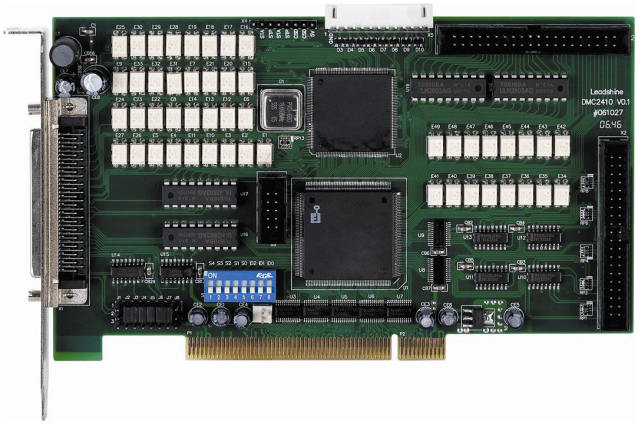
深圳市雷泰控制技术有限公司
SHENZHEN LEADTECH CONTROL TECHNOLOGY CO., LTD

DMC2410 PCI bus 4 axes motion control card

Software Manual

Version 1.1



Limited Warranty:

For a period of one year from the date of original purchase, LEADTECH will repair or replace without charge controls and accessories which our examination proves to be defective in material or workmanship. This warranty is valid if the unit has not been tampered with by unauthorized persons, misused, abused, or improperly installed and has been used in accordance with the instructions and/or ratings supplied. This warranty is in lieu of any other warranty or guarantee expressed or implied.

LEADTECH shall not be held responsible for any expense (including installation and removal), inconvenience, or consequential damage, including injury to any person or property caused by items of our manufacture or sale. (Some countries and U.S. states do not allow exclusion or limitation of incidental or consequential damages, so the above exclusion may not apply.) In any event, LEADTECH's total liability, under all circumstances, shall not exceed the full purchase price of the control. Claims for purchase price refunds, repairs, or replacements must be referred to LEADTECH with all pertinent data as to the defect, the date purchased, the task performed by the control, and the problem encountered. No liability is assumed for expendable items such as fuses. Goods may be returned only with written notification including a LEADTECH Return Authorization Number and any return shipments must be prepaid.

Safety Notice

Only qualified personnel should attempt to start-up, program or troubleshoot this equipment. This equipment may be connected to other machines that have rotating parts or parts that are controlled by this equipment. Improper use can cause serious or fatal injury.

Precautions

WARNING: Do not touch any circuit board, power device or electrical connection before you first ensure that no high voltage is present at this equipment or other equipment to which it is connected. Electrical shock can cause serious or fatal injury.

WARNING: Be sure that you are completely familiar with the safe operation and programming of this equipment. This equipment may be connected to other machines that have rotating parts or parts that are controlled by this equipment. Improper use can cause serious or fatal injury.

WARNING: The stop input to this equipment should not be used as the single means of achieving a safety critical stop. Driver disable, motor disconnect, motor brake and other means should be used as appropriate.

WARNING: Improper operation or programming may cause violent motion of the motor shaft and driven equipment. Be certain that unexpected motor shaft movement will not cause injury to personnel or damage to equipment. Peak torque of several times the rated motor torque can occur during control failure.

CAUTION: The safe integration of this equipment into a machine system is the responsibility of the machine designer. Be sure to comply with the local safety requirements at the place where the machine is to be used. In Europe these are the Machinery Directive, the Electromagnetic Compatibility Directive and the Low Voltage Directive. In the United States this is the National Electrical code and local codes.

CAUTION: Electrical components can be damaged by static electricity. Use ESD (electrostatic discharge) procedures when handling this drive.

Contents

Chapter 1 Introduction

DMC2410 is an ASIC based, pulse type motion controller. This controller is made into three blocks: motion ASIC, PCI card, software motion library. Users can access motion ASIC via our software motion library under Windows 98, 2000, XP driver. Our software motion library provides one-stop function for controlling motors. All the motion parameters' calculations are done via our library.

For example, if you want to perform a one-axis point to point motion with a trapezoidal speed profile, just fill the target position, speed, and acceleration time in one function. Then the motor will run as the profile. It takes no CPU resources because generation of every control cycle pulse is done by the ASIC. The precision of target position depends on the motor driver's closed loop control performance and mechanical parts' precision of the motion table, not on motion controller command because the motion controller is only responsible for sending correct pulses counts via a desired speed profile. So it is much easier for programmers, mechanical or electrical engineers to find out problems and debug.

In order to help customers using the DMC2410 easily, Leadtech provides the MOTION2410 software, which has many kinds of motion control functions and test functions. Using the Motion2410 software, customers can learn the movement control functions and the hardware system quickly, and test the performance of the motion system expediently.

Besides, Leadtech provided some VB, VC program samples for customers to copy those samples as program module in applications directly.

This manual describes how to use functions of DMC2410 card to meet your requirements. It is divided into six chapters:

Chapter1, "Introduction", gives an overview of the product software's features and applications.

Chapter2, "Software Installation", describes how to install the DMC2410.

Chapter3, " Software Application Program", describes how to use VB、VC writing application software with DMC2410's functions.

Chapter4, "Using Motion2410", describe how to utilize a Microsoft Windows based utility program Motion2410 to configure and test running the DMC2410.

Chapter5, "Basic Functions of Motion Control", describes some basic functions of motion control in detail.

Chapter6, "Programming Example", shows 2 examples of program with VC and VB.

Chapter7, "DMC2410 Function Library", a very comprehensive explanation of DMC2410 function library is given.

（The hardware content please refer to the DMC2410 hardware manual）

Chapter 2 Software Installation

Leadtech provides the following software for DMC2410:

1. Drivers;
2. Motion control functions DLL;
3. Demonstration software Motion2410;
4. Some program source codes for DMC2410, which are writing with VB and VC.

2.1 The DMC2410 card drivers install

Install the DMC2410 drivers process is the same as any other card drivers install in PC (such as MODEM card, graphics card). There are two ways to install DMC2410 card drivers in Windows 2000/XP operating system.

2.1.1 Install drivers first then install DMC2410 card

1. Starts the PC;
2. Put the DMC2410 CD to the CD-ROM, in the corresponding directory, as g:\INF, find the regist2k.bat document, double clicks it;
3. Close down the PC, inserts DMC2410 card to an available PCI slot. The details please refer to the hardware manual of DMC2410.
4. Starts the PC, The system will prompt discover the new hardware and pop-up the new hardware install guide like Figure 2-1. Select “Install from list or specific location (Advanced)”, Click “next”.

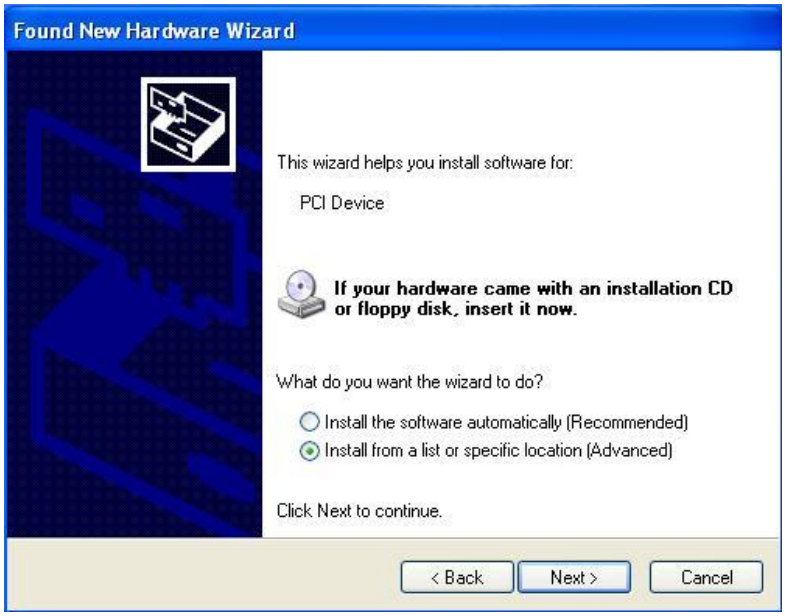


Figure 2-1 New hardware install guide

5. Select “Search for the best driver in these locations” and “Include this location in the search”, click “Browse”, the interface is shown as Figure 2-2.
6. Select folder INF in the “Browse for folder” window, click “OK”.

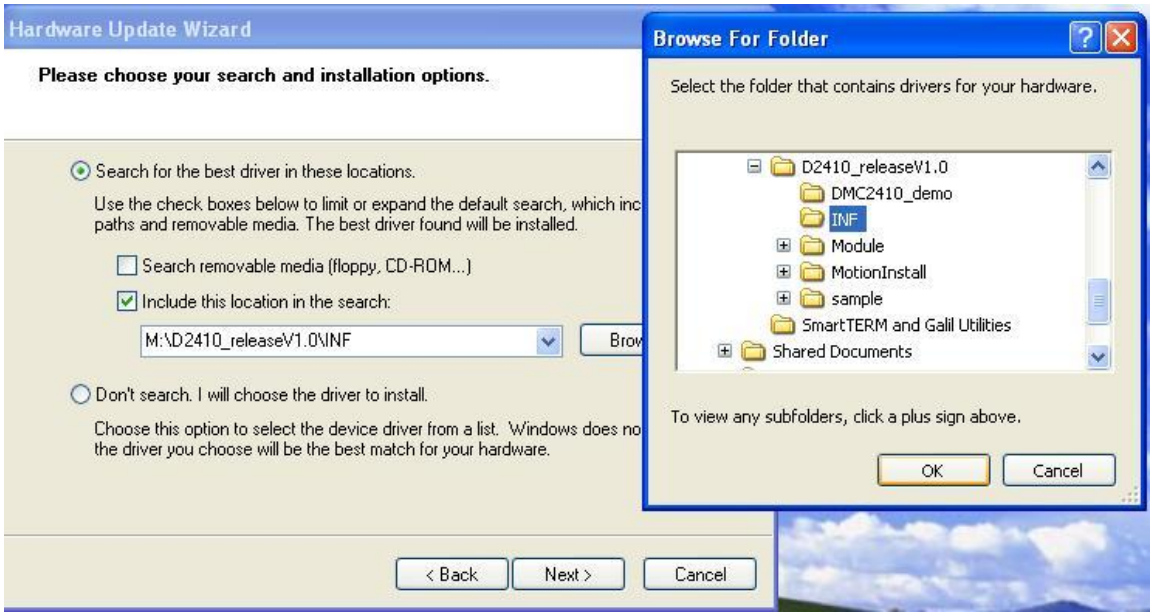


Figure 2-2 Browse and select catalog INF

7. Click”Next” to continue installation, see Figure 2-3.

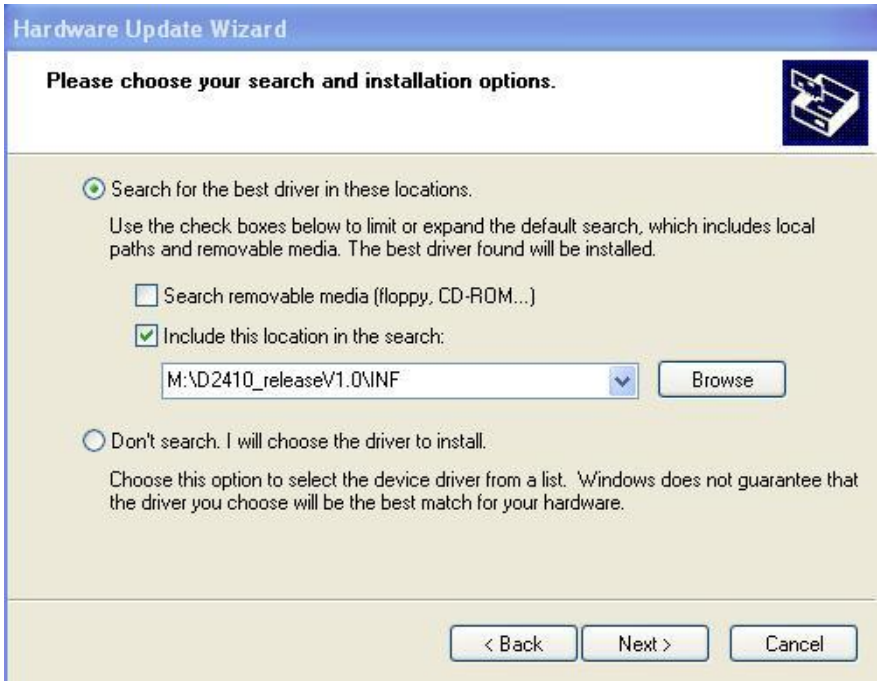


Figure 2-3 Start to install driver

8. As figure 2-4 shown, DMC2410 driver is installing;
9. Click “Finish” to finish installation as figure 2-5 shown.

Figure 2-4 Driver is installing

Figure 2-5 DMC2410 driver installation is finished

10. DMC2410 can use normally.

2.1.2 Install DMC2410 card first then install driver

1. Open the cover of PC, insert DMC2410 card. The details please refer to the hardware manual.
2. Starts the PC;
3. Plese refer the step 4~9 of the frist method.
4. In the corresponding directory, as g:\INF, find the regist2k.bat document, double

clicks it;

5. DMC2410 can use normally.

2.2 Install Motion2410 software and program examples

Put the DMC2410 CD to the CD-ROM, copy the motion2410 which in corresponding directory to any appoint position of the PC disk, executing the motion2410.exe you can test DMC2410 card and the motion system, also you can learn functions of DMC2410.

