2nd International School on Intelligent Signal Processing in Frontier Research & Industry Paris 2014

List of Computing and Practical Labs

Each afternoon from 2 to 4.45pm
will be held 3 parallel sessions: computing Labs,
practical Labs and dedicated master classes.

The computing and practical labs will gather 2
students per session and the training staff that
organizes this session will introduce them to the
proposed Lab or computing exercise(s).

Some of this session may be indeed divided in
two consecutive sessions (part 1 and 2), each
one being normally performed per day.

Some sessions (indeed most of them) include a
tutorial on an experiment or project to which
they are related. This will give the students to
learn about other experiments or projects than
those discussed in the plenary lectures. This is
briefly mentioned in the present summary here
below of these Lab sessions.

The third part of the students will attend the
dedicated master-class sessions that will mainly
be a Q&A sessions with the lecturers of the day
on the lecture of the morning or in anticipation
of the last afternoon session.
This session may also include presentation of
additional material complementary to what is

given in the plenary lectures or to cover some questions from the students.

The first session on July 15 will be devoted to the presentation of all the 11 Computing Labs and 10 Practical Labs by their authors.

The participants will have to choose the "*Menu of Labs*" they want to perform. Students must choose sessions that do not correspond to their expertise and/or current work.

The organizers will prepare accordingly the overall work plan and schedule for the whole duration of the school. The first parallel session will thus start on July 16,

Summary

Computing Classes:

CL1: Introduction to GPU Programming by P. Kestener (CEA)

CL2: Introduction to FPGA Programming with the CASPER Tool Flow by G. Cotter & A. Pollak (Oxford)

CL3: Introduction to analogue integrated circuit design from schematic to layout by: F. Calmon, P. Pittet, L. Quiquerez (CNRS-INL)

CL4: CAD-based test bench for 3D ASIC designs by R. Patti (Tezzaron-USA)

CL5 & CL6: Multi/Many-Core Programming with Intel Xeon Phi Coprocessors; Fundamentals of Parallel Programming using Intel's Many Integrated Core (MIC) Architecture by R. Iope et al (SPRACE, BR)

CL7: RooFit Statistical Analysis Framework of Experimental Data (Application to the Silicon PM Data Analysis) by N. D'Ascenzo & V. Saveliev (NRNU-Ru)

CL8: Applying Sparsity to Astrophysics by J. Bobin, F. Lanusse (Cosmostat-CEA)

CL9: Level-1 Pixel based tracking trigger algorithm for LHC upgrade, by C.S. Moon (CNRS/APC)

CL10: Gravitational waves based exercises by A. Petiteau, H. Inchauspe, E. Plagnol (CNRS/APC)

CL11: Initiation to LABVIEW by M.L. Clark (CNRS/APC)

Practical Labs:

Lab1&2: Hands-on Silicon Photomultipliers by R. Santoro, C, Mattone (Insubria U., Como, It)

Lab3: TARGET ASIC Lab for CTA by G. Cotter, F, de Andrea (Oxford, UK)

Lab4: Timepix; a hybrid pixel detector (Part 1) by A, Pellegrino, E. Schioppa, J. Visser (NIKHEF, NL)

Lab5: Timepix; a hybrid pixel detector (Part 2) by A, Pellegrino, E. Schioppa, J. Visser (NIKHEF, NL)

Lab6&7: Field Programmable gate arrays – Detecting Cosmic Rays (Part 1) by D. Cussans and S. Dasgupta (Bristol U. UK)

Lab8: Introduction to a future space Double Sided Strip Detector (DSSD) based Compton telescope: Principle, Overview of space constraints & Lab tests by P. Laurent, Y. Dolgorouky, M. Khalil (AIM-CEA Saclay & APC/CNRS)

Lab9: Principles of interferometry for gravitational waves detection by H. Halloin and C. Buy (APC-Paris-Diderot University)

Lab10: Silicon Strip telescope for HEP Beam tests by M.T. Jones (Purdue University, USA)

CL1
introduction to GPU programming
Pierre Kestener (CEA/Maison de la Simulation)

This lab will give an introduction to GPU programming and will somehow complete the lecture given during the session on Massive Parallel Computing.

CL2
introduction to FPGA programming with the CASPER Tool Flow
Garret Cotter and Alexander Pollak (Oxford)

The Collaboration for Astronomy Signal Processing and Electronics Research (CASPER) is an international collaboration which aims to streamline and simplify the design flow of radio astronomy instrumentation. To achieve this goal, CASPER promotings design reuse through the development of platform-independent, open-source hardware and software
.
The CASPER group have developed a number of general-purpose, FPGA-based processing platforms, which can be programmed using an intuitive graphical interface, based on Matlab's Simulink environment and Xilinx System Generator. In this practical, participants will familiarize themselves with the CASPER tool flow by constructing a simple program that flashes LEDs, counts and adds numbers on demand (though participants are encouraged to experiment with any of their own ideas!). The firmware created can be run on a "ROACH" board, which is based around a Xilinx Virtex 5 FPGA.

CL3

Introduction to analogue integrated circuit design from schematic to layout

Francis Calmon, Remy Cellier Patrick Pittet, Laurent Quiquerez
CNRS-INL – Lyon Institute of Nanotechnology

This lab introduces the methodology and design flow implementing Cadence CAD-Tools for the design of analogue integrated circuits in a 0.35µm CMOS technology.

The first step is focused at schematic level for the design of a conventional analogue cell. A parametric study of this cell is performed by circuit simulations (e.g. static, transient, small signal and noise analysis) to determine effects of some key parameters (such as transistor sizes) on the cell characteristics. Additionally, the links between the different types of simulation and the extracted electrical performances will be detailed.

The second step deals with the drawing of the cell in the different physical layers of the technology called ''layout''. A special attention is paid on verification issues: design rule checker, layout versus schematic, parasitic extraction in order to simulate the effective physical implementation of the integrated circuit with its extracted parasitics.

