

SilverPakT Integrated Controller/Driver and Motor



Sample Code

Version 1.02

Lin Engineering
1990 Russell Ave, Santa Clara, CA 95054
(408) 919-0200

www.linengineering.com

sales@linengineering.com

Sample code

General Parameters

```
//TMCL Sample Setup Program
StallValue = 0      // StallGuard Threshold (0 = off)
Velocity = 2047    // Max Velocity
Acceleration = 1100 // Max Acceleration
ustep = 4          // 4 = 16X u-stepping
pdiv = 3           // pdiv @ default
rdiv = 8           // rdiv
current = 600      // 600 mA Peak
Hcurrent = 300     // 300 mA Peak
decay = 0          // 2048 = on, 0 = off

//Initialization of above parameters
SAP 6, 0, current    //Setting Current
SAP 203, 0, decay    //Switch Mixed Decay
SAP 5, 0, Acceleration //Setting max acceleration
SAP 205, 0, StallValue //Setting StallGuard Threshold
SAP 140, 0, ustep    //Setting u-stepping
SAP 143, 0, Hcurrent //Max Hold Current
SAP 154, 0, pdiv     //Initializing pdiv
SAP 4, 0, Velocity   //Setting maximum Velocity
SAP 138, 0, rdiv     //Ramp Divisor
ROR 0, 1049          //Rotate Right Continuously
WAIT TICKS, 0, 1000 //Wait for 1000 msec before issuing next command
MST 0                //Stop Motor # 0
STOP                 //End program
```

Potentiometer Control #1 – Run continuously while varying the speed via Pot

```
//TMCL POT Control for Silverpak 17T. This will move step motor
//up to top speed when pot is turned to maximum, when pot is 0, motor is 0 speed

//Parameters
StallValue = 0      //StallGuard Threshold (0 = off)
Velocity = 2047    //Max Velocity
Acceleration = 1000 //Max Acceleration
ustep = 1          //4 = 16X u-stepping
pdiv = 8           //pdiv @ default
rdiv = 8           //rdiv
current = 600      //600 mA Peak
Hcurrent = 300     //300 mA Peak
decay = 2048       //2048 = on, 0 = off
speed = 500        //Rotation Speed of ROR Command

//Connect pot: 1 side to +5V output, pin 3. Other side to GND, pin 12 or 13,
//and the middle wire to the input line, pin 14. Pot resistance acts as a voltage divider

//Initialization of above parameters
SAP 6, 0, current    //Setting Current
SAP 203, 0, decay    //Switch Mixed Decay
```

```

SAP 5, 0, Acceleration //Setting max acceleration
SAP 205, 0, StallValue //Setting StallGuard Threshold
SAP 140, 0, ustep //Setting u-stepping
SAP 143, 0, Hcurrent //Max Hold Current
SAP 153, 0, rdiv //Initializing rdiv
SAP 154, 0, pdiv //Initializing pdiv
SAP 4, 0, Velocity //Setting maximum Velocity
SAP 138, 0, 2 //Ramp Mode
main:
GIO 0, 1 //read pot
CALC MUL, 17 //Run calculations to make it a valid number
CALC DIV, 10
WAIT TICKS, 0, 5 //Small Delay
AAP 132, 0 //transfers result to target position
JA main //Infinite Loop
STOP //End of Program

```

Potentiometer Control #2 – CW and CCW continuous rotation, and stop motion

```

//TMCL POT Control for Silverpak 17T. This will move step motor
// when pot is 0, motor is rotating CW, when pot is half way, motor is stopped,
// when pot is max, motor is rotating CCW (depending on connection of motor)

```

```

StallValue = 0 //StallGuard Threshold (0 = off)
Velocity = 2047 //Max Velocity
Acceleration = 1000 //Max Acceleration
ustep = 1 //4 = 16X u-stepping
pdiv = 8 //pdiv @ default
rdiv = 8 //rdiv
current = 600 //600 mA Peak
Hcurrent = 300 //300 mA Peak
decay = 2048 //2048 = on, 0 = off

```

```

//Connect pot: 1 side to +5V output, pin 3. Other side to GND, pin 12 or 13,
//and the middle wire to the input line, pin 14. Pot resistance acts as a voltage divider

