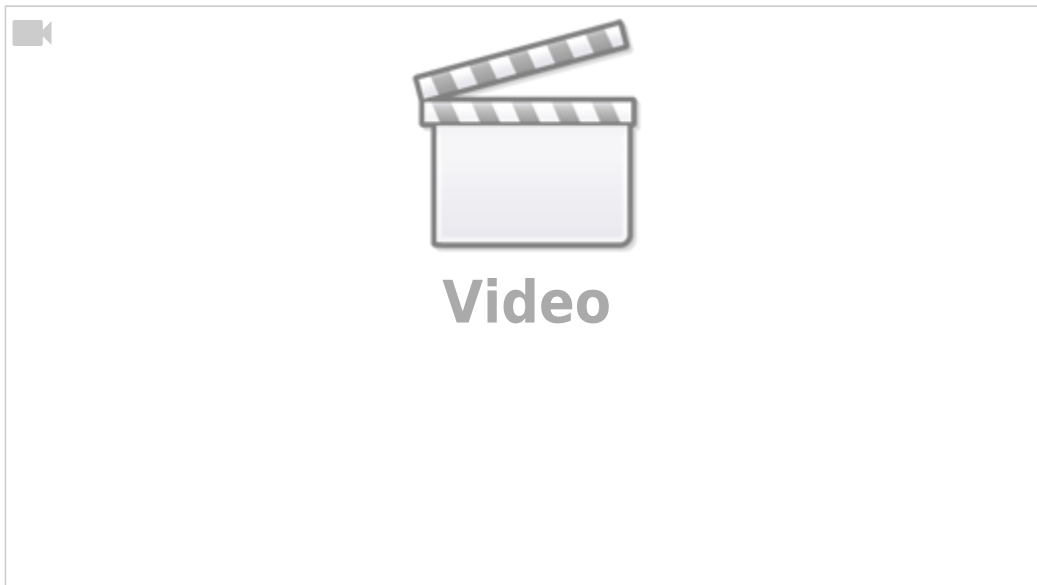
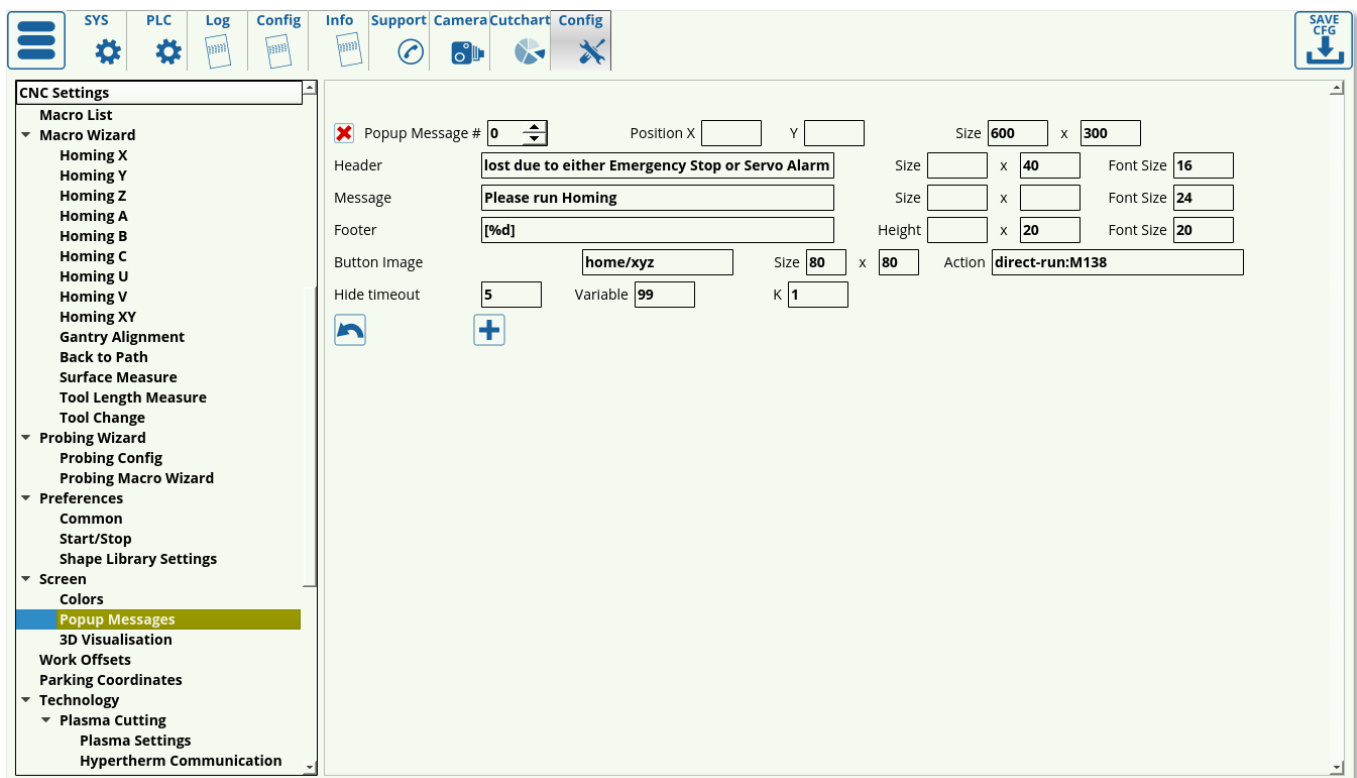


