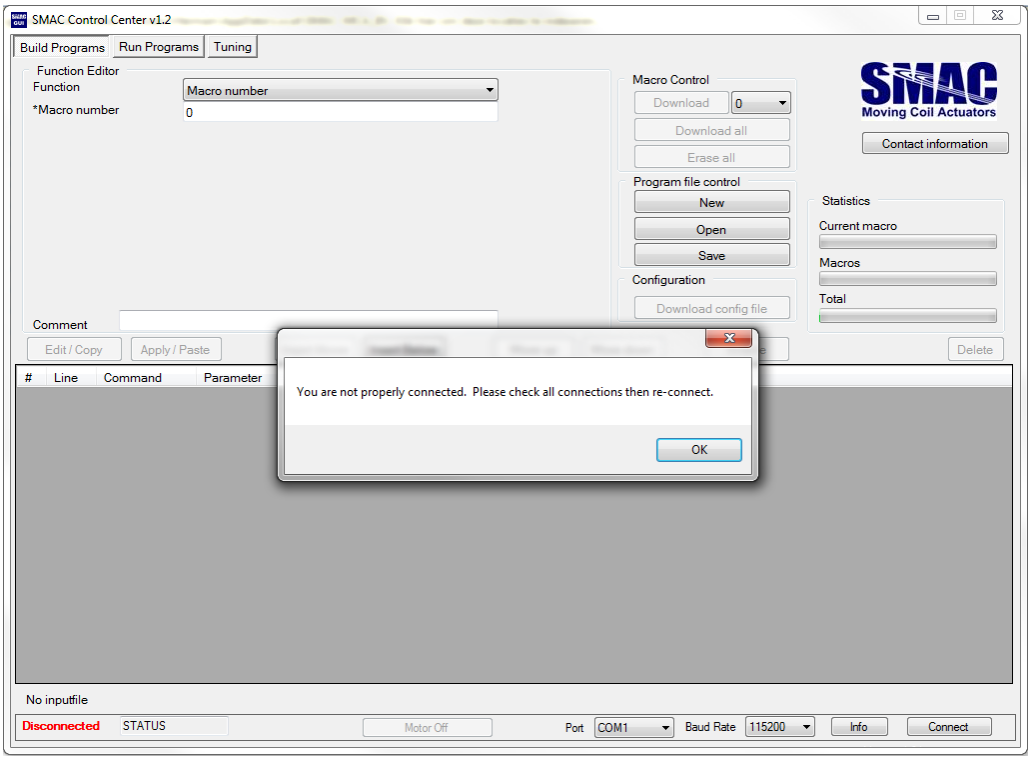
SMAC Control Center Getting Started v1.2

Starting the application

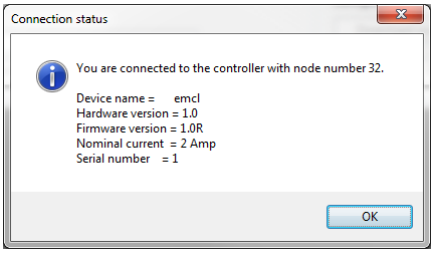
The application can be started by double clicking the installed shortcut on the desktop which looks like this:

At the first start you may get an error message that you are not properly connected. To correct this you have to setup the right communication parameters and then connect to the controller.

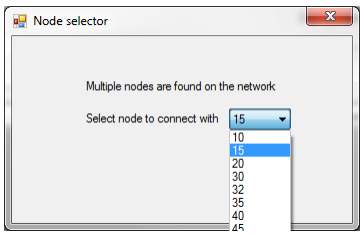


Connecting to the controller

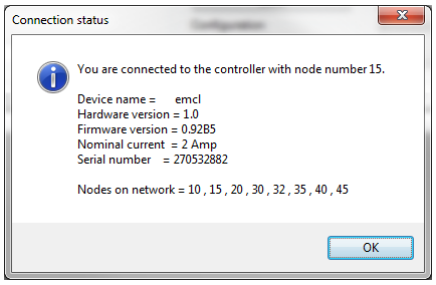
1- Make sure that the controller and the PC are properly connected by a serial cable.
2- Make sure that the controller is powered.
3- Select the right port from the list box in the bottom of the screen.
4- Hit Connect in the bottom right of the screen.
5- SMAC Control Center now checks which controllers are present.
6- If only one controller is present then the following Connection status window will appear



7- If more devices are present then you need to select one of the devices from a list in the following dialog:



After selection of the node the following Connection status window will appear:



Note that this window also lists an overview of the available nodes on the network.

8- After closing the Connection status window the connected node is shown in the lower left of the screen and the text of the Connect button changes to Disconnect.
The Connection status window can be shown again by hitting the Info button in the lower right of the screen.
9- If you want to connect to another node on a daisy-chained network first hit Disconnect, hit Connect again and select the other node.

Writing a program

A program consists of one or more macros. In total 60 macros are available for free use. A macro is identified by its number, the valid range of numbers is 0 to 59.

Macro 0 is a special macro that will always be executed when power is applied to the controller.

Macros can be called or jumped to from other macros. Macros that are called must contain a “Macro return” instruction that will give control back to the calling macro. Macro calls can be nested, up to a maximum nesting level of 9 calls.

