The folder "logiciels_temps_reel_IUT" contains a pack of software used in lab work with students of bachelor’s degree and engineering school in the areas of **signal processing** and **servo control**.

We noted that some students modified the codes of the files “.m”. For this reason the files are provided in pseudo code (files “.p”)

Software works with I/O board, **in real time**. It is therefore not simulation software: it actually realize, for example, correctors in a feedback loop, MATLAB allows easy comparison between the experimental and theoretical results.

These software have been tested with a "National Instruments - PCI 6221-37 pins" board; they use the **RealTime Windows Target** toolbox and **RealTime Workshop** toolbox (Matlab and Simulink releases R2008 to R2010) or **Simulink Coder** (release R2011).

On the face of it, all boards recognized by the RealTime Windows Target toolbox could be used, but we have not verified this claim.

Software can be classified into four categories:

- Plotter
- Feedback control
- Process identification
- Signal processing

Plotter : «Table Traçante»

This software is designed to record responses from slow systems.

You can:

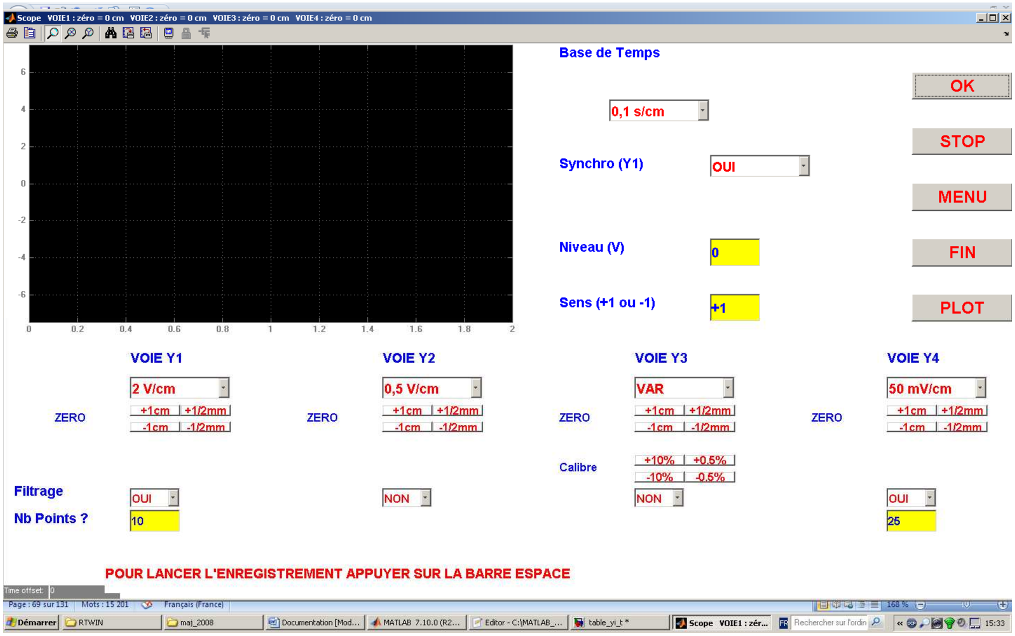
- Plot Y = f (t), 1 to 4 channels simultaneously.
  Tensions viewed can be referenced to ground or be differential.
- Plot Y = f (X).

Tensions can also be referenced to ground or differential.

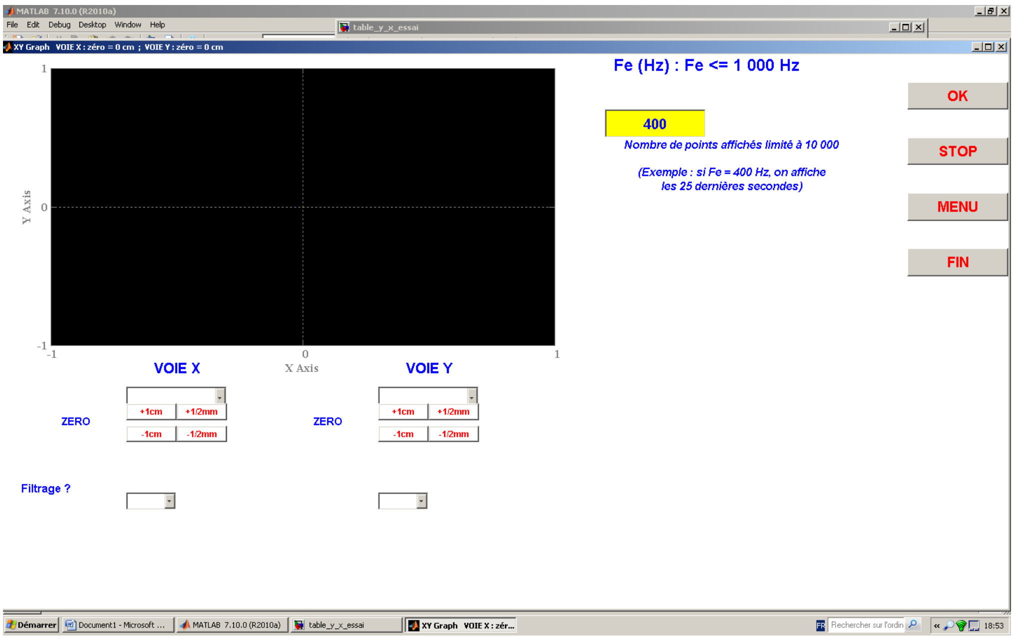
Mode Y = f (t), the time base can vary from 20 ms / cm to 20 s / cm.

