POLITECNICO DI MILANO

mecc





Introduction to PoliArd

Damiano Milani

Politecnico di Milano
Department of Mechanical Engineering

PoliArd Project

PoliArd is a complete environment for implementing control logics on real-time hardware.

It is based on **Simulink software** and **Arduino hardware**.

It is developed and maintained by the *Department of Mechanical Engineering* and it is free to use for students.

It includes:

- **PoliArd Board**: a real-time controller board
- **PoliArd Library**: a Simulink blockset
- **PoliScope**: an acquisition application

Official website: http://mecsys.mecc.polimi.it/mechlab

Outline

The PoliArd Project

- PoliArd Board

- PoliArd Simulink Library

- PoliScope

- Examples

Outline

The PoliArd Project

- **PoliArd Board**

- PoliArd Simulink Library
- PoliScope
- Examples

The Board

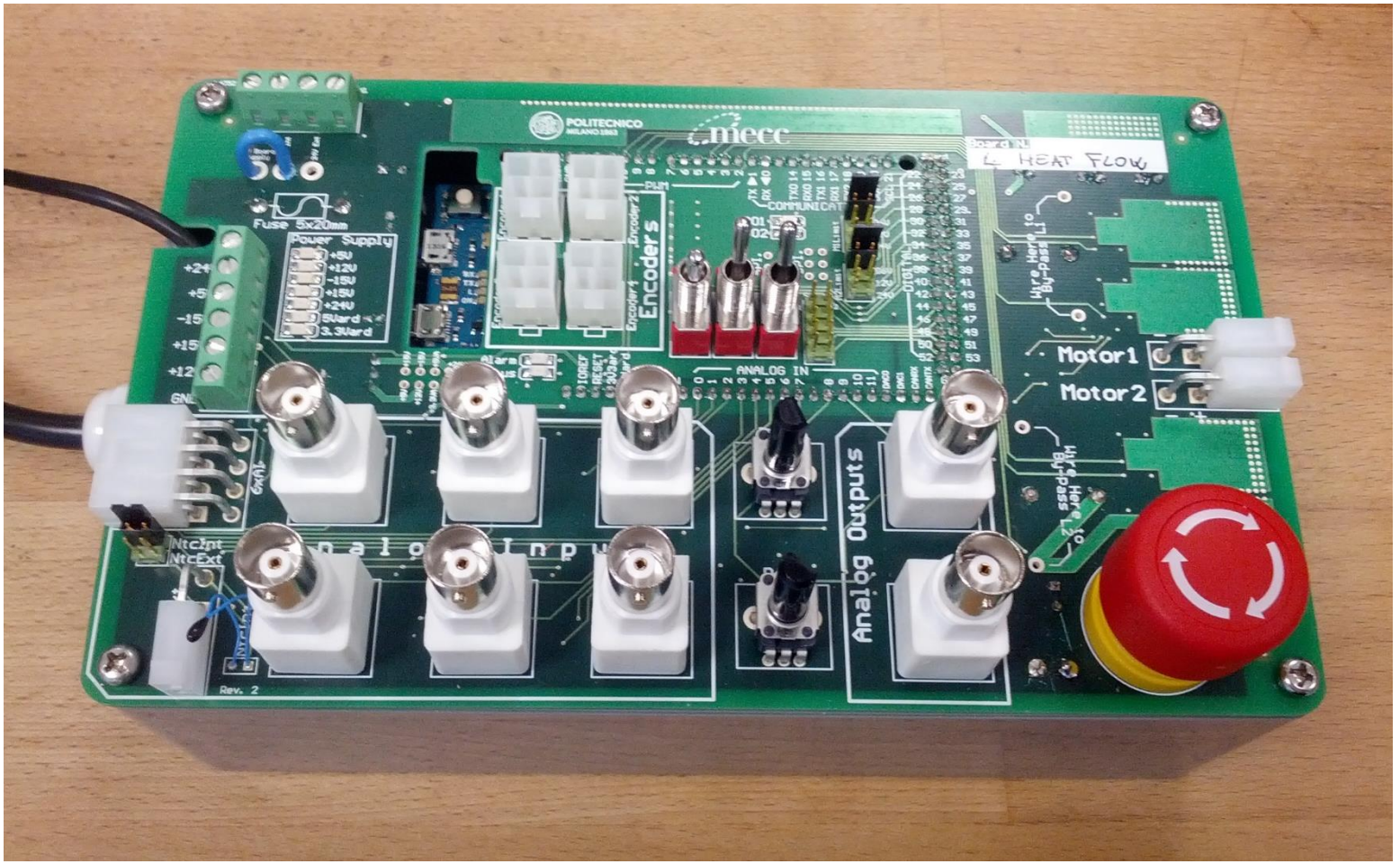
PoliArd Board allows to perform real-time control logics thanks to an **Arduino Due** board (32-bit ARM Cortex-M3).

PoliArd Board presents several interfaces for generic input/output, motors, encoders…

You can find the board schematics at
http://mecsys.mecc.polimi.it/mechlab/files/poliard/PABoard.zip