CL4

CAD-based test bench for 3D ASIC designs

Robert Patti (rpatti@tezzaron.com)

In this lab students will be exposed to a 3D integrated circuit design flow and its related CAD tools. The students will use
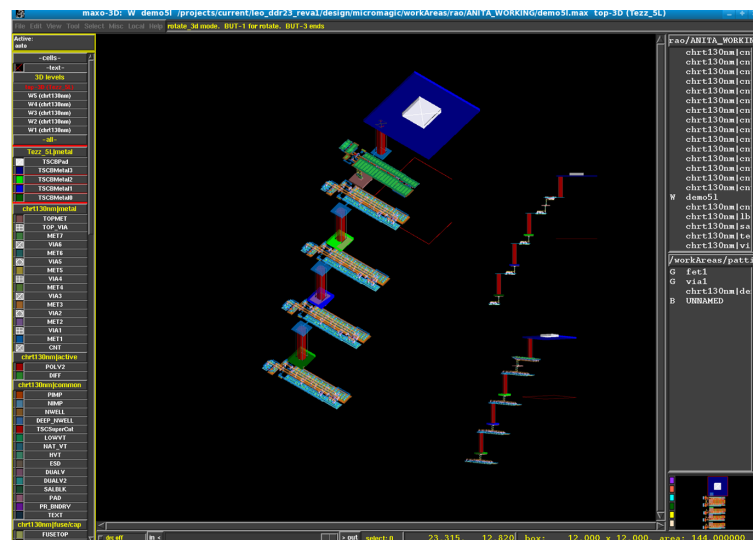
the latest CAD tools to create a simple 3D IC design, implementing it from schematic through design verification. Prior experience with CAD tools or integrated circuit design is not required.

CAD Tools used in course:

MicroMagic MAX, SUE

Mentor ELDO, Calibre

Cadence VDI

CL5 and CL6

Multi/Many-Core Programming with Intel Xeon Phi Coprocessors

Fundamentals of Parallel Programming using Intel's Many Integrated Core (MIC) Architecture

Rogério Iope, Allan Szu, Beraldo Leal, Eduardo Bach, Gabriel Winckler, Marcio Costa

Heterogeneous Computing Working Group - Center for Scientific Computing

São Paulo State University (UNESP) - São Paulo, Brazil

Overview

The Intel Xeon Phi coprocessor, the first product of Intel's Many Integrated Core (MIC) Architecture, is a new accelerator technology developed by Intel to enable performance gains for highly parallel computing workloads that possesses several interesting and appealing features, including the ability to use familiar programming models such as OpenMP and MPI. This hands-on training session is a comprehensive, practical introduction to the Xeon Phi architecture and programming models, aiming to demonstrate the processing power of the Intel Xeon Phi product family.

Participants will have access to a heterogeneous computing system equipped with Intel Xeon processors and Intel Xeon Phi coprocessors, as well as Intel software development tools. The computing system, a state-of-the-art server with two Intel Xeon processors (16 cores, 2 threads/core) and three Intel Xeon Phi coprocessors (171 cores, 4 threads/core), is hosted at the Center for Scientific Computing of the São Paulo State University, in Brazil. It will be fully dedicated for the Massively Parallel Computing training sessions of the INFIERI 2nd International Summer School. The step-by-step hands-on activities have been planned to provide easy to follow instructions in order to allow the participants to have a real - though very introductory - experience on using a powerful 716-thread manycore system.

Learning Goals

Attendants of these hands-on labs will start issuing simple command-line tools to get basic information about the Intel Xeon Phi coprocessors, then learn how to monitor what resources are being used and access their operating systems by establishing ssh sessions with them. Trainees will thus verify that the Intel Xeon Phi coprocessor is an IP-addressable PCIe device - managed by an independent environment provided by

the MIC Platform Software Stack (MPSS) - that runs the Linux operating system.

Following the introductory part, participants will learn how to compile and run simple C/C++ applications directly into the coprocessors, and then compile and run example codes based on shared-memory parallelism with OpenMP and Cilk Plus and distributed-memory parallelism with MPI. They will also work on MPI application examples that should be executed simultaneously on the Xeon processors and the Xeon Phi coprocessors, explore the use of Intel libraries TBB and MKL, and develop insights on tuning parallel applications.

Proposed Agenda

The hands-on activities are split into two 3-hour sessions. Participants will work on a predefined set of exercises that address specific topics that progressively help them getting familiar with the Intel Xeon Phi coprocessor hardware, programming models and development tools. Exercises have been developed in such a way that the learner proceeds from one topic to the next at his/her own speed.

Session 1: Exploring the Xeon Phi coprocessor architecture

The first session will start with an introductory lecture providing a concise overview about the Xeon Phi coprocessor and a live demonstration on how to access the remote system. The practical activities will then allow participants to apply the learned concepts with lab exercises using the Intel Xeon Phi coprocessors installed on the server.

Topics:

- Overview of the Intel Xeon Phi hardware architecture
- Overview of Xeon Phi system software and programming models
- Compiling and running trivially simple applications

- High performance test-drive

Session 2: Exploring the Xeon Phi application development tools

The second session builds on information learned during the first session, providing extra coverage with exercises that highlight task parallelism using OpenMP and Cilk Plus, stress MPI specific key concepts, and demonstrate the use of Intel optimized libraries TBB and MKL. The session concludes with examples and exercises that show how to improve the performance efficiency of applications developed for the Xeon Phi.

Topics:

- Task parallelism using OpenMP and Cilk Plus
- Intel MPI programming models and hybrid OpenMP/MPI
- Using Intel libraries TBB and MKL
- Application performance tuning: optimizing a real-world code example

CL7

RooFit Statistical Analysis Framework of Experimental Data
(Application to the Silicon PM Data Analysis)

Nicola D'Ascenzo (ndasc@mail.desy.de)
Valeri Saveliev (saveliev@mail.desy.de)

The Silicon Photomultiplier (SiPM) is a novel type semiconductor photo-detector developed for the detection of low photon fluxes. It consists of a fine array of micro-sensors (microcells) operated in limited Geiger mode with

the use of a special quenching method.   The extremely high internal amplification gain gives the possibility to detect the space distributed single quantum of light (photons) with high efficiency.

As a result the unique feature of the SiPM is the detection of very low photon flux up to a single photon in the form of single photon spectra (Figure 1). If taken in the proper way, such spectra are carrier of many fundamental features of the photon physics and of the physics of SiPM itself.

