

# Write simple applications without programming

The user "guzzj" of the GRIX site ([www.grix.it](http://www.grix.it)) asked if we could use the Theremino system to measure RPMs (revolutions per minute) of internal combustion engines.

This article will demonstrate that with the Theremino system you can build this application (and other similar) even without programming.

## **Preliminary analysis**

To design an application, you must first analyze the problem well and spend some time to think without writing anything.

In this case I did a little research on the net and I also asked "guzzj" about the engines and what to expect from this project.

So in the next two or three chapters we will see only calculations, tables and considerations that will help us to frame the subject.

## Considerations

The voltage socket on the distributor should be a negative voltage from 12 to 16 volts, or from 6 to 10 volts for Porsche and Beetles.

For two-stroke engines, which have no battery, approaching the input wire to the high voltage cable that comes from the reel (before any distributor).

It would not hurt if possible, to see the screen even the battery voltage to check that there are no problems on the system of charging. If the charge controller does not work well you may have voltages on the distributor even up to 20 volts

The strobe guns already do this, but unfortunately they are not accurate below 1000 rpm and sometimes makes it hard to make the measurements.

The range should be between 400 and 12000 rpm, and are asked as accurately as possible between 400 and 5000 rpm.

Normally the duration of the spark goes from 1.3 to 2 milliseconds.

## Useful parameters and calculations

The 2-stroke engines may have 1, 2, 3 or 4 cylinders.

The 4-stroke engines may have 1, 2, 3, 4, 6, 8, 10 or 12 cylinders.

2-stroke engines: an ignition every revolution of the shaft, for each cylinder.

4-stroke engines: an ignition every 2 rev. of the shaft, for each cylinder.

### Calculations for 4-stroke engines

Number ignitions per minute = rev. per minute \* number of cylinders / 2

Number ignitions per second = rev. per minute \* number of cylinders / 120

Rev. per minute = number ignitions per minute \* 2 / number of cylinders

Rev. per minute = number ignitions per second \* 120 / number of cylinders

### Calculations for 2-stroke engines

Number ignitions per minute = rev. per minute \* number of cylinders

Number ignitions per second = rev. per minute \* number of cylinders / 60

Rev. per minute = number ignitions per minute / number of cylinders

Rev. per minute = number ignitions per second \* 60 / number of cylinders

## Frequencies and RPMs

Giri per minute	Engines 4 stroke (Cylinders and frequency in Hz)								2-Stroke Engines (Cylinders and frequency in Hz)			
	1	2	3	4	6	8	10	12	1	2	3	4
20	0.1666	0.3333	0.5	0.6666	1	1.3333	1.6666	2	0.3333	0.6666	1	1.3333
50	0.4166	0.8333	1.25	1.6666	2.5	3.3333	4.1666	5	0.8333	1.6666	2.5	3.3333
100	0.8333	1.6666	2.5	3.3333	5	6.6666	8.3333	10	1.6666	3.3333	5	6.6666
200	1.6666	3.3333	5	6.6666	10	13.333	16.666	20	3.3333	6.6666	10	13.333
500	4.1666	8.3333	12.5	16.666	25	33.333	41.666	50	8.3333	16.666	25	33.333
1000	8.3333	16.666	25	33.333	50	66.666	83.333	100	16.666	33.333	50	66.666
2000	16.666	33.333	50	66.666	100	133.33	166.66	200	33.333	66.666	100	133.33
5000	41.666	83.333	125	166.66	250	333.33	416.66	500	83.333	166.66	250	333.33
10000	83.333	166.66	250	333.33	500	666.66	833.33	1000	166.66	333.33	500	666.66
20000	166.66	333.33	500	666.66	1000	1333.3	1666.6	2000	333.33	666.66	1000	1333.3

**Frequency of ignitions per second according to the revolutions per minute of the crankshaft and to the type of engine.**

This table will not serve us for calculations in the software but only to determine which frequencies in cycles per second (Hz) should we expect.

It took a bit 'of time to fill in the table, but now we know that we must measure frequencies from about 1 Hz to 1 KHz or so. With a frequency-meter to achieve accuracy at such low frequencies you should measure for a long time.

