cirklo.org scientific facility management

EKRANO

2015

Contents

1 – Ekrano: What is it?

Ekrano is a multiple sensor data acquiring unit, based on open-source hardware and firmware. Basically, it operates over an Arduino Ethernet platform, connected to a maximum of six analogue sensors, recording all the data read in a Micro SD card which also holds a configuration text file. Besides, it has its own http server which allows the user to directly connect to the platform and read the raw values of each analogue pin. This server presents its information in XML format and can be used to configure the platform with a list of commands sent through an HTTP GET, as it will be described later. It is possible to get a list of the last 10000 lectures (recommended maximum). An early implementation of an actuation system, using Arduino's digital pins and programmed through this server, is also available. This last feature is not fully implemented yet, as the configuration is lost if the Arduino is shut down.

To store values, the time must be defined before, otherwise it is only possible to get the values in real-time. It can be done by introducing a direct HTTP GET command with the actualized time. It has to be done every time Ekrano is reset.

Since Ekrano has an internal HTTP server answering with an XML file, it is possible to access it through a browser and get its info in real-time.

In this manual, we intend to explain how to mount and use the Ekrano system by yourself. Three appendix with useful information were added. The first one shows examples of you should see in a web-browser, depending on what you've requested. The second gives you a variety of analogue sensors you can assemble to use with this system. Appendix three includes some drawings of the plastics which make the box and in appendix four, the material and suppliers we've used can be found, listed by utility.

Feel free to change and use our platform at your will!

Figure 1: Ekrano

2 – How does it work?

2.1 – The unit

The system is exclusively composed by an Arduino Ethernet, a Micro SD Card, a 9V DC power supply, a network connection and up to six low current analogue sensors fed with up to 5V DC.

The Arduino Ethernet can be replaced by an Arduino Uno and an Ethernet Shield (which is a more expensive and physical space consumer option).

We suggest a 4GB Micro SD Card and a 9V power supply which can feed up to 660 mA. To make it easier the sensor connection to the ADC you can use the available screw shields.

Material used on appendix IV.

2.2 – Aether Firmware

This firmware, named Aether and developed by us, is what makes Ekrano what it is. It was developed in Arduino language and is open to anyone to change and use. I can be found on our website and has to be uploaded to Arduino Ethernet. To do so, if you're using Arduino Ethernet, you should need a USB to serial device to do the upload. We suggest the USB2SERIAL Light by Arduino (in some distributors a pack of Arduino Ethernet plus USB2SERIAL is available). Otherwise, if you're using the option Arduino Uno plus Ethernet Shield, you just need an USB cable.

You should install the Arduino IDE, which can be downloaded at http://www.arduino.cc/ . The installation provides the drivers for any of the Arduinos that may be used. Then, just follow the next instructions:

- Connect your Arduino to the computer and wait until the system recognizes it
- Unzip the downloaded Aether folder
- Inside the new folder, open the file with the same name as the folder, with Arduino IDE
- In "Tools -> Board", select the Arduino board you're using (Ethernet or Uno) and in "Tools -> Port" select the port in which the Arduino board is connected
- To finish, just press the "Upload" button or in "File -> Upload"

If the result is positive, you now have the Ekrano system ready to use. Just need one final step: configuring your network settings on the Micro SD card.

2.3 – SD Card Configuration

To work, Ekrano has to be configured by reading a text file, obligatory named as *cinfo.txt*, which should contain a line with exact same number of character and format as the next example:

000.000.000.000;7500;90-A2-DA-0D-C9-AA;192.163.064.015;192.163.064.253;192.163.064.254

000.000.000.000 is the IP of an external data server, like Netsensors. Let it be 000.000.000.000 if you don't intend to use this.

90-A2-DA-0D-C9-AA is the platform MAC Address. It is advised to use the MAC Address suggested in a sticker placed in Arduino Ethernet board

192.163.064.015 is the fixed IP of Ekrano

192.163.064.253 is the DNS Address

192.163.064.254 is the Gateway Address

2.4 – Ekrano Box

A container for Arduino was also designed. The schematics can be found on our website, if you wish to reproduce it. It protects the Arduino board and holds the sensor cables.

To build it you'll need four PMMA plastic boards, drilled and cut as the sketches on appendix III, four spacers and screws to hold all the set. We chose to use stainless steel screws to hold the spacers and nylon ones to hold the Arduino and the cable holder. The figure 2, 3 and 4 shows photos of the mounted structure.