The goal of the statistical analysis of the dedicated single photons spectra is the investigation of the low photon flux photon statistics law, the proof of the Poisson nature of the low photon flux (Poisson Distribution) and the analysis of the non ideality of the experimental single photon flux spectra.

The statistical analysis of the experimental spectra of the SiPM gives the possibility to study the statistical nature of the detection of the single photons by fine structures of elementary pixels with single photon detection capability. The statistical fluctuation of the elementary cells signal contributes to the overall resolution of the single photon spectrum.
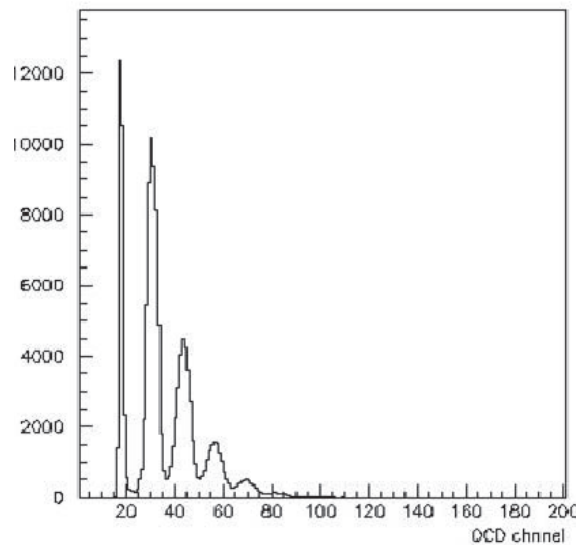

Fig 1: Spectrum of a low Photon flux in SiPM

The statistical analysis allows investigating the main characteristics of the SiPM as the resolution and the specific features of the SiPM as the optic crosstalk and methods of its suppression.

The statistical analysis of the experimental spectra of the SiPM is based on the modern statistical analysis framework of the Particle Physics community: RooFit.

The exercises propose the development of the analysis chain with fitting procedures for a precise description of the experimental spectra of the SiPM and the extraction of the fundamental properties of the low photon flux detection and other features of the SiPMs.

Advanced exercises include the analysis of the SiPM spectra for medical application as Positron Emission Tomography (PET) on the basis of fully digital detection of the signals from positron annihilation by a pair of LySO crystals

CL8
Applying Sparsity to Astrophysics
Jerome Bobin, Francois Lanusse (Cosmostat-CEA)

Exercises based on new statistical Mathematics tools using the sparsity concept as developed by the Cosmostat group at CEA, are applied to astrophysics. They will introduce the students to these new concepts that are as well applicable to several other fields as for instance Medical Imaging or Particle Physics. This Lab will be complementing the lectures by Jeorme Bobin and Julien Girard in the Massive Parallel Session.

CL9

Level-1 Pixel based tracking trigger algorithm For LHC upgrade

*Chang-Seong Moon (APC-Paris-Diderot/CNRS)*

The Pixel Detector is the innermost detector of the tracking system of the Compact Muon Solenoid (CMS) experiment at CERN Large Hadron Collider (LHC). It precisely determined the interaction point (primary vertex) of the events and possible secondary vertices due to heavy flavours (b and c quarks); it is part of the overall tracking system that measures the direction of charged particles; it measures the track part close to the interaction point.

The Level-1 (real-time) Pixel based tracking trigger is a novel trigger system that is currently being studied for the LHC upgrade. A major aspect is to develop real time algorithms able to cope with very high rate and high flux data flux in a very harsh environment, "noisy" so to speak environment due in particular to pile-up of events on the top of the rare physics events the experiment is looking for.

The goal of adding the pixel information already at the real time level of the selection is to help reducing the total level-1 trigger rate while keeping an high trigger efficiency.

This is quite an innovative and challenging objective for the High Luminosity detector upgrade and the special case here addressed is the CMS experiment at LHC.

This computing session will provide as exercises the development of a fast pixel track reconstruction and pixel track matching with other level-1 physics objects using ROOT based simulation framework.
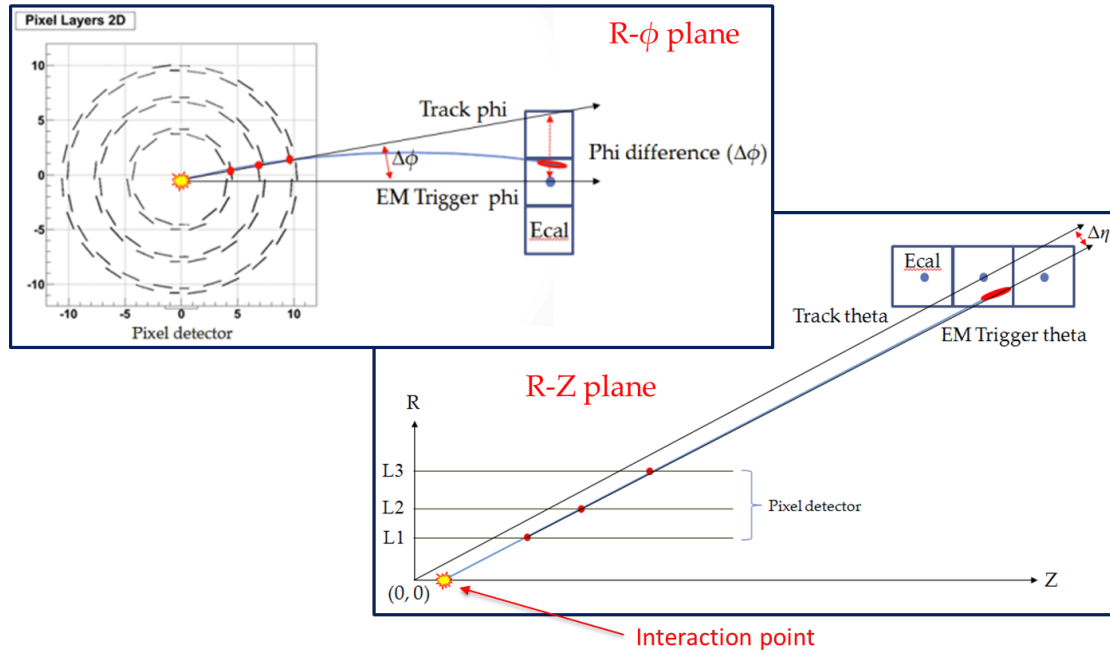
Above schematic shows the regional pixel tracking matching with the level-1 electromagnetic calorimeter object using a signal window range in both the transverse (R-φ) and longitudinal views (R-Z) with respect to the beam axis.