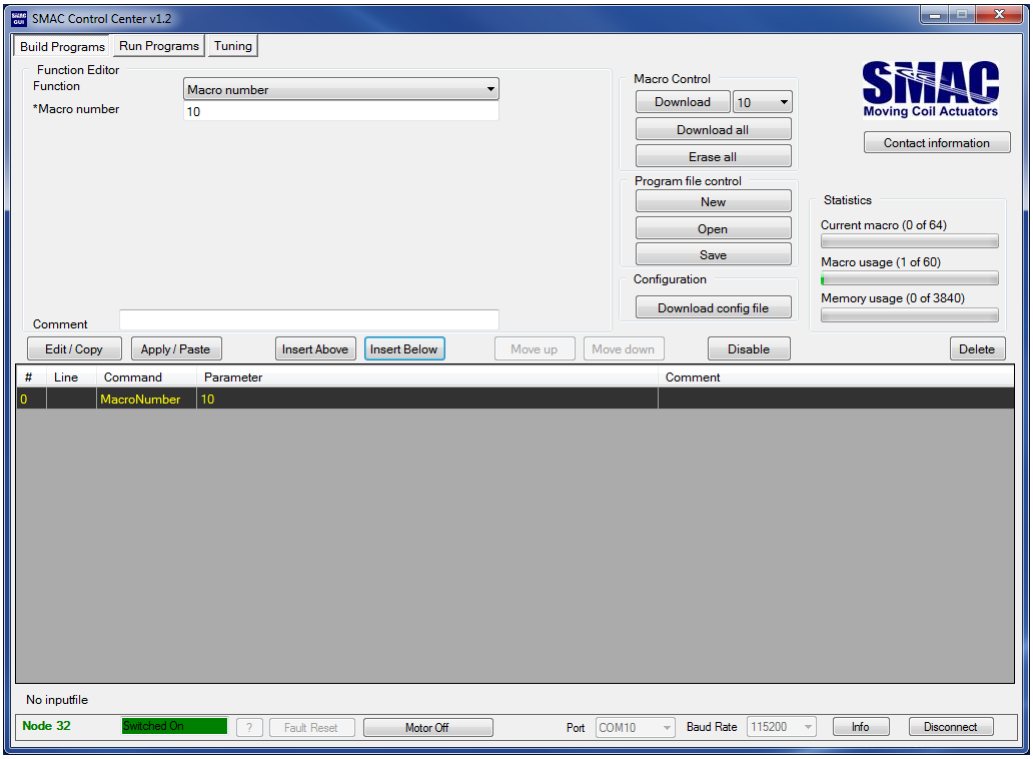
Macros are created with the “Function Editor” that is on the “Build programs” tab of SMAC Control Center.

As an example we will create 2 macros:

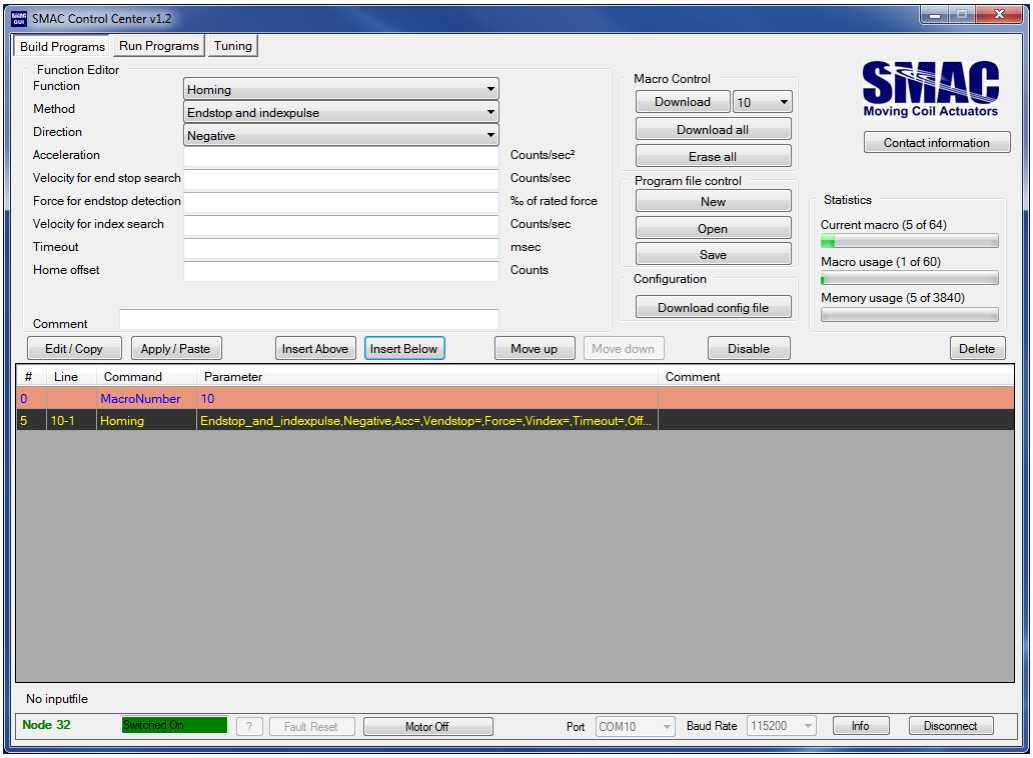
Macro 10 that will perform a homing

Macro 20 that moves the actuator 100 times to position 1000 and back to position 0.