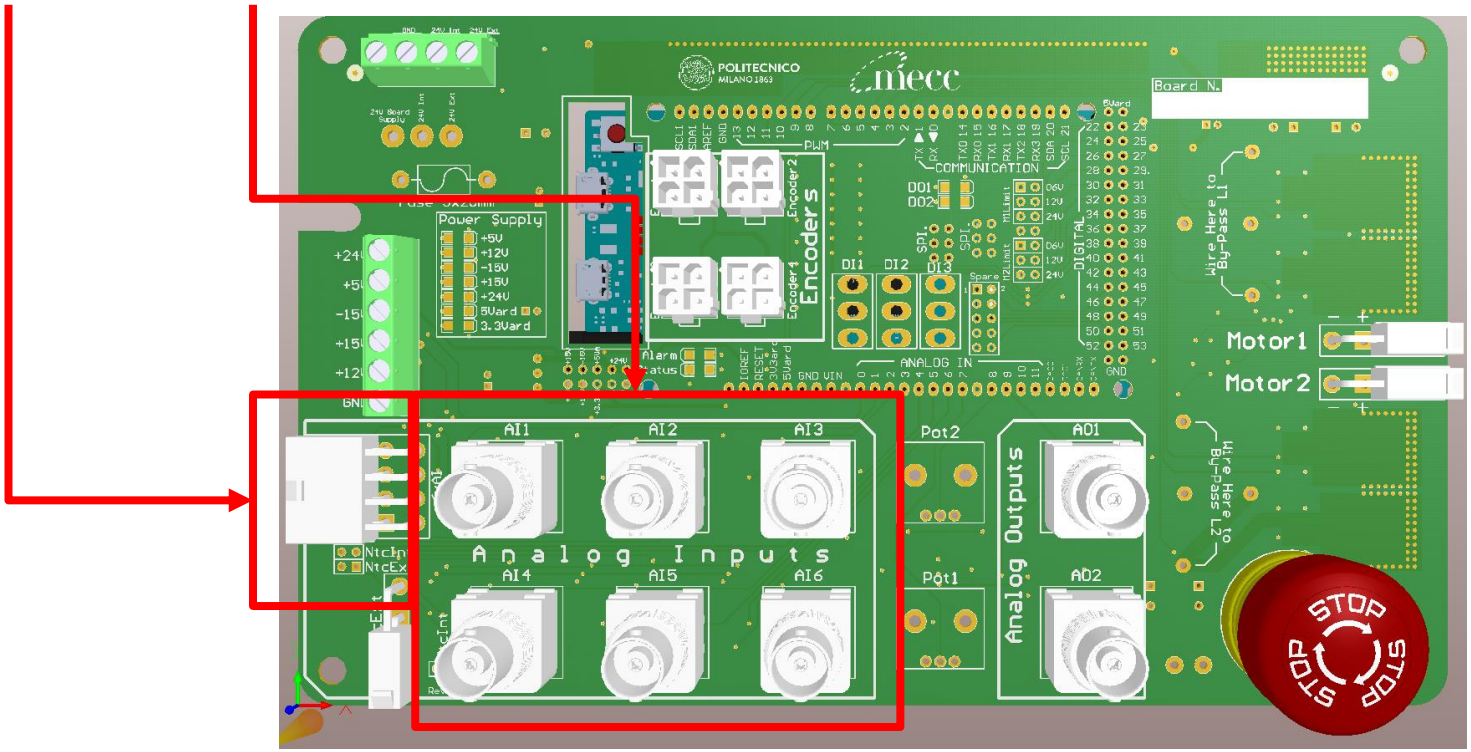
PoliArd Board

The Board



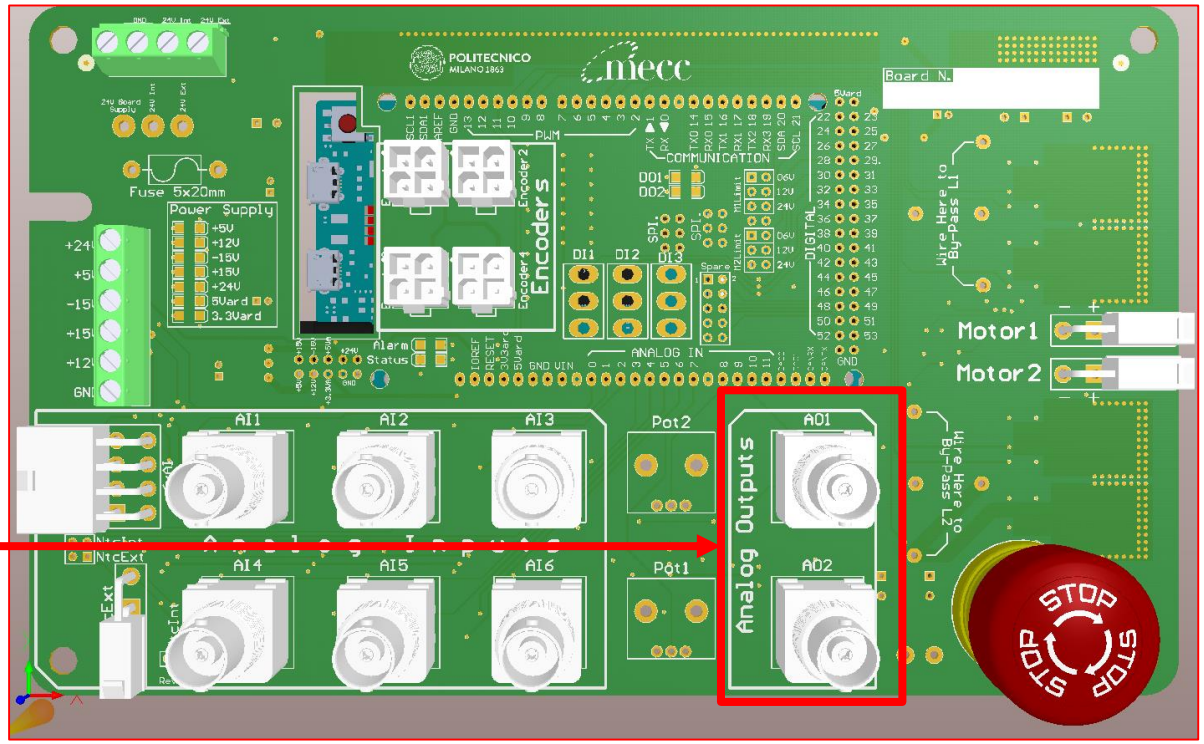
Analog Inputs

- 6 Analog Inputs (12-bit ADC)
- ±10 V
- 2x4 Mini-Fit or BNC



Analog Outputs

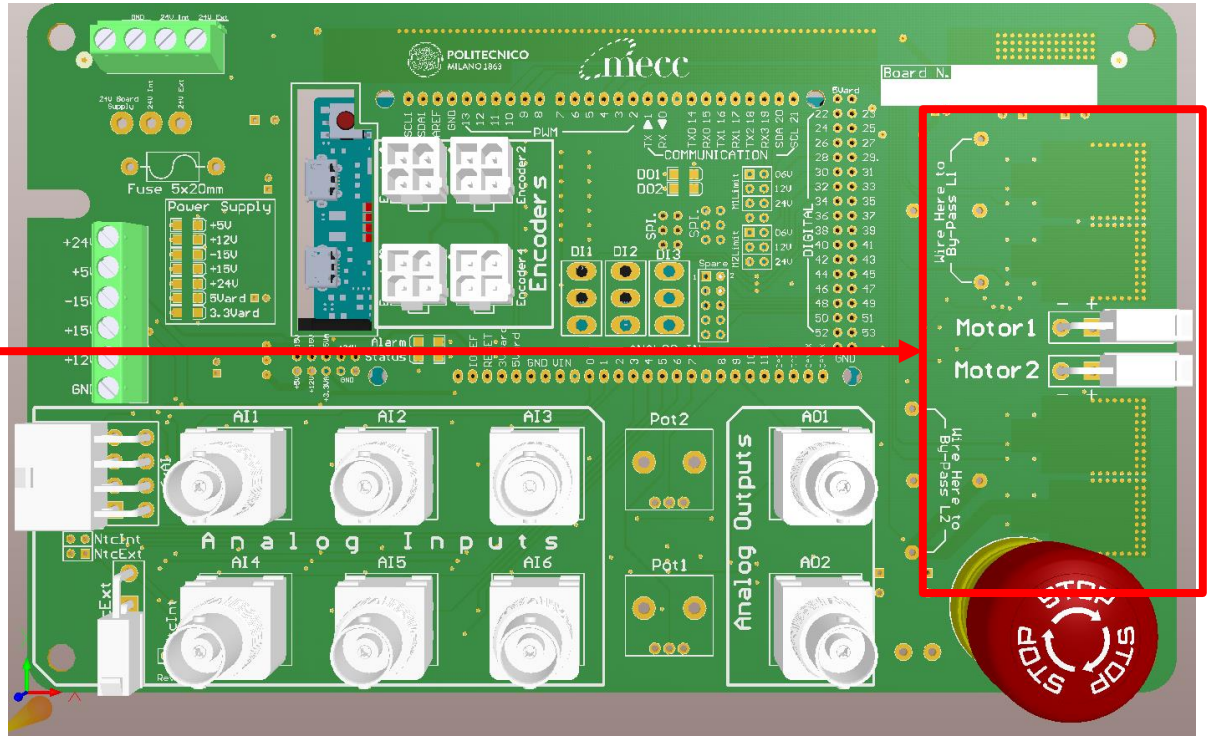
- 2 Analog Outputs (12-bit DAC)
- ±10 V
- BNC