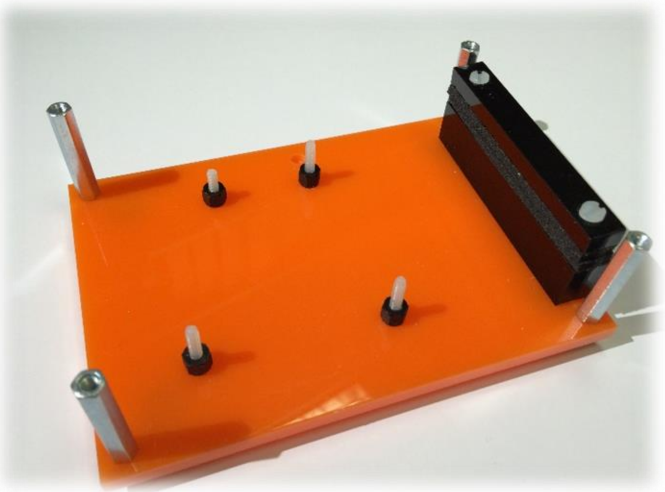

Figure 2: Ekrano box bottom

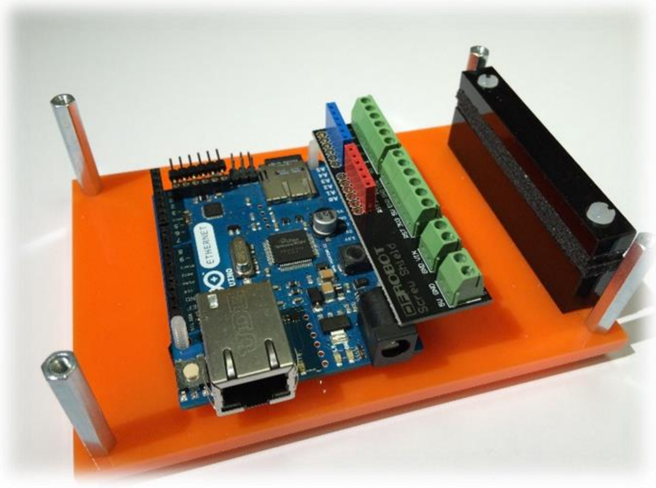

Figure 3: Ekrano with no top cover

Figure 4: Ekrano assembled

In appendix IV, you can find the suppliers, with direct link to the product, to whom we've acquired the materials.

2.5 – The sensors

Ekrano can read up to six analogue sensors which its output voltage is not higher than 5V.

Some sensors, as for temperature, humidity and electrical current have been calibrated by us and used effectively. These and its instructions can be found in Appendix II.

3 – HTTP Server

Ekrano is a completely autonomous platform thanks to the HTTP server. The server answers back with a considerable amount of information, formatted as XML. In this, it is given the MAC Address, the current time of Ekrano, the state of the data server connection, the number of data server writings and reading failed in the connection, the delay time between sensor readings, an error code, the output of the digital pins used in actuation and the values of every sensor.

ADDRESS shall be the IP address of one unit. To simply access this information, http://ADDRESS can be introduced in a web browser address bar. And a page, with information like in appendix I is reproduced.

3.1 – Tag analysis

Next, the tags used to pass the information shall be explained. The beginning and ending will be shown, with a brief description of the information that normally lies between those.

<Devices-Detail-Response></Devices-Detail-Response> - inside this tags lies everything that matters, i.e., this is just a container of all the information explained next.

<MACAddress></MACAddress> - the MAC Address of the accessed Ekrano unit

<Time></Time> - the time configured on Ekrano, printed as year-month-day hour:minute:second. When the platform is turned on, the date is 1970-1-1 and must be changed to start the data acquisition.

<ConnectingServer></ConnectingServer> - can be 0 or 1 and indicates if the connection to na external data server is turned on or off. At start, this option is turned off and can be changed.

<WritesFailed></WritesFailed> - indicates the number of writings which were not accomplished when connecting to the data server. Essentially used for debugging.

<ReadsFailed></ReadsFailed> - indicates the number of readings which were not accomplished when connecting to the data server. Essentially used for debugging.

<Delay></Delay> - the time between each data recording and, if turned on, connection to the data server.

<Error></Error> - a six digit binary code which indicates if an error occurred. Essentially used for debugging.

<DigitalOutput></DigitalOutput> - a seven digit binary code in which each one represents the state (on/off) of the digital pins used for actuation, ordered sequentially (2, 3, 5, 6, 7, 8 and 9).

<ADC<x>></ADC<x>> - where <x> stands for the ADC number (0 to 5) and includes the tags <Value>, <ThresMin>, <ThresMax>, <ActuationMin>, <ActuationMax>, <Hysteresis> and <IOPinout>.