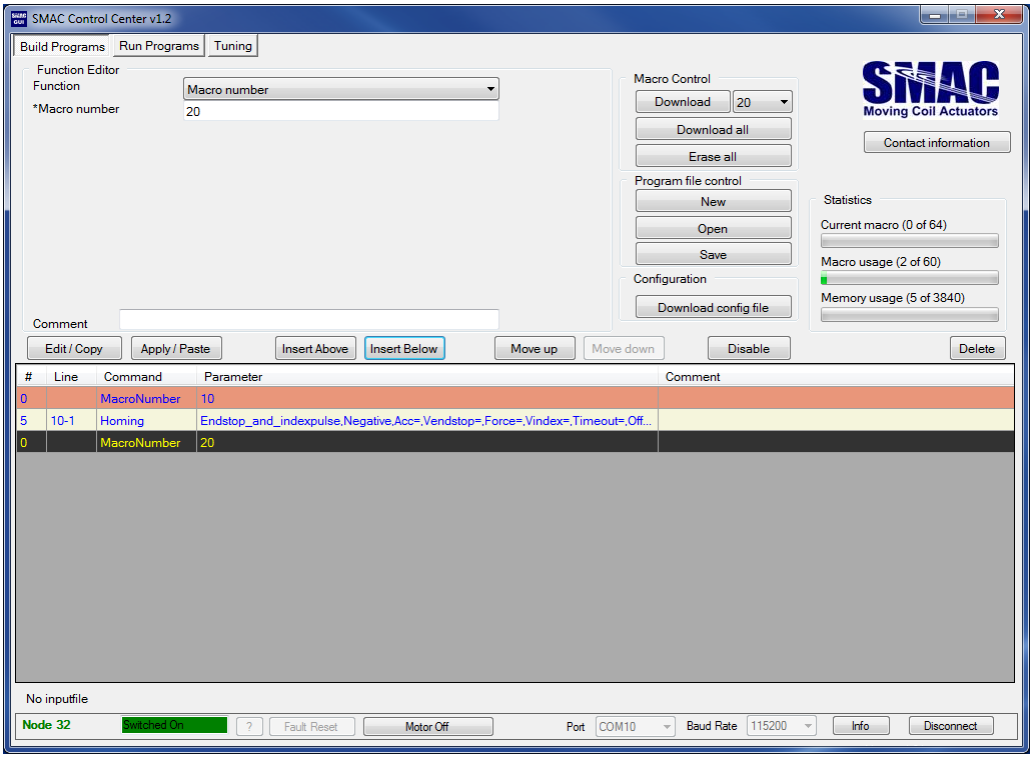
To start the definition for macro number 10 select “Macro number” from the dropdown list in the function editor and fill in 10 for the macro number. Enter a description of what this macro will do in the Comment field and then hit the button “Insert Below”.
The line will now show up in the program listing area.



First command in the macro is a Homing command. Select Homing from the function dropdown and leave the rest of the parameters open as in the screenshot below and then hit “Insert below”

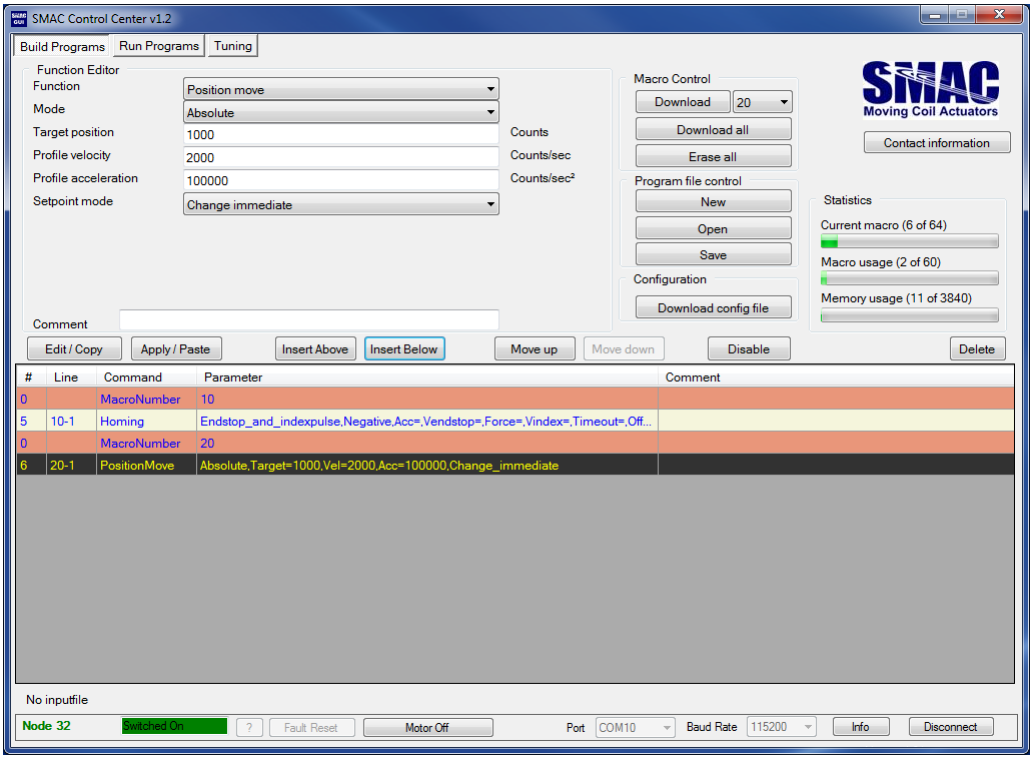


Leaving parameters unfilled means: Take the value that is currently in the controllers working memory. After power up these values are copied from non volatile memory to the controllers working (volatile) memory. Parameters that are mandatory are marked with an * in front of the name. Now define Macro 20 and hit “Insert below”



Add a position move function:

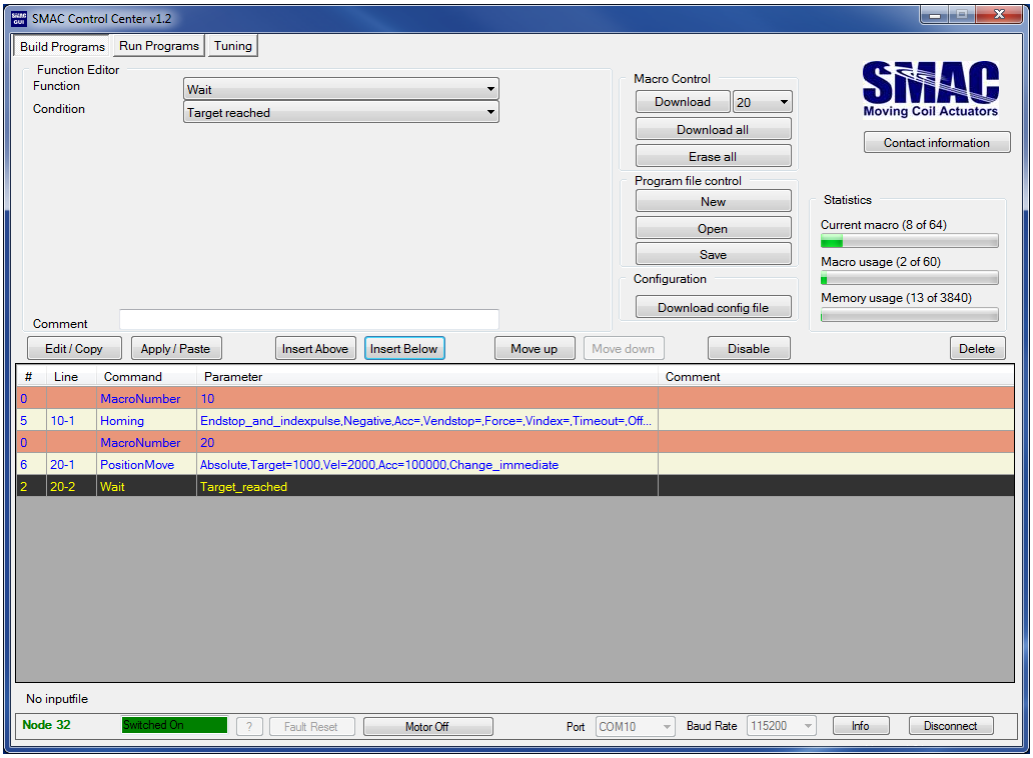
- Select “Position move” from the function list
- Enter the parameters as in the screenshot below
- Hit “Insert below”



Now enter a Wait for target reached instruction:

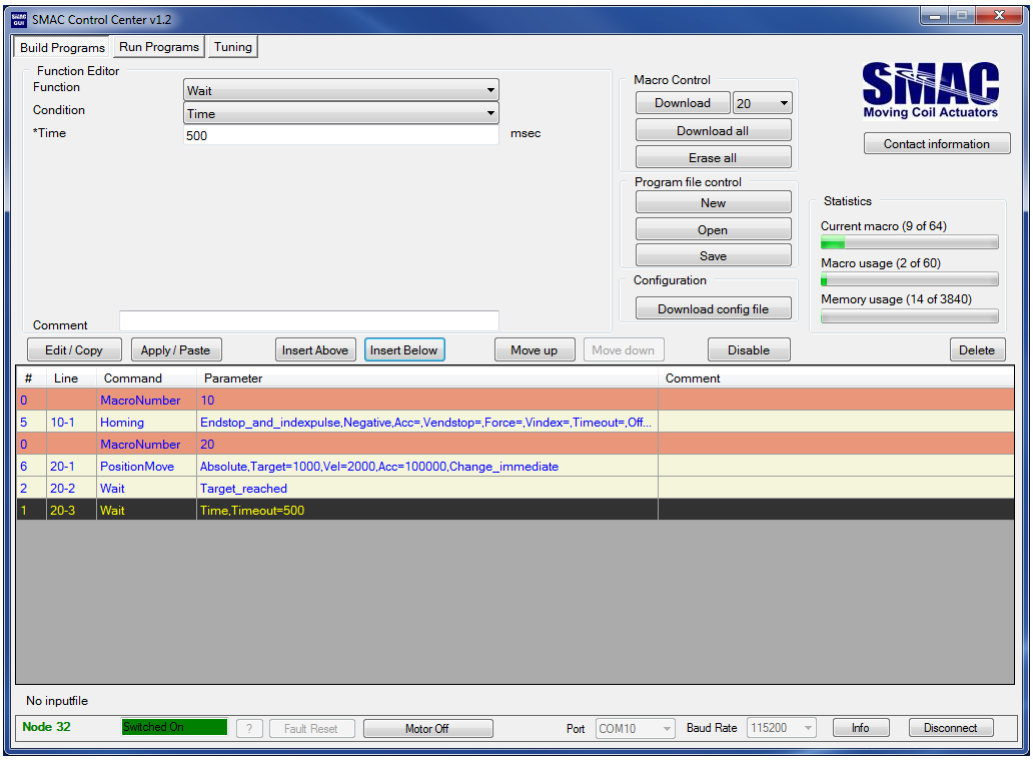
- Select “Wait” from the function list
- Select “Target reached” as condition
- Hit “Insert below”

The program now looks like:

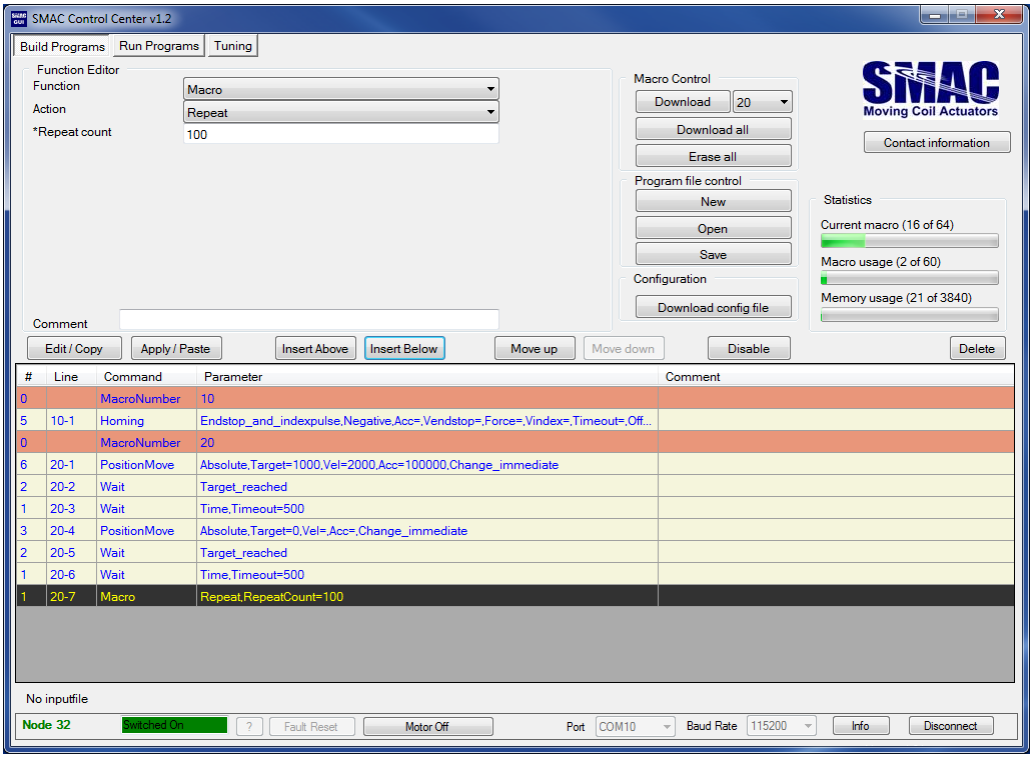


Add another wait instruction that waits 0.5 second:

- Select “Wait” from the function list
- Select “Time” as condition
- Enter 500 as value for the time
- Hit “Insert below”



Now complete the program as in the screenshot below:



Note that the second position move has no values for velocity and acceleration. In this case the parameters are taken from the previous move. Also note that the Repeat count is a mandatory parameter marked with *

Modifying a program

To modify a program use the buttons Edit/Copy, Apply/Paste, Insert Above, Insert below, Move up, Move down and Delete.

First you need to make a program line current (black) by clicking on it.

Then the function of the buttons is:

- Edit/Copy:
  Copy the current program line to the Function editor. This can also be done by double clicking the program line. After making the edits you need to hit Apply/Paste to copy the contents of the Function editor to the current line.
- Apply/Paste:
  Copy the contents of the Function editor to the current line.
- Insert Above:
  The contents of the Function editor is copied into a new program line above the current line.
- Insert Below:
  The contents of the Function editor is copied into a new program line below the current line.
- Move up:
  The current line is moved one line up.
- Move down:
  The current line is moved one line down.

- Disable:
  The current line is not downloaded in the controller and shown in light grey color. After clicking the text on this button changes to Enable.
- Delete:
  This can also be done by hitting the Delete button on the keyboard. The current line is removed from the program.

To copy a program line to another location in the program:

- Double click the program line to copy.
  This will copy the contents of the program line to the Function editor.
- Click the destination program line.
  This will make the destination program line current (black).
  Hit Apply/Paste to overwrite the destination line or Insert above/below to create a new program line.

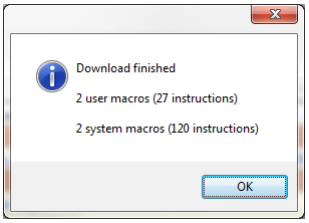
Saving a program

Now the program is complete and can be saved: After clicking "Save" standard save dialogs will help you to select a filename and save location. The default suffix of SMAC Control Center files is .scc. You also get an option to save the file in Composer format (.mlm suffix). These files can be opened and modified with Composer but note that modifications made in composer will not be visible in SMAC Control Center. Use this option only in case functions are needed that cannot be programmed in SMAC Control Center.

After the file has been saved the full pathname of the .scc file will be shown below the program listing area.

Download a program in the controller

To download the created macros in the controller click the "Download all" button.
After a successful download the following message is displayed:



Macros are always stored in the controllers non-volatile memory. The system macros (with the reserved macro numbers 60 .. 63) are also downloaded in the controller when necessary. The system macros hold some commonly used functions.