To have a precision of one percent to 10 Hz should do measures of 10 seconds. So we should be content with an accuracy of 10% and a response in the order of seconds. A frequency meter would not work so well, it would be inaccurate and slow.

Fortunately the theremino Master module has the ability to **measure the period**, that is the time that elapses between two successive ignitions. This time is measured in the PIC micro-controller firmware by means of a hardware interrupt and is accurate up to sixteenths of a microsecond.

From the time you can then **derive the frequency**. The conversion from a frequency period is carried out in the HAL application. Just choose "Period" as a kind of Pin and check the box "Convert to frequency".

In the next chapter we will see how to make these settings that will allow us to **simultaneously obtain the maximum speed of response and accuracy**.

## Set a Pin to measure the period

The screenshot shows the Thermano HAL - V7.2 software interface. The main window displays a table of pins and their configurations. The table has columns for Type, ID, Subtype, Dir., Slot, Value, and Notes. The 'Period' pin (ID 1) is highlighted in blue, indicating it is selected. The 'Pwm\_fast' pin (ID 2) is highlighted in yellow, indicating it is active. The 'Unused' pins (IDs 3-12) are in white. The right side of the interface shows the 'Proprietà del Master' and 'Proprietà del Pin' panels. The 'Proprietà del Pin' panel shows the selected pin type as 'Period', slot 1, max value 1000, min value 0, and response velocity 30. The 'Proprietà del Master' panel shows the master name as 'HWLAB', repetition rate (fps) 702, error rate 0.00, and data exchange speed checked. The 'Frequenza' panel shows the 'Converti in frequenza' checkbox checked, with max frequency 1000 Hz and min frequency 0 Hz.

Type	ID	Subtype	Dir.	Slot	Value	Notes
Master	1	HWLAB				
Slave	1	MasterPins				Firmware V5.0
Pin	1	Period	get	1	300.0	
Pin	2	Pwm_fast	set	2	300.0	
Pin	3	Unused				
Pin	4	Unused				
Pin	5	Unused				
Pin	6	Unused				
Pin	7	Unused				
Pin	8	Unused				
Pin	9	Unused				
Pin	10	Unused				
Pin	11	Unused				
Pin	12	Unused				

Here you see the Pin1 configured as "Period". Pin 2 (Pwm\_Fast) served to generate a signal and do without engine tests.

Period using the main interrupt the PIC, for which you can configure as a "Period" just a Pin at a time (and Pin 11 and 12 are not valid).

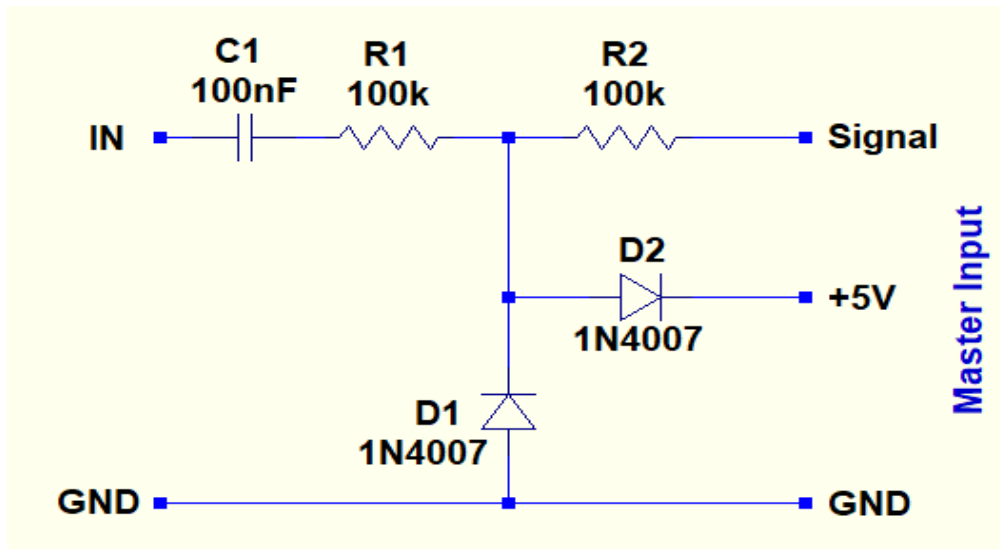
Starting with an empty configuration you must select the Pin chosen line (which turns blue) and then go to the "Pin Properties", click on the "Pin Type" and choose "Period" (if the hardware you choose Period\_pu should require a pullup current, but this is not the case).