In order to help customers to develop application program with DMC2410, Leadtech offered some typical source codes of VC and VB, such as Single axis motion, Home motion, Interpolated motion, Position capture, and Position comparison and so on. Consumers can copy them from the corresponding CD directory "sample" to your program project.

Chapter 3 Motion2410 Demo Software

Motion2410 software offered four functions: parameter setup, I/O test and motion test. According to the interface information, consumers can opiate basic motion: such as point to point motion, linear interpolation, circle interpolation, encoder test and I/O test.

Before using Motion2410, please refer to the DMC2410 hardware manual checking hardware system carefully.

The Motion2410 main interface is shown as Figure 3-1. First, user should choose the number of card. If only one DMC2410 card used, the card's number is 0. Then, click the function button which on the main page to go into next interface.

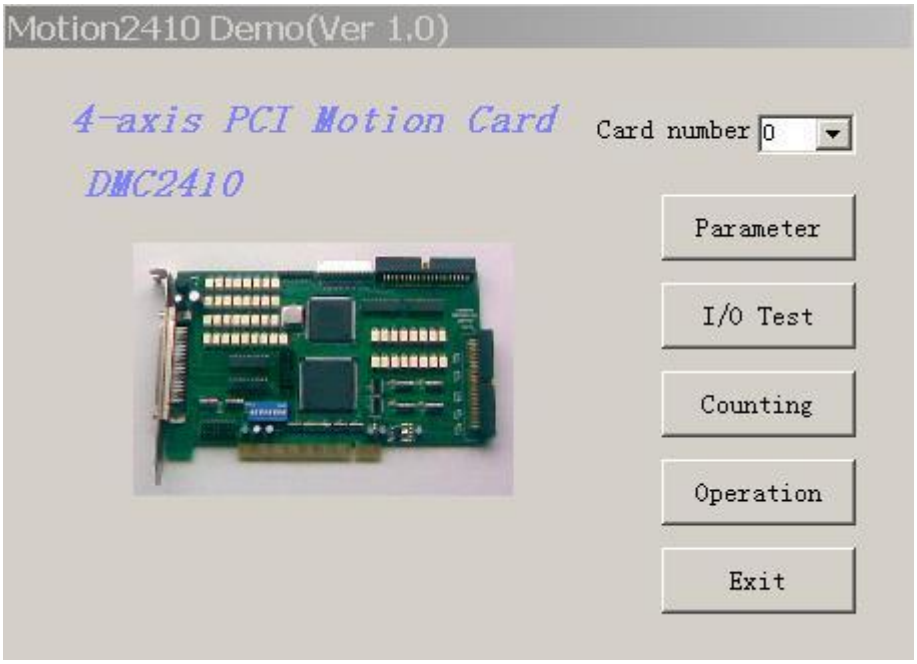


Figure 3-1 Demo software's main page

3.1 Parameter setup operation

Parameter setup interface has 5 sheets: Pulse signal setup, Home parameter setup, Encoder's countering setup and End-limit signal setup.

When click "load" button, the parameters which saved in the data file of Motion2410 will load into DMC2410 card. When click "Save" button, the current parameters which is in the DMC2410 card will save into the data file.

3.1.1 Pulse signal setup

Pulse signal setup include: Pulse mode, active level of pulse and active level of direction signal selection. See figure 3-2.

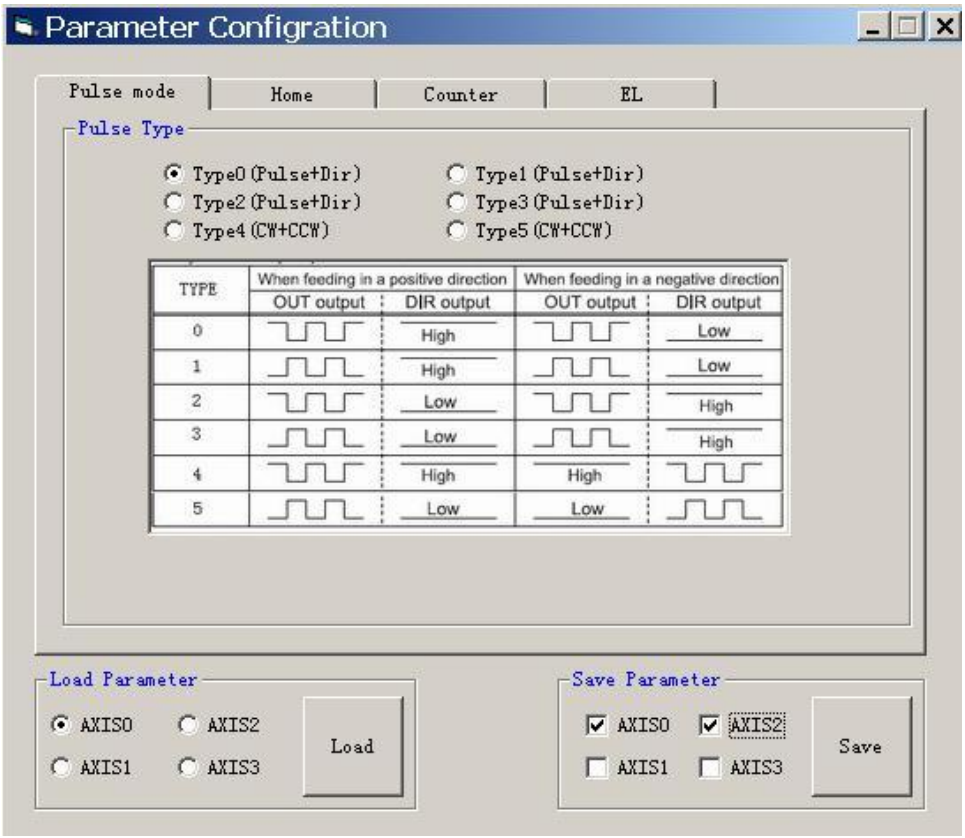


Figure 3-2 The page of Pulse signal setup

3.1.2 Home parameter setup

Home parameter setup include: Home mode, Home speed and Home signal's active level selection. See figure 3-3.

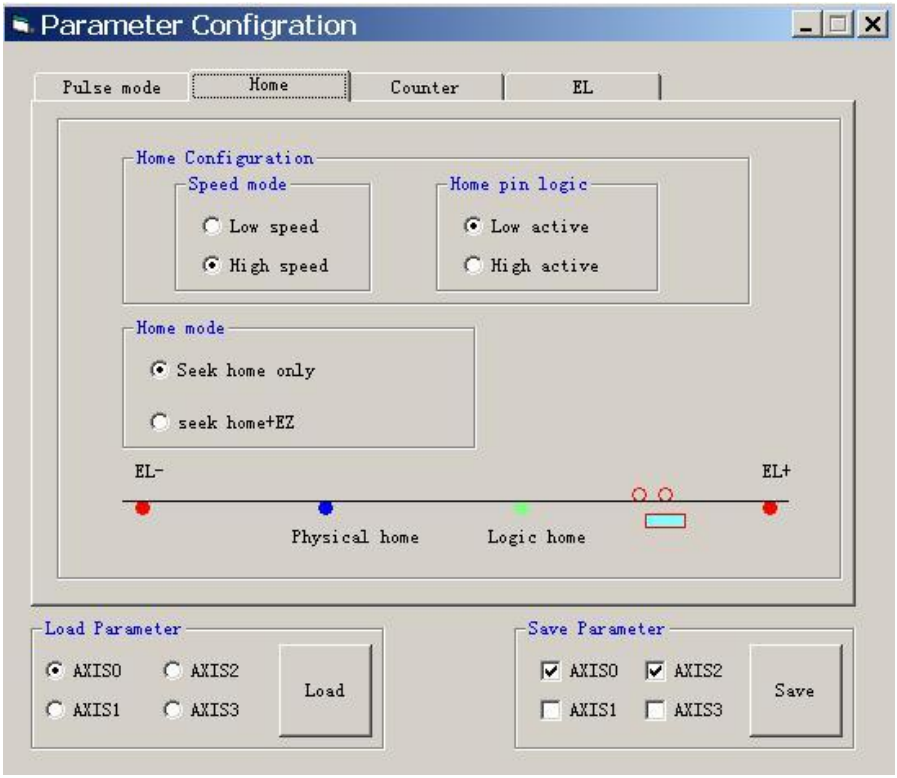


Figure 3-3 The page of Home parameter setup

There are 2 Home modes for selection.

Seek Home only: The motion table go back to the Home sensor, when the Home signal active, the motion table is stopped immediately.

Seek Home +EZ: The motion table goes back to the Home sensor. When the Home signal is active, the motion table is slow down; When EZ signal come, the table is stopped immediately.

3.1.3 Encoder's countering setup

Encoder's countering setup include: Encoder input signal mode selection, EZ signal configuration, position latch signal configuration. See figure 4-4.

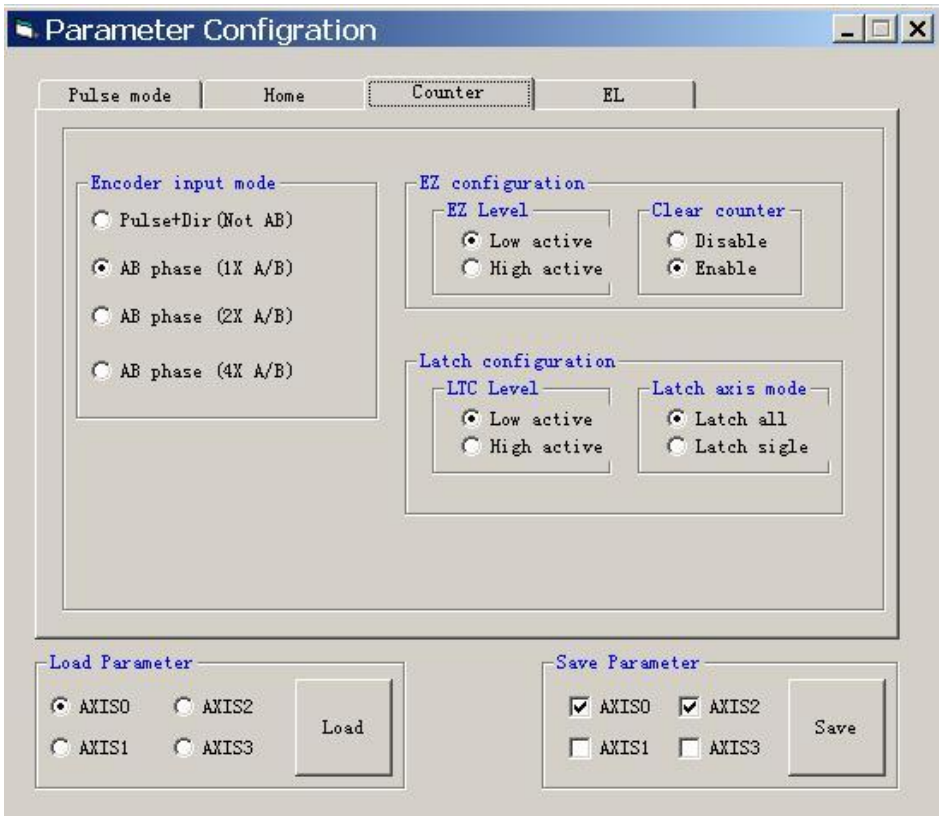


Figure 3-4 The page of Encoder's countering setup

3.1.4 End-limit signal setup

End-limit signal's parameters include: End-limit signal control mode and EL signal's active level. See figure 3-5.

There are 2 mode of the End-limit signal control the motion table stop. The first is the motion table stop immediately when an EL signal is active; the second is the motion table is slow down and stops when an EL signal is active.