PoliArd Board

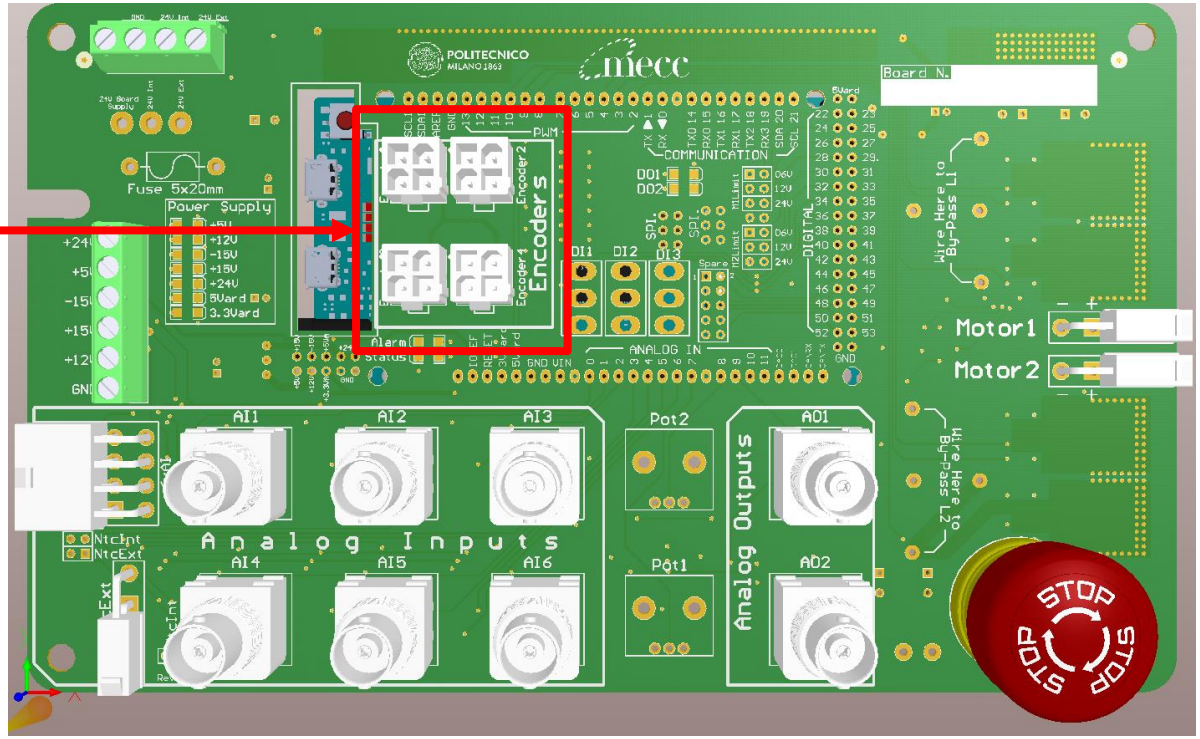
Motor driver

- 2 DC motor drivers
- 6, 12 or 24 Vdc
- PWM controlled
- Current sensor
- 2x1 Mini-Fit



Encoder interface

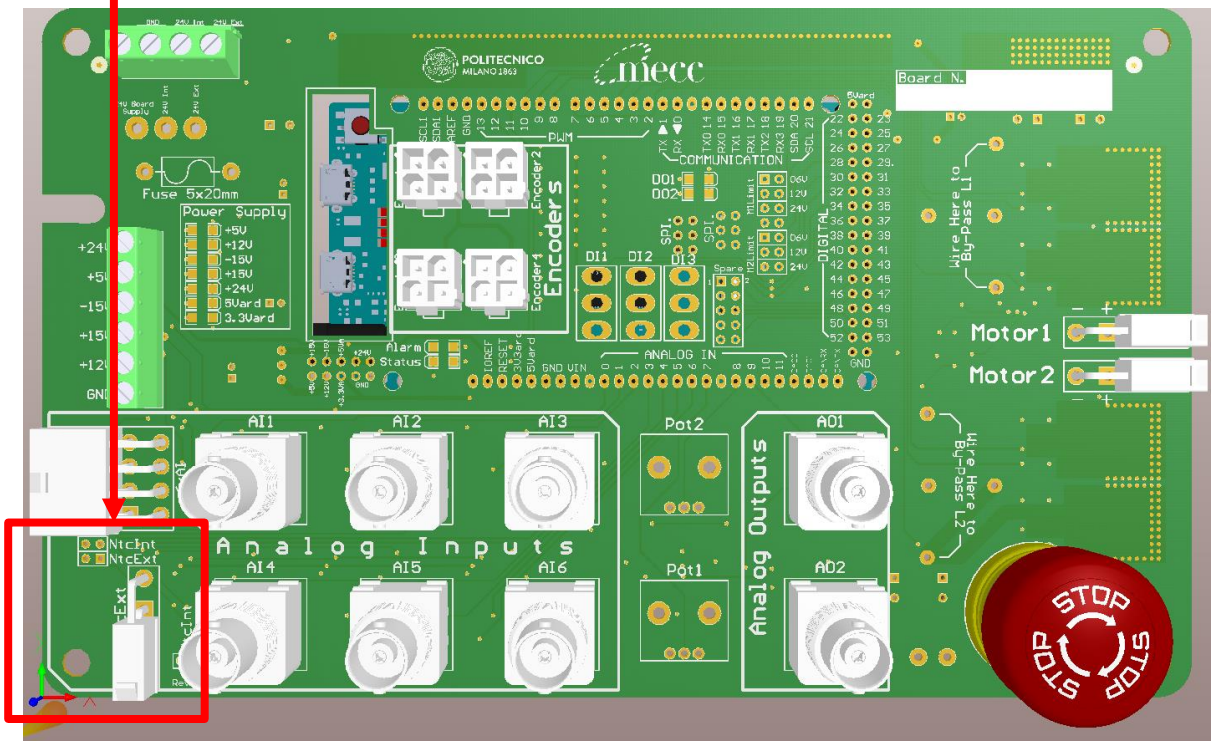
- 4 encoder interfaces
- 2x2 Mini-Fit



PoliArd Board

NTC

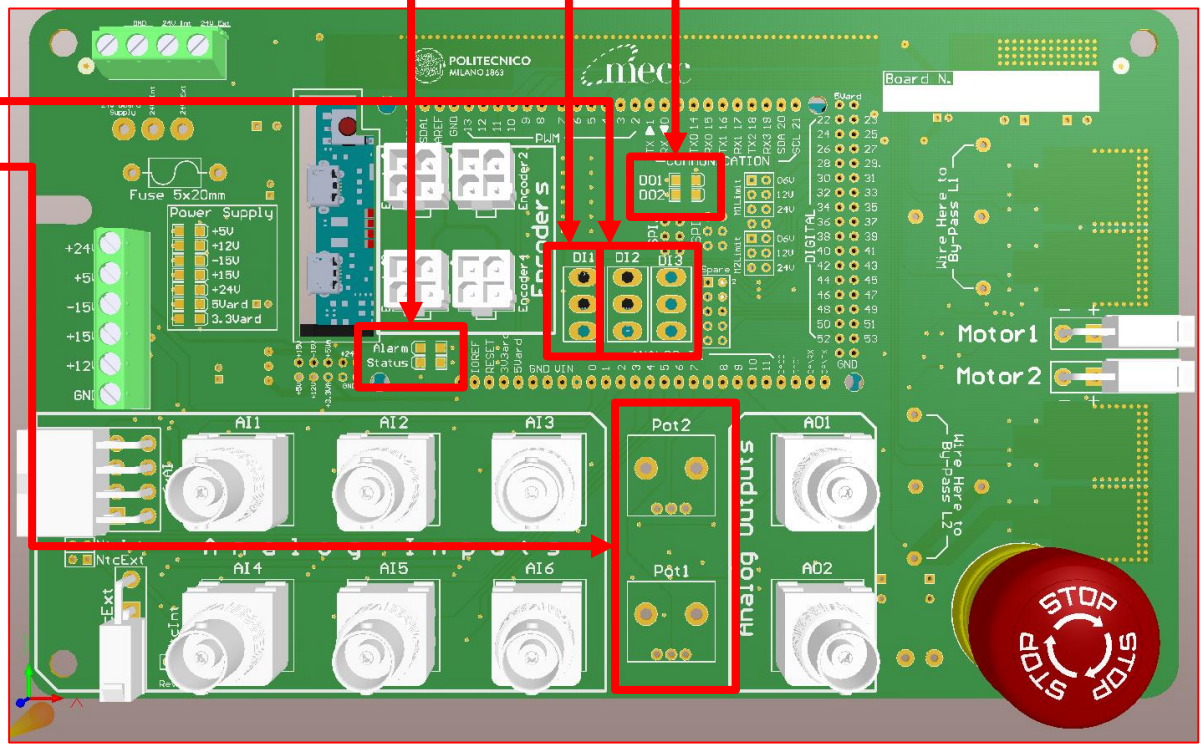
- Internal
- External (2x1 Mini-Fit)