Then you must check the box "Convert frequency". We recommend that you leave everything else as it appears until you know the system better (maybe studying the HAL instructions and those of the Master).

From this moment in the Value column you will now see the frequency measured and this value will also be available for other applications on the chosen slot (in this case the Slot 1).

## Input circuit - Theory

This scheme is designed to function as a universal input. The two input wires are connectable to each type of engine.



### operation

The capacitor C1 removes the DC component for which it does not matter if the input voltage is negative or positive.

The resistor R1 (a quarter of watt) limits the current and protects the subsequent components even with input voltages of hundreds of volts.

The diodes D1 and D2 limit the negative voltage from 0.7 volts to 5.7 volts positive and could withstand currents thousands of times higher than those that occur with input voltages of hundreds of volts.

The diodes D1 and D2 may also be the 1N4001 or 1N4002, or any other type that can withstand at least 100 mA and 20 volts. They would also be good Schottky diodes or 1N4148 and 1X8055.

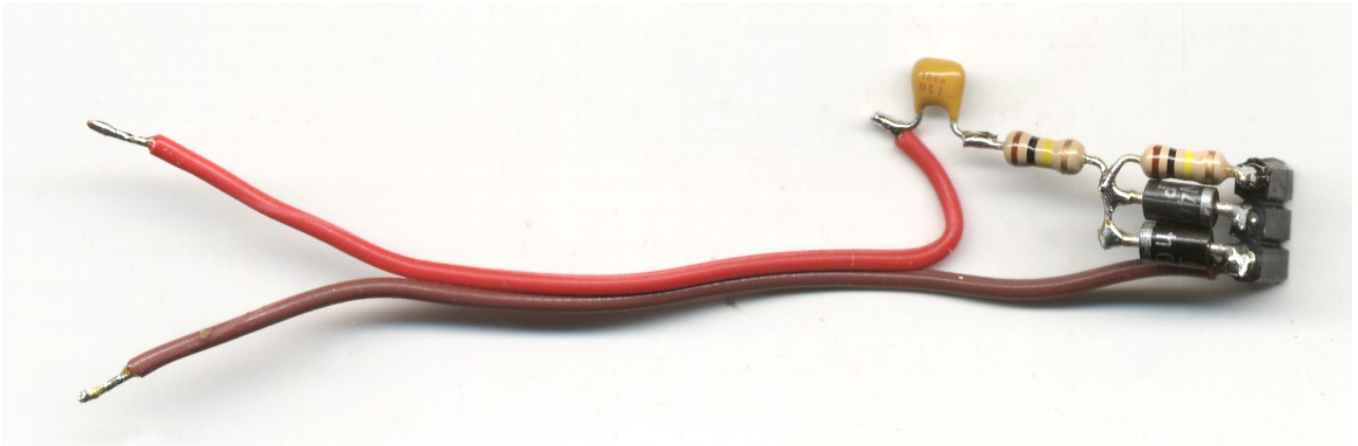
The resistor R2 limits the current that can go towards the PIC. Since the PIC limited inputs from 0.3 volts negative and positive 3.6 volts, the maximum current towards the PIC will be 20 uA. These 20 uA are much lower than the 100 uA that could create problems (interruption of USB communication).

For the operation of this circuit the input voltage excursion must have a minimum of 3 or 4 volts. The maximum allowable voltage is hundreds of volts or even thousands of volts for a few seconds.