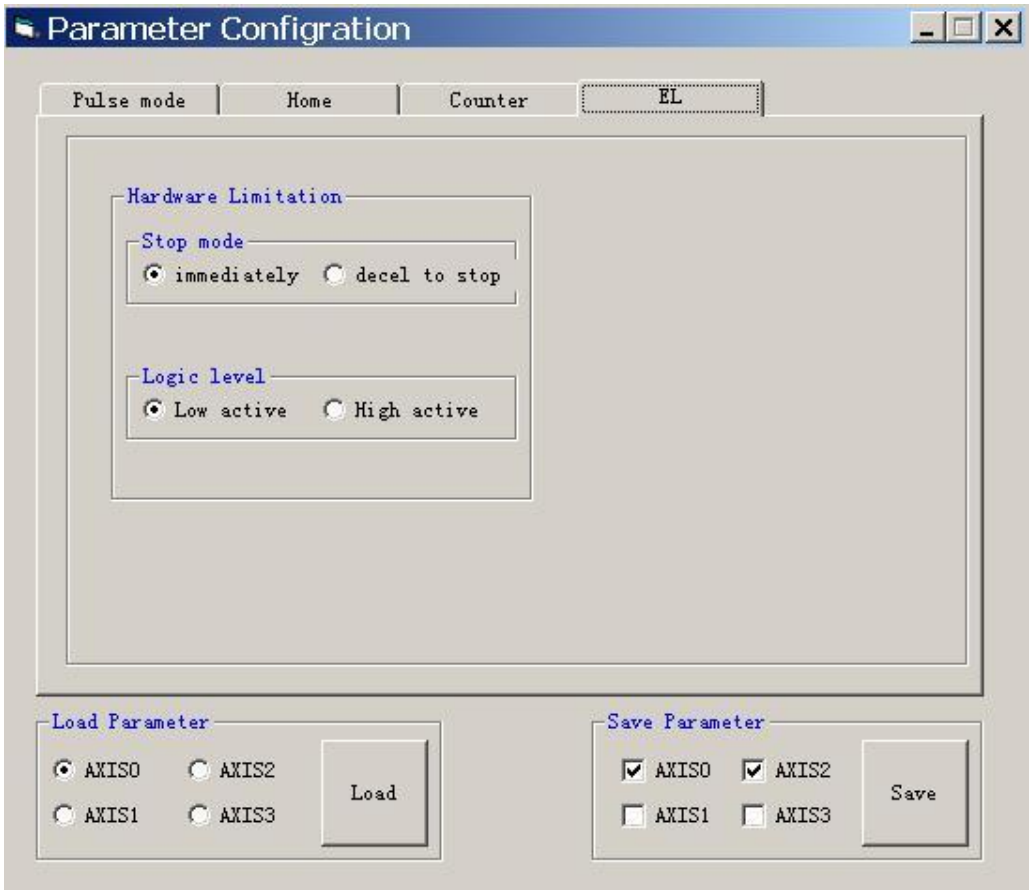


Figure 3-5 The page of End-limit signal setup

3.2 I/O test

I/O test interface is used to test I/O state. Each indicator light denotes an I/O state, green means ON, red means OFF. Press buttons of Output, the output level can be changed. The I/O test interface is shown as figure 3-6.

3.3 Encoder Counting

The page of Encoder Counting is shown as figure 3-7. It is used to test encoders counting.

In Latched Data display box, encoder's position or command's position which is saved in the latch are displayed, when a LTC signal is active.

3.4 Motion test

Motion test interface is shown as figure 3-8.

Axis Selection: Which axis will be test, the axis should be chosen. If multi-axis interpolation will be testing, more axes will be chosen.

Operation type: there are 5 motion types can be chosen.

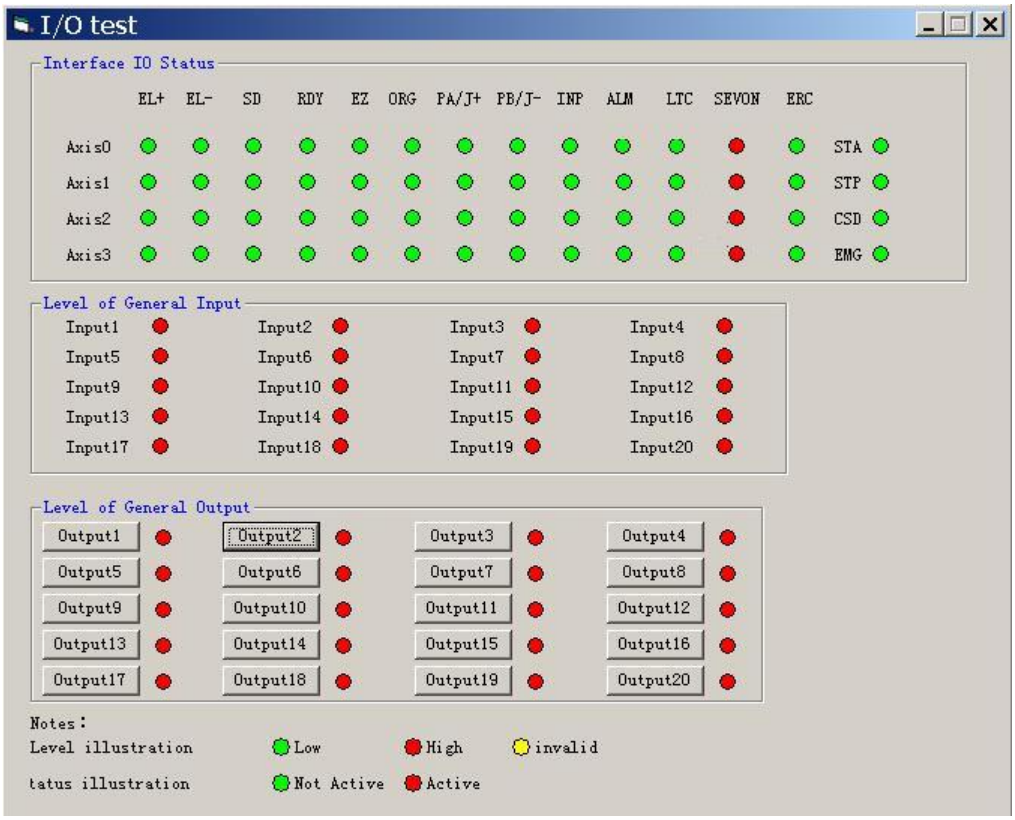


Figure 3-6 The page of I/O test

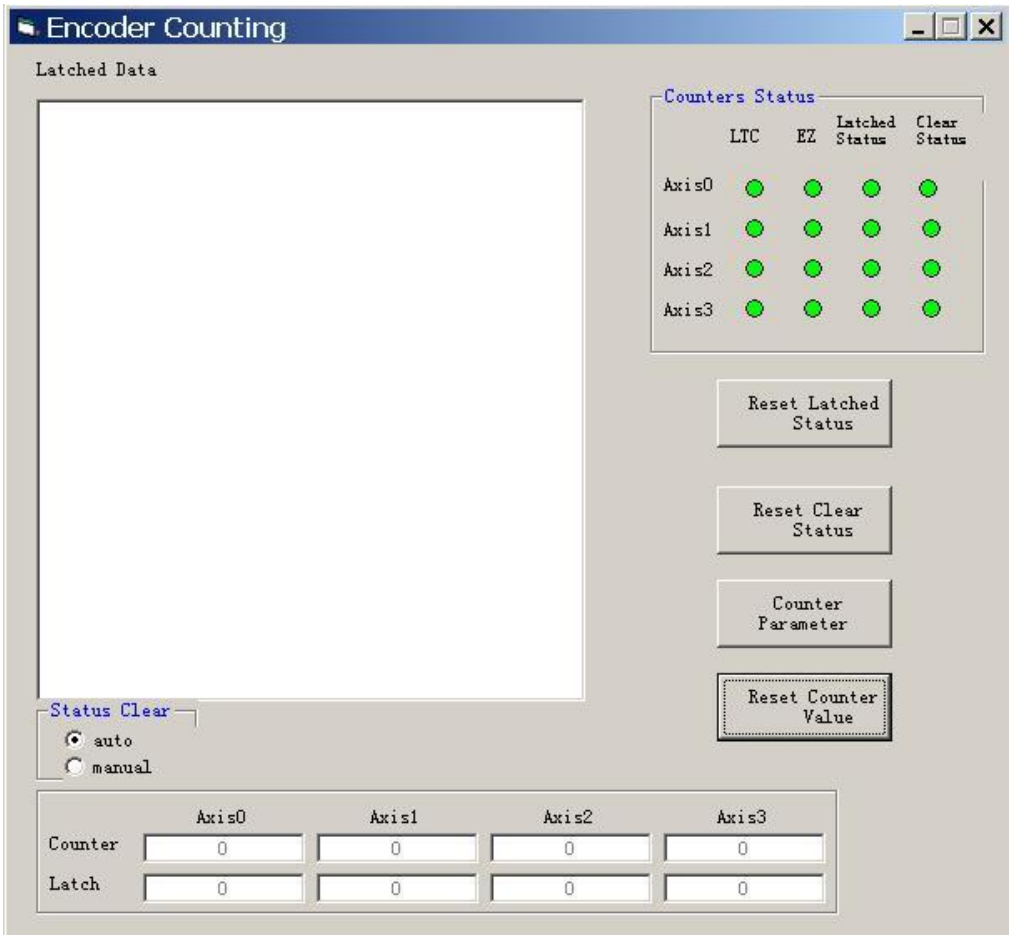


Figure 3-7 The page of Encoder Counting

Single axis parameter: Minimum velocity, maximum velocity, acceleration time, time of

accelerating S-curve part, motion distance, motion direction, and deceleration time should be setting, the unit of distance is pulse, speed is p/s, time is s.

Interpolation motion parameters are same as single axis motion parameter.

Speed curve: setup trapezoidal speed profile or setup S-curve speed profile.

Position source: Select command position or encoder feedback position of 4 axes will display on the below.

Load Parameter button: Load parameters from the data file, it would be a default parameters list loaded unless you had saved the parameters setting to the data file previously.

Save Parameter button: save current parameters setting into a existent data file.

Start button: Execute motion command with current parameters.

Stop button: Stop the motion command which is executing.

Clear position button: Clear command position register.

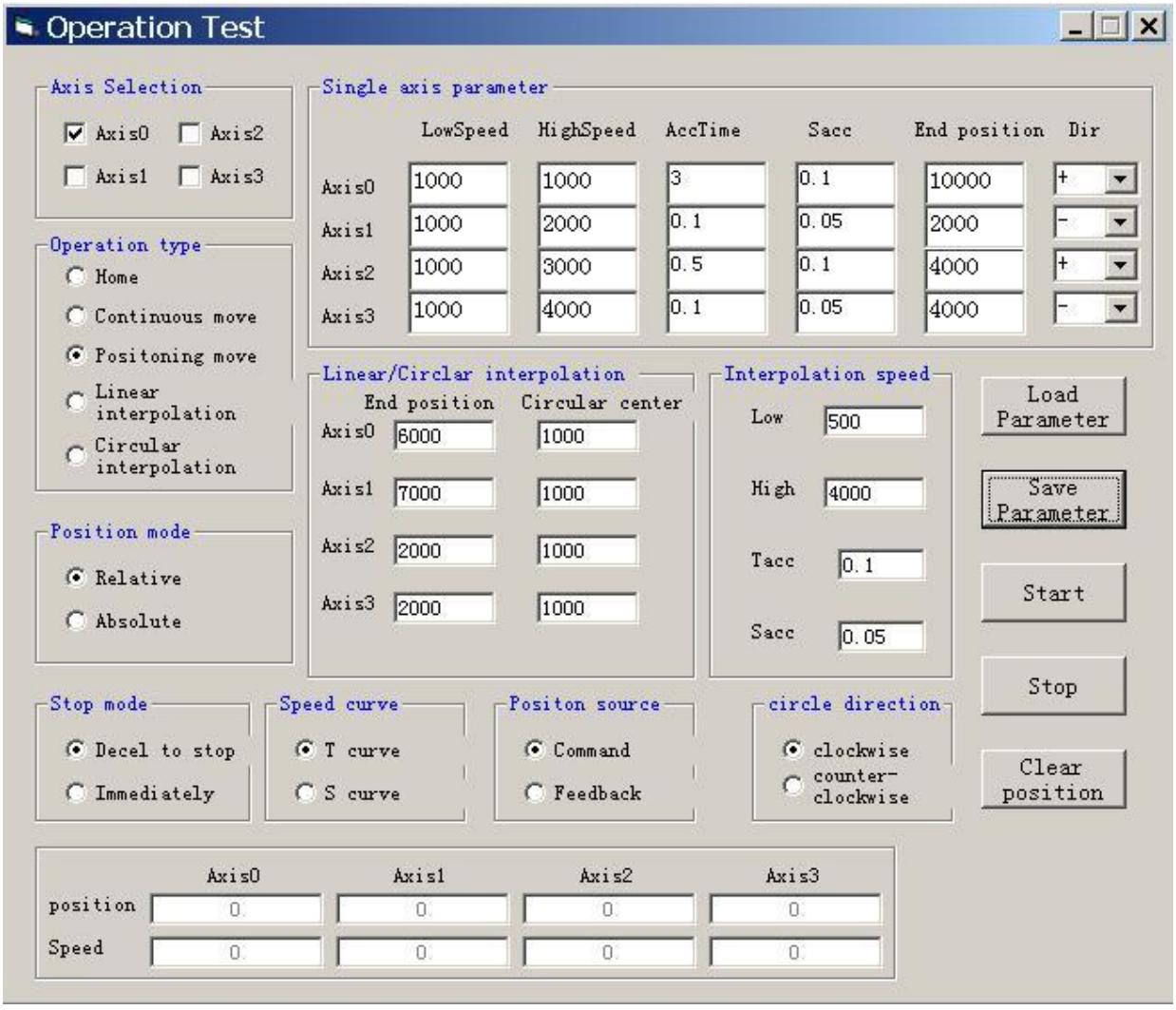


Figure 3-8 The page of motion test

Chapter 4 Motion Control Functions

In this chapter, main functions of DMC2410 will be described. The list of all DMC2410 functions is in Chapter 6, .the parameters and return value of functions are described in detail.