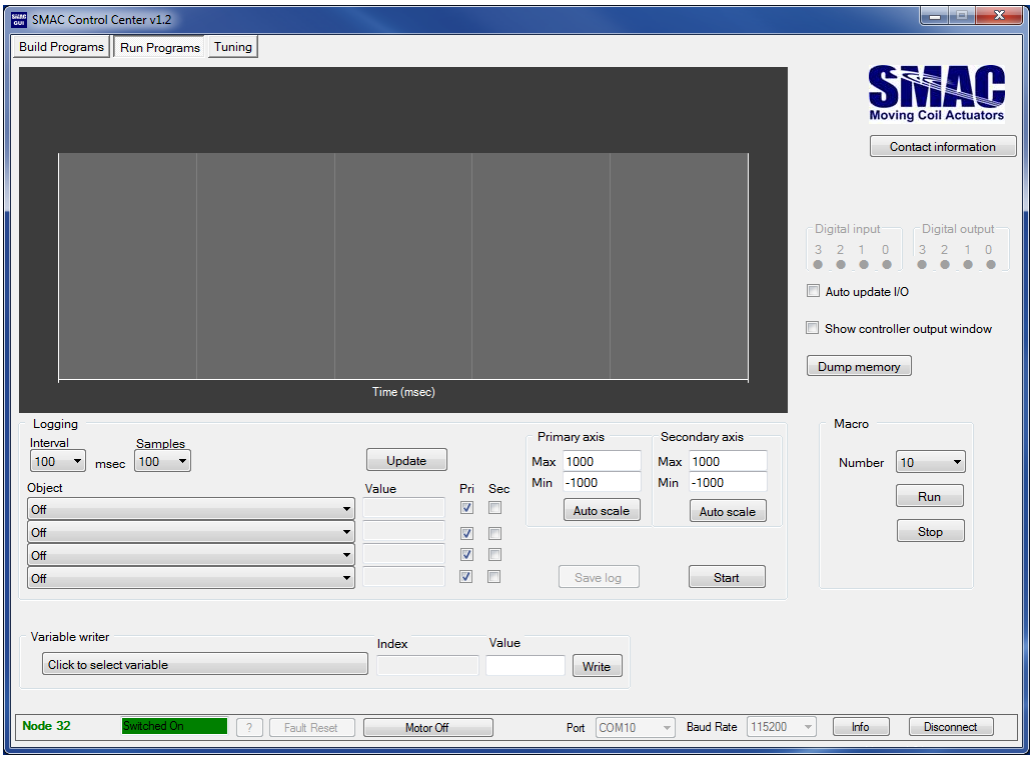
Download configuration file

Now download the configuration file in the controller. Configuration files have a .mlc suffix are created with MotionLab. Standard configuration files are available here: http://www.smac-mca.nl/Configfiles.htm. Advise: always start with a standard config file.

To download a configuration file hit the “Download config file” button and follow the instructions on the screen.

Run the program

Switch to the “Run Programs” tab, select Macro Number 10 from the dropdown and hit the Run button. Assuming that the actuator has been connected in the correct way the actuator now will move to the home position.



Now select Macro number 20 and hit Run. The actuator now will move to position 1000 and back to position 0 a total of 100 times.

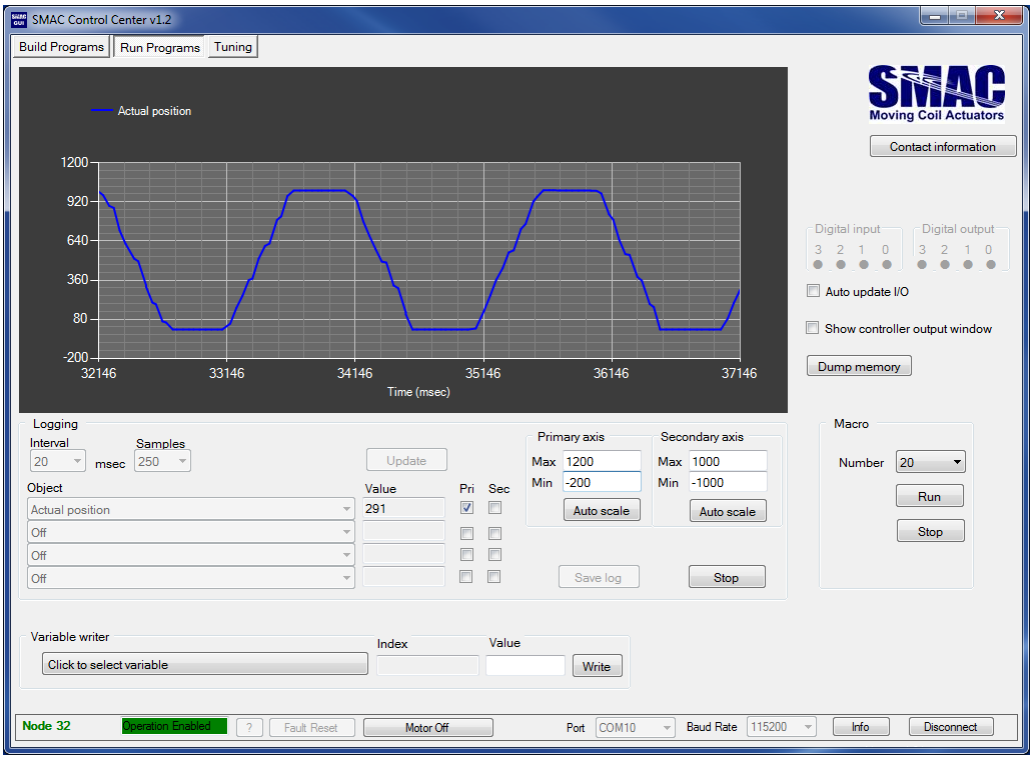
Logging and Charting

It is possible to show up to 4 variables live in a chart and at the same time log these variables in a .csv file.

To start logging:

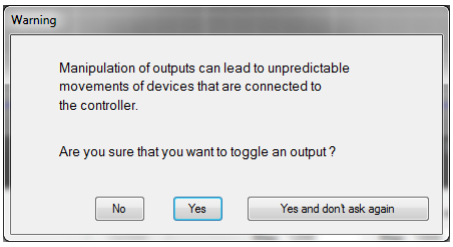
- Select Interval. This is the interval at which variables are requested from the controller
- Select Samples. This is the number of samples that is shown in the chart
- Select at least 1 variable (object) and set a checkmark to select the Yaxis in the chart (there are 2 Y-axes defined: The primary (left) and secondary (right).
- Hit Start: The logging will start. You can manually set the limits for each Y-axis or hit the Auto scale button any time to automatically scale the Y-axis.

Logging of Actual position with the sample program and macro 20 running will look like this:



Digital I/O

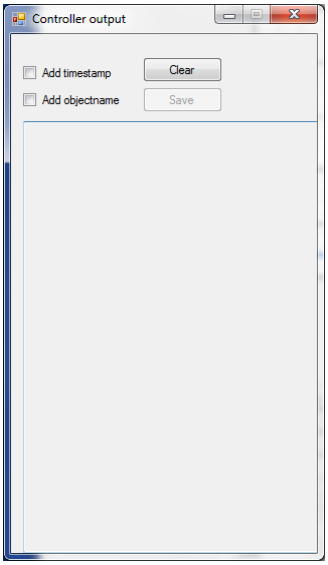
If the option “Auto update I/O” is checked then SMAC Control Center will poll the status of the digital I/O every 100 msec and the status of the inputs and outputs will be shown on simulated green LED’s. You can toggle Digital outputs by clicking on an output LED. After clicking you will get this message:



Make your selection, the “Yes and don’t ask again” option only is valid the rest of this SMAC Control Center session.

