRO. The system is able to select the wanted tracks at the full 40 MHz hit rate.

Finally we also performed a test with random hits and a random pattern bank. We downloaded the hits through the EDRO, we collected the roads found by the AM board on disk and we compared them to what expected by the simulation of the system. By generating enough random inputs and patterns, we verified that the EDRO and the AM board produce correct results for all possible hits and patterns.



(a)

(b)

Figure 6: **EDRO-AM Straight tracks test** - (a) 2D Hits on four planes (8 coordinates) from straight tracks generated by EDRO internal generator (b) Selected events by AM board loaded with diagonal tracks patterns

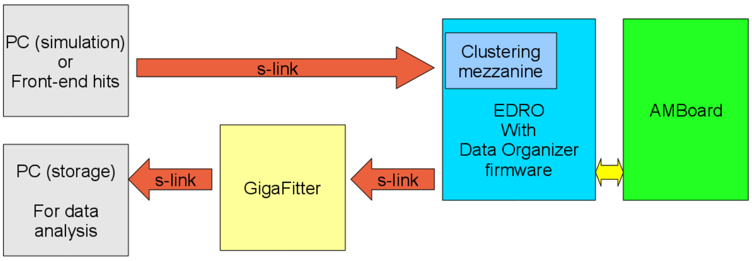


Figure 7: **Vertical Slice core demonstrator** - The EDRO with Data Organizer function and FTK_IM mezzanines, attached to AM Board for pattern recognition and GigaFitter for track fitting

5. Future developments towards a demonstrator

In a later stage, when the Vertical Slice will become a demonstrator, the EDRO will have also the functionality of the Data Organizer. The Data Organizer is a smart database that associate the roads found by the Associative Memory to the corresponding hits. With the Data Organizer function it will be possible to add the second stage of the FTK algorithm: the linear track fitting stage. The EDRO will output data to the GigaFitter board to perform track fits (see figure 7).

The size and shape of the detector region covered by the demonstrator will be chosen to favor the selected physics case. We studied the size of a bank in a single barrel ($|\eta| < 1$) detector wedge 45° large in the transverse plane. We use the 3 pixel layers and the 3 inner axial SCT layers with a road size defined by 48 pixels in the transverse plane (2.4 mm in ϕ), 40 SCT strips (3.2 mm) and 36 pixels along the beam (14 mm in z). The road size was thin enough to allow a sustainable rate of fakes. We found the bank efficiency shown in figure 8 as a function of the number of stored patterns. With 2 Millions patterns per wedge, we can extrapolate a 90% efficiency bank for a half barrel with roughly 2 prototype AM boards (640 kpatterns/board). We may choose to have a more symmetric configuration both in the transverse plane and along the beam, to measure well the beam spot and the primary vertex positions. In this case we would prefer to use 2 or 3 AM boards to cover 2 or 3 towers separated in the transverse plane, all of them centered around $\eta = 0$.

We are studying a configuration with a tower 45° large in ϕ and covering a reduced η region of $|\eta| < 0.4$. We expect it will require a bank of about 600 kpatterns, that corresponds to a single prototype AM board for an efficiency of 90%. The road size in this case has been reduced in the transverse plane and increased in the z direction to reduce further the fake rate. We used a road size of 20 SCT strips (1.6 mm) and 144 pixels in z (the whole module).

We can do a demonstrator with 2 or 3 AM boards and corresponding EDROs, or even more if the physics case and the length of the run suggest it.