4.1 Initialization and close DMC2410 card

Before using DMC2410 card in a system, the function of d2410_board_init will be called to allocate PC system resource and init its parameters. Sameness, after using DMC2410 card, the function of d2410_board_close should be called at the end of the application program to close DMC2410 card and release PC's resources.

.

Example：Init and close DMC2410 card

(Using standard C language for example. Same as all examples below)

```
……
CardCount = d2410_board_init();     // init DMC2410
if(CardCount == 0)                  // if return value is 0,
{
    printf("\n Have not find DMC2410 card");
    getch();
    return();
}
……
d2410_board_close();                // close DMC2410
……
```

4.2 Pulse output mode

DMC2410 uses pulse command to control the servo / stepping motors via the drivers. The pulse command consists of two output signals: OUT and DIR. The pulse command output of DMC2410 has two mode：

(1) PUL/DIR mode (single pulse output mode);

(2) CW/CCW mode (dual pulse output mode).

In PUL/DIR mode, the PUL signal indicates the motor's rotating speed, and the DIR indicates the motor's rotating direction, as the Figure 4-1:

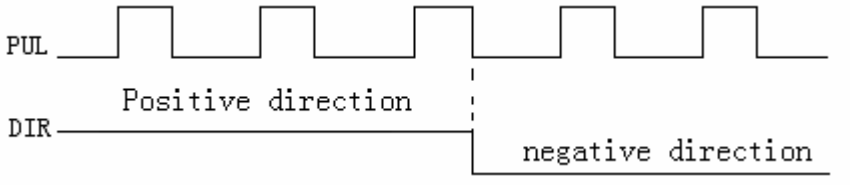


Figure 4-1 The PUL + DIR pulse output mode

In CW/CCW mode, the PUL pulse signal indicates the motor in positive direction, the DIR pulse signal indicates the motor in negative direction, and the pulse frequency relate to the motor speed as the Figure 4-2.

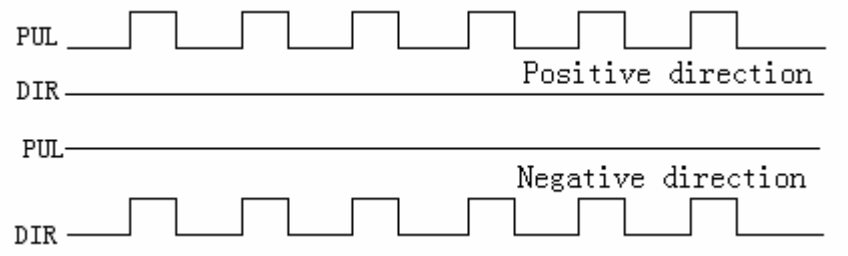


Figure 4-2 The CW/CCW pulse output mode

Example：Set the pulse output mode

```
……
d2410_set_pulse_outmode (0,0)；// set axis 0 is PUL/DIR mode, the raise edge of PUL0- is
                                active, the low level of DIR0- is positive direction.
d2410_set_pulse_outmode (1,4)；// set axis 1 is CW/CCW mode, raise edge is active.
……
```

4.3 Motion Control Modes

DMC2410 can make motors run according to a specific speed profile, path trajectory and synchronous condition with other axes. The following sections describe the motion control modes of this motion controller could be performed.

4.3.1 Coordinate system

The Cartesian coordinate system and pulses for the unit of length are used. The physical length depends on mechanical parts and motor's resolution. For example, if the motor is installed on a ball screw, the pitch of the ball screw is 10 mm and the pulses needed for a round of the motor are 10,000 pulses. We can say the physical unit of one pulse is equal to 10 mm/10,000 pulse = 1 micro-meter/pulse.

4.3.2 Absolute and relative position move

There are two kinds of commands to locate target positions in the coordinate system: absolute and relative. Absolute command means that to give the motion controller a position of the coordinate, the motion controller will control a motor to move that position of the coordinate. Relative command means that to give the motion controller a distance, the motion controller will control a motor to move by the distance from current position. During the movement, you can specify a speed profile.

4.3.3 Trapezoidal speed profile

A trapezoidal speed profile means the acceleration and deceleration course follows a first-order linear velocity profile (constant acceleration rate). The profile chart is shown as

Figure 4-3.

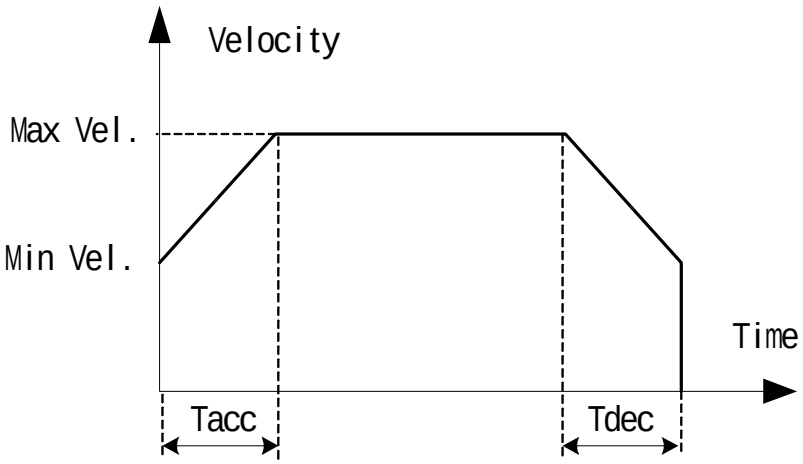


Figure 4-3 A trapezoidal speed profile

Sometimes, the profile looks like a triangle because the desired distance is smaller than the area of given speed parameters. When this situation happens, DMC2410 motion controller will lower the maximum velocity but keep the acceleration rate to meet the distance requirement, and the pinnacle of speed profile is eliminated automatically. The chart of this situation is shown as Figure 4.4.

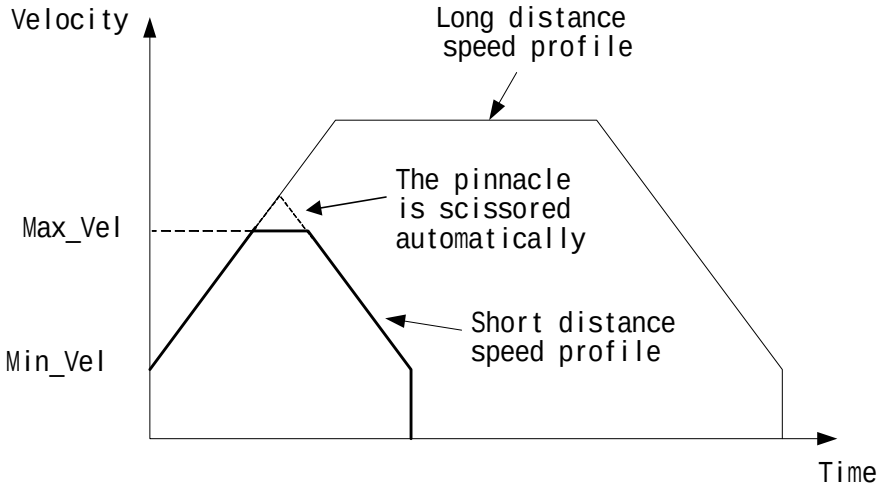


Figure 4-4 Automatically adjust speed profile

This kind of speed profile could be applied on velocity mode, position mode in one axis or multi-axes linear interpolation and two axes circular interpolation modes.

4.3.4 S-curve speed profile

S-curve means the speed profile in acceleration and deceleration course follows a second-order curve. In order to speed up the acceleration and deceleration during a motion, a linear part is inserted between the upper side of s-curve and lower side of s-curve.

S-curve improves the speed of acceleration and deceleration course compare with the trapezoidal profile, also reduces the vibration of the motion table at the start and end points of acceleration and deceleration course.

S-curve speed profile and its acceleration chart are shown as Figure 4-5.

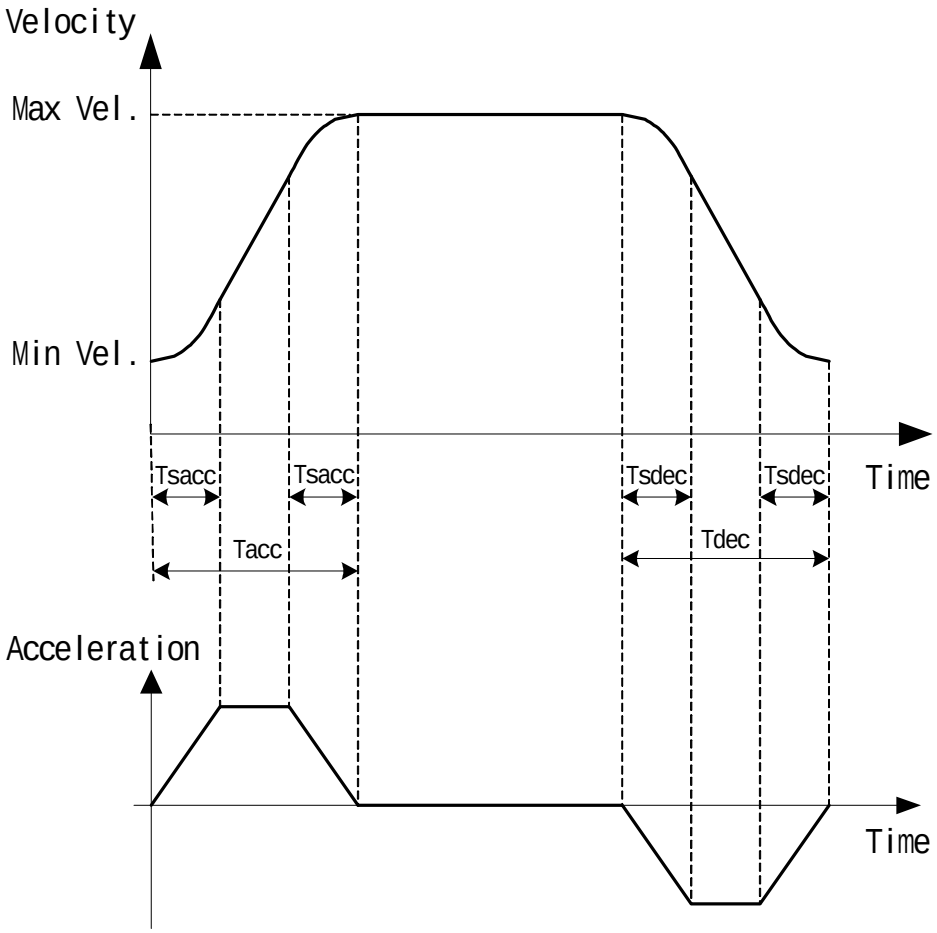


Figure 4-5 S-curve speed profile and its acceleration chart

In Figure 4-5,

Min Vel: Start and end velocity in PPS

Max Vel: Maximum velocity in PPS

Tacc: Acceleration time in second

Tdec: Deceleration time in second

Tsacc: S-curve part in acceleration in second

Tsdec: S-curve part in deceleration in second

The S-curve profile motion functions are designed to always produce smooth motion. If the time for acceleration parameters combined with the final position don't allow an axis to reach the maximum velocity (for example: the moving distance is too small to reach MaxVel), then the maximum velocity is automatically lowered.

The rule is to lower the value of MaxVel and the Tacc, Tdec, Tsacc, Tsdec

automatically, Min Vel and acceleration unchanged. (See Figure 4-6)

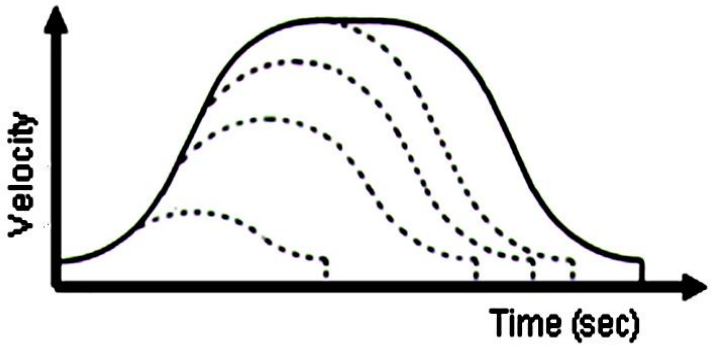


Figure 4-6 Automatically adjust S-curve

This kind of speed profile could be applied on velocity mode, position mode in one axis or multi-axes linear interpolation and two axes circular interpolation modes of DMC2410.