CL10

Gravitational waves based exercises
Antoine Petiteau, Henri Inchauspe, Eric Plagnol
(APC-Paris-Diderot/CNRS)

The exercises will start with a broad presentation of the Gravitational Universe and of the eLISA concept. We will present the Gravitational Waves sources that eLISA will "hear" and the detection techniques eLISA uses. The exercise will then focus on the detection of Galactic Binaries and the parameter estimation process that is presently used.

CL11
Initiation to LABVIEW
Miles Lindsay Clark (APC-Paris-Diderot/CNRS)

QU'EST CE QUE LABVIEW?

LABVIEW est un environnement de développement de programme.

LABVIEW est un langage de programmation graphique qui crée des programmes sous forme de diagramme.

Un programme LABVIEW est appelé « instrument virtuel » (VI)

LABVIEW est cependant accès sur l'acquisition de données, le contrôle d'instruments par port série, parallèle, GPIB, carte E/S…, ainsi que l'analyse, la présentation et le stockage de ces données.

La programmation sous LABVIEW est modulaire, Labview possède des bibliothèques de VI intégrés qui sont directement incorporables dans vos VI développés de manière spécifique.

COMMENT FONCTIONNE LABVIEW?

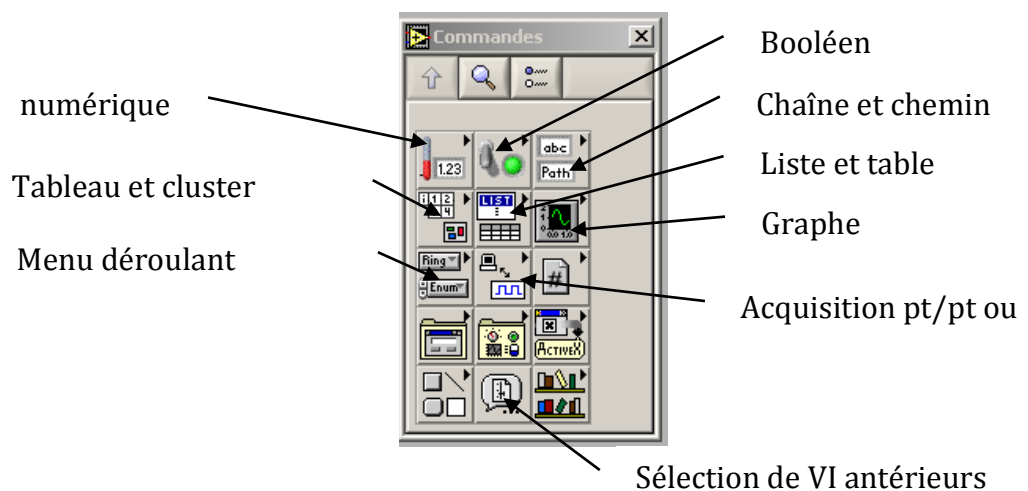Un programme Labview comporte 2 éléments principaux :

1. Une face-avant.
2. Un diagramme.
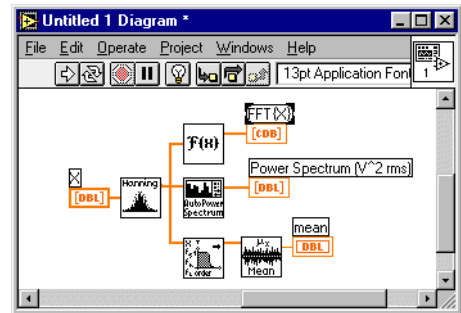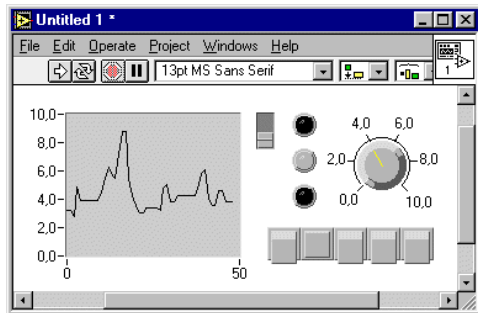
- La face-avant d'un programme est l'interface utilisateur du VI contenants des entrées (les commandes) et de sorties (les indicateurs) du programme.
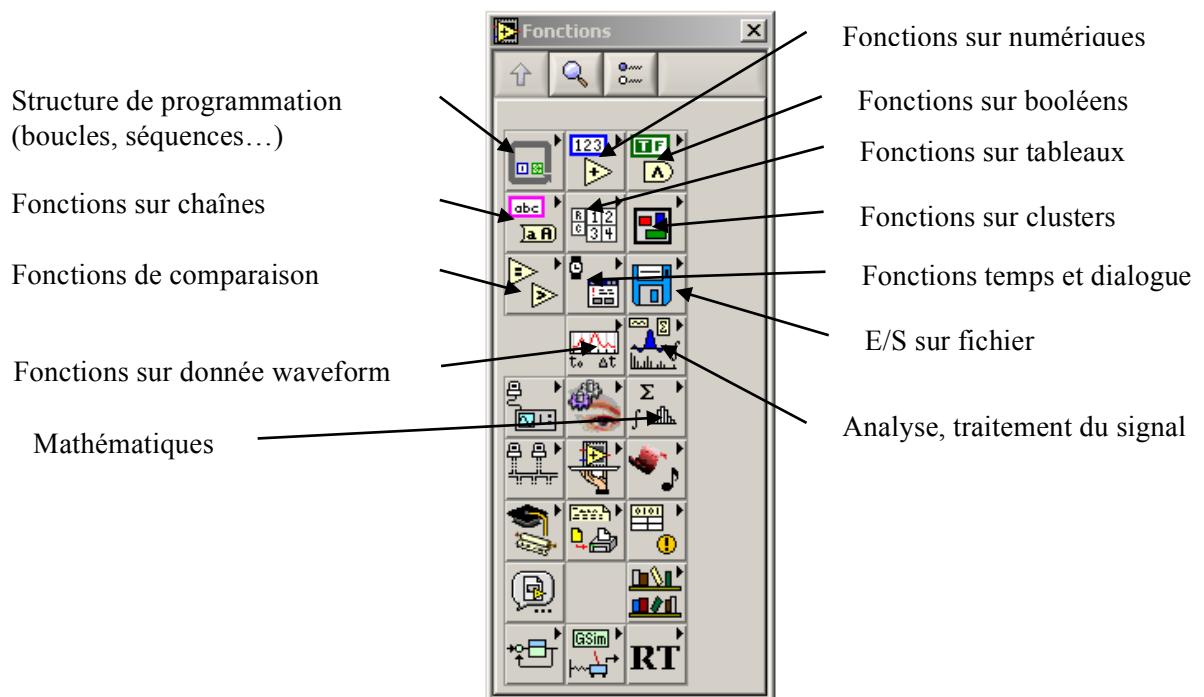
Les commandes et indicateurs peuvent être des afficheurs numériques, des commutateurs booléens, des jauges, des vu-mètres, des boutons poussoirs, des graphes…
Ces commandes et indicateurs sont accessibles par la palette de commandes (clic droit dans une fenêtre de face avant.)