The voltage ranges vary from 2V/cm to 20 mV / cm, more than one mode "variable" (uncalibrated).

A figure is obtained comparable to the front face of an oscilloscope:



In XY mode, the sampling frequency, limited to 1000 Hz, is at the discretion of the user; only 10 000 points are visible. We obtain the following figure:



The "PLOT" button which appears at the end of the base sweep time or when you stop recording by XY "STOP" button, allows for curves in a window-type figure, with all the tools related to the figures, particularly data cursor. Until the window is not closed, the curves are added.

You can save a curve in a ". mat" file, so that you can recall it later and display it, possibly with a delay.

The following instructions, which can be placed in the file "startup.m", allow to obtain a printing as 1 screen graduation = 1 cm print :

```
set (0, 'DefaultFigurePaperOrientation', 'landscape')
set (0, 'DefaultFigurePaperType', 'a4letter')
```

set (0, 'DefaultFigurePaperUnits', 'centimeters')
set (0, 'DefaultFigurePaperPosition', [0.5 0.5 25.75 18.32])

Feedback control

This category includes 3 software

- Programmed analog controller
- Digital controller
- Reconstruction of the flux of an asynchronous machine

It is possible to record the response of the closed loop system and print the response, **without interrupting the process control.**

-1 - Programmed analog controller

Terms of use

The sampling period T_E must be small compared to all the time constants of all process + controller.

Feasible controller

We can achieve the following regulators:

- **PID** - parallel structure
 (**P** if Ti = inf and Td = 0, **PD** if Ti = inf, **PI** if Td = 0)
- **PIR :** PI and delay
- **Smith Predictor**
- **R (s)** any
- **Tachometer feedback** (secondary feedback loop)

Only the scheme of controller changes..

• PID

The PID has an additional option: we can vary each parameter by thrust ("+" or "-" square). This allows, for example, find the critical regime.

• PIR

This correction is used in the case where the process is a delayed first order system : $G(s) = \frac{A}{1+\tau s}.e^{-Ts}$ and provides, in a closed loop, a transfer function of the form:

$$W(s) = \frac{1}{1+\frac{\tau}{\alpha}s}.e^{-Ts}$$

• Smith Predictor

This correction is used if the process is delayed:

$$G(s) = G_0(s).e^{-T.s}$$

Let $R_0(s)$ the controller obtained in the case where the process would not introduce delay. (For example, a PID controller)

$R_0(s)$ and $G_0(s)$ are defined by their numerator and denominator.

The transfer function of the controller is:

$$\mathbf{R(s) = \frac{R_0(s)}{1 + R_0(s).G_0(s).(1 - e^{-T.s})}}$$

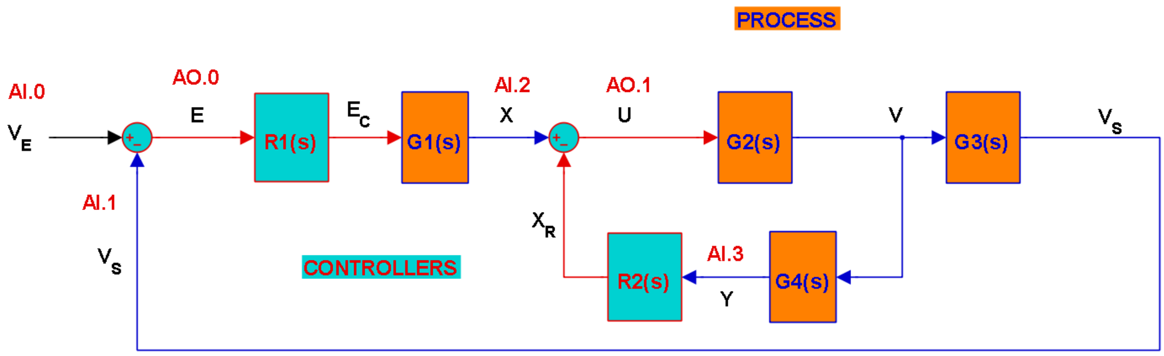
• <u>R(s) any</u>

The user programs the transfer function R(s) of the controller.