4.3.5 Velocity mode

Velocity mode means the pulse command is continuously output until a stop command is issued. In other words, the motor will run without a target position or desired distance unless it is stopped by other reason.

The output pulse accelerates from a starting velocity to a specified maximum velocity. It can be followed by a trapezoidal or S curve acceleration profile. The pulse output rate is kept at maximum velocity until another velocity command is set or a stop command is issued. The velocity can be overridden by a new speed setting. Notice that the new speed could not be a reversed speed of original running speed. The speed profile of this kind of motion is shown as Figure 4-7.

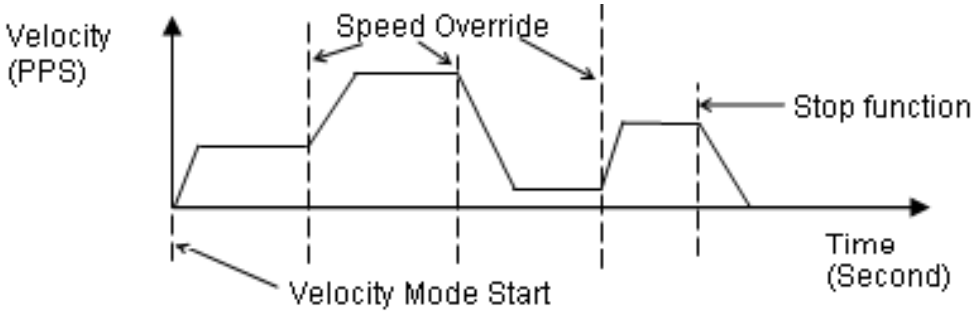


Figure 4-7 Change speed in the velocity mode

Example：Change speed and stop motion in Velocity mode with a trapezoidal speed profile.

```
……
d2410_set_profile(0,500,1000,0.1,0.1);   //Set trapezoidal profile parameters
d2410_t_vmove(0,1);                      //Axis0 moving in velocity mode, direction is positive
 if(“condition of change speed”)         // If the condition is true, then change speed
 {
    Curr_Vel= 1200；                      //Set new speed
    d2410_change_speed(0,Curr_Vel)；      //change speed
```

```
 }
if("condition of stop")                    // If the condition is true, then stop moving
     d2410_decel_stop(0,0.1);              // speed slow-down, deceleration time is 0.1 S
……
```

Related functions:

d2410_variety_speed_range
d2410_imd_stop
d2410_emg_stop
d2410_read_current_speed

4.3.6 One axis position mode

Position mode means the motion controller will output a specific amount of pulses which is equal to the desired position or distance. The unit of distance or position is pulse internally on the motion controller. The minimum length of distance is one pulse.

Besides position via pulse counts, DMC2410 motion controller provides two types of speed profile to accomplish positioning: the trapezoidal curve and the S-curve. Users can call respective functions to perform that.

4.3.7 Position Override Function

Position override means that change target position after a positioning command is issued. If the new target position is behind current position, the motor will slow down, and then reverse to new target position. If the new target position is far away from current position on the same direction, the motion will remain its speed and run to new target position. If the override timing is on the deceleration of current motion and the target position is far away from current position on the same direction, it will accelerate to original speed and run to new target position. See Figure 4-8.

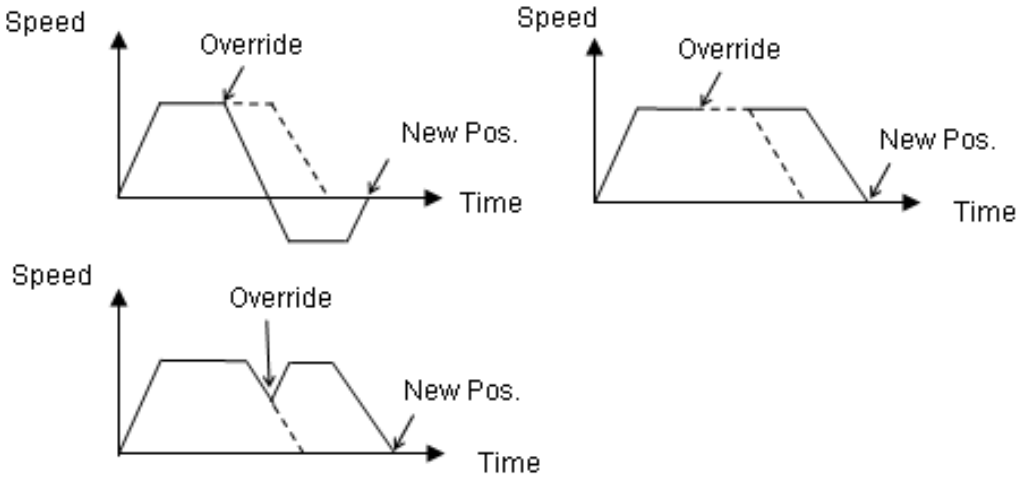


Figure 4-8 Change target position

Example：Change target position

```
……
d2410_set_profile(0,500,6000,0.01,0.02)；    // set trapezoidal speed profile
d2410_ex_t_pmove(0,50000,0)；                // axis 0 moving 50000 pulses of relative distance
If(“a condition of change target position”)   // if condition is true, then change target position
{
   d2410_reset_target_position(0,55000)；     // change target position to 55000 pulses
}
……
```

4.3.8 Two axes linear interpolation position mode

“Linear interpolation between multi-axes” means these axes start simultaneously, reach their ending points at the same time, and the ratio of speed of every axis is a constant value. Assume that DMC2410 control a motion table moving from (0,0) to (20,9). The pulses output from X or Y axis remains 1/2 pulse difference according to a perfect linear line. The linear interpolation results are shown as Figure 4-9.

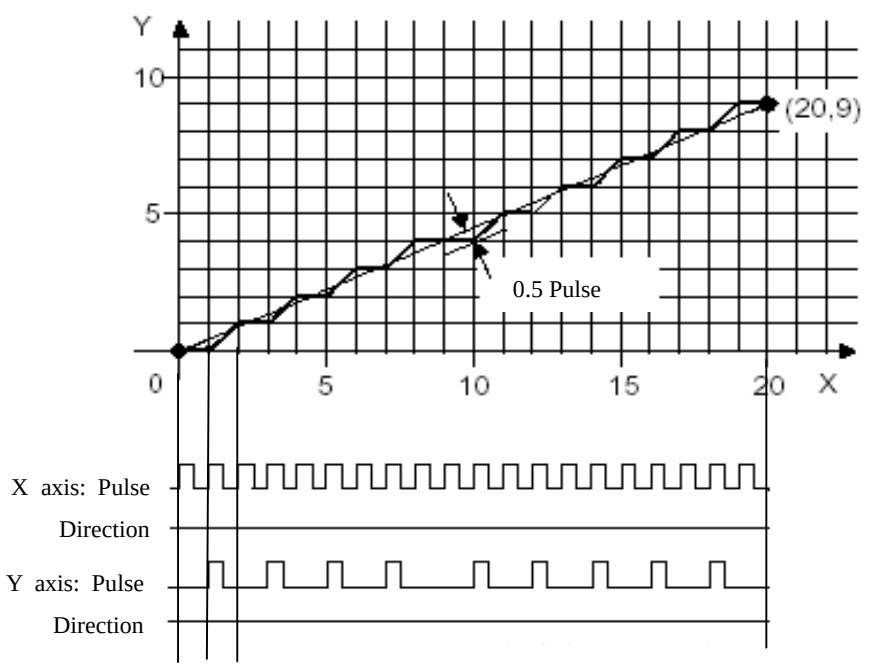


Figure 4-9 Linear interpolation tracks

As Figure 4-10 shown, 2-axis linear interpolation means to move the XY position from P0 to P1. The 2 axes start and stop simultaneously, and the path is a straight line.

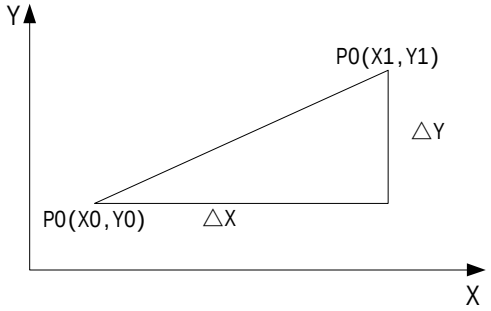


Figure 4-10 Linear interpolation parameters

The speed ratio along X-axis and Y-axis is ΔX/ΔY, respectively, and the vector speed of linear interpolation is:

$$\frac{\Delta P}{\Delta t}=\sqrt{\left(\frac{\Delta X}{\Delta t}\right)^2+\left(\frac{\Delta Y}{\Delta t}\right)^2}$$

When 2-axis linear interpolation functions are called, the Min velocity and Max velocity should be defined with vector speed.

Example：Axis X and axis Y linear interpolation with relative position.

```
……
short AxisArray[2];
AxisArray[0]=0;              // setup axis X as No. 0 axis
AxisArray[1]=1;              // setup axis Y as No. 1 axis
d2410_set_vector_profile(1000,5000,0.1,0.2);    // vector speed: Vmin =1000, Vmax =5000pps
                                                // acceleration time=0.1, deceleration time=0.2s
d2410_t_line2(AxisArray[0],30000,AxisArray[1],40000,0);       // ΔX=30000, ΔY=40000 pulse
……
```

Related functions:

d2410_set_vector_profile

d2410_t_line3

d2410_t_line4

4.3.9 Two axes circular interpolation mode

Circular interpolation can be executed between any 2 axes of DMC2410. The direction of circular interpolation has clockwise and anti-clockwise, and the speed of circular interpolation is tangential speed. See Figure 4-11.

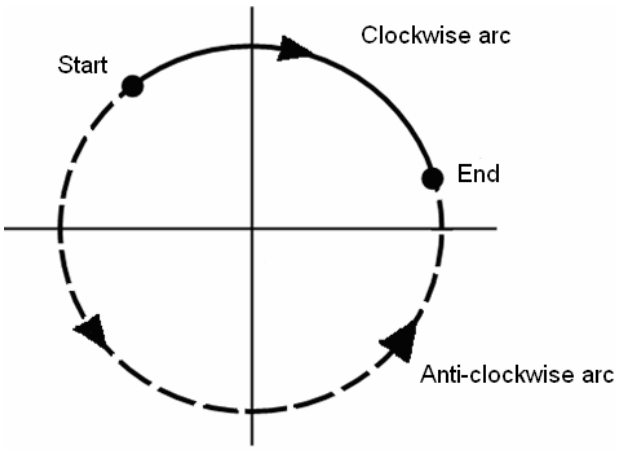


Figure 4-11 Circular interpolation

Example：Axis X and axis Y circular interpolation with relative position.

```
……
short AxisArray[2];
long cen_pos[2] = {5000, 0};       // setup the centre point（5000, 0）of the circle
long end_pos[2] = {5000, -5000};// setup the end point （5000, -5000）with relative position
```

```
AxisArray[0]=0;                    // setup axis X as No. 0 axis
AxisArray[1]=1;                    // setup axis Y as No. 1 axis
d2410_set_vector_profile(1000,3000,0.1,0.1); // Tangential speed: Vmin=1000,Vmax=3000pps
                                             // acceleration time=0.1, deceleration time=0.1s
d2410_rel_arc_move (AxisArray, end_pos, cen_pos, 0);    //clockwise circular interpolation
  ……
```

4.3.10 Continuous motion

Continuous motion means a series of motion command can be run continuously. A new command can be set right after previous one without interrupting it. DMC2410 motion controller can make it possible because there is a command buffer (reregister) inside.

Example：Axis X and axis Y continuous motion

DMC2410 control an x-y table to move a curve as Figure 4-12 shown. First, the table moves from the origin to point C1, then begin continuous motion from C1 to C2, C3, C8, C1.