Controller output window

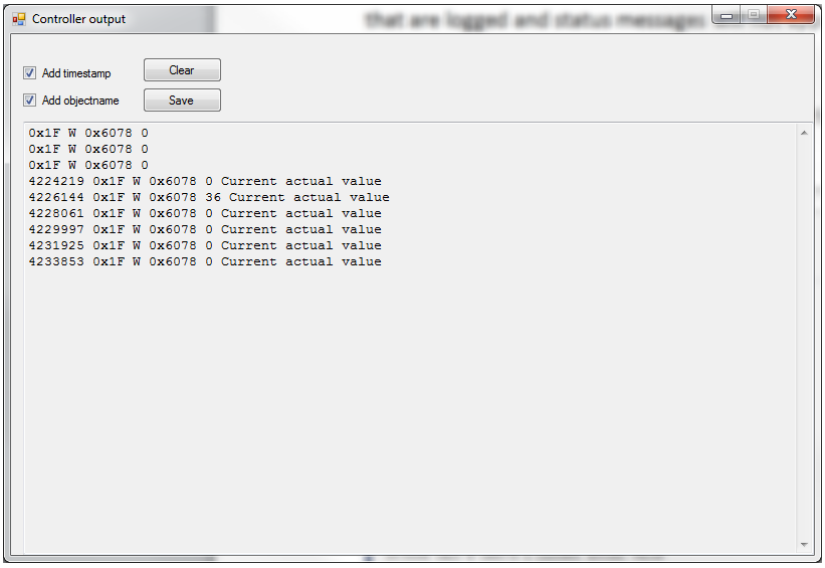
If the “Show controller output window” option is checked then the following window will appear:



This window lists all messages that are received from the controllers that are not requested by SMAC Control Center. In general this will be the response on “Get variable” functions. Note that variables that are logged and status messages will not appear here because these messages are handled by SMAC Control Center.

The window can be expanded to the right.

If we add a “Get variable” function in macro 20 for the actual current at the end of the move then the display looks like this:

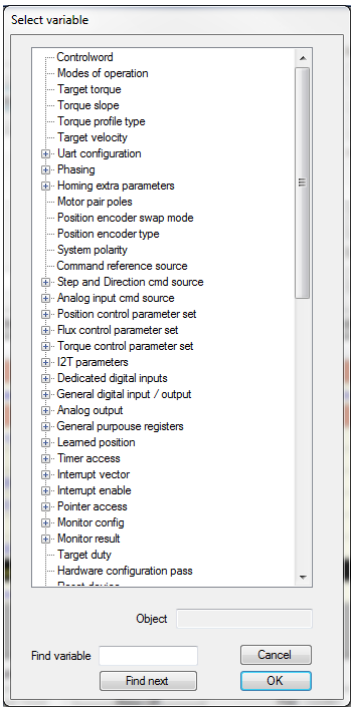


The timestamp is msec since the start of the program.

The window is always active in the background so all unsolicited messages since the start of the program are captured here (also from other nodes).

Variable selector

For some functions (Get and Set variable, If function and selection of log items) it is necessary to select a variable. This is done with a “Select variable” dialog shown below:



All available variables and their meaning are listed in the “Embedded Motion Control Library” Document.
The variable selector makes it easier to find a variable if you know (part of) the name or (part of) the index.
To find the variable Target position:
- Type target in the Find variable box and hit enter, Target torque is highlighted now but that is not the variable we want. Hit Find next (or Enter) until the Target position variable is selected and then click OK.

To find general purpose register W30:
- Type w30 in the Find variable box and hit Enter. The requested is found right away: hit OK.

The text in the Find variable box is not case sensitive. You can also type (part of) a hexadecimal variable address here (with or without a leading 0x).

The “Select variable” dialog is context sensitive. When used in “Set variable” function all writeable variables are shown and in “Get variable” all readable.

Variable writer

The variable writer (in the lower left of the Run programs tab) can be used to directly write a value to any writeable variable in the controller.

First select the variable with the variable selector.
Next type a value in the value field.
Hit Enter or click the Write button to write the value in the controller.

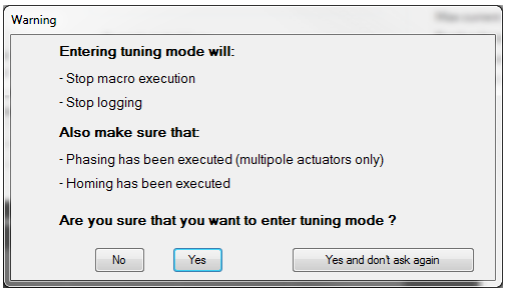
Tuning

The tuning tab can be used to fine tune the Position and Current control loops.