<Value></Value> - the value read at the current moment in the specified ADC. 7 sal comprehended between 0 and 1023 which can be linearly converted to a voltage by dividing this value by 204.6.

<ThresMin></ThresMin> - the minimum alarm threshold defined in the data server for the specified ADC. Not a functional information.

<ThresMax></ThresMax> - the maximum alarm threshold defined in the data server for the specified ADC. Not a functional information.

<ActuationMin></ActuationMin> - the value in which the digital pin assigned to this ADC will be turned on if going below it.

<ActuationMax></ActuationMax> - the value in which the digital pin assigned to this ADC will be turned on if going over it.

<Hysteresis></Hysteresis> - the absolute hysteresis value (not a percentage).

<IOPinout></IOPinout> - this codes indicates which digital pins are associated with this ADC in case of actuation taking part.

If the command to get the last y values recorded is sent, the formatting will not change with the exception of the <ADC<x>> which shall be replaced by <LastData>. Inside this tag there will be y <Data<z>> tags. <z> is the number from 0 to y. Each of this tags will include the date and values of the six analogue pins at that time.

3.2 – Commands

The easiest way to manually configure the platform is using an internet browser and sending commands through the address bar. As before, ADDRESS shall be the IP address of one unit.

To send a command, it is only needed to type http://ADDRESS/comand=value.

Let's turn this simpler giving an example to each of commands available. **It's is important to state that the value has to always have the same amount of characters as the example.**

Time – time configuration

This is an essential one, especially if data logging is needed.

http://ADDRESS/time=20150504163000

Meaning: the time is set to the 4th of May, 2015, 16 hours, 30 minutes and 0 seconds.

Delay – delay time, between recordings, configuration

http://ADDRESS/delay=0300

Meaning: sets the delay between each recording of values and, if turned on, data server connections to 300 seconds. The default value is a random number between 0 and 1023.

Sconn – turns on or off the function to connect to a data server

http://ADDRESS/sconn=0

Meaning: the function which connects Ekrano to a data server is turned off. To turn off, the "0" should be replaced by "1". This function is only used if connecting to an external dataserver.

ldata – requests a chosen amount of data recorded to be shown

http://ADDRESS/ldata=0010

Meaning: request the last 10 lines of recorded data to be shown in the web browser. Something similar to what is shown in appendix I shall appear.

prog – used to configure the actuation values of a chosen analogue pin

http://ADDRESS/prog=2,0010,064,1000,001,030

Meaning: the actuation in the analogue pin 2 is set to be on when the value is inferior to 10 in the code 64 pin configuration and when the value is superior to 1000 in the code 1 pin configuration. The hysteresis is of 30 units.

The pin configuration is a little bit trickier and need some more explanations. It is a decimal value between 0 and 127.

There are 7 digital output pins available to be used. 2, 3, 5, 6, 7, 8 and 9. May it be converted in a 7 digit binary code where the first element starting at right is 2, the second is 3, the third is 5 and so on, being 9 the seventh element. Calculating the decimal value of each digit, the following table is created.

Digital Output Pin	9	8	7	6	5	3	2
Decimal Value	$2^6 = 64$	$2^5 = 32$	$2^4 = 16$	$2^3 = 8$	$2^2 = 4$	$2^1 = 2$	$2^0 = 1$

Each digital output pin has a value that is used to define which one or ones are used in the actuation of a certain analogue pin read.

Let's see some examples:

If digital output 8 is to be used in the actuation of the lower threshold of the analogue read 0, and the digital output 5 to its higher threshold, 32 should be used to define the first and 4 to define the last.

http://ADDRESS/prog =0,0014,032,9000,004,030

This command is configuring the Ekrano to act on the analogue read 0, when the value is below 14, on pin 8 (configured with 32) and when the value is over 900, on pin 5 (configured with 4). The hysteresis value is 30.

You may be asking why such a complex way to associate a digital output to an analogue input. Couldn't it just be the real digital output identification? Yes, it could, but this way it is possible to associate multiple digital output pins to a single input. How? As shown next!

Imagine you want to associate three outputs (3, 7 and 8) to the lower threshold value (let it be 100) and five outputs (3, 5, 6, 7 and 8) to the higher threshold value (800, just as example) of the analogue input pin 4. The hysteresis shall be 25.

We know the value of each digital output. To 3 the value is 2, to 7 is 16 and to 8 is 32. To associate these three outputs to a single analogue input threshold it is only needed to associate the sum of this numbers. 2 + 16 + 32 = 50.