Popup messages



This is an expanded manual that serves to introduce the user to the concept of popup messages beyond the basics described in the [MyCNC Configuration Dialogs](#) manual.



The following settings are available:

- Popup Message Number
- Position (X&Y) on the screen
- Size of the popup window (in pixels)
- Header (title) of the popup message
- Header size and font size
- Message (body) of the popup window which allows to present additional information to the user

- Message size and font size
- Footer of the popup message
- Footer size and font size
- Button image (specify the file location)
- Button size
- Button action to run a specific macro when the button is clicked
- Timeout to hide the popup window (in seconds)
- Variable number to display (usually in footer, through [%d]) - useful to display a changing variable, like time in seconds, etc.
- Coefficient K to multiply the variable by. Useful to convert tiny incremental changes (such as fractions of a second) or large rapidly changing numbers into numbers that can be easily read by the user.

Popup messages are controlled by writing a **1** or a **0** into global variables #9100-9163 (1 to bring up the popup, 0 to close it). Therefore, the user can set up to 64 possible popup messages within the myCNC software by going into Settings > Config > Screen > Popup Messages and assigning the necessary messages there. Afterwards, the message can be brought up either through a PLC procedure, like so:

```
gvarset(9160,1);
```

or by using a G-code command (for example, within a macro) such as

```
G10 L80 P9160 Q1
```

which will write a 1 in the global variable #9160.

Examples of popup messages implementation

Mandatory homing procedure

The below section is an excerpt from the following manual: [How to add mandatory Homing after Emergency Button and-or Servo ready alarm](#)

Please consult the full manual for additional information on mandatory homing setup.

Mandatory Homing procedure handler can be implemented with **Software PLC**:

The screenshot shows the CNC configuration software interface. The top menu bar includes SYS, PLC, Log, Stat, Info, Support, and Config. The left sidebar shows the CNC Settings tree, with Software PLC selected. The main area displays the PLC Sources list, including BV17, HANDLER_EXIT, HANDLER_GCODE_START, HANDLER_GCODE_STOP, HANDLER_INIT, HANDLER_SERVO_OFF, HANDLER_SERVO_ON, CHARGE_PUMP, HOMING_HANDLER (highlighted), LEDS, MESSAGE_BOX, MODBUS_TEST, OIL_CHANGE_COUNTER, and POPUP_HANDLER. The right pane shows the main() function of the HOMING_HANDLER program, which starts with exit(99); and contains logic for monitoring axes flags X, Y, Z, and C, calculating a homing flag, and setting status variables.

```

Name: HOMING_HANDLER  Aliases:
main()
{
  exit(99);

  a=0;
  do{

    hx=gvarget(7391); //monitor axes flags X, Y, Z and C
    hy=gvarget(7392);
    hz=gvarget(7393);
    //hc=gvarget(7396);

    a++; if(a>9){a=1;};

    b=hx+hy*10+hz*100;
    //b=hx+hy*10+hz*100+hc*1000;
    c=a*10000000;
    gvarset(99,c+b); //build variable to display which axis is not ready

    home_old=home;
    home=hx+hy+hz; //check if any of axis is not ready

    _HANDLER_INIT: OK
    _HANDLER_SERVO_OFF: OK
    _HANDLER_SERVO_ON: OK
    CHARGE_PUMP: OK
    HOMING_HANDLER: OK
    LEDS: OK
    MESSAGE_BOX: OK
    MODBUS_TEST: OK
    OIL_CHANGE_COUNTER: OK
    POPUP_HANDLER: OK
  }
}

```

NOTE: The homing procedure handler is OFF by default as it has the exit(99); line in the beginning of the program which immediately terminates the PLC as soon as it is started. In order to enable the homing procedure handler, add two forward slashes in front of the exit(99); line to comment it out so that the PLC can proceed uninterrupted:

This screenshot is identical to the previous one, but with a red annotation. A red box highlights the line `// exit(99);` in the main() function, and a red text box to its right says "Add '//' to enable homing handler".

There are Homing flags situated in global variables 7391-7399

Address	Global variable	Description
7391	X axis homing flag	
7392	Y axis homing flag	

Address	Global variable	Description
7393	Z axis homing flag	
7394	A axis homing flag	
7395	B axis homing flag	
7396	C axis homing flag	

The flags are set automatically to “1” if

1. The myCNC software has just loaded
2. Emergency button has been pressed
3. Corresponding Servo Ready alarm has been triggered

Software PLC Homing procedure handler monitors the flags and stops running G-code if any of the flags is set to 1.

[HOMING_HANDLER.plc](#)

```

main()
{

    //exit(99);

    a=0;
    do{

        hx=gvarget(7391); //monitor axes flags X, Y, Z and C
        hy=gvarget(7392);
        hz=gvarget(7393);
        //hc=gvarget(7396);

        a++; if(a>9){a=1;};

        b=hx+hy*10+hz*100;
        //b=hx+hy*10+hz*100+hc*1000;
        c=a*10000000;
        gvarset(99,c+b); //build variable to display which axis is not ready

        home_old=home;
        home=hx+hy+hz; //check if any of axis is not ready
        //home=hx+hy+hz+hc; //check if any of axis is not ready

        if (home!=0) //if any of axis is not ready, then ...
        {
            prg=gvarget(6065);
            if (prg!=0) //Check if Program running is started and Stop it
            immediately
            {
                gvarset(0xffffffff,1); //Stop Program
            };

            gvarset(9100,1); //display the message #1 on the screen, if any of