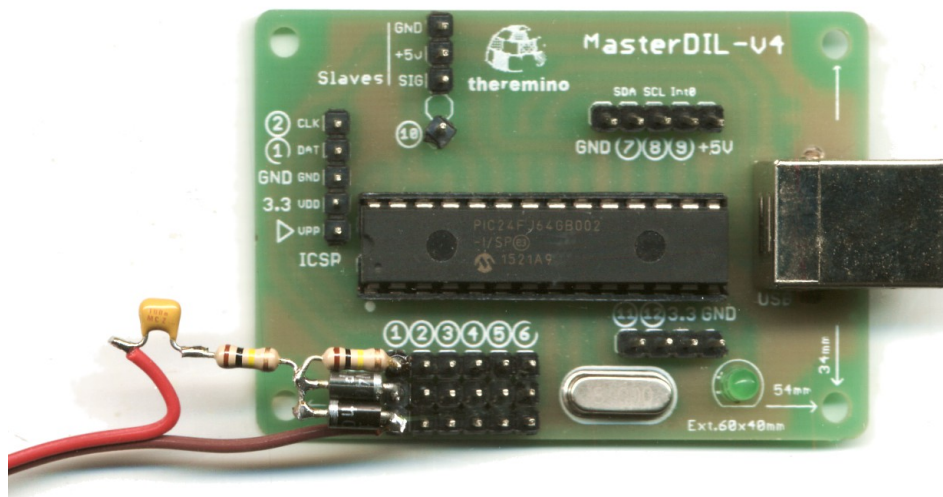
## Input circuit - Practice

The wire GND (brown) must be connected with a clamp (any metallic piece that is also connected to the metal chassis and engine).

The wire IN (red) must be connected to distributor or even just approached the high voltage wire that comes out of the coil.



The three wires are connected to a right of Theremino Master Module Input pins. Maybe you do not notice much but there is a female connector, you do not have to solder the components to the Master.



Everything should be then placed in a small plastic box. It makes a square hole for the USB connector, first with a drill (10 mm) and then widens with a flat file by 10 millimeters.

Finally, make the holes for the bushings from the opposite side. Pass the wires from a hole would be crude and unreliable, the wires could move and tear components.

## Try the basic operation

Before proceeding you should check that the connection to the engine, the master module and the HAL applications work well and produce a frequency proportional to the engine rpm.

It should be verified that the measurement of the frequency is stable, quickly respond and will not occur or uncertainties hops throughout the useful field, from the minimum number of revolutions up to the maximum.

The signal of the distributor can be taken in various ways and may assume different shapes and voltages depending on the vehicle model. So you will have to check that the input circuit that functions well in all cases.

The input circuit proposed should also work on cars with six volt battery and on mopeds but there may be cases of unstable measurements, which measurement is interrupted or that suddenly doubles frequency.

If you experience any frequency doubling you could add a capacitor between Signal and GND to limit the bandwidth.

Try a capacitor of 1000 pF.

If you picks up the signal approaching the input wire to the high voltage cable then this capacitor too could reduce the sensitivity and prevent operation. In this case, to make a few turns on the high voltage wire or reduce the capacitor to 100 pF.

## Coefficients for the calculation

We have already checked the Master and Application HAL. At this point we should have a figure that represents the frequency of the ignitions in cycles per second.

We therefore need constant multiplied by the frequency and get the rpm.

These constants are calculated for different engines with the following two formulas:

- ◆  $\text{RPM} = \text{frequency} * 120 / \text{number of cylinders (four-stroke engines)}$
- ◆  $\text{RPM} = \text{frequency} * 60 / \text{number of cylinders (two-stroke engines)}$

Engines 4 stroke (Cylinders)								2-Stroke Engines (Cylinders)			
1	2	3	4	6	8	10	12	1	2	3	4
120	60	40	30	20	15	12	10	60	30	20	15

**Coefficients to be multiplied by the frequency of ignitions per second in order to obtain the revolutions per minute of the crankshaft.**



## Coefficients directly HAL

You can enter the coefficients directly into the HAL application and get the revolutions per minute even without additional applications.

To enter the coefficients you change the box "Max value".

Engines 4 stroke (Cylinders)							
1	2	3	4	6	8	10	12
120000	60000	40000	30000	20000	15000	12000	10000

**Coefficients to be included in the value "Max" box HAL to get revolutions per minute of the crankshaft.**

2-Stroke Engines (Cylinders)			
1	2	3	4
60000	30000	20000	15000

**Coefficients to be included in the value "Max" box HAL to get revolutions per minute of the crankshaft.**

The number of RPM that will read in the column "Value" is a bit 'small and therefore not very visible, but for the first tests may suffice.