To associate 3, 5, 6, 7 and 8 we must use the sum of the correspondent values. 2 + 4 + 8 + 16 + 32 + 64 = 126.

In the end, the command should be:

http://ADDRESS/prog=4,0100,050,0800,126,025

Appendix I – Examples

Example of Usual Webpage Appearance


```
<Devices-Detail-Response>
	<MACAddress>90:A2:DA:F:9F:1A</MACAddress>
	<Time>2015-5-18 11:53:0</Time>
	<ConnectingServer>0</ConnectingServer>
	<WritesFailed>0</WritesFailed>
	<ReadsFailed>0</ReadsFailed>
	<Delay>348</Delay>
	<Error>000000</Error>
	<DigitalOutput>0000000</DigitalOutput>
	<ADC0>
		<Value>380</Value>
		<ThresMin>0</ThresMin>
		<ThresMax>1023</ThresMax>
		<ActuationMin>0</ActuationMin>
		<ActuationMax>1023</ActuationMax>
		<Hysteresis>10</Hysteresis>
		<IOPinout>0:0</IOPinout>
	</ADC0>
	<ADC1>
		<Value>377</Value>
		<ThresMin>0</ThresMin>
		<ThresMax>1023</ThresMax>
		<ActuationMin>0</ActuationMin>
		<ActuationMax>1023</ActuationMax>
		<Hysteresis>10</Hysteresis>
		<IOPinout>0:0</IOPinout>
	</ADC1>
	<ADC2>
		<Value>375</Value>
		<ThresMin>0</ThresMin>
		<ThresMax>1023</ThresMax>
		<ActuationMin>0</ActuationMin>
		<ActuationMax>1023</ActuationMax>
		<Hysteresis>10</Hysteresis>
		<IOPinout>0:0</IOPinout>
	</ADC2>
	<ADC3>
		<Value>370</Value>
		<ThresMin>0</ThresMin>
		<ThresMax>1023</ThresMax>
		<ActuationMin>0</ActuationMin>
		<ActuationMax>1023</ActuationMax>
		<Hysteresis>10</Hysteresis>
		<IOPinout>0:0</IOPinout>
	</ADC3>
	<ADC4>
		<Value>359</Value>
		<ThresMin>0</ThresMin>
		<ThresMax>1023</ThresMax>
		<ActuationMin>0</ActuationMin>
		<ActuationMax>1023</ActuationMax>
		<Hysteresis>10</Hysteresis>
		<IOPinout>0:0</IOPinout>
	</ADC4>
	<ADC5>
		<Value>379</Value>
		<ThresMin>0</ThresMin>
		<ThresMax>1023</ThresMax>
		<ActuationMin>0</ActuationMin>
		<ActuationMax>1023</ActuationMax>
		<Hysteresis>10</Hysteresis>
		<IOPinout>0:0</IOPinout>
	</ADC5>
```

```
</Devices-Detail-Response>
```


Example of Webpage Appearance After Using Command ldata=0010


```
<Devices-Detail-Response>
	<MACAddress>90:A2:DA:F:9F:1A</MACAddress>
	<Time>2015-5-18 11:59:5</Time>
	<ConnectingServer>0</ConnectingServer>
	<WritesFailed>0</WritesFailed>
	<ReadsFailed>0</ReadsFailed>
	<Delay>348</Delay>
	<Error>000000</Error>
	<DigitalOutput>0000000</DigitalOutput>
	<LastData>
		<Data0>20150513162959;0372;0366;0361;0353;0333;0367</Data0>
		<Data1>20150513163727;0400;0393;0388;0381;0363;0399</Data1>
		<Data2>20150513164455;0394;0387;0382;0376;0357;0394</Data2>
		<Data3>20150513165223;0405;0398;0393;0386;0369;0404</Data3>
		<Data4>20150513165951;0402;0395;0389;0383;0366;0400</Data4>
		<Data5>20150513170719;0375;0368;0364;0357;0337;0373</Data5>
		<Data6>20150513171447;0366;0360;0356;0350;0327;0361</Data6>
		<Data7>20150513172215;0397;0389;0383;0376;0360;0391</Data7>
		<Data8>20150513172943;0377;0370;0365;0358;0338;0371</Data8>
		<Data9>20150518115848;0401;0396;0392;0386;0370;0397</Data9>
		<Data10></Data10>
	</LastData>
</Devices-Detail-Response>
```

Appendix II – Sensors

Temperature sensor

The suggested temperature sensor is a Betatherm 30K6A309I NTC Thermistor. It has a beta value of 4143K and a resistance of 30000 Ω at 25ºC.