```

```

home alarm activated
  gvarset(8160,2); //set XHC Homing display

}else
{
  gvarset(9100,0); //hide the message #1, if everything's ok
  if (home_old!=home) //just home
  {
    gvarset(8160,0); //set XHC Homing display
  };
};

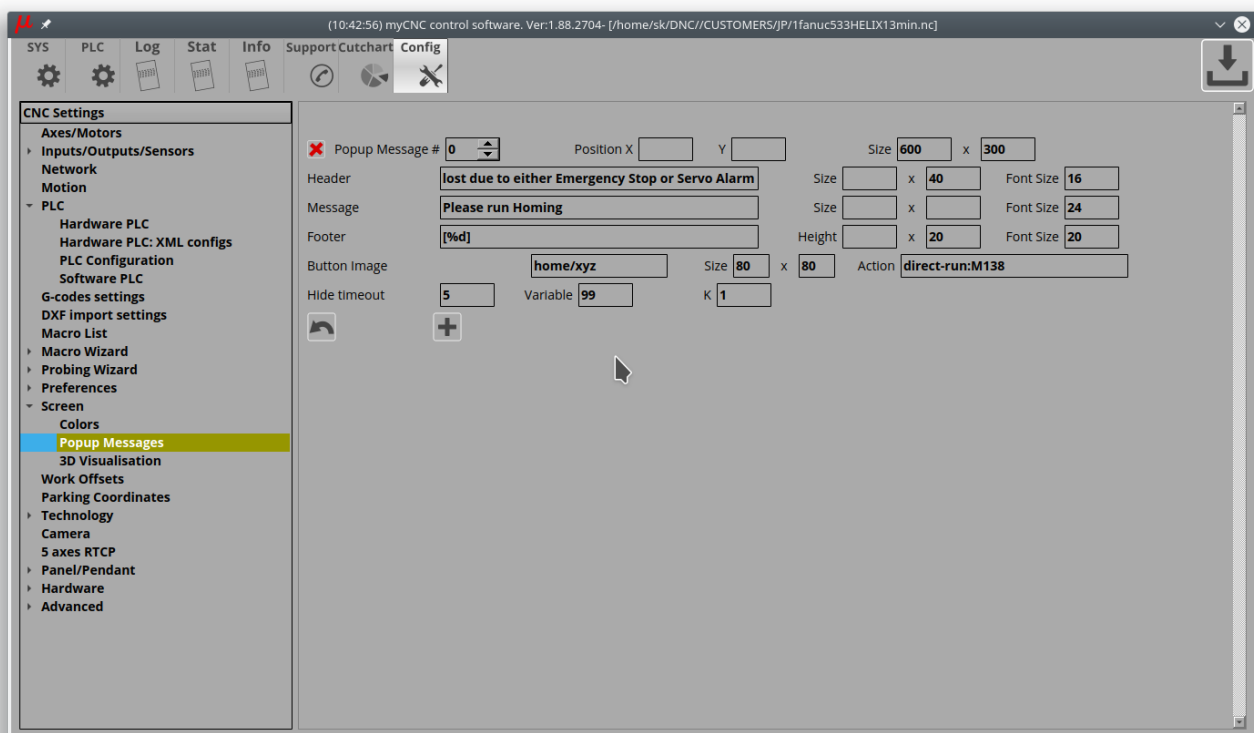
}while(1);

exit(99);
};

```

Writing a “1” to global variable #9100 will show **Popup Message #0**. Content to show in the Popup message should be programmed in the cnc-variables.xml configuration file.

myCNC software has configuration widget to add and setup a Popup Message Widget.



- Press “+” button to add new Popup Message description
- **Popup message #** - ID number of popup message to show/hide. Global Variable numbers 9100...9200 are used to show/hide the popup messages.
For example, variable number **9114** is used to show Popup message with ID **14**
- **Position X,Y** - used to define Popup widget position on the screen. Popup message will be shown in the center, if Position x,y are not defined.

- **Size** - defines size of Popup message widget
- **Header, Message, Footer** - Popup message widget contains 3 sections to show a message. You can define text for each section (Header, Message, Footer). Section Size (width, height) and Font size can be defined for each section as well.
- **Button Image, Size, Action** - Popup message may have a button at the bottom of the popup widget. A button image file, button size and action should be defined to get the button appeared in the widget.
- **Hide timeout** - (under developing) Popup message may disappear automatically after given timeout
- **Variable** - One of text lines (header, message or footer) may contain C-style format section to print variable value. If this section is present, then a variable, defined here, is printed. For example, Variable #99 will be printed in "Footer" section according to the screenshot above.
- **K** - ratio to print the variable value. For example, if "K=0.001" and Variable value is "5000", the value "5" will be printed ($5000 \cdot 0.001 = 5$)