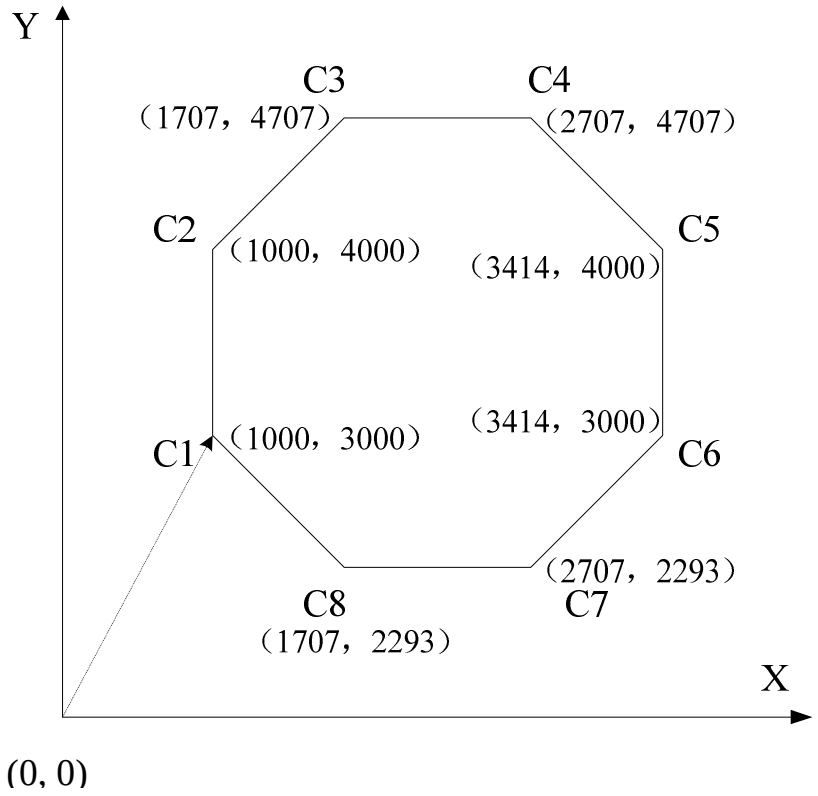


Figure 4.12 Continuous motion paths

```
……
WORD Axis[2];
int i;
long pos_array[8][2] = { {1000, 3000}, {1000, 4000}, {1707, 4707}, {2707, 4707}, {3414, 4000},
                         {3414, 3000}, {2707, 2293}, {1707, 2293} }; // set points' coordinates
 Axis[0] = 0;
 Axis[1] = 1;
 d2410_set_vector_profile (500, 3000, 0.01, 0.01);                    // set speed profile
 d2410_t_line2(Axis[0], pos_array[0][0], Axis[1], pos_array[0][1], 1);
                                                                      // move from point O to point C1
```

```
while ( (d2410_check_done(Axis[0])==0 ) || (d2410_check_done(Axis[1])==0) )
      {  }                                          // waiting axis 0,1 stop
for (i=1; i<8; i++)
  {
   while ( (d2410_prebuff_status(Axis[0]) == 1) || (d2410_prebuff_status(Axis[1]) == 1) )
         {  }                                       // waiting the reregister is empty
   d2410_t_line2 (Axis[0], pos_array [i][0]- pos_array [i-1][0], Axis[1], pos_array [i][1]- pos_array
                  [i-1][1], 0);                     // writing new command
  }
……
```

Note:

1. Continuous motion must use the relative position mode;
2. If reregister is not empty, the new command can't be written, otherwise the new command will be false.

4.3.11 Home Return Mode

Home return is to search for a zero position point on the coordinate. Sometimes, ORG, EZ or EL sensor is used as a zero position on the coordinate. After system power-on, the program needs to find a zero point of each axis.

DMC2410 have many home modes. After home return is completed, the target counter will be reset to zero at the desired condition of home mode, such as a raising edge when ORG input.

Example1： Home with low speed

In this home mode, the motor moves with a constant low speed, when the ORG signal is active, the motor stop immediately. Then set the position as zero point. See Figure 4-13.

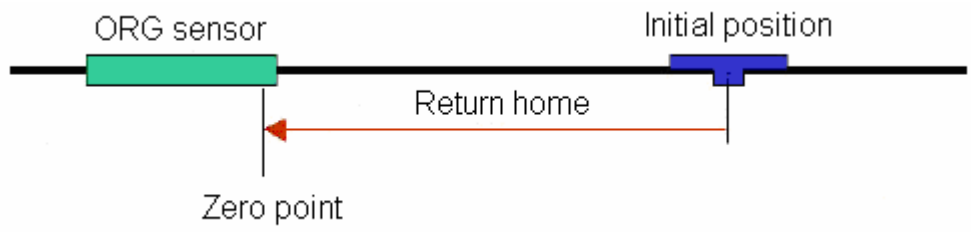


Figure 4.13 Home with low speed

```
……
d2410_set_HOME_pin_logic (0,0,1);          // Set axis 0 's ORG signal is low level active
d2410_config_home_mode(0,0,1);             // Set Home mode: stop home when ORG is active
d2410_set_profile(0,500,1000,0.1,0.1);     // Set axis 0 's speed profile
d2410_home_move(0,2,0);                    // begin home, speed is 500 pps, direction is minus
while (d2410_check_done(0) == 0)           // waiting for home stop
      {  }
d2410_set_position(0,0);                   // set axis 0 's command counter=0
……
```

Example2： Home used EZ signal

In this Example, the EZ signal is used for set zero point. Step 1, the motor return home with high speed; Step 2, the motor slow-down when ORG signal is active, and stop until EZ signal is active. See Figure 4-14.

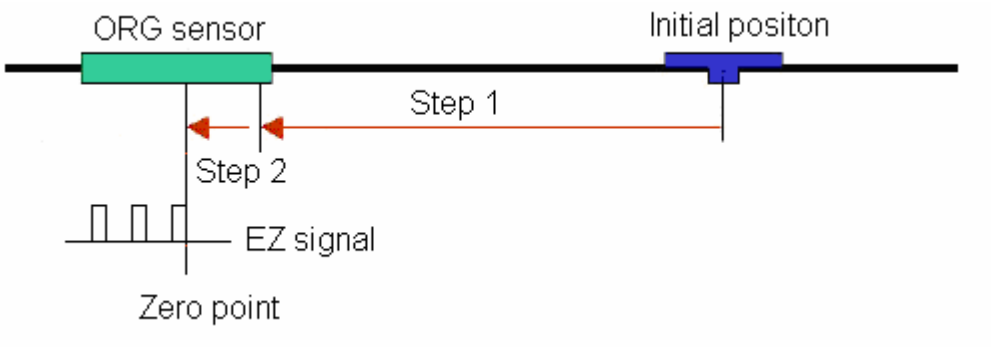


Figure 4-14 Home used EZ signal

```
……
d2410_set_HOME_pin_logic(0,0,1);          // Set axis 0 's ORG signal is low level active
d2410_config_home_mode(0,1,1);            // Set Home mode: stop home when first EZ is active
d2410_set_profile(0,500,1000,0.1,0.1);    // Set axis 0 's speed profile
d2410_home_move(0,2,1);                   // begin home, Vmax=1000, Vmin=500 pps,
                                          // direction is minus
while (d2410_check_done(0) == 0)          // waiting for home stop
      {  }
d2410_set_position(0,0);                  // set axis 0 's command counter=0
      ……
```

4.4 Encoder input and it's functions

4.4.1 Encoder signal mode

Each axis of DMC2410 has an up/down counter for checking the current position. The counter counts signals input from EA and EB pins. The card can accept 2 kinds of pulse input: (1) CW/CCW mode; (2) 90°phase differential signal mode.

In CW/CCW mode, pulse from EA causes the counter to count up; otherwise, pulse from EB causes the counter to count down.

In this mode, EA and EB signals are used. If the EA signal is 90°phase leading compare with EB signal, it will be consider as positive direction. If the EA signal is 90°phase lagging compare with EB signal, it will be consider as negative direction.

If a rotary encoder has 2000 pulses per circle, and the multiplied factor 4x is selected, then the value read from the counter will be 8000 pulses per round or -8000 pulses per round.

Example： Read encoder position data

```
……
d2410_counter_config (0,3);              // set axis 0 encoder is 4 times frequency counting.
d2410_set_encoder(0,0);                  // reset axis 0's counter
X_Position = d2410_get_encoder(0);       // read axis 0's encoder data
……
```

4.4.2 Position capture

The DMC2410 support capturing the position of an axis's encoder triggered by LTC input. This function is widely used in auto measure devices, the LTC signal usually come from a probe.

Example：Position capture

```
int key=0;                                          // the flag of capture state
long LatchStatus;                                   // the flag of trigger state
long xValue[100],yValue[100],zValue[100];           // the array for captured position data
int g_count=0;                                      // reset a counter
d2410_config_latch_mode(0,1);                       // capture 4 axes data at same time
while(g_count<100)                                  // capture 100 points
{
   LatchStatus = d2410_get_latch_flag(0);           // read the state of the latch
   if ( ( (LatchStatus & 0xF00) != 0) && key==0)
   {                                                // 4 axes of Card 0 have captured new data
      xValue[g_count] = d2410_get_latch_value(0);   // read captured data
      yValue[g_count] = d2410_get_latch_value(1);
      zValue[g_count] = d2410_get_latch_value(2);
      g_count++;                                    // the counter add 1
      key = 1;                                      // set the flag of Key
   }
   else if ( ( (LatchStatus & 0xF) == 0) && key==1)
   {
      d2410_reset_latch_flag(0);                    // reset the latch
      key = 0;                                      // reset the flag of Key
   }
}
```

4.4.3 Position compare

The DMC2410 provide two positions of an axis for the position compare. When the current position of the axis is equal or less or more than the positions which setting by program, the CPM pin will output a signal to indicate that a position compare event has occurred.

Related functions:

d2410_config_CMP_PIN

d2410_read_CMP_PIN
d2410_write_CMP_PIN
d2410_config_comparator
d2410_set_comparator_data

4.5 general digital I/O control

DMC2410 can control 20 general digital input and output, 20 general digital output signals, which can be use to input switches, sensors signals, and to control relay, electromagnetic valve, signal lamp devices.

Example：Using a switch control a LED, the circuit diagram is shown as Figure 4-15.

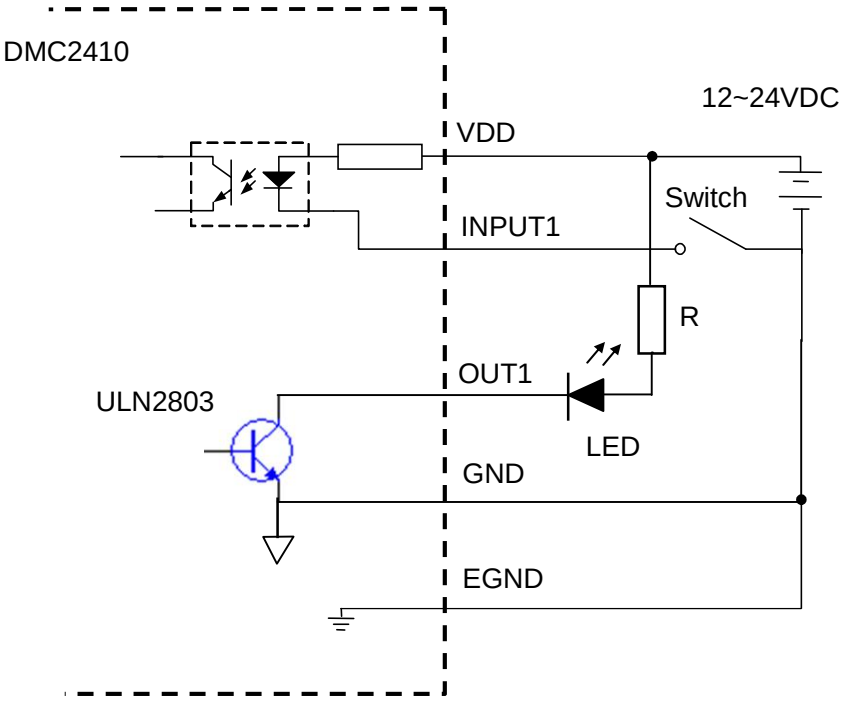


Figure 4-15 Using a switch control a LED

```
……
If(d2410_read_inbit(input1)==0)      // read the switch state
 {
     d2410_write_outbit(out1, 0);     // if the switch close, output out1=0, the LED light
 }
else
 {
     d2410_write_outbit(out1, 1);     // if the switch open, output out1=1, the LED dark
 }
 ……
```

Related functions:

int d2410_read_outbit(WORD cardno,WORD bitno)

long d2410_read_inport(WORD cardno)
long d2410_read_outport(WORD cardno)

4.6 AC servo Motor Interface

The DMC2410 provides RDY, INP, ALM, SEVON and ERC signals for AC servo motor driver's control interface. RDY, INP, and ALM are used for feedback the servo driver's status. The SEVON and ERC are used to control servo motor driver.

4.6.1 RDY signal

When a servo motor driver is ready to move, it will send a RDY single to its motion controller. DMC2410 can check the RDY input to decide if send pulses to the motor.

Related functions:

d2410_read_RDY_PIN
d2410_read_inbit

4.6.2 INP signal

Usually, servo motor driver with pulse input has a position deflection counter to detect the deflection between the input pulse command and feedback counter. The driver controls the motion of servo motor to minimize the deflection until it becomes 0. Theoretically, the servo motor operates with some time delay from command pulses. Accordingly, when the motion controller stops outputting pulses, the servo motor does not stop but keep running until the deflection counter become zero. At this moment, the servo driver sends out the in-position signal (INP) to the controller to indicate the motor stops running.