We've a made a calibration for it, which works very well between 0 and 50ºC. It makes it ideal to measure environmental temperature.

To do it, it is needed one of this sensors and a 22000 Ω resistor.

They should be connected as the shown in the next figure:

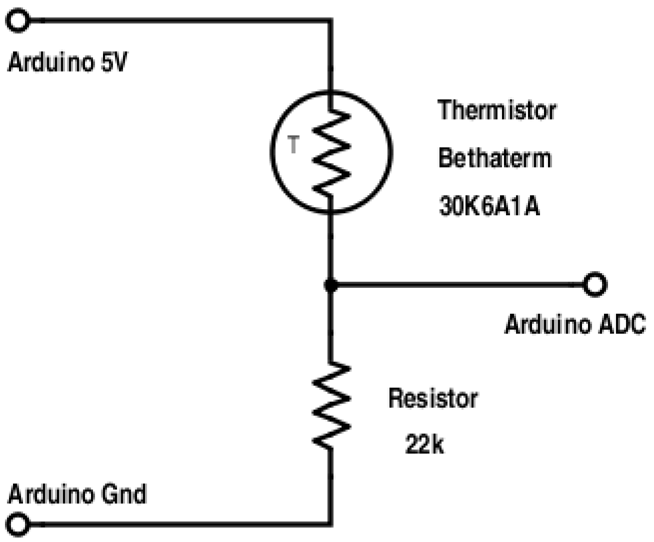

Figure 5: Temperature Sensor Circuit

As known, Ekrano will read a value between 0 and 1023. With the sensor well connected to a specific ADC, to obtain the real temperature the next formula has to be applied:

$$T = 0.09188\, b - 14.8511$$

where T is the temperature in ºC and b the value read at the ADC.

To simplify all the all the mounting and soldering the cables, a small PCB board was designed at Fritzing and the schematic is available at our website.

FALTA IMAGEM DO SENSOR

Figure 6: Temperature Sensor Assembled

Relative Humidity Sensor

To the effect of relative humidity measurement an analogue humidity sensor was used. We suggest the Honeywell S&C HIH-4010-001. It measures the relative humidity between 0 and 100%, giving a linear output voltage between 0 and 5V. It is easily implemented, just by adding a 100000 Ω resistor, as shown in the next figure.

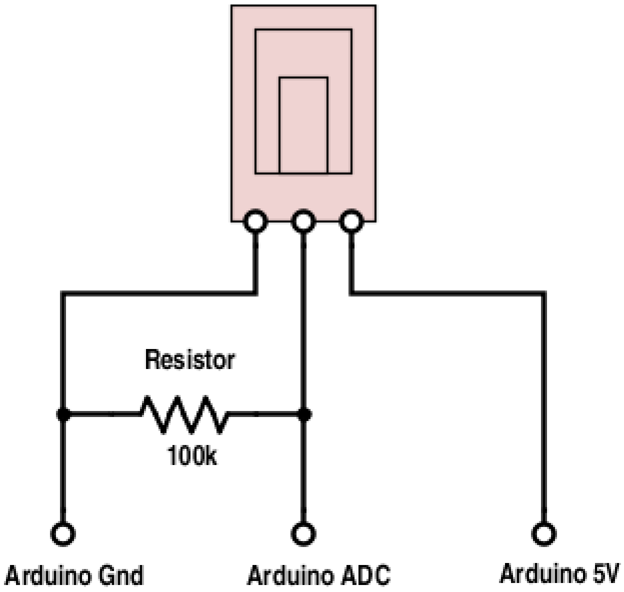

Figure 7: Relative Humidity Sensor Circuit

Figure 4: Temperature Sensor Assembled

With the sensor well connected to a specific ADC, to obtain the real relative humidity the next formula has to be applied:

$$H = 0.15767\, b - 25.81$$

where H is the relative humidity in % and b the value read at the ADC.

Like for the temperature sensor, a PCB board is also being designed to this one but not yet finished.

FALTA IMAGEM DO SENSOR

Figure 8: Relative Humidity Assembled

Electrical Current

For measuring the electrical current we suggest the use of non-invasive AC current sensor (100ª max. That way it is only needed to place the sensor around one cable (line or neutral) and measure its current (AC).

To make the sensor values readable by Ekrano, we've developed a circuit which converts it to up to 5V. The circuit is represented in the next figure.