numérique

Tableau et cluster

Menu déroulant

Booléen

Chaîne et chemin

Liste et table

Graphe

Acquisition pt/pt ou
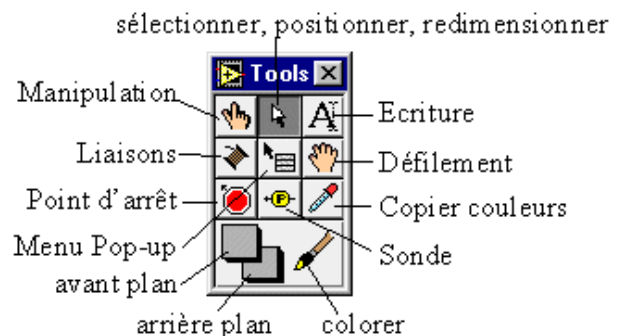
Sélection de VI antérieurs

- Le diagramme contient le code graphique du programme Labview (VI). La programmation est graphique selon une logique de flux de données. Les commandes et indicateurs de la face avant figurent sous forme de terminaux dans la fenêtre diagramme et sont à connecter entre eux selon le traitement désiré.

Ce traitement s'effectue à l'aide de fonctions

Structure de programmation (boucles, séquences…)

Fonctions sur chaînes

Fonctions de comparaison

Fonctions sur donnée waveform

Mathématiques

Fonctions sur numériques

Fonctions sur booléens

Fonctions sur tableaux

Fonctions sur clusters

Fonctions temps et dialogue

E/S sur fichier

Analyse, traitement du signal

accessibles par la palette fonctions (clic droit dans la fenêtre diagramme).

Pour connecter, sélectionner les terminaux, mettre en forme la face avant comme de la fenêtre diagramme, on dispose de la palette d'outils (Menu : Fenêtre>>palette d'outils)

sélectionner, positionner, redimensionner

Manipulation

Liaisons

Point d'arrêt

Menu Pop-up

avant plan

Ecriture

Défilement

Copier couleurs

Sonde

arrière plan   colorer

LA   PROGRAMMATION

GRAPHIQUE.

Bien que ce langage soit totalement graphique, on retrouve sous Labview toutes les structures classiques des langages de programmation textuel (C, Basic....)

- Boucles While, For
- Séquençage (qui permet l'exécution séquentielle de sous-VI)
- Aiguillage IF, CASE...

Les types de données

Labview traite de données structurées classiques :
- numérique
- booléen
- chaîne
- liste
- graphe
- tableau

Un tableau est une donnée qui regroupe des données de même type.
- cluster

Un cluster est une donnée qui regroupe un ensemble de données de types différents.
- waveform.

Une waveform est un type de donnée important sous Labview (essentiel dès que l'on aborde l'acquisition sous Labview).
Une wavefom est un cluster particulier qui contient 3 données :

Un instant inital (to) numérique
un pas (Δt) numérique
un tableau de valeur

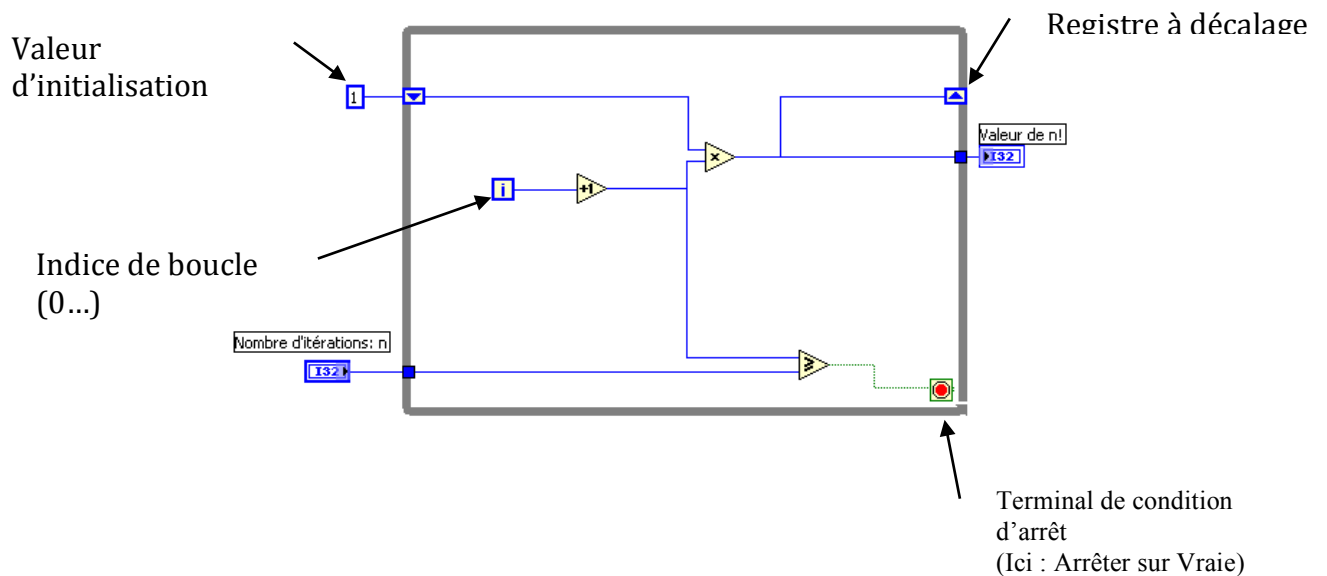Une waveform est le type de donnée utilisé pour étudier les signaux.