If you want you can double-click on the Pin 1 line and open the simple HAL Oscilloscope display turns into a chart.

In the event to display the revolutions per minute you were using an additional application (eg Theremino Buttons) these multipliers will be put in the application and MaxValue box it will be restored to its natural value ie 1000.

The value 1000 is the standard value used in Theremino system to indicate the maximum.

In Theremino system 0 and 1000 do not indicate the minimum and maximum measurable but they are values that determine a scale ratio. Also MinFreq and MaxFreq are not the minimum and maximum measurable but the two points on the scale of frequencies corresponding to MinValue and MaxValue.

If there is no clear these proportions always put all Min = 0 and Max = 1000 all.

## Improve data stability

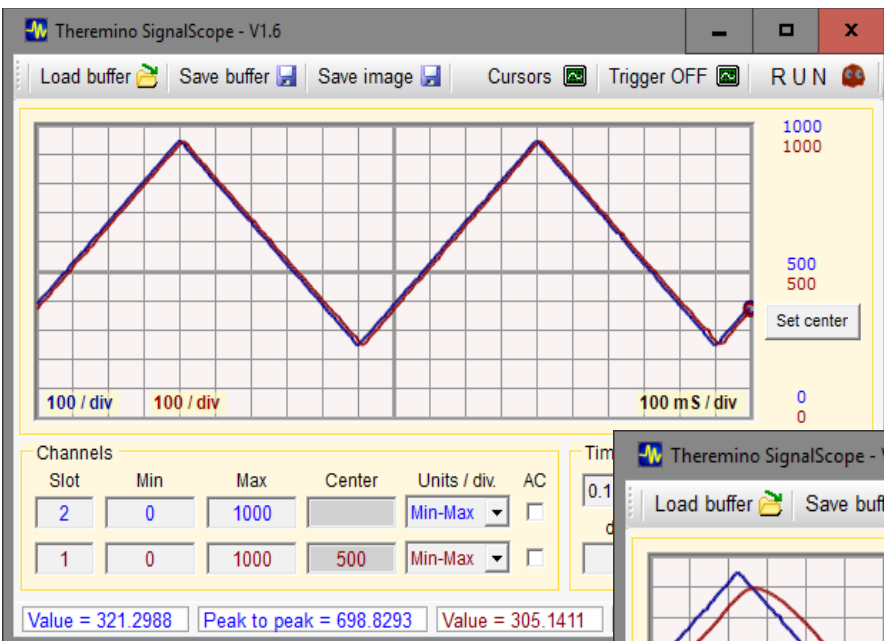
You can try to change the "Speed" value Pin used (the HAL application).

Lowering "Speed" increases the stability of the measures but the response becomes slower.

Pressing the "response rate" button gives you more promptly and more stability. For more information read the instructions of the HAL application.

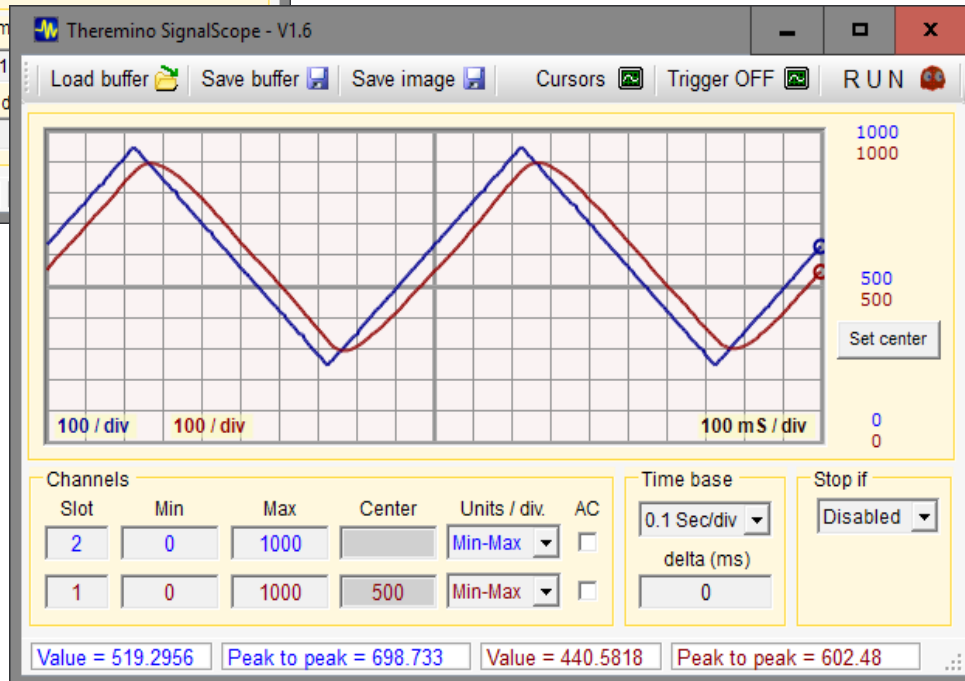
With the SignalScope application you can make more precise measurements and record for a while and then revise and enlarge every detail.