Popup message example is shown below



M660

M660 PLC procedure is supposed to be a handler for manual tool change. It shows a "Manual Tool Change" message, then waits in a loop till tool changed. There can be a software flag (global variable register) indicating tool changed or the procedure can wait till a hardware button (connected to the controller input) is pressed. Below is an example of a manual tool change handler M660.plc:

```
main()
{
    gvarset(1999,1); //set flag
    timer=0;
```

```

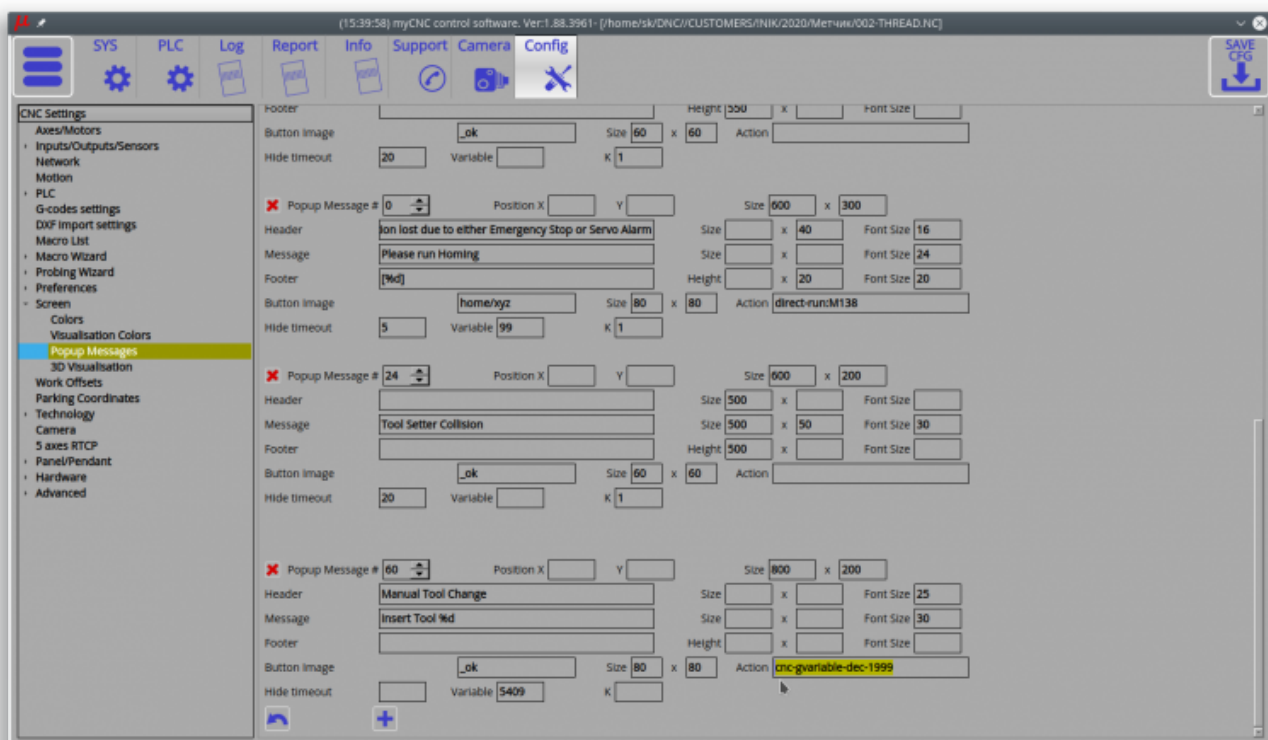
flag=1;
do{
  timer++;
  if ((timer&0xff)==0) //check every 0.25 sec
  {
    gvarset(9160,1); //show the Manual Tool Change Message #60
    flag=gvarget(1999); //check the flag, if flag<=0, then tool changed
    and a job should be resumed
  };

}while(flag>0);