• <u>Tachometer feedback</u>

The transfer function G(s) process can be decomposed into three blocks:

$$G(s) = G_1(s).G_2(s).G_3(s)$$



The user programs the transfer functions $R_1(s)$ and $R_2(s)$.

-2 – Digital controller

This software allows you to realize digital controllers in a sampled loop : the sampling period T_E may be the same order of magnitude as the time constants of the process.

The controller is programmed directly in Z:

$$C(z) = \frac{NUM(z)}{DEN(z)}$$

Two "Scope" are used: the first to view the analog (reference and process output), the second to see the sampled signals.

-3 - Reconstruction of the flux of an asynchronous machine

This software is especially dedicated to students in their final year of engineering training.

This is based on measurements of voltages and currents of each phase to rebuild flux, in real time, in a squirrel-cage asynchronous machine. To find out if the reconstruction is correct, one can determine from the flux speed of the machine, and compare the result with the actual measurement of the speed obtained with a tachogenerator.

We can then consider feedback control called "sensorless".

Process identification

In order to choose the type of correction to be used in a feedback loop, the first step is to identify the process.

This category includes 3 software

- Identification
- Impulse response
- Model

-1 - Identification

We can :

- Determine the transfer function of a process without delay:

$$G(s) = \frac{a_0 + a_1.s + ... + a_m.s^m}{s^k.\left(1 + b_1.s + ... + b_n.s^n\right)} \qquad (m \le n + k)$$

- Determine the transfer function of a process with a pure delay:

$$G(s) = \frac{a_0 + a_1.s + ... + a_m.s^m}{s^k.\left(1 + b_1.s + ... + b_n.s^n\right)}.e^{-T.s} \qquad (m \le n + k)$$

- Get a Strejc model : we seek here a model of the form:

$$G(s) = \frac{A}{(1 + \tau.s)^n}.e^{-T_R.s}$$

-2 - Impulse response

This software can obtain experimentally the impulse response of a process.

The input signal, provided by the software (AO.0 output) is either white noise or a pseudo-random binary sequence (PRBS). The correlation between the response of the process and the input signal provides the impulse response and thus to trace back the transfer function of the process.

-3 - Model

This is to compare the responses of the process and model.

The model can be realized either with analog simulator transfer functions (with a simulated delay), simulated using SIMULINK, or twice.

Signal processing

This category includes 5 software

- Digital filters
- FFT
- reverse FFT

• Harmonics
• Synthesis of active filters (does not work in real time)

-1 - Digital filters

MATLAB + SIMULINK are used to calculate the samples of the output signal:

$$\mathbf{y_n = \sum_{k=0}^{P} a_k . x_{n-k}} \quad \textbf{ou} \quad \mathbf{y_n = \sum_{k=0}^{P} a_k . x_{n-k} - \sum_{r=1}^{Q} b_r . y_{n-r}}$$

As for the study of any filter, the digital filter must be introduced from a signal generator for example, and display the response of the filter on an oscilloscope.

We can make FIR or IIR filters.

With the PCI-6221 board, can be achieved for example a 50-th order FIR filter operating at 50 kHz.

It is also possible to observe the impulse response of these filters, the pulses being delivered by the program and can be displayed on an oscilloscope using a second output.

-2 - FFT

This software allows you to realize :
• Simulations with programmed signals
• Spectral analysis of a real signal (acquisition)
• Spectral analysis of two real signals: typically with and without anti aliasing filter

Samples can be weighted by different windows:
• Rectangular
• Bartlett
• Hann
• Hamming
• Blackman
• Flattop

After the acquisition of $N = 2^m$ points, we start the calculation of the FFT of the (or more) signal (weighted or unweighted)

It can display monolateral or bilateral spectrum, in volts or dB.

We can finally obtain, in a text file, the amplitude (complex) lines.