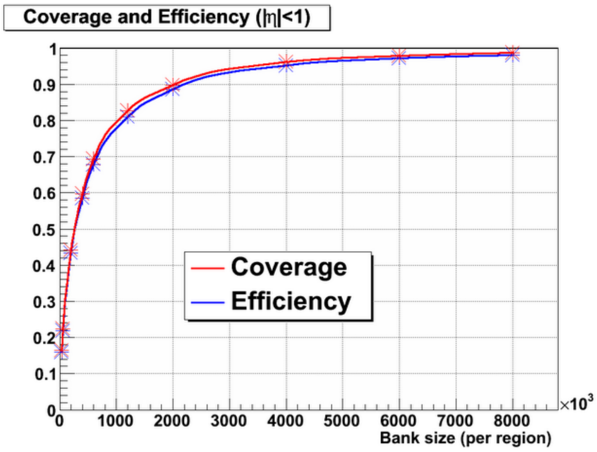


Figure 8: **Vertical Slice bank efficiency** - pattern bank efficiency and coverage as a function of bank size for a slice with $|\eta| < 1$ and 45° wide in ϕ. Coverage and efficiency are computed over a sample of tracks (truth) generated within the acceptance of the pattern bank. The coverage is defined as the percentage of truth tracks with at least one matching pattern. The efficiency is defined as the percentage of truth tracks with at least one matching reconstructed track.

6. Conclusions

The road to build the FTK processor is long and requires many steps. The Vertical Slice is an essential step in this development as it will be the environment in which we develop, test and demonstrate hardware, software and ideas behind FTK.

The “baby FTK” we built, the system made by EDRO + FTK_IM + AM Board, is a first stage that will evolve into the Vertical Slice. We have shown that the hardware for the “baby FTK” is built and stable. We have successfully performed simple yet comprehensive tests for functionality, data integrity and stability at full speed. We have developed all the necessary software tools for development, maintenance and control of our hardware, with enough flexibility to evolve when the Vertical Slice demonstrator will be completed.

We have a plan for early installation inside the ATLAS TDAQ environment, as described in section 1, and plans for future development toward the Vertical Slice demonstrator (section 5). We will start to move our hardware to CERN as early as September 2011. Once the “baby FTK” is installed it will evolve in the more complex Vertical Slice demonstrator and, depending on the LHC schedule, it is possible to contribute also to the physics program before the completion of FTK. Some physics cases for which even a small FTK demonstrator might be interesting (primary vertex detection, slow ionizing particles selection) are under study. We are studying and producing pattern banks suited for both development and physics studies purposes.

[1] A. Andreani, et al., The fasttrack real time processor and its impact on muon isolation, tau and b-jet online selections at atlas, in: Conference Record of Real Time Conference (RT), 2010 17th IEEE-NPSS, 2010, pp. 1 – 8.

[2] S. Bettarini, et al., The SLIM5 low mass silicon tracker demonstrator, Nucl. Instrum. Meth. A623 (2010) 942–953. doi:10.1016/j.nima.2010.08.026.

[3] G. Batignani, et al., The associative memory for the self-triggered SLIM5 silicon telescope, in: Nuclear Science Symposium Conference Record, 2008. NSS ’08. IEEE, 2008, pp. 2765–2769. doi:10.1109/NSSMIC.2008.4774945.

[4] A. Annovi, M. Beretta, A Fast General-Purpose Clustering Algorithm Based on FPGAs for High-Throughput Data Processing, Nucl.Instrum.Meth. A617 (2010) 254–257. arXiv:0910.2572, doi:10.1016/j.nima.2009.10.046.

[5] S. Amerio, et al., GigaFitter: Performance at CDF and perspective for future applications, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment 623 (1) (2010) 540 – 542, 1st International Conference on Technology and Instrumentation in Particle Physics. doi:DOI: 10.1016/j.nima.2010.03.063.

[6] A. Annovi, M. Bagliesi, A. Bardi, R. Carosi, M. Dell’Orso, P. Gannetti, G. Iannaccone, F. Morani, M. Pietri, G. Varano, A pipeline of associative memory boards for track finding, Nuclear Science, IEEE Transactions on 48 (3) (2001) 595 –600. doi:10.1109/23.940125.

[7] A. Annovi, et al., A VLSI processor for fast track finding based on content addressable memories, IEEE Trans. Nucl. Sci. 53 (2006) 2428–2433. doi:10.1109/TNS.2006.876052.

[8] A. Annovi, M. Beretta, E. Bossini, F. Crescioli, M. Dell’Orso, P. Giannetti, M. Piendibene, I. Sacco, L. Sartori, R. Tripiccione, Associative memory design for the FastTrack processor (FTK) at ATLAS, in: Real Time Conference (RT), 2010 17th IEEE-NPSS, 2010, pp. 1 –3. doi:10.1109/RTC.2010.5750451.