Pour chacun de ces types de données, il existe des fonctions dédiées qui s'appliquent sur ces données. Ces fonctions sont accessibles par la palette fonctions de la fenêtre diagramme.

Remarque importante :

Que ce soit en face avant, comme dans la fenêtre diagramme, vous disposez d'une aide contextuelle sur chaque élément (fonctions, indicateurs, commandes..). Pour y accéder, Il faut taper : CTRL+H et placer le curseur de la souris sur l'élément en question.
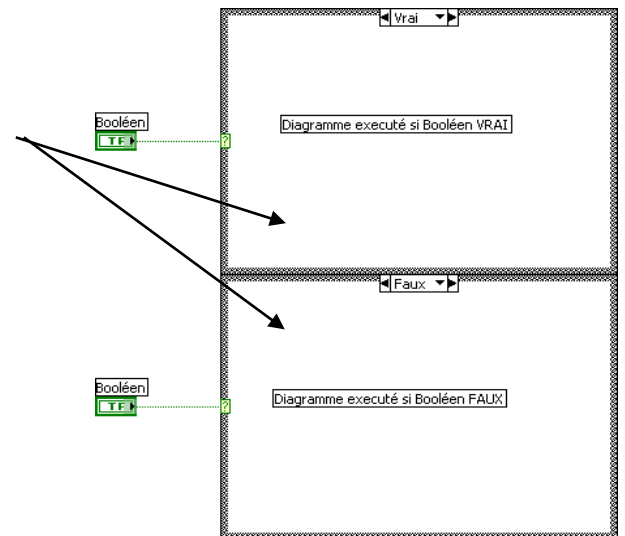

Les boucles while et For

Exemple de boucle WHILE

Valeur d'initialisation

Registre à décalage

Valeur de n!

Indice de boucle (0...)

Nombre d'itérations: n

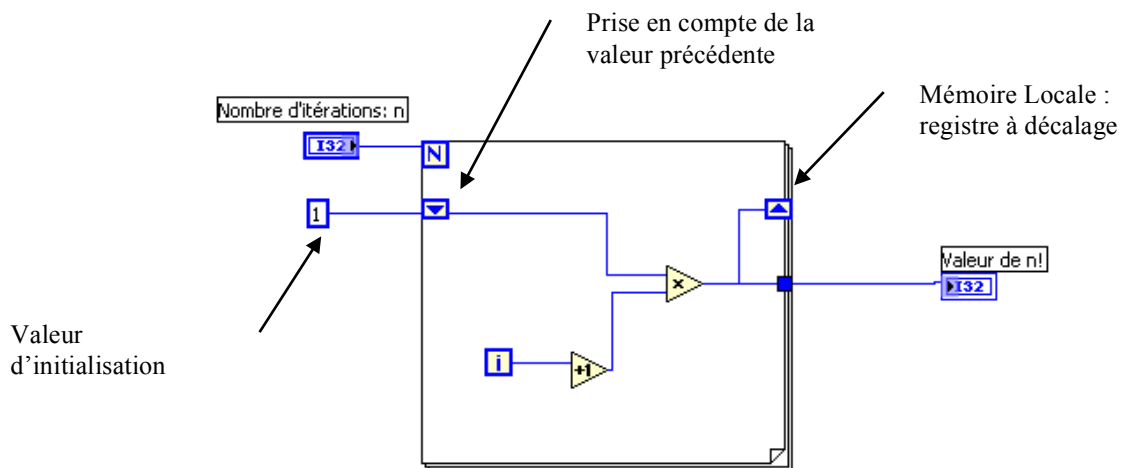Terminal de condition d'arrêt
(Ici : Arrêter sur Vraie)

Exemple de boucle FOR

Exemple de structure conditionnelle IF THEN

Les 2 sous-diagrammes se superposent et sont exécutés de manière exclusive

Booléen
Diagramme executé si Booléen VRAI

Booléen
Diagramme executé si Booléen FAUX

T
r

Prise en compte de la valeur précédente

Mémoire Locale : registre à décalage

Nombre d'itérations: n

Valeur de n!

Valeur d'initialisation

aTravail demandé.

1. Construire un VI qui simule une acquisition d'une température comprise entre 2 valeurs T_{min} et T_{max} saisies par l'utilisateur, par l'intermédiaire d'un glisseur à 2 curseurs. Pour cela, vous utiliserez la fonction numérique *Nombre aléatoire (0-1).*

2. Encapsuler ce VI pour former un sous-VI : « AcqT.vi »

3. Créer un VI utilisant « AcqT.vi », qui réalise l'acquisition de N points (N à saisir par l'utilisateur en face avant) effectuées toutes les 100ms, affiche ces points sur un graphe déroulant, et calcule pour l'ensemble de ces points la valeur minimal, maximale et moyenne. Ce VI doit fonctionner tant que l'utilisateur n'a pas cliquer sur un bouton STOP en face avant.

4. Créer un second graphe déroulant qui affiche la moyenne mobile de ces N points.

5. Enregistrer au fur à et mesure, cette séquence de N points dans un fichier « c:\data.xls ».

LAB 1 & LAB 2

Hands-on Silicon Photomultipliers

Romualdo Santoro, Cristina Mattone (University of Insubria, Como, Italy)

Silicon Photomultipliers (SiPM) are state-of-the-art light detectors, featuring single photon sensitivity and unprecedented photon number resolving capability. SiPM are in rapid evolution, with an extremely high potential in basic Science (High Energy and Nuclear Physics, Biology, Environmental Science), diagnostics in Medicine and industry.