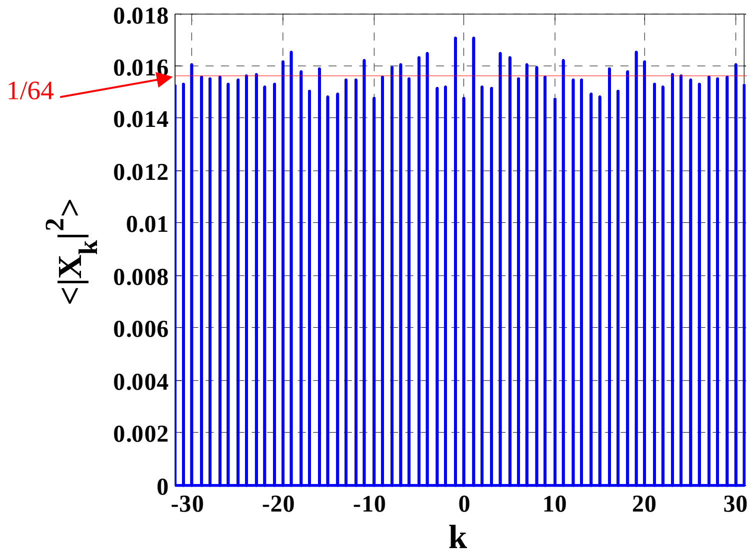
A small additional tool was developed: estimation_DSP.

This tool can be used to estimate the power spectral density of a random signal, white noise for example ("periodogram").

We record, using the software "FFT", the signal x with a great number of points (2^{16} points or 2^{15}); x is then cut into slices of 2^6 points, for example, and we calculate the FFT of each slice and then average $\left|X_k\right|^2$:

[S, K] = estimation_DSP (x, 64);

Here are the result obtained for white noise, power 1 V^2, sampled at 64 Hz:



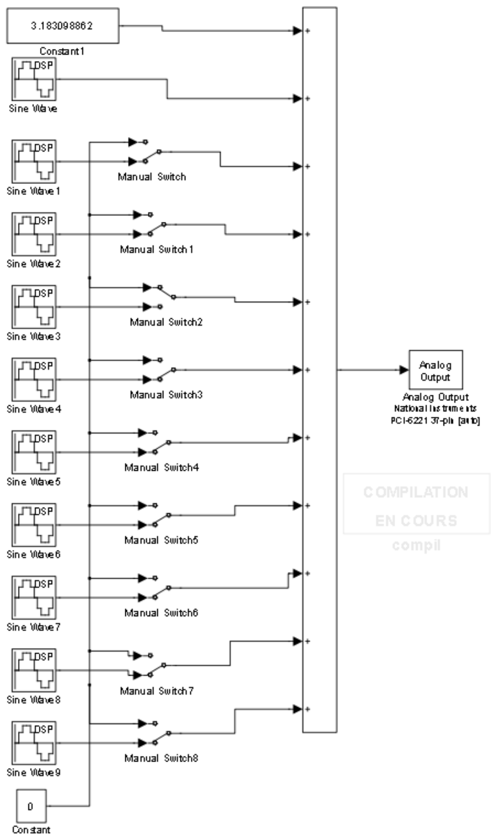
-3- Reverse FFT

This is to reconstruct a signal, visible on an oscilloscope as it is available on an output of the acquisition board, from its bilateral spectrum.

This spectrum is programmed: the F_E sampling frequency is 5 kHz. The total number of samples of the signal, or the total number of spectral lines is N = 5000.

-4 - Harmonics

This is to reconstruct a signal, visible on an oscilloscope, gradually adding the harmonics:



-5 - Synthesis of active filters

This software uses MATLAB + SIMULINK and SIMPOWERSYSTEM toolbox
The core consists of a library comprising 5 basic filters:

- Low-pass and high pass first order
- Low-pass and high pass second order (Sallen - Key structure)
- Bandpass second order (Rauch structure)

All filters are such that the amplification bandwidth is + 1 or - 1 (gain bandwidth of 0 dB).

The software allows for the synthesis of filters:

- Lowpass
- High pass
- Band-pass

Using the approximation functions of:

- Butterworth
- Chebychev

Can be obtained, or do the following studies:

1. Transfer function of the filter
2. Bode plots (gain and argument) theoretical, and comparison with the template
3. Theoretical step response
4. Theoretical impulse response
5. Poles and zeros positions
6. Simulation and practical realization using operational amplifiers (TL 084 by default)
7. Same study that -6 - but using the normalized values rounded to the nearest theoretical values (choice in E6, E12, E24, E48 or E96 series)
8. Monte Carlo analysis: random distribution of the value of components (for a given series), allowing to have an idea of the dispersion of results if a large production is achieved.
9. Sensitivity: the value of a component can be varied from a minimum value to a maximum value with chosen step.

It is possible to change the operational amplifier, by defining the following parameters:

- Differential continuous amplification
- cutoff frequency of the follower assembly
- Slew-rate
- Supply voltages
- Differential Input Resistance
- Output Resistance

Installation

The folder « logiciels_temps_reel_IUT » contains a file **« installation.m » :** run this file.

All the necessary files are copied in a user-defined folder.

Then run the file **« liste_logiciels »** to reach by a simple clic all the files describe previously.