

PLC Examples

NOTE: The myCNC team recommends utilizing the examples provided in this manual (as well as other manuals in this documentation) as a starting point for your machine setup. When possible (and applicable), it is recommended to keep changes to a minimum. In general, using these examples as the basis for your PLCs/macro commands allows for an easier setup process.

Spindle Speed control for DAC

SPN.plc

```
#define command    var00
#define parameter  var01
//set Spindle speed control via DAC channel #1
//Spindle Speed is given in **eparam** register

main()
{
    value=eparam;
    if (value>0xFFFF) {value=0xFFFF;}; //fix if given value is out of range
    0...0xffff
    if (value<0) {value=0;};
    dac01=val;                                //setup DAC value

    /**Set Spindle Speed** is asynchronous operation.
    //It's better to inform myCNC Software New Spindle Speed applied.
    //Send information about new Spindle Speed to myCNC Software
    message=PLCCMD_REPLY_TO_MYCNC;
    //Command code to send to myCNC software
    command=PLC_MESSAGE_SPINDLE_SPEED_CHANGED;
    //Message code
    parameter=eparam; //New Spindle Speed information
    timeout=timer+10; do {timer++;}while(timer<timeout);
    //Delay to push the Message to myCNC Software

    gvarset(7371,eparam);
    //myCNC register #7371 contains actual Spindle Speed.
    //Another way to inform myCNC software about new Spindle Speed
    //(to display on it DRO for example)

    exit(99); //normal exit.
};
```

Spindle Speed control for ET10_DAC Examples

SPN.plc

```
#define command      var00
#define parameter    var01
//set Spindle speed control via ET10 DAC channel #1
//Spindle Speed is given in **eparam** register

main()
{

    command=0x32; //EXT_ET10_DAC_OFFSET; set ADC offset register address
    parameter=0x800-(eparam/2)+(1<<12);
    //0x800 - is the middle of 12bits range - represents 0V
    //Eparam contains 12bits DAC value in 10V range,
    //ET10 DAC setup in +10V...-10V range, so need half range (/2)
    //Encoder channel number is given in high 12 bits of 16bit word -
    //Channel #1 is (1<<12).

    message=PLCCMD_SET_CNC_EXTVAR;
    //setup Message register with command for access to [[External CNC
    Variables]]
    texit=timer+2;do{timer++;}while(timer<texit);
    //2ms delay to push the command from PLC to myCNC Core

    /**Set Spindle Speed** is an asynchronous operation.
    //It's good to inform myCNC Software New Spindle Speed applied.
    //Send information about new Spindle Speed to myCNC Software
    //myCNC register #7371 contains actual Spindle Speed.

    gvarset(7371,eparam);
    timer=30; do { timer--; } while (timer>0); //Delay to push the
    Message to myCNC Software
    //A way to inform myCNC software about new Spindle Speed (to display
    on it DRO for example)

    exit(99); //normal exit.
};
```

Spindle Speed control Example for ET15 DAC

SPN.plc

```
#include vars.h

#define ADC_CHANNEL 1
```

```
main()
{
    speed=eparam/2;

    command=0x32; //set ADC offset register address
    parameter=0x800+(speed)+(ADC_CHANNEL<<12);
    message=PLCCMD_SET_CNC_EXTVAR;    timer=2;do{timer--;}while(timer>0);

    gvarset(7371,eparam);    timer=20;do{timer--;}while(timer>0);

    exit(99); //normal exit.
};
```

ET15 Spindle control examples

Spindle Speed control Example for ET15 DAC

SPN.plc

```
#include vars.h

#define ADC_CHANNEL 1

main()
{
    speed=eparam/2;

    command=0x32; //set ADC offset register address
    parameter=0x800+(speed)+(ADC_CHANNEL<<12);
    message=PLCCMD_SET_CNC_EXTVAR;    timer=2;do{timer--;}while(timer>0);

    gvarset(7371,eparam);    timer=20;do{timer--;}while(timer>0);

    exit(99); //normal exit.
};
```

M03 - Spindle ON Example for ET15 DAC

M03.plc

```
//Turn on Spindle clockwise
#include pins.h
#include vars.h
```

```
#define ADC_CHANNEL 1

main()
{
    timer=0;
    proc=plc_proc_spindle;

    speed=eparam/2;

    command=0x32; //set ADC offset register address
    parameter=0x800+(speed)+(ADC_CHANNEL<<12);
    message=PLCCMD_SET_CNC_EXTVAR;    timer=2;do{timer--;}while(timer>0);

    portset(OUTPUT_SPINDLE);

    gvarset(7370,1); timer=30;do{timer--;}while (timer>0); //Spindle
State
    gvarset(7371,eparam); timer=30;do{timer--;}while (timer>0); //Spindle
Speed Mirror register

    //delay after spindle started
    timer=spindle_on_delay; do{timer--;}while (timer>0); //delay for
Spindle reach given speed

    exit(99);    //normal exit
};
```

M05 - Spindle OFF Example for ET15

M05.plc

```
//Turn on Spindle clockwise
#include pins.h
#include vars.h

#define ADC_CHANNEL 1

main()
{

    portclr(OUTPUT_SPINDLE);
    portclr(OUTPUT_CCW_SPINDLE);

    proc=plc_proc_idle;

    speed=0;

    command=0x32; //set ADC offset register address
```

```
parameter=0x800+(speed)+(ADC_CHANNEL<<12);  
message=PLCCMD_SET_CNC_EXTVAR;    timer=2;do{timer--;}while(timer>0);  
  
gvarset(7370,0);  timer=30;do{timer--;}while(timer>0); //Spindle  
State  
gvarset(7371,0);  timer=30;do{timer--;}while(timer>0); //Spindle  
Speed  
  
timer=spindle_off_delay;  
do { timer--; } while (timer>0);  
  
exit(99);  //normal exit  
};
```

From:

<http://docs.pv-automation.com/> - **myCNC Online Documentation**

Permanent link:

http://docs.pv-automation.com/plc_examples/m03_et10_dac

Last update: **2022/03/29 15:54**