In the next two images we see the effect of variation of the parameter "Speed"



Speed = 100

Speed = 30

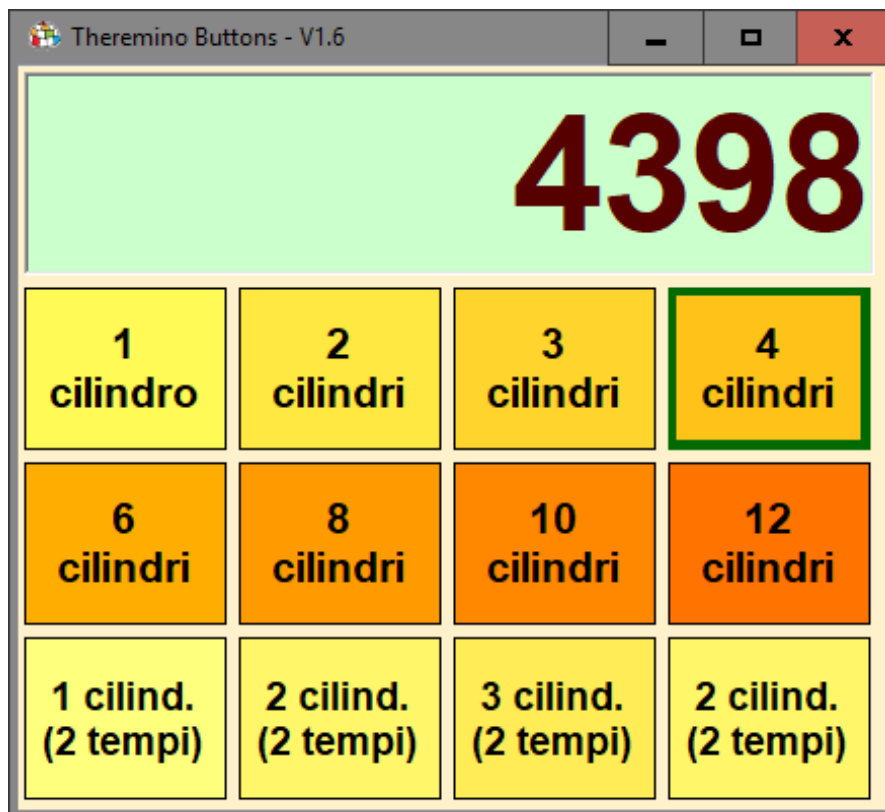


## Build a user interface

A turn meter needs a large display and large, easy-to-use keys, so we will also build a user interface.

The fastest method provided by the system is the "Buttons" application. With Buttons you can create simple applications, without options, without menus and without files, but can perform tasks that are not too complex like this.

You do not need to know a programming language, changes are made by selecting default boxes or writing in a text file.