Related functions:

d2410_config_INP_PIN
d2410_get_rsts
d2410_check_done

4.6.3 ALM signal

The ALM input receives the alarm signal output from the servomotor driver. The signal immediately stops DMC2410 to generate pulses or stop it after deceleration.

Related functions:

d2410_config_ALM_PIN
d2410_axis_io_status

4.6.4 SEVON signal

DMC2410 can send a single form SEVON pin to a servomotor driver, let it into standby state.

Related functions:

d2410_write_SEVON_PIN
d2410_read_SEVON_PIN

4.6.5 ERC signal

The ERC (Deflection counter clear) signal can immediately stop the servomotor by resetting the deflection counter to zero. ERC usually is inserted in the following 4 situations:

(1) Home return is complete;
(2) The end-limit switch is active;
(3) An alarm signal stops PULSE and DIR signals;
(4) An emergency stop command is issued by software operator.

Related functions:

d2410_config_ERC_PIN

4.7 Set scale factor

The scale factor allows each axis to be scaled into engineering units for ease of use. It is a multiplication factor that is applied to all motion variables for that axis (speed, acceleration, move distances, etc). By default the scale factor is 1

For example, an X-Y table uses stepping motors which is 400 pulses per revolution, and the pitch of ball screw is 5 mm. The stepping motor rotates 80 pulses, the table moves 1 mm. So the scale factor is 80 pulse/mm.

After using scale factors on an X-Y table which have different screw pitches, a circle track can be done.

The encoders of DMC2410 also can be set scale factors. Using the position loop control function, the position error can be compensated automatically.

Related functions:

d2410_set_equiv
d2410_get_equiv
d2410_get_position_unitmm
d2410_set_position_unitmm

d2410_read_surrent_speed_unitmm
d2410_get_encoder_unitmm
d2410_set_encoder_unitmm
d2410_arc_move_unitmm
d2410_rel_arc_move_unitmm
d2410_pulse_loop

4.8 Multiple DMC2410 Cards Operation

The software function library support maximum up to 8 DMC2410 cards that is one PC can control 32 motors. Since DMC2410 has the characteristic of Plug-and-Play, users not need to care about setting the based address and the IRQ level of the card.

No. 1 of cards controls No. 0 ~ 3 of axes, No. 2 of cards controls No. 4 ~ 7 of axes, and so on. User can use MOTION2410 testing software to check the number of axes and the number of cards.

Related functions:

d2410_set_t_move_all
d2410_start_move_all
d2410_set_sync_option
d2410_set_sync_stop_mode
d2410_config_CSTA_PIN

Chapter 5 Examples of Program Design

In the chapter, two simple motion control program written by VB and VC are described. The program control axis 0 moving a distance with a trapezoidal speed profile.

Note：Before writing program, two steps have been done as following:

1. PC has installed Visual C++ 6.0 or Visual Basic 6.0;
2. DMC2410 has been installed exactly, and the hardware has been test by Motion2410 Demo software.

5.1 Example of Visual Basic 6.0

（1）Create a new folder, such as E:\test1;

（2）Open Visual Basic 6.0;

（3）Create a new project with “Standard EXE”, add 2 buttons “START” and “STOP” in the form. Set “START” button name as CB_Start, and “STOP” button name as CB_Stop. See figure 5-1;

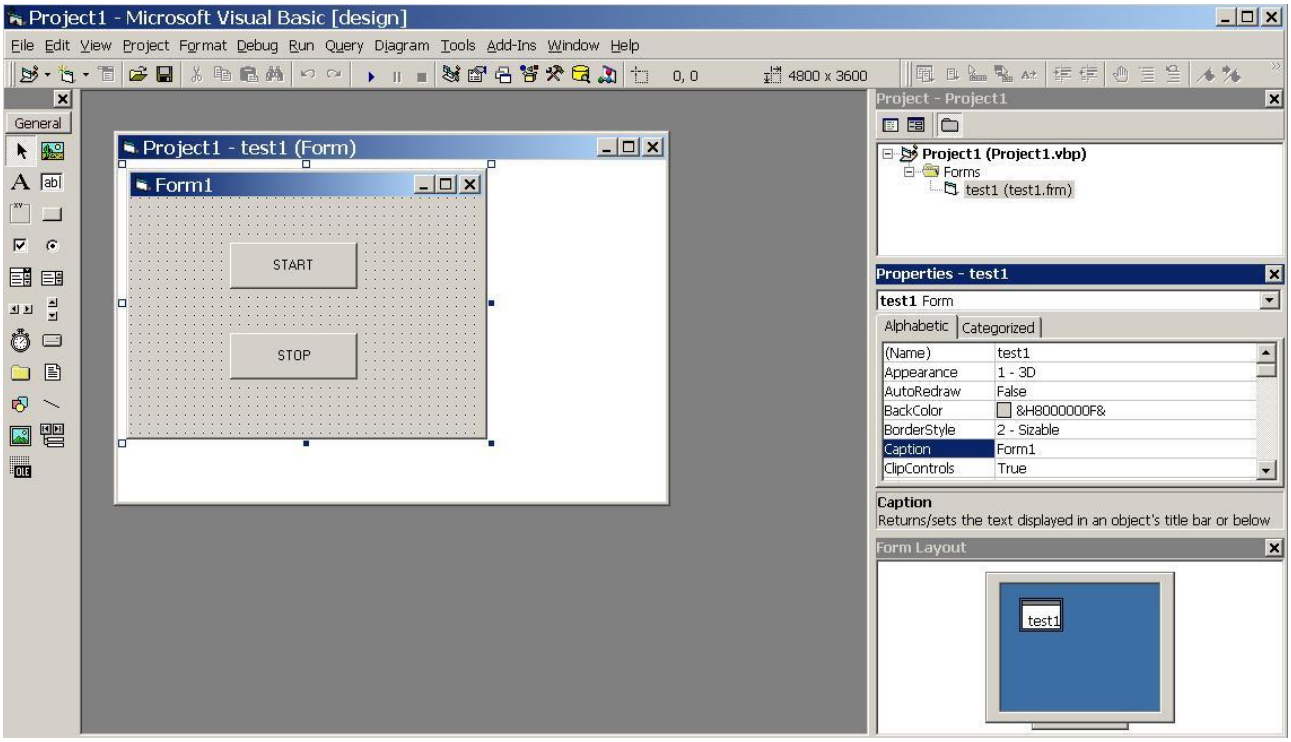


Figure 5-1 Creates a new form

（4）Save the project into folder of E:\test1;

（5）Copy DMC2410.bas from the folder “driver” of Leadtech DMC2410 CD to the folder “test1” ;

（6）Click “Project”->“Add Module”->“Existing”, choose DMC2410.bas in folder “test1”, and add it into the project. See figure 5-2;

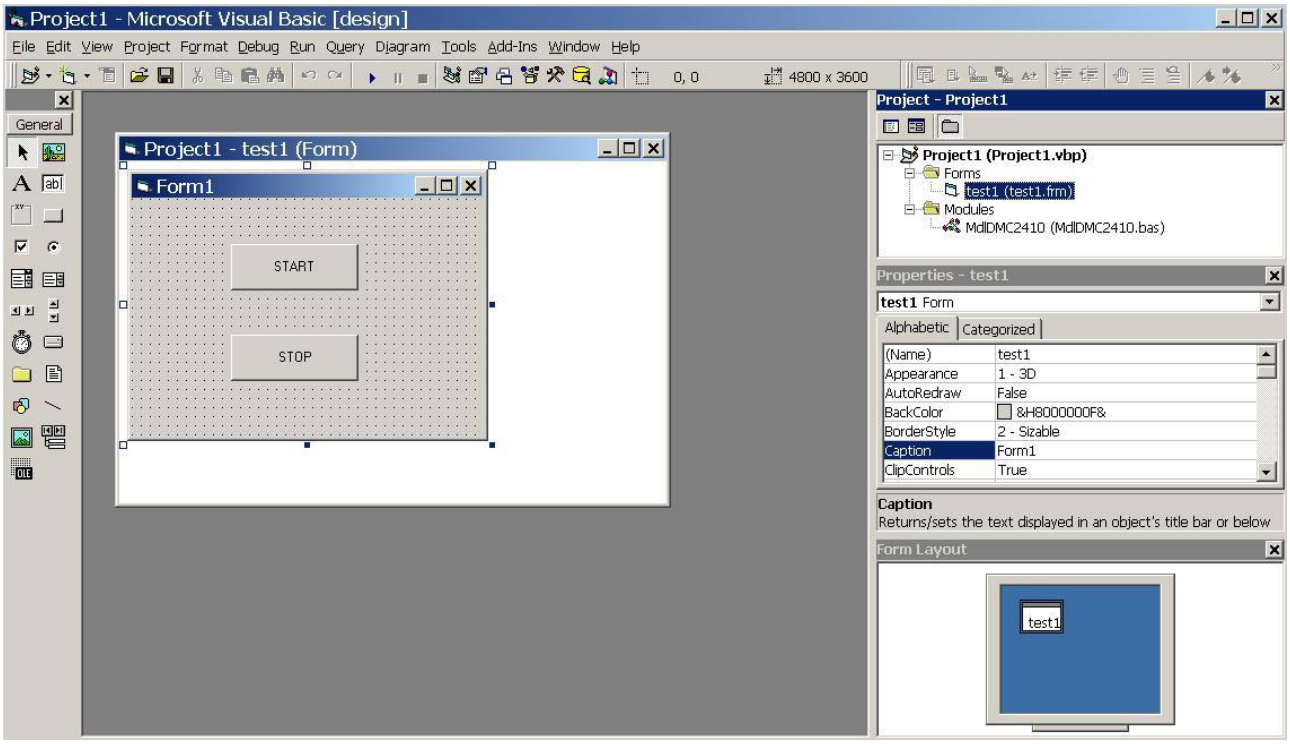


Figure 5-2 Add Module of DMC2410 into the project

（7）Double click the Form, add code in “Form_Load” function as below:

d2410_board_init

Choose “UnLoad” event, add code in “Form_Unload” function as below:

d2410_board_close

Double click “START” button, add codes in “CB_Start_Click” function as below:

d2410_set_profile 0,500,5000, 0.01,0.01

d2410_t_pmove 0,200000,0

Double click “STOP” button, add code in“CB_Stop_Click” function as below:

d2410_decel_stop 0, 0.01

See figure 5-3;

（8）Press button of VB to execute the program. Press “START” button, the axis 0 will move 200000 pulses. When the axis 0 is moving, the axis will stop after the “STOP” button is pressed. See figure 5-4;