The students will be guided to the discovery of this class of detectors: after an introductory tutorial on the sensor and the system, the major figures of merit will be introduced and basic measurements carried out. As a further step, trainees will have the possibility to perform one of the following experiments:

1. *characterization of the statistics of photons emitted by a LED*
An advanced experiment providing a deep insight into the SiPM world. The multi-photon spectrum will be analyzed to gather information about the statistics of light and the major detector effects;

2. *basic gamma spectrometry with SiPM coupled to scintillators*: measurement of linearity & resolution, background subtraction techniques and characterization of different crystals (LYSO, BGO, CsI);
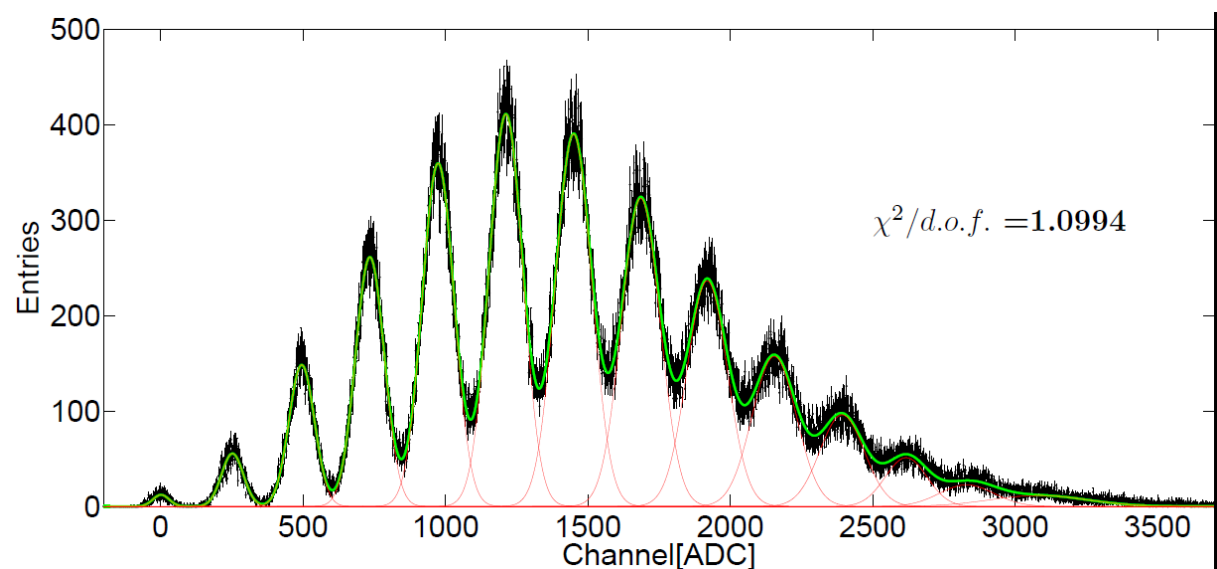
3. measurement of the absorption coefficient of materials for gamma rays;

4. measurement of the counting capabilities of a SiPM based system in response to a Poissonian random trigger.

Two experimental set-ups will be available and can run in parallel. Each experiment will be run over 2 sessions.
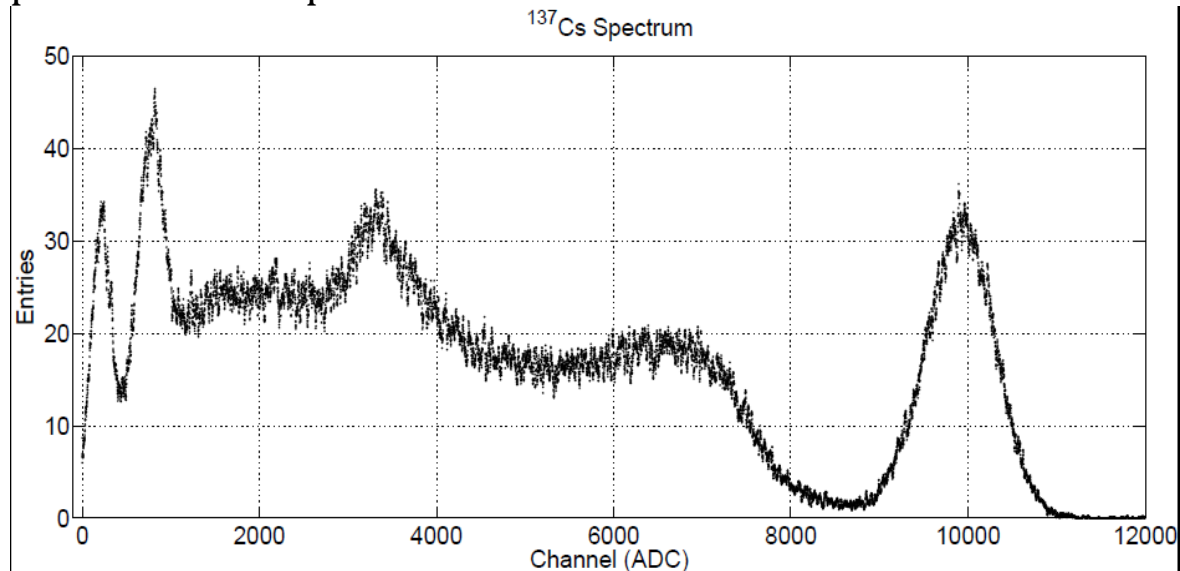
¹³⁷Cs Spectrum recorded with a 6x6 mm² SiPM coupled to a CsI crystal

A typical multi-photon spectrum. Peaks correspond to the number of detected photons. The area underneath every peak measures the relative frequency.

LAB 3

TARGET ASIC Lab for CTA

Garret Cotter, Franco de Andrea (Oxford)

This practical Lab will give an introduction to the signal processing from the photo-sensors that will be used to read out the Compact High Energy Camera (CHEC) in development. They are fed to low-noise shaping pre-amplifiers and thence to 1 GS/s (i.e. 1 ns time-bin) digitising ASICs (called TARGET). In the backplane trigger decision will be made. This is work in progress and among the photosensors: SiPMs are an option.
This Lab will include a tutorial on the CTA project which is one of the major project in the years to come in High Energy Astrophysics.