PoliArd Board

User I/O

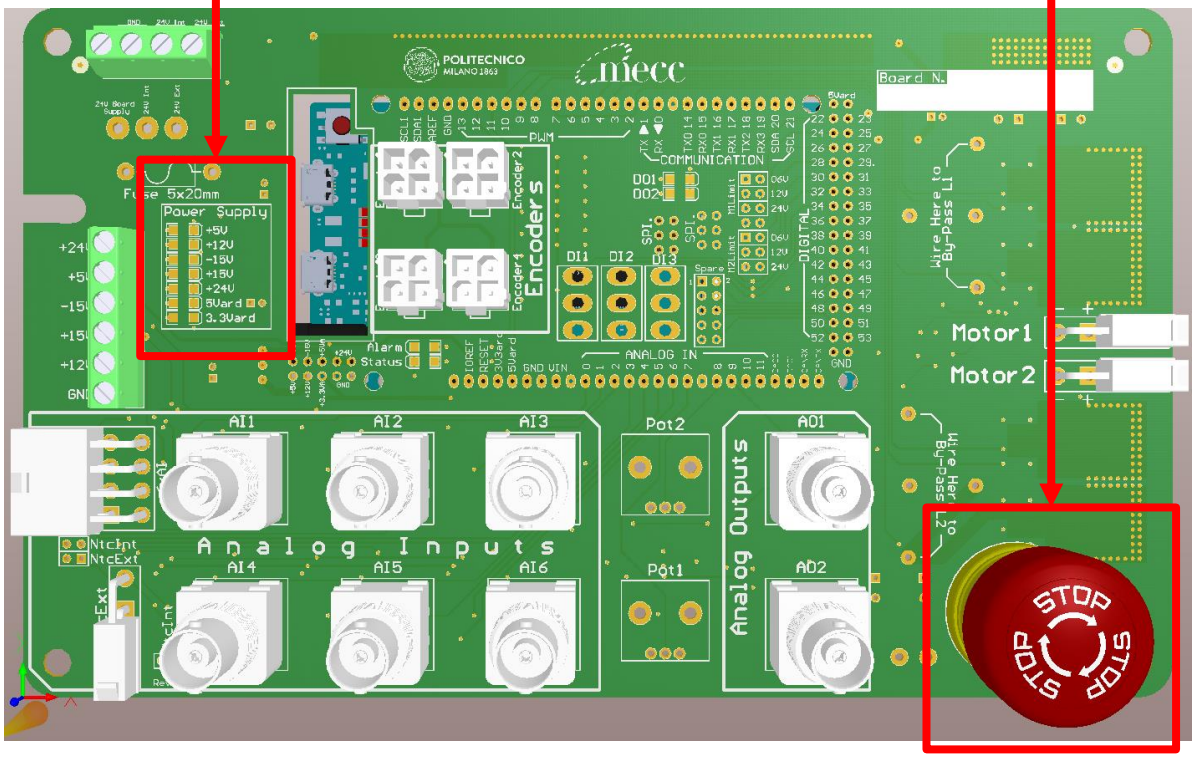
- System Status/Alarm LEDs
- 2 configurable LEDs
- Enable switch
- 2 configurable switches
- 2 potentiometers



PoliArd Board

Power

- Emergency button
- Power Status LEDs



Outline

The PoliArd Project

- PoliArd Board
- **PoliArd Simulink Library** 
- PoliScope
- Examples

PoliArd Simulink Library

Installing the PoliArd Library you can easily implement control logics on the PoliArd Board using a custom Simulink blockset.
The supported version is ***MATLAB 2015a*** 64bit.

First you need to download the Mathworks ***support for Arduino hardware***, following these instructions:
http://mathworks.com/help/supportpkg/arduino/ug/install-support-for-arduino-hardware.html
and download the **Simulink** package for **Arduino Due.**

* Note that during this step and in the next one, some Antivirus software may freeze the installation processes.

Library Installation

You can download the required files from
http://mecsys.mecc.polimi.it/mechlab/files/poliard/PALibrary.zip
and run the MATLAB file

`install_PALibrary.m`

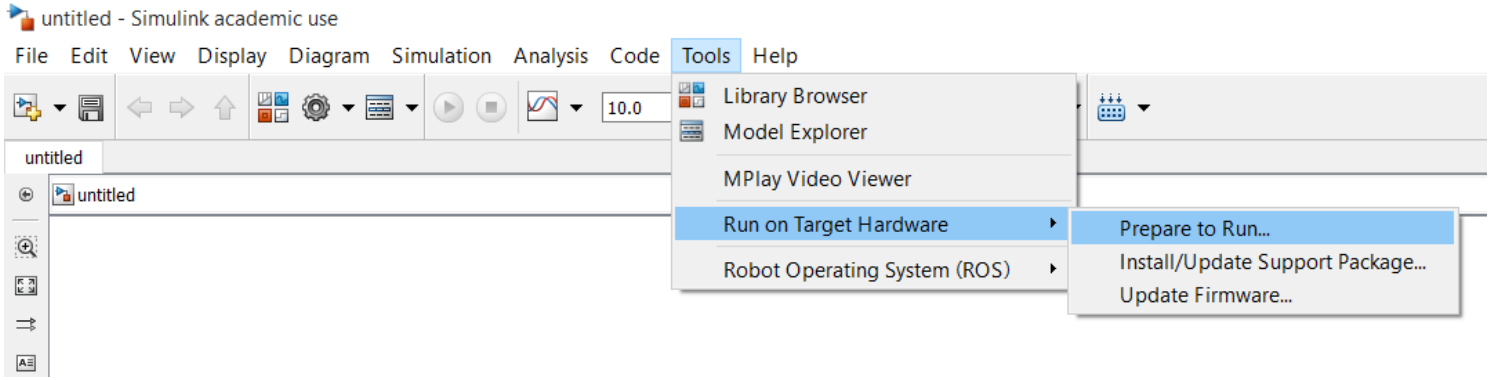
following the instructions.
With the same file you can **remove** or **update** from the official repository the library files.

Requirements (supported and tested):

- Microsoft Windows OS 64bit
- MATLAB 2015a 64bit (available with the campus license)
- Arduino Due support package for Target Hardware

Simulink Target Hardware

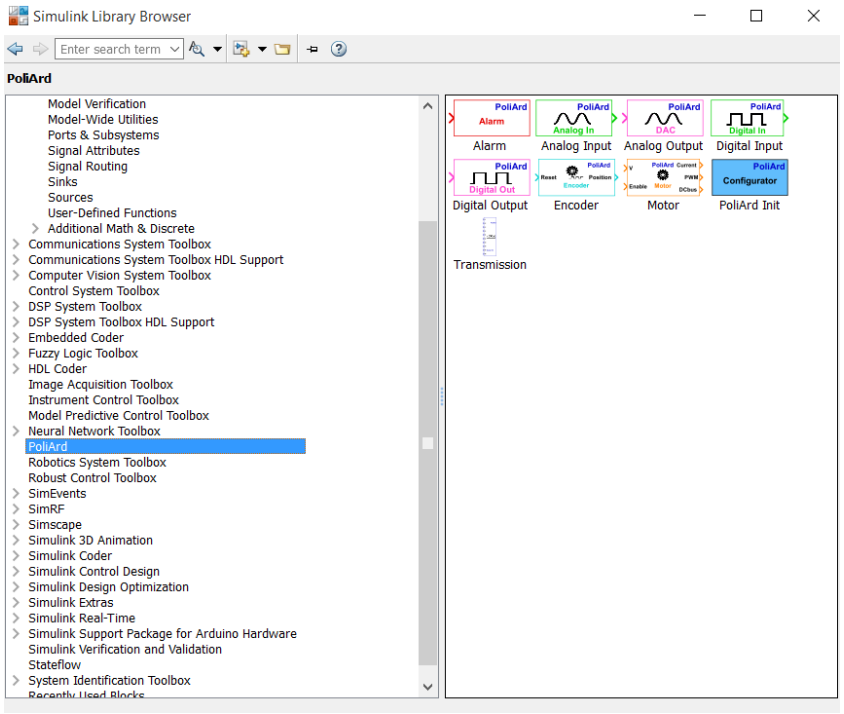
To implement the control logics on the PoliArd Board, you need to **setup any Simulink model:** go to *Tools* menu, *Run on Target Hardware* and *Prepare to Run…* From the new window select *Arduino Due* and check ***Enable overrun detection***.