```

```

//Initialization of above parameters

```

```

SAP 6, 0, current //Setting Current
SAP 203, 0, decay //Switch Mixed Decay
SAP 5, 0, Acceleration //Setting max acceleration
SAP 205, 0, StallValue //Setting StallGuard Threshold
SAP 140, 0, ustep //Setting u-stepping
SAP 143, 0, Hcurrent //Max Hold Current
SAP 154, 0, pdiv //Initializing pdiv
SAP 4, 0, Velocity //Setting maximum Velocity
LOOP:
GIO 0, 1 // get status of analog input
COMP 0 // compare to value 0
JC EQ, GORIGHT // rotate right if equal
COMP 1015 // compare to value 1015 (1023 is highest)
JC GE, GOLEFT // rotate left if greater or equal to 1015
JC LT, STOPMOTOR // stop motor if less than 1015
JA LOOP // loop continuously
STOP
GORIGHT: //Rotate right if pot is at min
ROR 0, 1000
GAP 1, 0
COMP 1000

```

```
JC GE, STOPMOTOR
JA LOOP
```

```
GOLEFT:                //Rotate left if pot is at max
  ROL 0, 1000
  GAP 1, 0
  COMP 0
  JC LE, STOPMOTOR
  JA LOOP
```

```
STOPMOTOR:            //Stop rotation if pot is in the middle
  MST 0
  JA LOOP
```

Potentiometer Control #3 – Position control

```
//TMCL POT Control for Silverpak 17T. This will move motor 2400 steps if pot is 0,
// and move back to position 0 (back by 10000 steps) if pot is max. It will not
// rotate if pot is in middle
```

```
//Parameters
StallValue = 0          //StallGuard Threshold (0 = off)
Velocity = 2047         //Max Velocity
Acceleration = 1000    //Max Acceleration
ustep = 1              //4 = 16X u-stepping
pdiv = 8               //pdiv @ default
rdiv = 8               //rdiv
current = 600          //600 mA Peak
Hcurrent = 300         //300 mA Peak
decay = 2048           //2048 = on, 0 = off
```

```
//Initialization of above parameters
SAP 6, 0, current      //Setting Current
SAP 203, 0, decay      //Switch Mixed Decay
SAP 5, 0, Acceleration //Setting max acceleration
SAP 205, 0, StallValue //Setting StallGuard Threshold
SAP 140, 0, ustep      //Setting u-stepping
SAP 143, 0, Hcurrent   //Max Hold Current
SAP 154, 0, pdiv       //Initializing pdiv
SAP 4, 0, Velocity     //Setting maximum Velocity
```

```
//Connect pot: 1 side to +5V output, pin 3. Other side to GND, pin 12 or 13,
//and the middle wire to the input line, pin 14. Pot resistance acts as a voltage divider
```

```
LOOP: GIO 0, 1         // get status of analog input
  COMP 0               // compare to value 0
  JC EQ, CLOSE_VALVE  // rotate right if equal
  COMP 500             // compare to value 500
  JC GE, OPEN_VALVE   // rotate left if greater or equal to 1015
  JC LT, STOPMOTOR    // stop motor if less than 1015
  JA LOOP             // loop continuously
  STOP
```

```
OPEN_VALVE:
  MVP ABS, 0, 10000
  JA LOOP
```

```
CLOSE_VALVE:
  MVP ABS, 0, 0
  JA LOOP
```

```
STOPMOTOR:
```

MST 0
JA LOOP

Left and Right Limit Switches

//TMCL Sample Setup Program. If no limit switches are tied to ground, no motion will occur.
//If the left limit switch is tied to ground, it will rotate left until the left limit switch goes high.
//If the right limit switch is tied to ground, it will rotate right until it goes high.

```
//Parameters
StallValue = 0           //StallGuard Threshold (0 = off)
Velocity = 2047         //Max Velocity
Acceleration = 1000     //Max Acceleration
ustep = 1              //4 = 16X u-stepping
pdiv = 8               //pdiv @ default
rdiv = 8               //rdiv
current = 600          //600 mA Peak
Hcurrent = 300         //300 mA Peak
decay = 2048           //2048 = on, 0 = off
```