LAB 4

Timepix; a hybrid pixel detector (Part 1)

Antonio Pellegrino, Enrico Schioppa, Jan Visser (NIKHEF)

Hybrid pixel detectors are used in many fields from high energy physics to industrial applications like X-ray Diffraction (XRD) for crystal structure analysis to determine the material. In this lab course, you will be introduced to the Timepix hybrid pixel detector. It consists of a 300µm thick silicon sensor bump-bonded on a Timepix read-out chip designed by the Medipix collaboration. The Timepix chip is a member of the Medipix photon counting chip family. In addition to its photon counting capabilities, it is also able to provide either Time of Arrival (ToA) information relative to

a trigger or Time over Threshold (ToT) information that provides information about the energy released by the photon or passing particle in the sensor medium.

Using a Timepix hybrid detector, we can extract information about the particle type we are detecting and the intensity variations across the detector by choosing the settings of the detector suitably. The settings of the detector that can be altered to obtain specific information are the following:

- Read-out mode; Counting, Time of Arrival, Time over Threshold
- Exposure time
- Threshold level

In this session, you be made familiar with setting up such a detector system and obtaining images and performing particle identification.

LAB 5

Timepix; a hybrid pixel detector (Part 2)

Antonio Pellegrino, Enrico Schioppa, Jan Visser (NIKHEF)

Hybrid pixel detectors are used in many fields from high energy physics to industrial applications like X-ray Diffraction (XRD) for crystal structure analysis to determine the material. In this lab course, you will be introduced to the Timepix hybrid pixel detector. It consists of a 300μm thick silicon sensor bump-bonded on a Timepix read-out chip designed by the Medipix collaboration. The Timepix chip is a member of the Medipix photon counting chip family. In addition to its photon counting capabilities, it is also able to provide either Time of Arrival (ToA) information relative to a trigger or Time over Threshold (ToT) information that

provides information about the energy released by the photon or passing particle in the sensor medium.

Using a Timepix hybrid detector, we can extract information about the particle type we are detecting and the intensity variations across the detector by choosing the settings of the detector suitably. The settings of the detector that can be altered to obtain specific information are the following:

- Read-out mode; Counting, Time of Arrival, Time over Threshold
- Exposure time
- Threshold level

In this session, you be made familiar with setting up such a detector system and obtaining images and performing particle identification.

LAB 6/7
Field Programmable gate arrays – Detecting Cosmic Rays (Part 1&2 same set-up)
David Cussans and Sudeshna Dasgupta (Bristol)

Abstract

Field-Programmable-Gate-Arrays (FPGAs) play an increasingly important role in instrumentation for particle physics. In this exercise students will develop firmware to read-out a simple scintillation detectors and hence detect cosmic-ray-muons

Description

A Xilinx FPGA evaluation board is coupled to discriminators attached to photo-multiplier-tubes instrumenting plastic scintillator.

Students will engage in one of more of the following activities

1) Write HDL code to fit into an existing framework. This code will do one or more of the following functions: Count pulses, Time-stamp pulses, Detect coincidences. These data will be read-out to a PC over Ethernet using a firmware module supplied.

2) Operate and modify a test-bench to allow simulation and verification of the design.

3) Write or modify a Python script to read and analyse the data.

The Xilinx ISE environment will be used for synthesis and ModelSim for simulation.

LAB 6/7
Field Programmable gate arrays – Detecting Cosmic Rays (Part 2 same set-up than 1)
David Cussans and Sudeshna Dasgupta(Bristol)

Abstract

Field-Programmable-Gate-Arrays (FPGAs) play an increasingly important role in instrumentation for particle physics. In this exercise students will develop firmware to read-out a simple scintillation detectors and hence detect cosmic-ray-muons

Description

A Xilinx FPGA evaluation board is coupled to discriminators attached to photo-multiplier-tubes instrumenting plastic scintillator.

Students will engage in one of more of the following activities

1) Write HDL code to fit into an existing framework. This code will do one or more of the following functions: Count pulses, Time-stamp pulses, Detect coincidences. These data will be read-out to a PC over Ethernet using a firmware module supplied.

2) Operate and modify a test-bench to allow simulation and verification of the design.

3) Write or modify a Python script to read and analyse the data.

The Xilinx ISE environment will be used for synthesis and ModelSim for simulation.

Lab 8
Introduction to a future space Double Sided Strip Detector (DSSD) based Compton telescope:
Principle, Overview of space constraints & Lab tests
Philippe Laurent, Youri Dolgorouky, Mohamad Khalil
(AIM-CEA Saclay & APC-Paris Diderot/CNRS)

This Lab will introduce the students to the specific use of Silicon strip detector for tracking devices in space. It will include a tutorial on the principle of this measurement, the specificity of this DSSD and the constraints imposed by such an experiment in space. The application of this Si tracking technology to space presents several important main differences and challenges w.r.t. the application of Si tracking devices for Particle Physics.

This Lab session will include a tutorial both on these technological specific challenges and on the spatial Compton Telescope experiment it aims to equip.

Lab 9
Principles of interferometry for gravitational waves detection
Hubert Halloin and Christelle Buy (APC-Paris-Diderot University)

Ground-based (VIRGO, LIGO,…) and future space-based (eLISA) interferometers for the detection of gravitational waves make use of Michelson-type interferometry to detect tiny variations of arm length. Due to the extreme precision of these instruments, the setting point has to be precisely controlled through the use of various feedback loops. The purpose of this Lab session is, first, to tune a 'classical' Michelson interferometer and assess its stability around the bright or dark fringe. In a second step, feedback control will be illustrated by setting up active arm-length stabilization, based on phase modulation/demodulation technique around the dark fringe. The effects of the loop parameters (gain, cut-off frequency) on the loop stability and control performance will be studied.

Lab 10
Silicon Strip telescope for HEP Beam tests
Matthew Jones (Purdue University)

This lab intends to provide hands-on experience working with silicon strip detector sensors, readout and data analysis.
Exercises include measuring thresholds and noise on the channels in the system, modelling the response to a radiation source and alignment and analysis of testbeam data previously recorded at Fermilab.