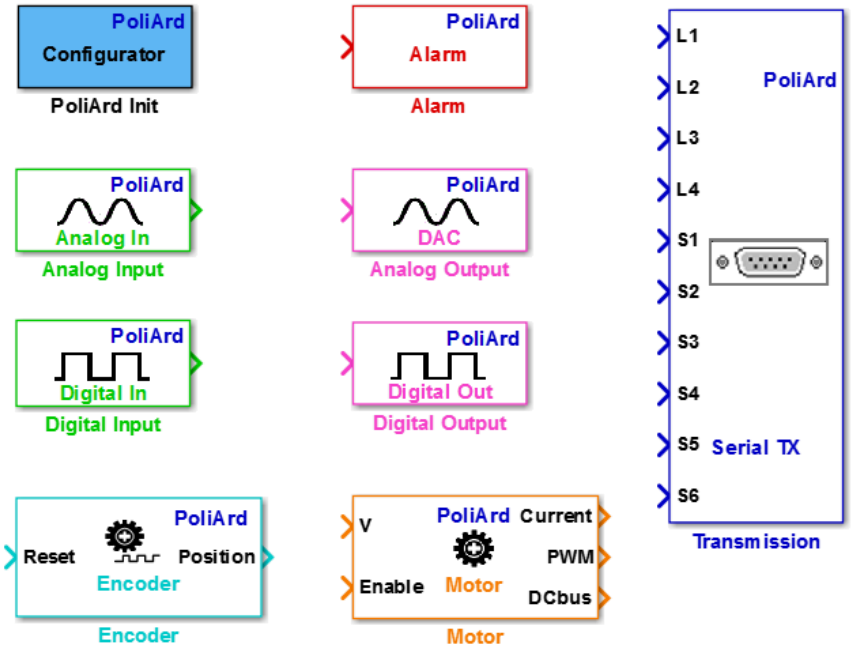


In the **Configuration Parameters** window you can set the fixed-step size (minimum value suggested 0.001) and the stop time (could be *inf*).

The PoliArd Library

After installing the PoliArd Library, a new following blockset will appear in the Simulink Library Browser:

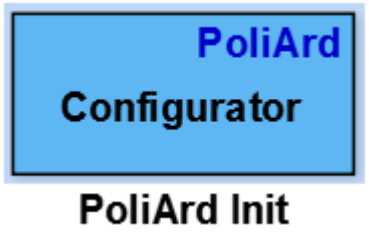




PoliArd Library

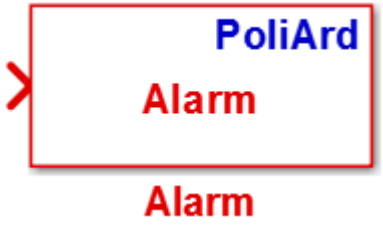
PoliArd init

This block ***must*** be present in any PoliArd model. It contains the initialization routines needed by the PoliArd Board.



Alarm

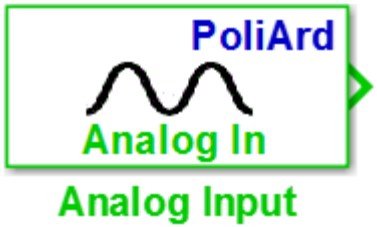
Useful to implement a custom alarm: when it is enabled, all the Digital Output, Analog Output and the motors are turned off. Only one Alarm block can be present in the same program.



Analog Input

Measures the voltage of a specified analog input pin. The block acquires the voltage as a ***digital value*** (12-bit ADC, 0-4095, minimum to maximum).

AI1-AI6: Input signals (±10 V)
AI7: NTC (0-3.3 V)
AI8-AI9: Potentiometers (0-3.3 V)
AI10: Vdc (0-3.3 V)

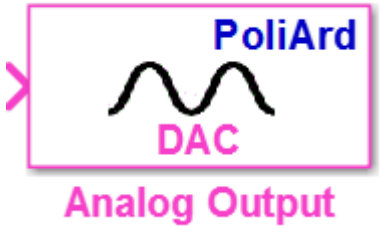


Do not assign the same pin number to multiple blocks within a model.

Analog Output

Set the value of a specified analog output pin. The block generates the voltage as a ***digital value*** (12 bit, 0-4095, minimum to maximum).

AO1-AO2: Analog Output (±10 V)

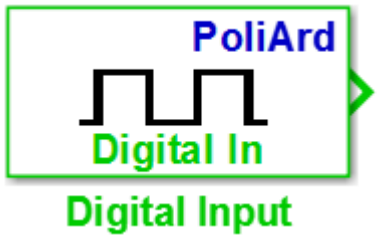


Do not assign the same pin number to multiple blocks within a model.

PoliArd Library

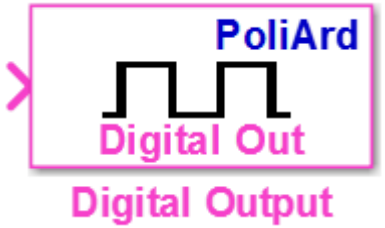
Digital Input

Get the logical value of a specified digital input pin.
DI2-DI3: Switch 2-3



Digital Output

Set the logical value of a specified digital output pin.
DO1-DO2: LED 1-2

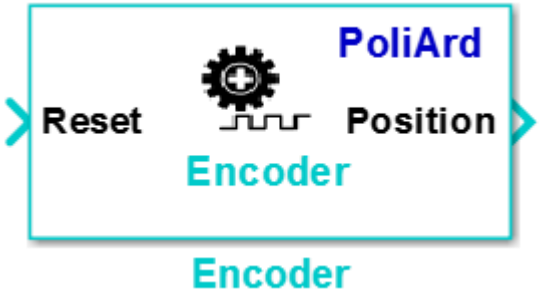


Do not assign the same
pin number to multiple blocks within a model.

Encoder

Get the *Position counter* value for the encoder selected.
If *Reset* is enabled, the counter is set to zero.

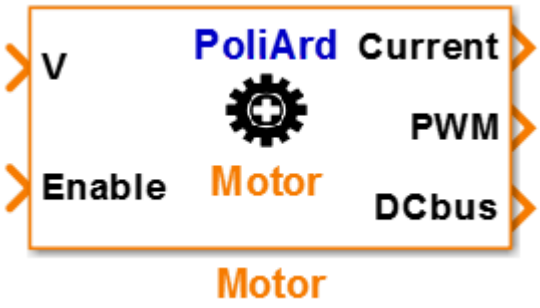
Encoder ID 1-4



Motor

Command the Motor selected, by imposing a proper voltage (input *V* signal is in volt).
Enable boolean signal turns on the Motor.
The block gets the measured *Current* (in ampere), the *PWM* (12 bit) and the *DCbus* (in volt) set internally.

Be sure to select the proper Vdc!

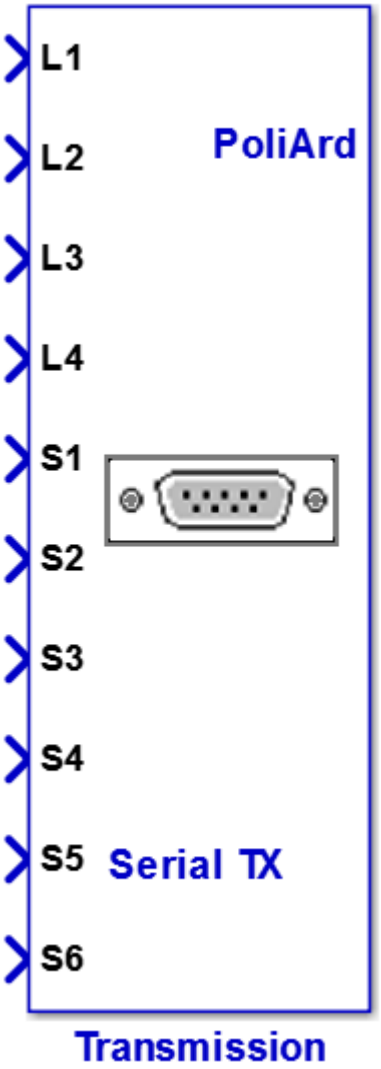


Motor ID 1-2

Transmission

Send data on serial port to PoliArd Scope software.
The transmission frequency must be lower than the loop frequency and must match the parameter set in PoliArd Scope.
The number of float signals must be specified in the block parameters.