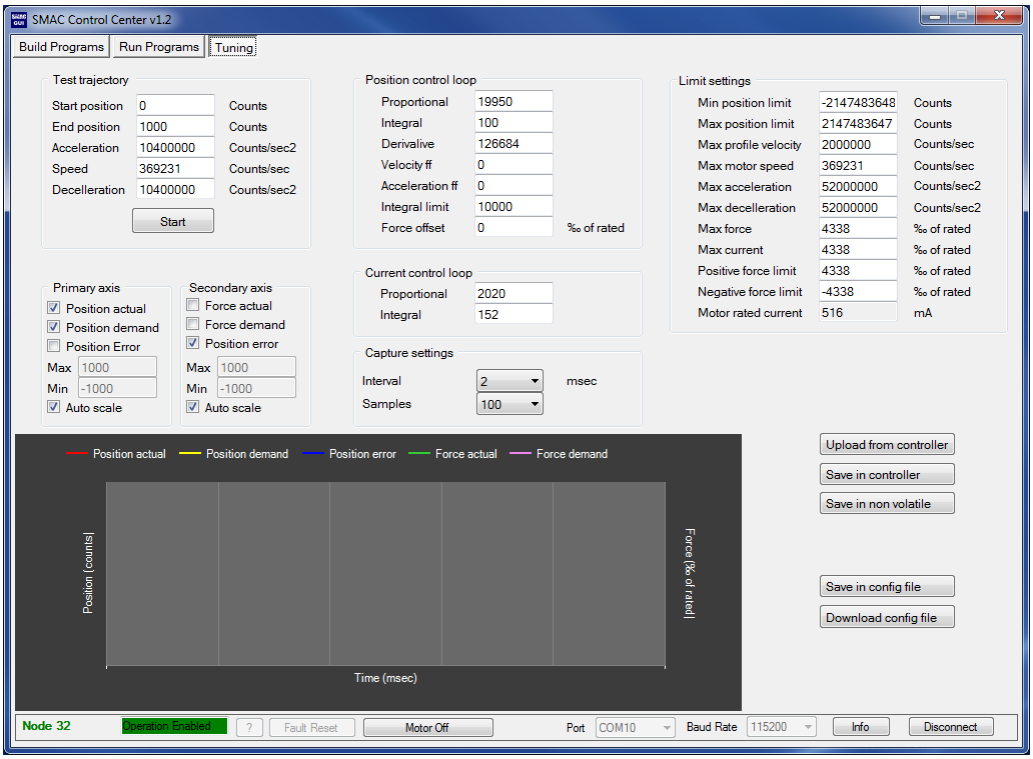
The standard configuration files from www.smac-mca.nl will work fine for an unloaded actuator and give a good starting point for fine tuning.

You will need to fine tune the actuator to compensate for added mass, springs and different mounting orientations.

On first start of the tuning tab the following warning is shown:



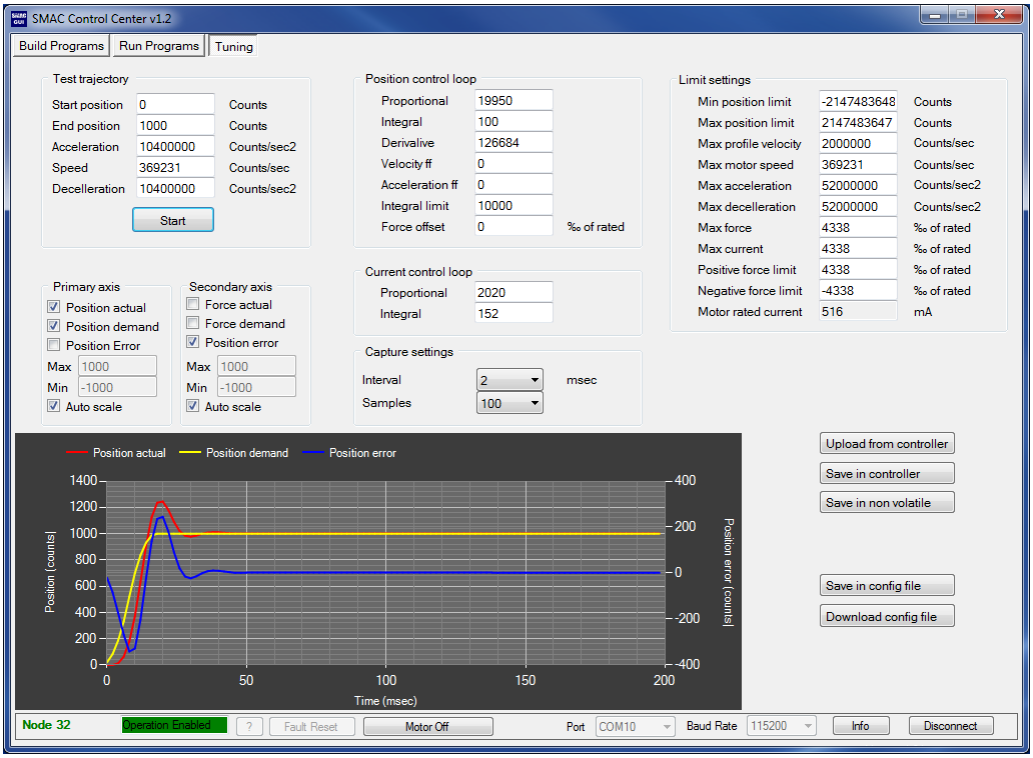
The “Yes and don’t ask again” selection is valid for this session of SMAC Control Center only.



On entering the tuning tab all actual tuning values are read from the controller.

First determine a test trajectory by setting the Start position, End position, Acceleration, Speed and Deceleration. Next hit Start.
After hitting start all tuning variables on this tab will be written into the controllers volatile memory. The actuator will then be moved to the start position with 50% of the Speed, Acceleration and Deceleration, wait 500 msec and then move to the end position at full speed, acceleration and

deceleration. During this last move data is captured in the controller and after the move the result is shown in the chart.



You can adjust the control loop parameters and hit start again to fine tune your application.

Capture settings:

Adjust interval and number of samples to match your application.

Limit settings:

All variables that limit motion are listed here and can be modified also.

Motor rated current is listed here to enable the user to convert the ‰ values into current.

Buttons:

- Upload from controller
  Read all values on the tuning tab from the controller.
- Save in controller:
  Write all values on the tuning tab to the controller (in volatile memory)
- Save in non volatile:
  Write all values on the tuning tab to the controller (in non volatile memory)
- Save in config file:
  To create a partial configuration file or to merge the current tuning configuration with an existing configuration file.
- Download config file:
  Writes an existing configuration file in the controllers non volatile memory. (same as on the Build programs tab)