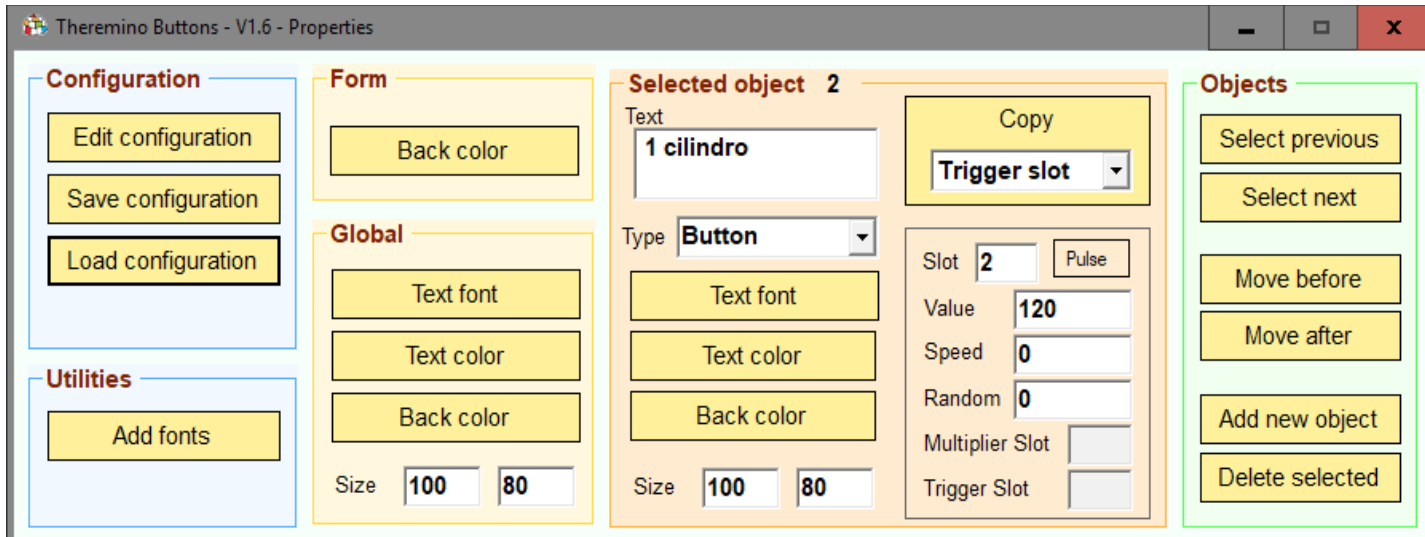
You can define text boxes and buttons, with size, color, font and text as you like. They can then be reordered and placed on multiple rows and columns. Finally, some simple behaviors can be associated with each control.

Applications with big buttons like this can be a great touch-screen panel to use with a tablet.

You can then build portable systems without wasting time to design printed circuit boards, LCD display and buttons. And it also has an easily editable panel in the future, as well as the batteries from the power supply to the law, the remote connection in WiFi and even an infinity of memory and computing power. The all in little volume, with a few watts of total consumption and (all counted) even saving money.

## Defining and editing controls

To open the editor you press the CTRL key and you "click" on a button or on any other area without buttons.



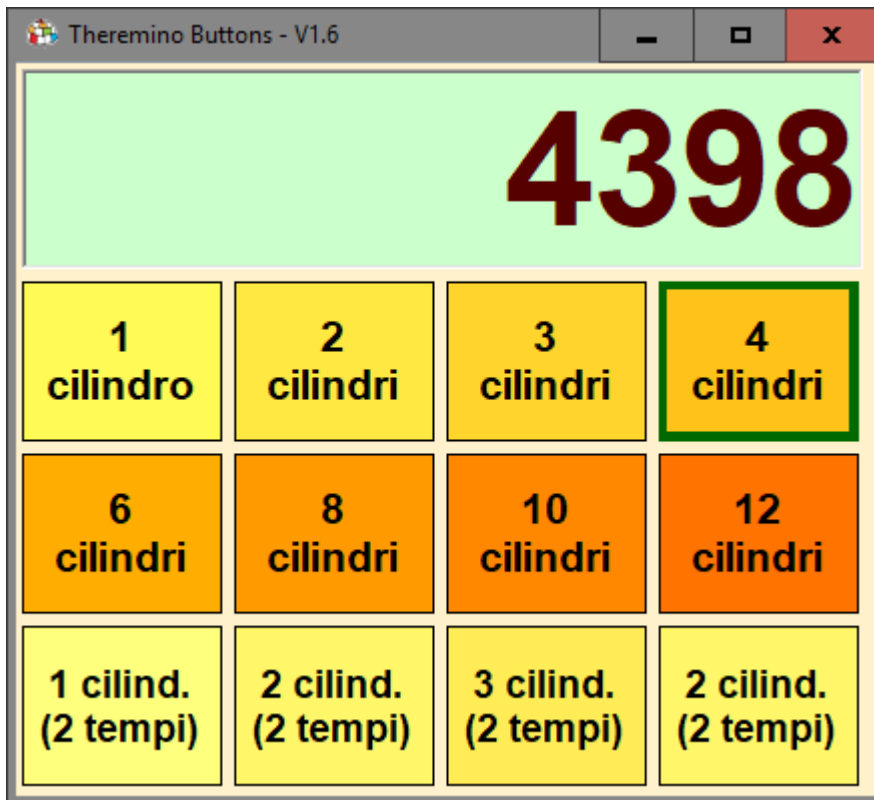
In some cases it may be convenient to edit the configuration files directly. To open it in a text editor and modify it press the "Edit configuration" button.