L1-L4: logical (boolean) values
S1-S6: float (single precision 32-bit) signals



Outline

The PoliArd Project

- PoliArd Board
- PoliArd Simulink Library
- **PoliScope**

- Examples

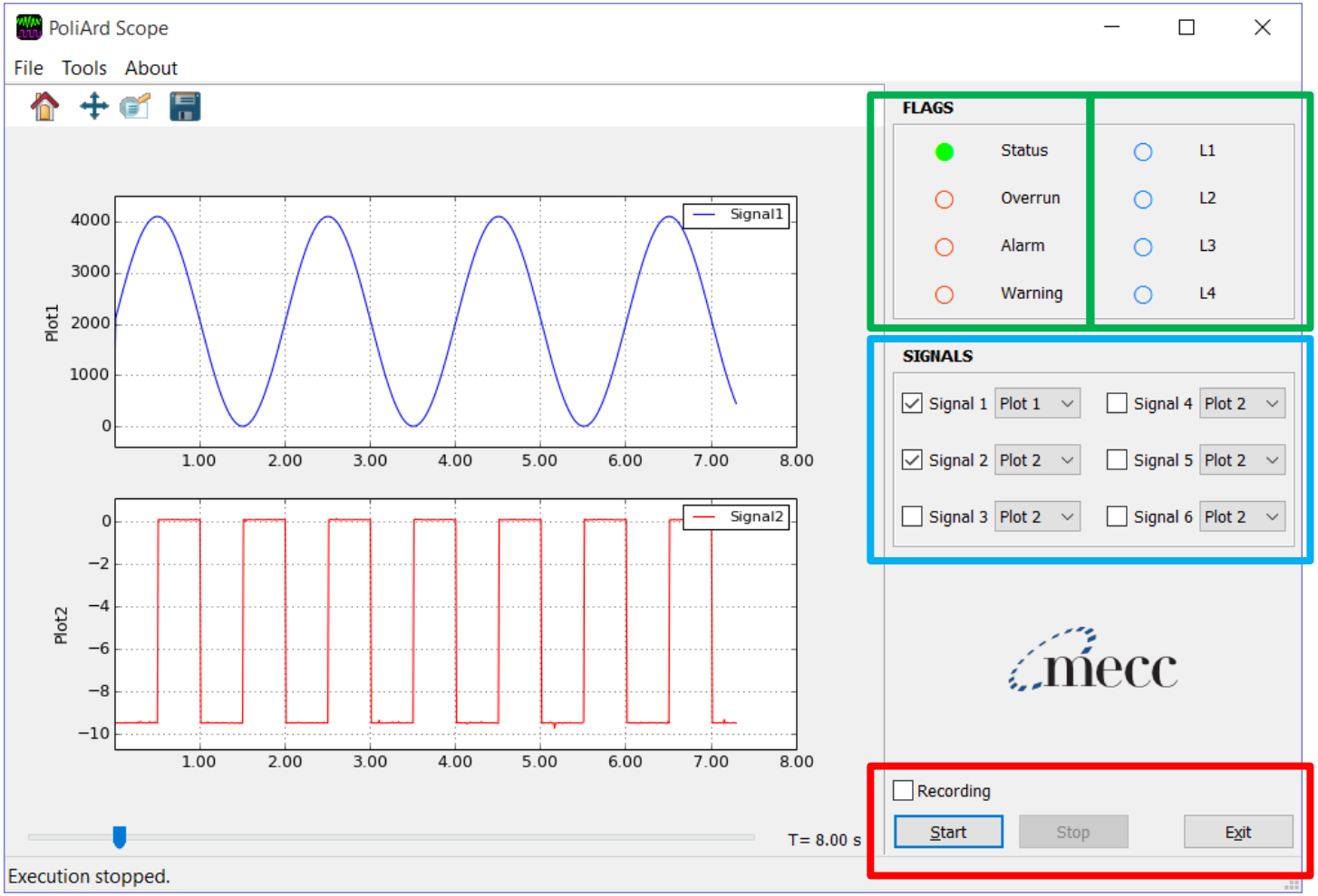
Installation

PoliScope is a **virtual oscilloscope** that reads, shows and records data from the PoliArd Board.

You can download the installer here:
http://mecsys.mecc.polimi.it/mechlab/files/poliard/PSInstaller.exe

Note:
You can install PoliScope even without installing PoliArd Library. Just plug the USB cable to your PC and you will be able to connect to the PoliArd Board.

Interface



Boolean flags visualization

Float signals configuration

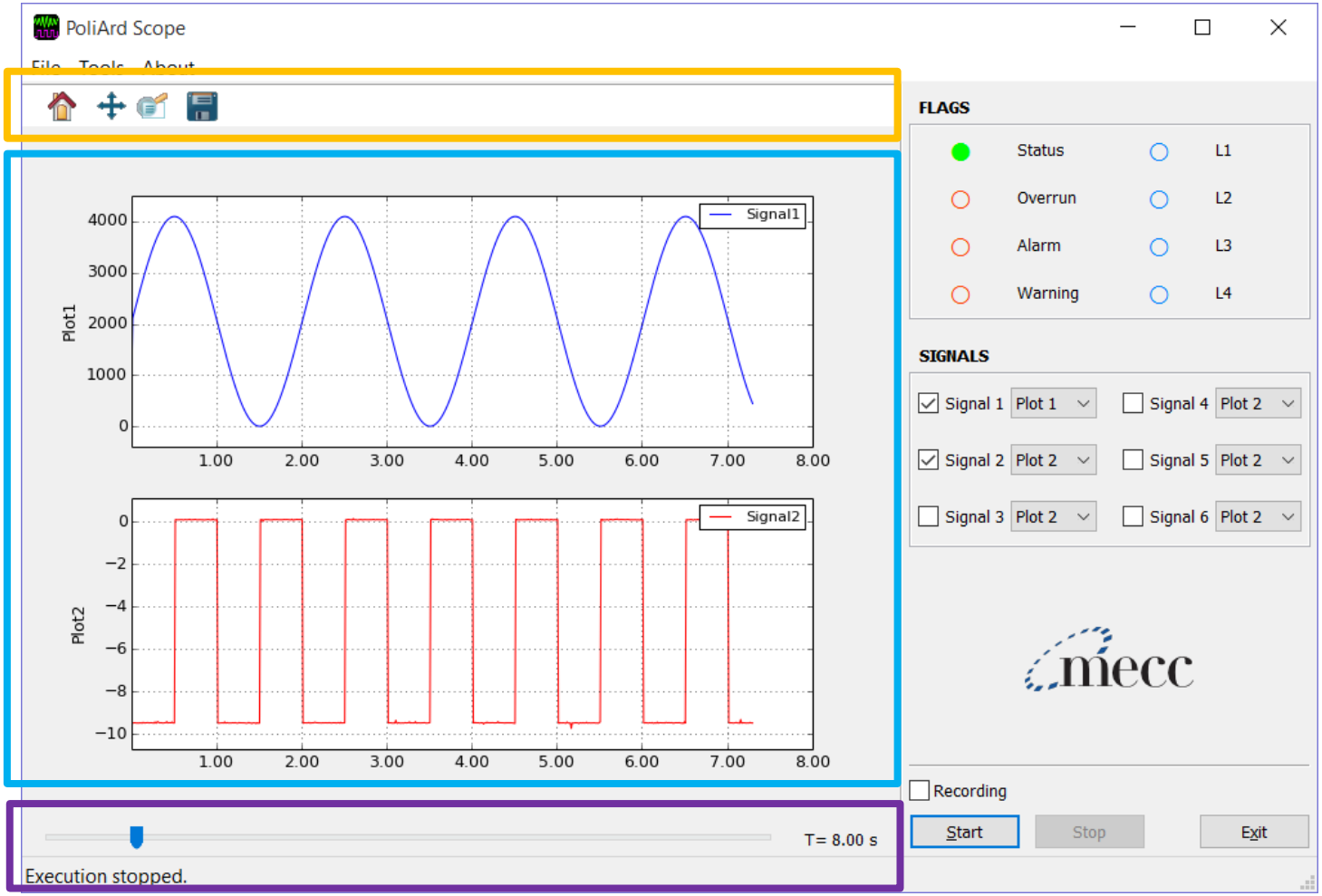
Acquisition controls

Interface

Plot toolbar

Plots for signal visualization

Time buffer slider



Recording mode

PoliArd Scope can write to file the data acquired.
Before starting the acquisition, check the ***Recording*** box.

PoliArd writes a **configuration file**, containing the details about the acquisition, and saves only the selected signals in multiple **data files**.

The configuration file and the data files are stored in a folder named with a unique **timestamp**.

N.B. When the acquisition is started, the board is **reset**!