//Connect a 3 position switch such that the Left Limit switch (Pin 11) is connected to one side
//of the switch, the Right Limit switch (Pin 4) is connect to the other side of the switch,
//and Ground (Pin 12) is in the middle position of the switch.

```
//Initialization of above parameters
SAP 24, 0, decel        //Set deceleration
SAP 6, 0, current       //Setting Current
SAP 203, 0, decay       //Switch Mixed Decay
SAP 5, 0, Acceleration  //Setting max acceleration
SAP 205, 0, StallValue  //Setting StallGuard Threshold
SAP 140, 0, ustep       //Setting u-stepping
SAP 143, 0, Hcurrent    //Max Hold Current
SAP 154, 0, pdiv        //Initializing pdiv
SAP 4, 0, Velocity      //Setting maximum Velocity
SAP 12, 0, 1           //activate right switch
SAP 13, 0, 1           //activate left switch
```

```
Loop:
CSUB CheckLeftLim      //Call subroutine
CSUB CheckRightLim    //Call subroutine
JA Loop
STOP

CheckLeftLim:
GAP 11, 0              //Get status of Left Limit switch
COMP 0
JC EQ, GoMotorLeft    //If equal to zero, rotate
COMP 1
JC EQ, StopMotor      //If equal to one, stop motor
RSUB

CheckRightLim:
GAP 10, 0
COMP 0
JC EQ, GoMotorRight   //If equal to zero, rotate
COMP 1
JC EQ, StopMotor      //If equal to one, stop motor
RSUB

StopMotor:
MST 0
RSUB
```

```

GoMotorRight:
    ROR 0, 1000
    JA CheckRightLim
GoMotorLeft:
    ROL 0, 1000
    JA CheckLeftLim

```

Monitor 3 inputs and execute based on high or low signal

//TMCL Sample Setup Program. This program will monitor the general purpose //input, the left limit switch, and the right limit switch. Upon sending a low //signal to any of these, it will execute a specific command: stop motor, run at faster speed, run at slower speed.

```

StallValue=0      //StallGuard Threshold (0 = off)
Velocity=2047     //Max Velocity
Acceleration= 1100 //Max Acceleration
ustep= 2         //4 = 16X u-stepping
pdiv = 3        //pdiv @ default
rdiv = 8        //rdiv
current = 1000   //1000 mA Peak
Hcurrent = 300   //300 mA Peak
decay = 0       //2048 = on, 0 = off

```

//Initialization of above parameters

```

SAP 6, 0, current      //Setting Current
SAP 203, 0, decay     //Switch Mixed Decay
SAP 5, 0, Acceleration //Setting max acceleration
SAP 205, 0, StallValue //Setting StallGuard Threshold
SAP 140, 0, ustep     //Setting u-stepping
SAP 143, 0, Hcurrent  //Max Hold Current
SAP 154, 0, pdiv      //Initializing pdiv
SAP 4, 0, Velocity    //Setting maximum Velocity
SAP 138, 0, rdiv      //Ramp Divisor

```

Loop:

```

GAP 10, 0             //get axis parameter of right limit switch
COMP 0               //compare the gotten value with zero
JC EQ, Stop_Motor    //Jump to "Stop_Motor" if equal

GAP 11, 0            //get axis parameter of left limit switch
COMP 0               //compare the gotten value with zero
JC EQ, Go_Faster     //Jump to "Go_Faster" if equal

GIO 0, 0             //get axis parameter of general purpose input
COMP 0               //compare the gotten value with zero
JC EQ, Go_Normal     //Jump to "Go_Normal" if equal

JA Loop              //Continuously loop back to "Loop"
STOP                 //End of program

```

Stop_Motor:

```

MST 0                //Stop motor #0
JA Loop              //Go back to original loop

```

Go_Faster:

```
ROR 0, 2000 //Rotate right continuously at a rate of 3000 pps
JA Loop //Go back to original loop

Go_Normal:
ROR 0, 100 //Rotate Right Continuously at a slower rate
JA Loop //Go back to original loop
```