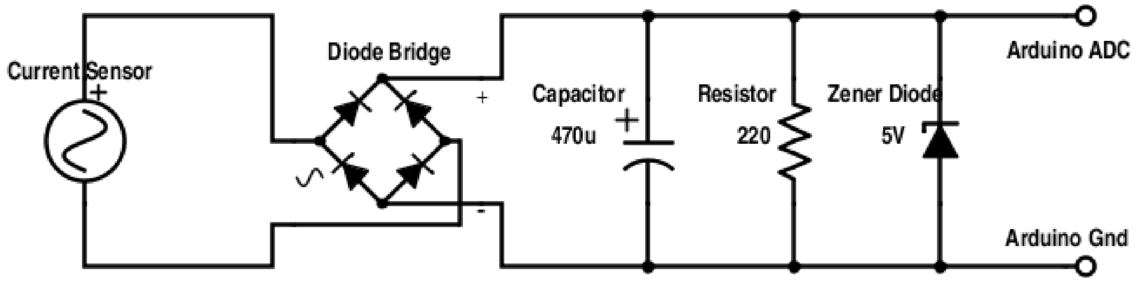

Figure 9: Current Sensor Circuit

If using a 220 Ω resistor, the maximum current readable is around 52 A. To convert the value read by Ekrano to a current:

$$C = 0.05063\ b$$

where C is the electrical current in A and b the value read at the ADC.

By changing that resistor to 150 Ω, it will be possible to read up to 76ª, losing some resolution. The formula to be applied also changes:

$$C = 0.07429\ b$$

where C is the electrical current in A and b the value read at the ADC.

Like for the temperature sensor, a PCB board is also being designed to this one but no yet finished.

FALTA IMAGEM DO SENSOR

Figure 10: Current Sensor Assembled

Soon, more sensor suggestions will be added…

Appendix III – PMMA Plastic Boards

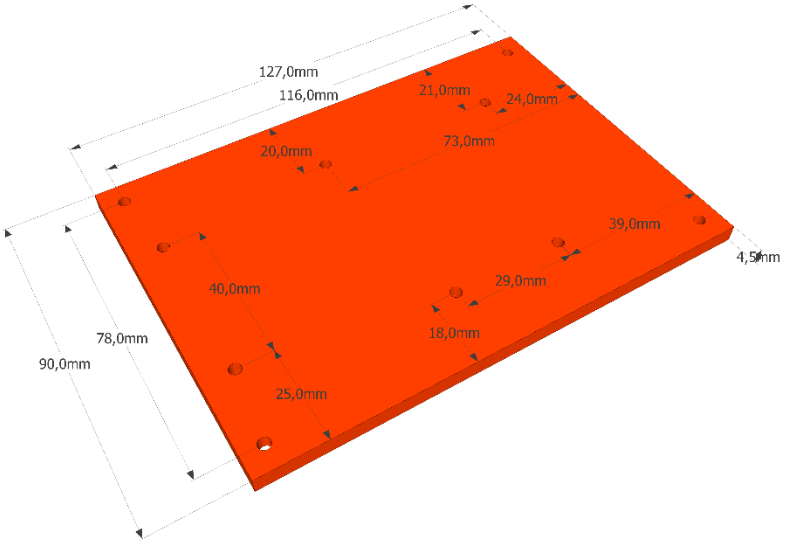

Figure 11: Bottom PMMA Plastic Board (top view)

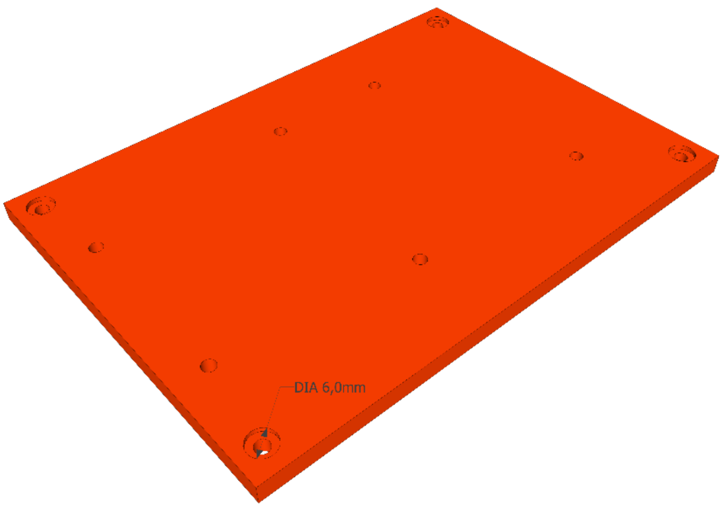

Figure 12: Bottom PMMA Plastic Board (bottom view)