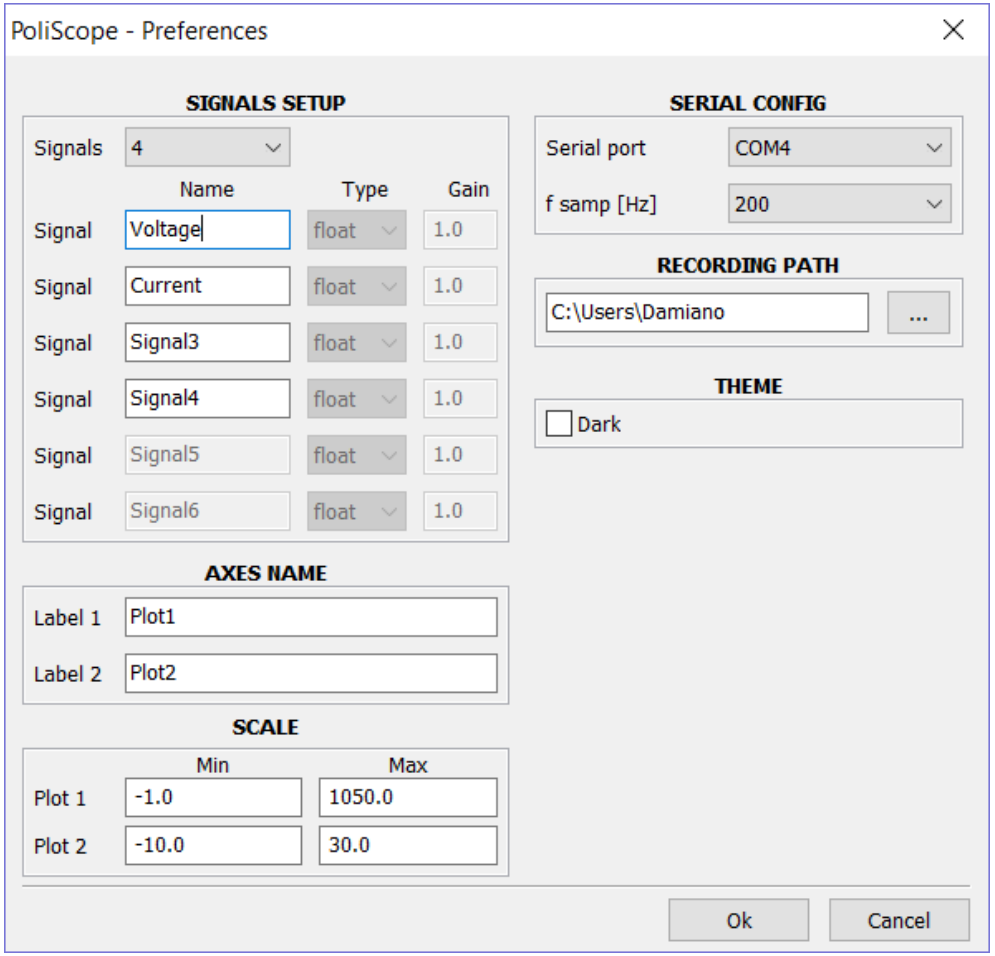
Configuration Panel

You can set the name of the signals and the y-axes.

The *Recording Path* is the base path for the saving folders.

Warning

- Before the acquisition be sure to select the proper COM port.
- The sampling frequency and the number of signals **must match** the parameters configured in the Transmission block!



PoliScope

You can import the saved data to MATLAB workspace.
Use the MATLAB function

PA_import.m

that allows you to load and merge the files recorded with PoliScope.
Syntax examples:

```
acq = PA_import                                  Wizard mode
acq = PA_import(acq_dir)                         Acq folder
acq = PA_import(acq_dir,show_plot)               Show plot (bool)
acq = PA_import(acq_dir,show_plot,save_name)     Save to .MAT
```

You can download this function from:
http://mecsys.mecc.polimi.it/mechlab/files/poliard/PA_import.zip

Outline

The PoliArd Project

- PoliArd Board

- PoliArd Simulink Library

- PoliScope

- **Examples**

Programming Procedure

1. Check that the **Emergency button** is pushed down and the **Enable switch** is down

2. Create your Simulink model, setting the **Target Hardware** options (Arduino Due, Enable overrun detection, Fixed-step size)

3. Implement your Simulink model using blocks from the **PoliArd Library** (always insert PoliArd Init block)

4. Download your control logic to the PoliArd Board using the **Deploy to Hardware** button. Check if the procedure is successful.

5. Open **PoliScope** and start the acquisition

6. Release the **Emergency button**

7. Turn on the **Enable switch** on the board

Shut-down procedure

1. Turn the **Enable switch** off
2. Press the **Emergency button**

Emergency procedure

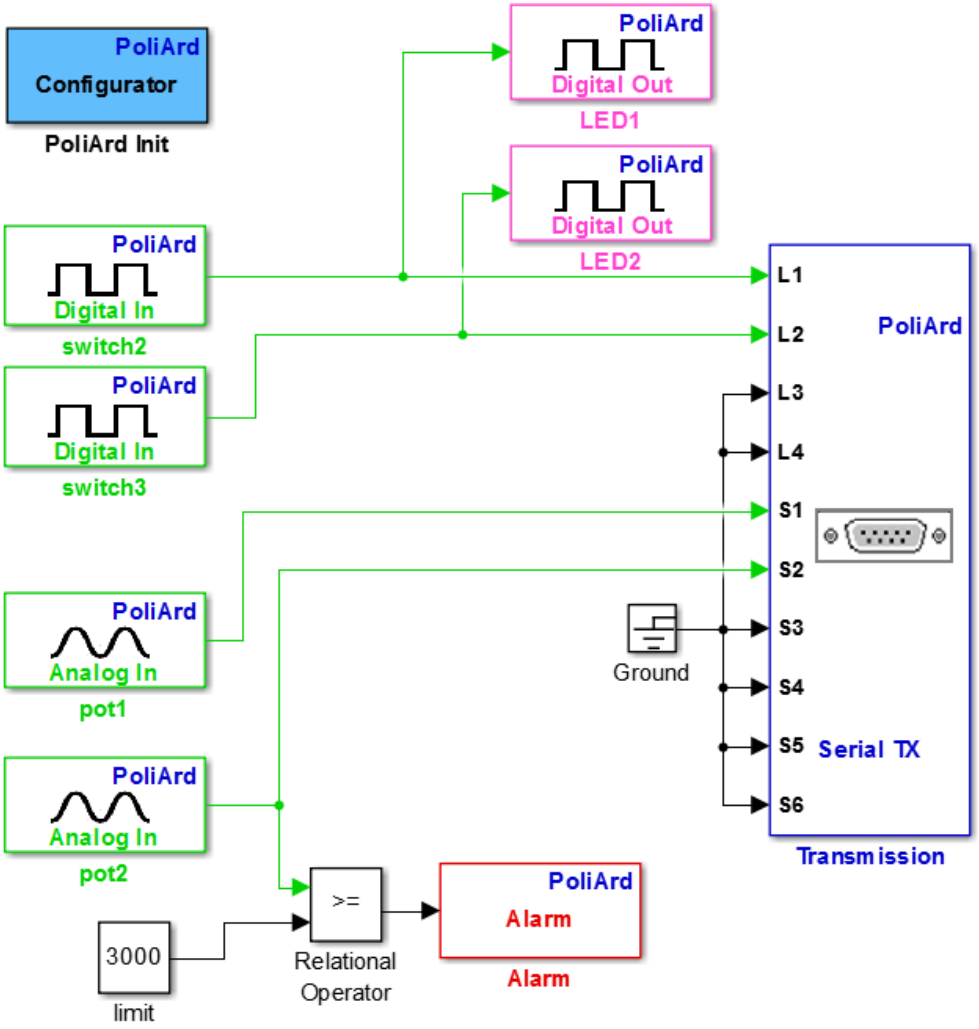
- Press immediately the **Emergency button**

Warnings

- The last downloaded control logic is **always resident and running** on the PoliArd Board!
- Always check the **Power Status LEDs** to be sure that the power supply is not faulty
- **Alarm** or **Overrun status** can lead to unpredictable behavior of the system
- When the acquisition is started in PoliScope the board is **automatically reset**
- Simulink cannot upload a new program to the PoliArd Board if a **serial connection** is opened with PoliScope