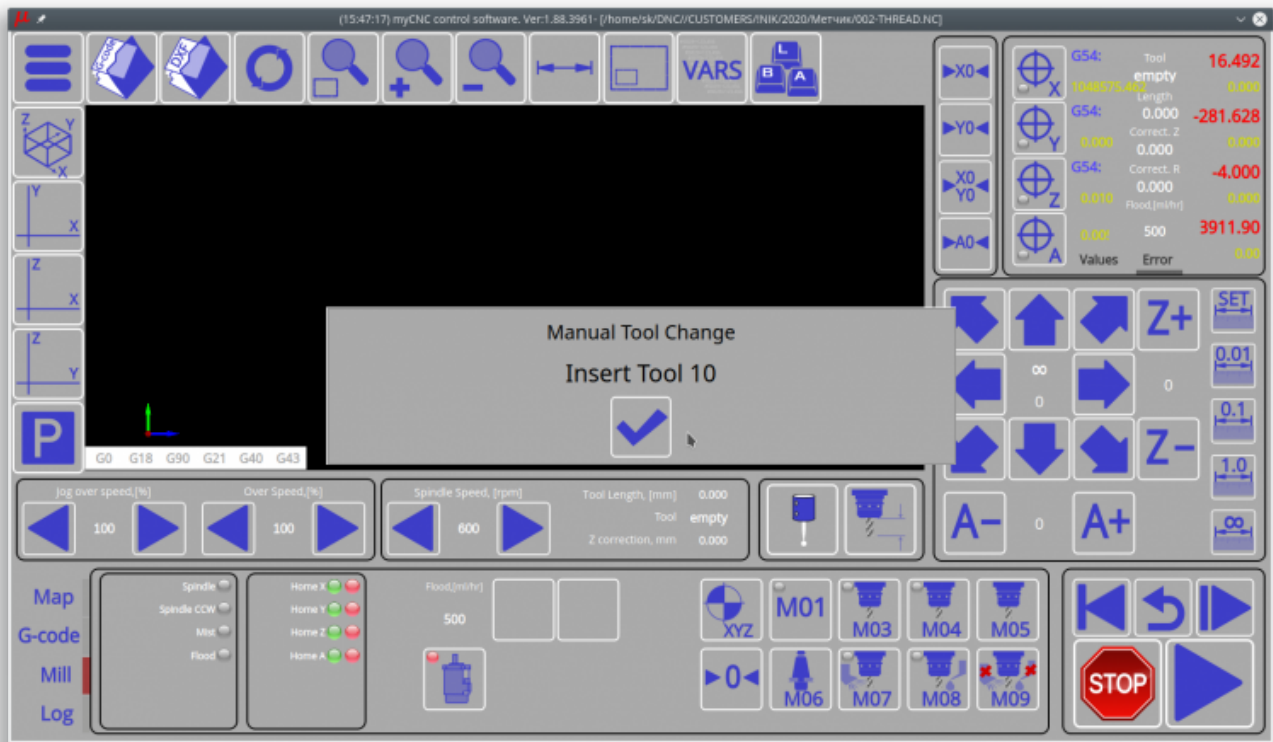
gvarset(9160,0); //clear the Message
exit(99);
};

```

Setup for message #60 in Settings > Config > Screen > Popup Messages:



Resulting popup message:



M604

```
#include pins.h
main()
{
    //clamp new tool
    portclr(OUTPUT_TOOL_CLAMP);

    timer=300;do{timer--;}while(timer>0);

    timer=2000;
    do
    {
        timer--;
        t=portget(INPUT_TOOL_CLAMPED);// 5
        if (t!=0)
        {
            exit(99);
        };
    }while(timer>0);

    gvarset(9124,1);
    timer=20;do{timer--;}while(timer>0);

    message=PLCCMD_MOTION_BREAK;
    timer=20;do{timer--;}while(timer>0);
```

```
exit(99);  
};
```

From:
<http://cnc42.com/> - **myCNC Online Documentation**

Permanent link:
http://cnc42.com/mycnc/popup_messages

Last update: **2020/11/19 11:59**