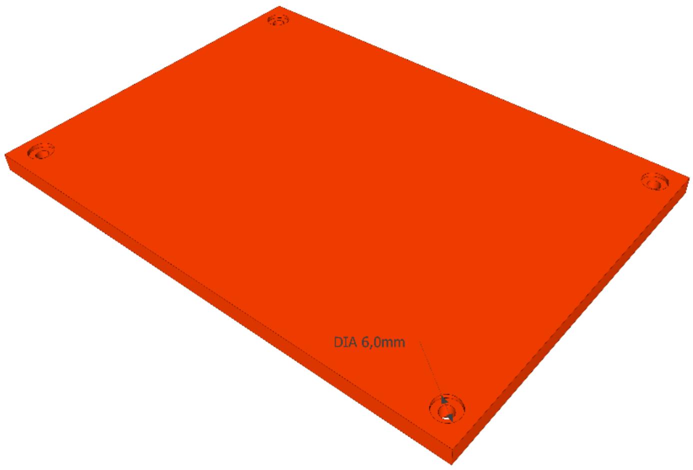

Figure 13: Top PMMA Plastic Board (top view)

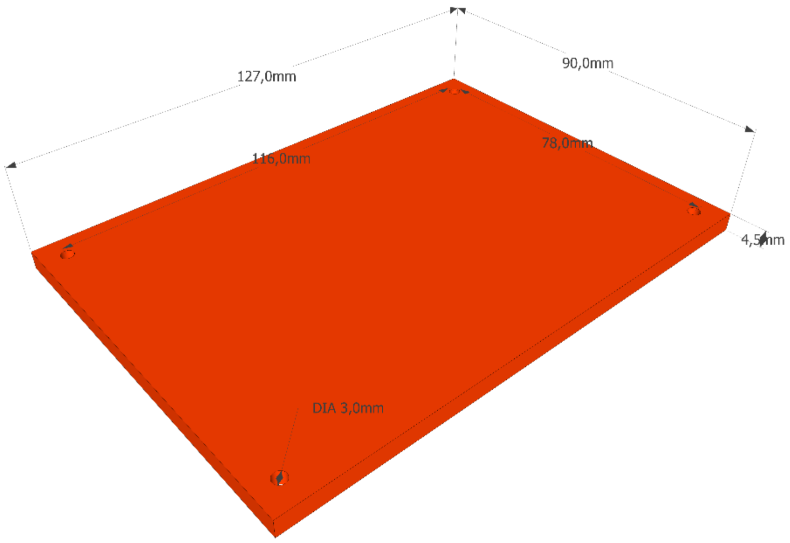

Figure 14: Top PMMA Plastic Board (bottom view)