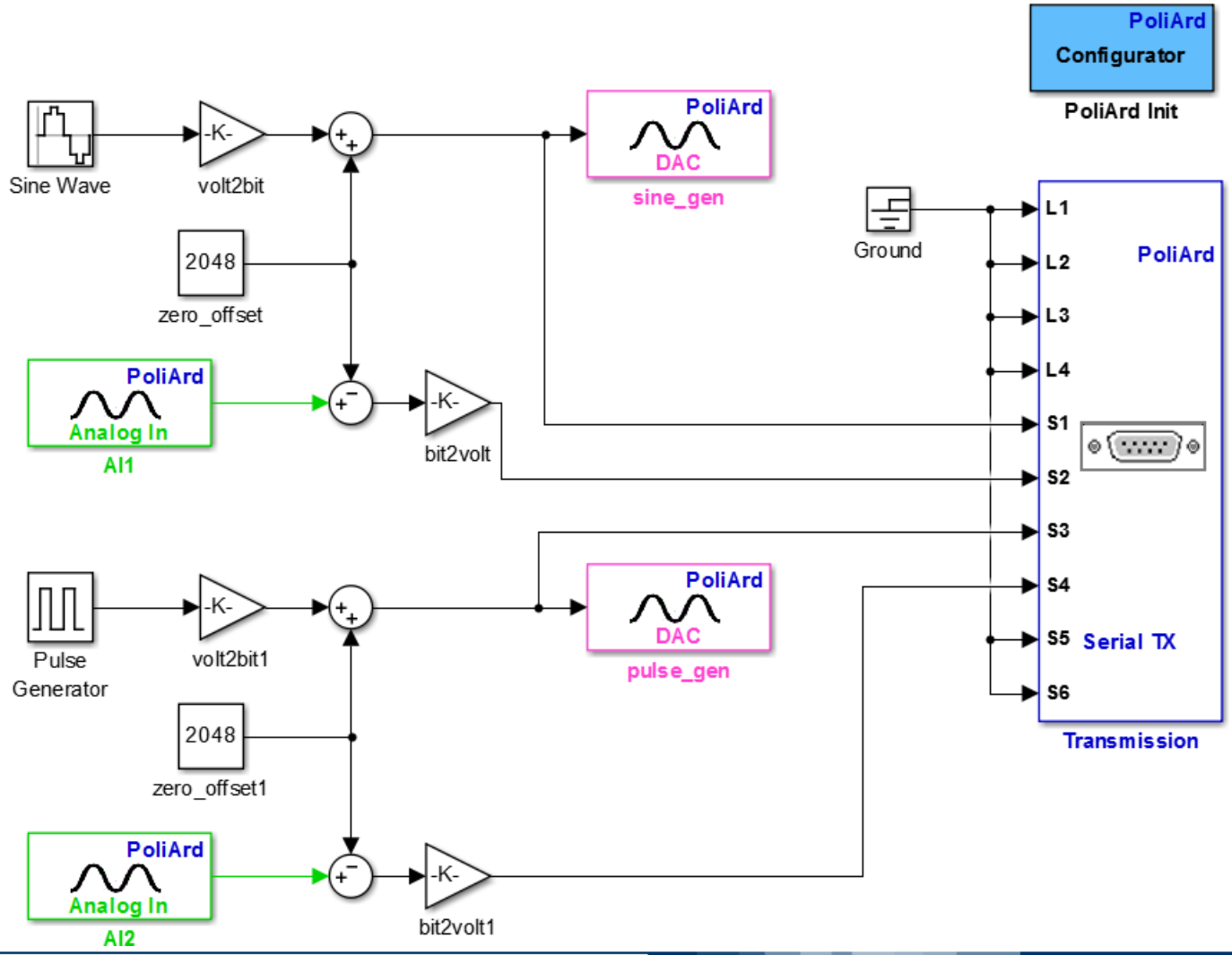
Examples

Digital and Analog signals



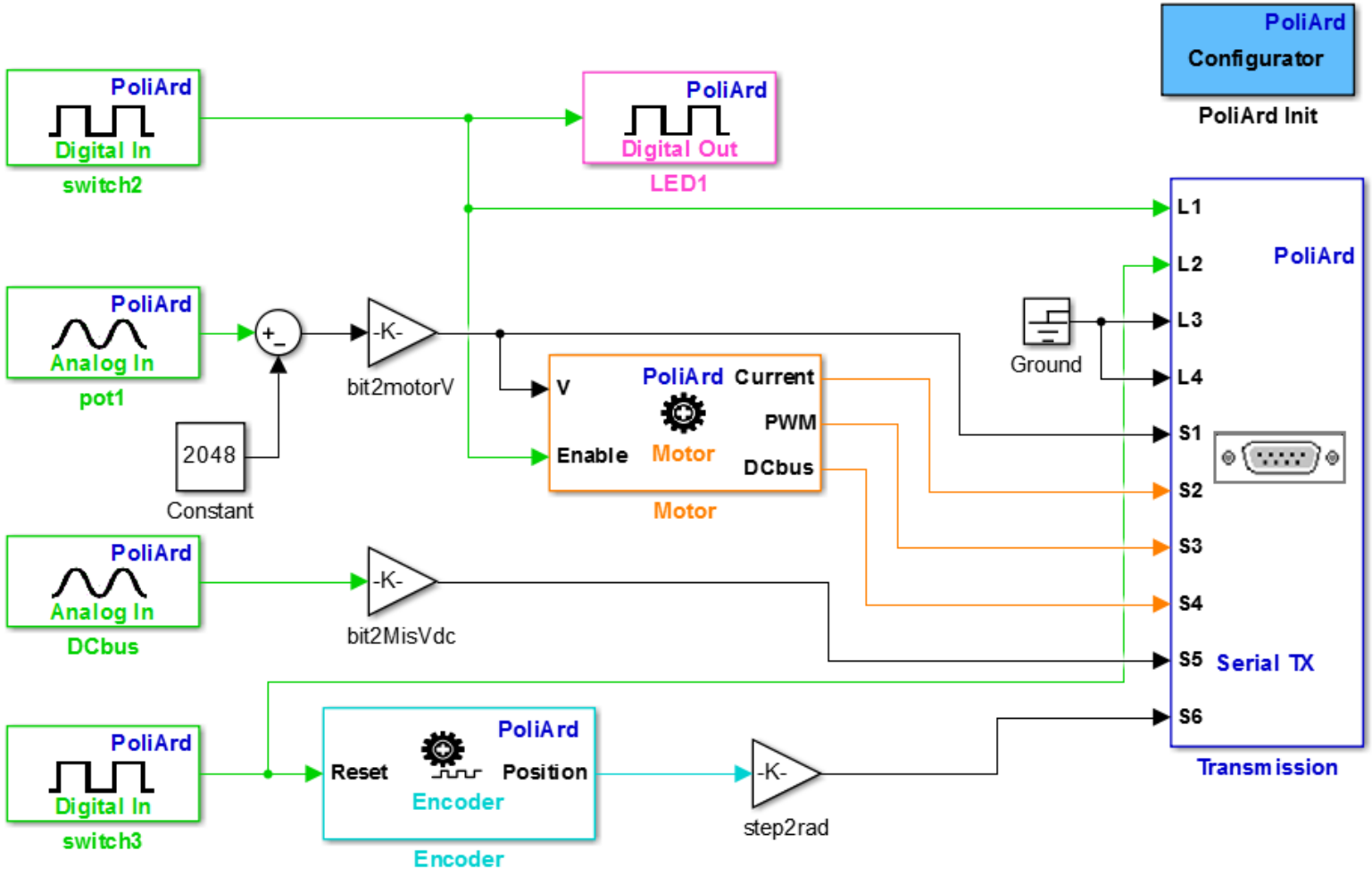
Examples

DAC and ADC



Examples

Motors and encoders



Acknowledgement

Please report any bug or suggestion to the **developers** of the project:

Andrea Costa
andrea.costa@polimi.it

Damiano Milani
damiano.milani@polimi.it

Davide Tarsitano
davide.tarsitano@polimi.it

Visit http://mecsys.mecc.polimi.it/mechlab to download the last revision of the software.