More information in the file "Theremino\_Buttons\_ITA.pdf" that you download from this page:

[www.theremino.com/downloads/foundations#buttons](http://www.theremino.com/downloads/foundations#buttons)

Following this link you will also find the "Buttons" application and, among its examples, also this RpmMeter application ready for use.

## Principle of operation



The twelve buttons from "1 cylinder" to "2 cylinders (2 times)", when pressed set in Slot 2 a multiplication coefficient.

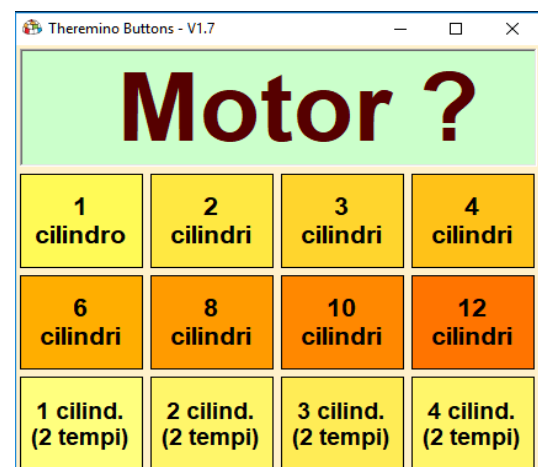
The multiplication coefficients are obtained from the table "Coefficients for the calculation" of the page 8.

The large text box at the top was set to read from Slot 1 the frequency (in Hertz) and multiply it by a coefficient which is located in Slot 2.

When you start the application you have to choose a type of engine otherwise the display always marks zero.

To remind the user, we wrote in the "Text" property the phrase "Motor ?".

This phrase is visible until the Multiplier Slot contains zero and as soon as you choose an engine it is replaced by the number of revolutions.



## Alternatives to "Buttons"

Those who wanted to create a more elaborate user interface could program themselves a dedicated application.

For this Rpm Meter the most suitable possibilities are:

### ◆ VBA (Visual Basic For Applications)

Those who know Excel well know how to write macros could find it simpler than DotNet. In VBA you can easily create panels and buttons and save files. You can also create charts with Excel. We have prepared a module to read and write slots that work with both Microsoft Office, Libre Office and OpenOffice.

<https://www.thereмино.com/en/downloads/foundations#office>

### ◆ VbNet (or Csharp for those who prefer)

We recommend not to create an application from zero, but starting from the [\*EmptyApp\*](#) sample application.

To program will require more time. For those who are not familiar with Visual Studio we have prepared the basic tips in these chapters:

<https://www.thereмино.com/en/downloads/notes-on-software#development>

<https://www.thereмино.com/en/downloads/notes-on-software#advices>

<https://www.thereмино.com/en/downloads/notes-on-software#events>

<https://www.thereмино.com/en/downloads/notes-on-software#visualstudio>

<https://www.thereмино.com/en/downloads/notes-on-software#tools>

## Conclusions

This article was not written to rob the place for a possible article "guzzj" but only to illustrate how to build "the thereminica" applications.

Not having a workshop we will do a few tests. So a complementary article "guzzj" with tests on different types of cars and motorcycles, practical tips and considerations for use, will be definitely helpful for those who will carry this equipment.

What the distributor be with you.

The system Theremino team.