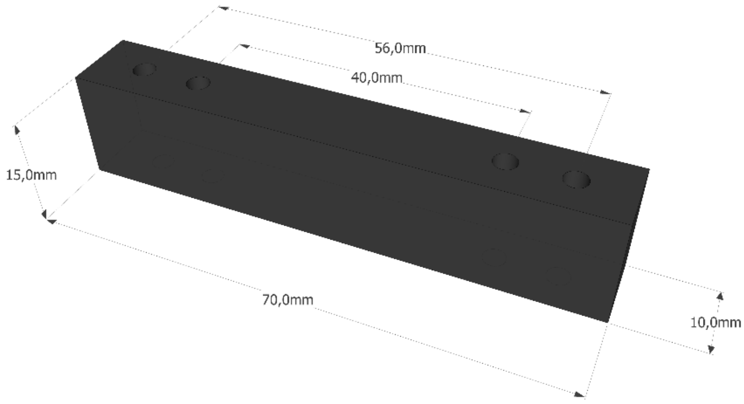

Figure 15: Bottom PMMA Plastic Cable Support

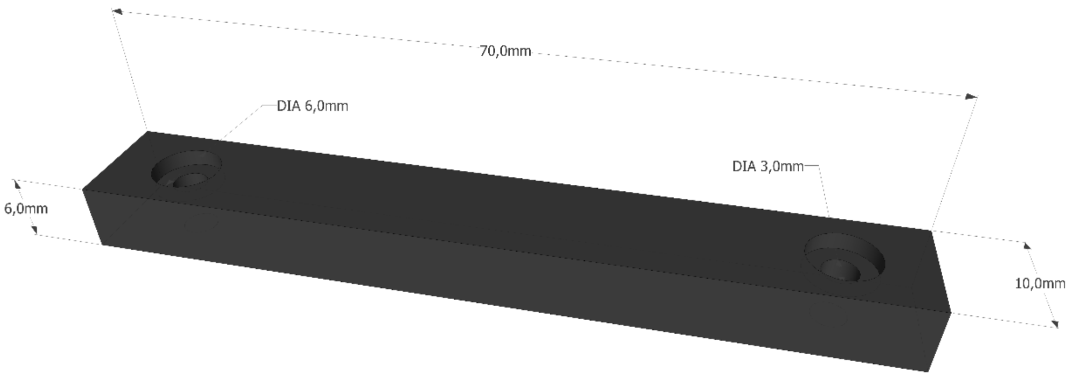

Figure 16: Top PMMA Plastic Cable Support

Appendix IV – Material List

Basic Ekrano

- Arduino Ethernet - http://www.ptrobotics.com/plataformaarduino-e-modelos-alternativos-equivalentes/901-arduino-ethernet.html - 1 unit
- USB2SERIAL Light - http://www.ptrobotics.com/conversores-usb/903-arduino-usb-serial-converter.html
- 9V Power Supply - http://uk.farnell.com/powerpax/sw4306/power-supply-plug-in-9v-0-66a/dp/1971789?categoryId=700000004429 - 1 unit
- 4GB Micro SD Card – any informatics store - 1 unit
- Screw Shield: http://www.ptrobotics.com/shields-proto-e-screw/861-screw-shield-for-arduino.html - 1 unit

Ekrano Box

Besides the PMMA plastic boards cut according to the draws in the previous appendix, it is needed:

- Metallic Spacer M3x30 - http://uk.farnell.com/ettinger/05-03-301/spacer-m3x30-vzk/dp/1466803?ost=1466803 – 4 units
- Stainless Screws M3x10 – any hardware store
- Nylon Screws M3x20 - http://uk.farnell.com/richco/nse-1207-m3-20/set-screw-slotted-csk-m3x20/dp/1261864?ost=1261864
- Nylon Hexagonal Nut - http://uk.farnell.com/ettinger/5-30-305/standoff-hex-nylon-f-f-m3-5mm/dp/1336154?ost=1336154 – 4 units
- Rubber Feet - http://uk.farnell.com/3m/sj5003grey/feet-stick-on-pk56/dp/1165075?ost=1165075 – 1 kit

Temperature Sensor

- Betatherm Thermistor – http://uk.farnell.com/betatherm/30k6a309i/thermistor-ntc/dp/9707280?ost=9707280 – 1 unit
- 22kΩ Resistor - http://uk.farnell.com/multicomp/mf25-22k/resistor-22k-0-25w-1/dp/9341544 - 1 unit

Relative Humidity Sensor

- Honeywell S&C HIH-4010-001 – http://uk.farnell.com/honeywell-s-c/hih-4010-001/sensor-humidity-3-4-5-8v/dp/2103660 - 1 unit
- 100kΩ Resistor - http://uk.farnell.com/multicomp/mf25-100k/resistor-100k-0-25w-1/dp/9341129 - 1 unit

Electrical Current Sensor

- Non-Invasive AC Current Sensor (100A max) – http://www.ptrobotics.com/sensores-de-corrente/2143-non-invasive-ac-current-sensor-100a.html
- Bridge Rectifier - http://uk.farnell.com/multicomp/am152/bridge-rectifier-1-5a-200v/dp/9381449?ost=9381449 – 1 unit
- Jack Connector - http://uk.farnell.com/pro-signal/mj-352w-0/connector-receptacle-3-5mm-phono/dp/1267400?ost=1267400 – 1 unit
- 470 uF Electrolytic Capacitor - http://uk.farnell.com/panasonic-electronic-components/eca-0jhg471/cap-alu-elec-470uf-6-3v-rad/dp/1848526?ost=1848526 – 1 unit
- Screw Terminal Block - http://uk.farnell.com/multicomp/mc24356/terminal-block-wire-to-brd-2pos/dp/2396250?ost=2396250 – 1 unit
- Zener Diode 5.1V - http://uk.farnell.com/vishay/bzx55c5v1-tap/diode-zener-500mw-5-1v-do-35/dp/1779203?ost=1779203 – 1 unit
- 220Ω Resistor - http://uk.farnell.com/multicomp/mf25-220r/resistor-220r-0-25w-1/dp/9341528 - 1 unit
- PCB Board - http://uk.farnell.com/roth-elektronik/re200-lf/pcb-eurocard-fr4-100-x-160mm/dp/1172148?ost=1172148 – 1 unit