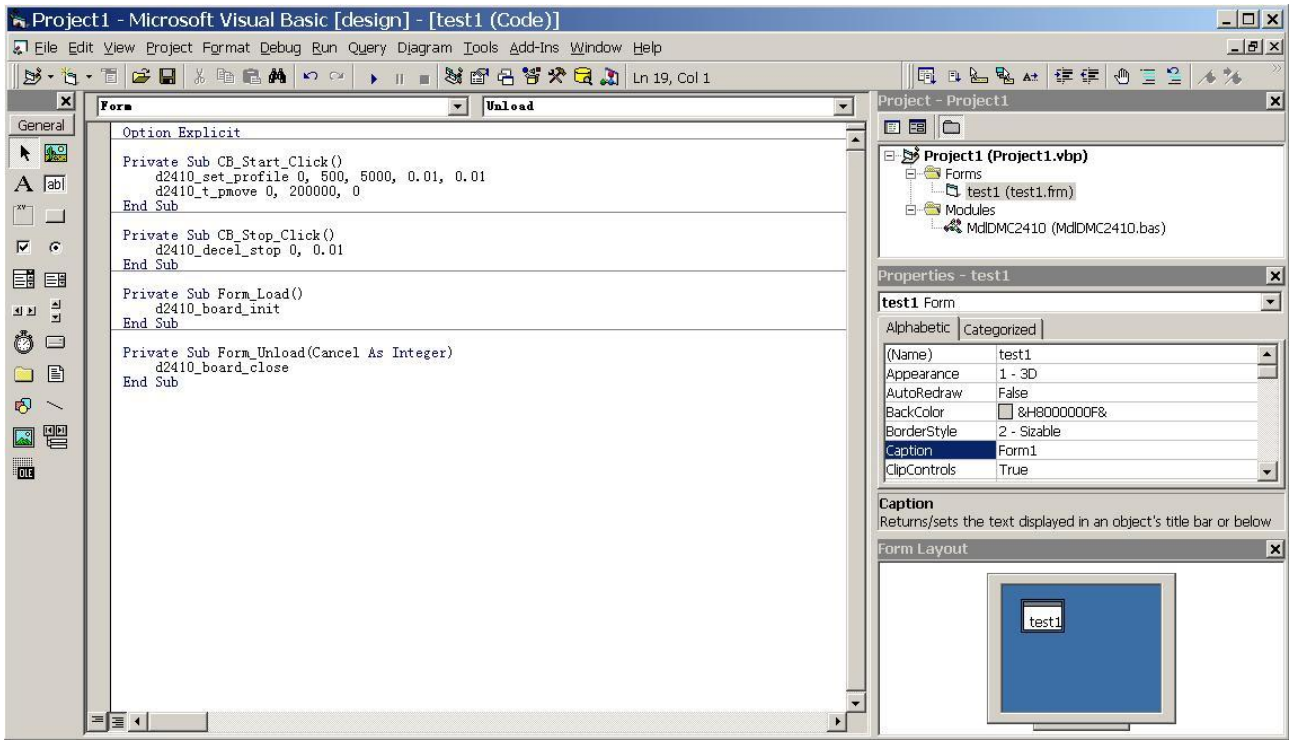


Figure 5-3 Add function of DMC2410 in program

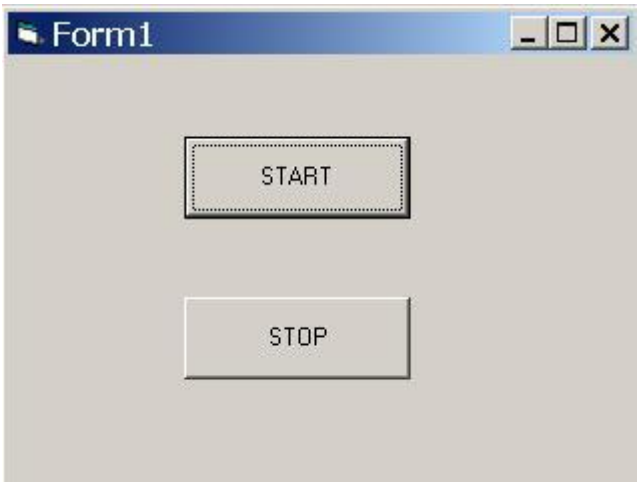


Figure 5-4 The page of the VB program

5.2 Example of Visual C++ 6.0

（1）Open Visual C++ 6.0;

（2）Create a new project;

（3）Choose MFC Apprized(exe);

（4）Set the location of saving the project, such as: E:\ ;

（5）Input the name of the project, such as: test1. See figure 5-5;

（6）Choose “Dialog based” type of application, press “Finish” button to create a project;

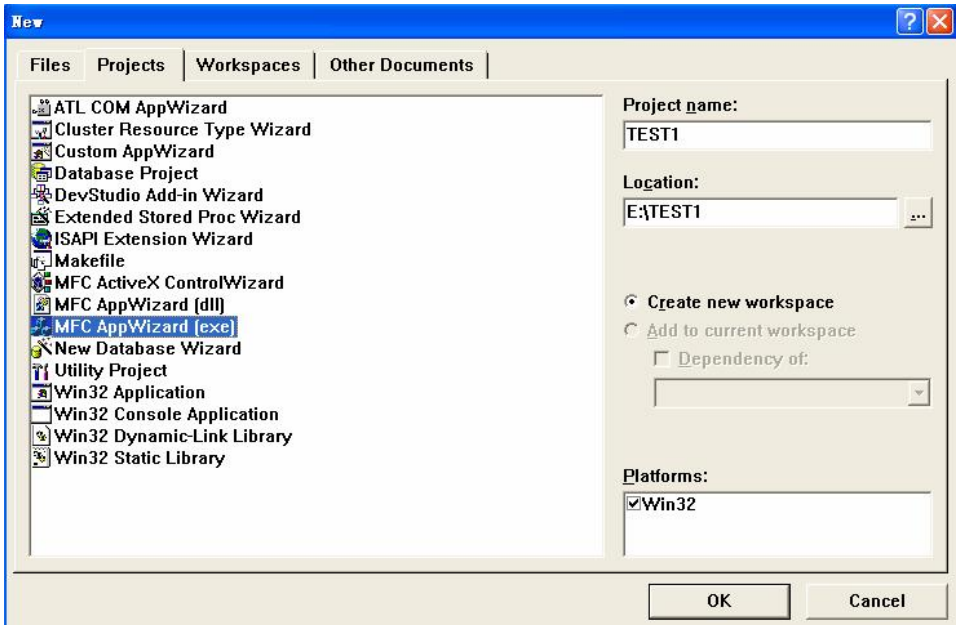


Figure 5-5 Creates a new project

（7） Add 2 buttons “START” and “STOP” in the dialog page. Set “START” button name as IDC_BUTTON_Start, and “STOP” button name as IDC_BUTTON_Stop. See figure 5-6;

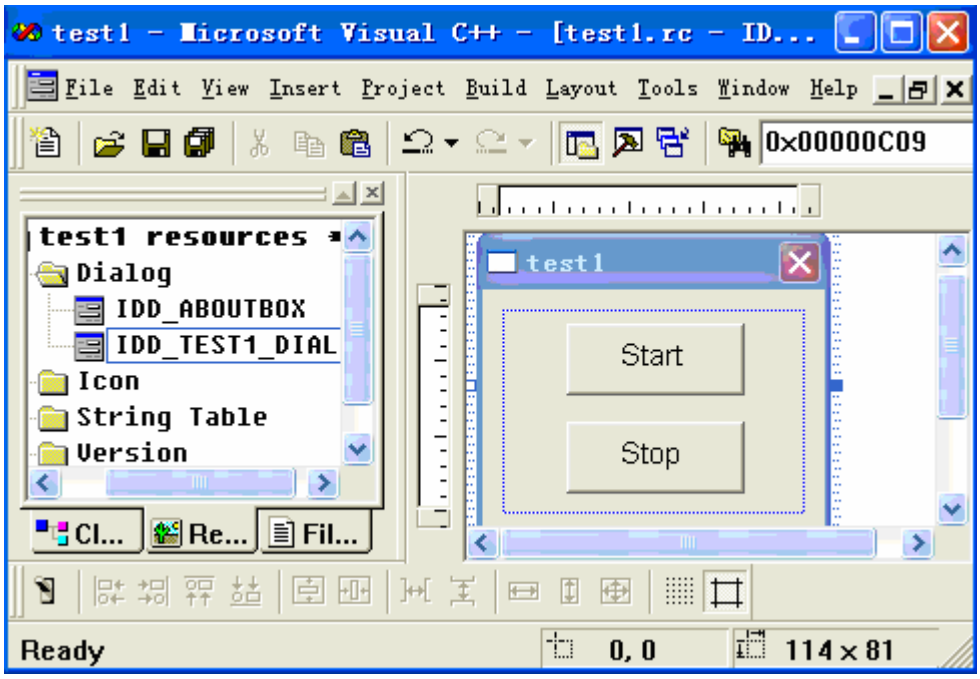


Figure 5-6 Add 2 buttons

（8）Copy DMC2410.h and DMC2410.lib from the folder “driver” of Leadtech DMC2410 CD to the folder “E:\test1” ;

（9）Click “Project”->“Add to project”->“Files”, Choose DMC2410.lib to add in the project;

（10）Open test1.cpp, add #include “DMC2410.h” at the head of program, as figure 5-7 shown;

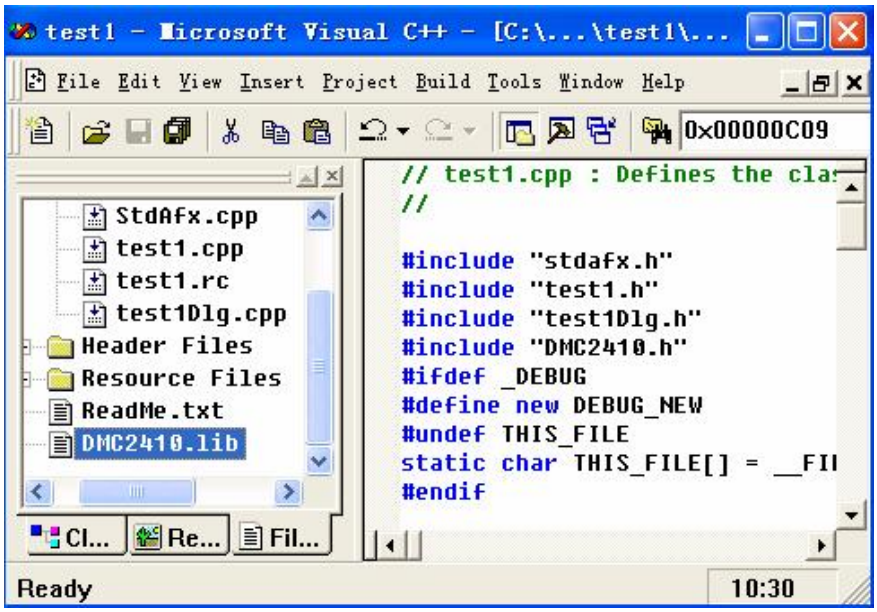


Figure 5-7 Add head file

（11）Add code in function of CTest1Dlg::OnInitDialog() as below:

d2410_board_init();

See figure 5-8;

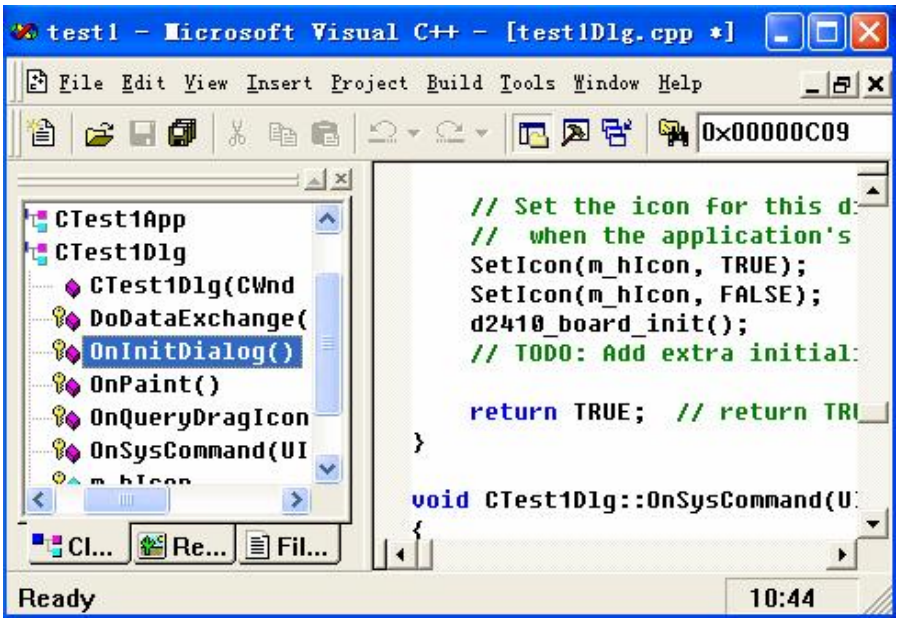


Figure 5-8 Add init function in program

（12）Add a class function “OnCancel” in Ctest1Dlg, and write codes in OnCancel function as below:

d2410_board_close();

CDialog::OnCancel();

See figure 5-9;

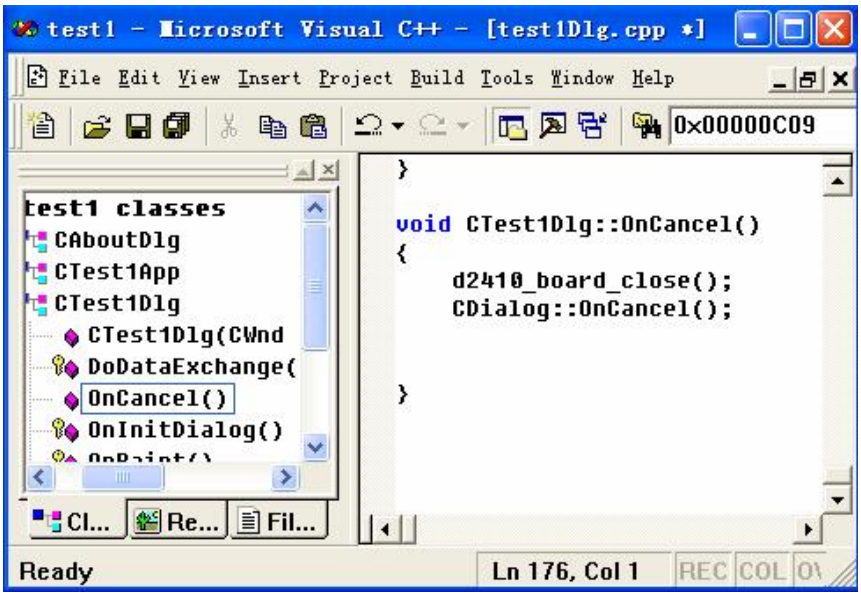


Figure 5-9 Add OnCancel function in program

（13）Double click “START” button, write codes in function of OnBUTTONStart() as below:

d2410_set_profile(0,500,5000, 0.01,0.01);

d2410_t_pmove(0,200000,0);

Double click “STOP” button, write code in function of OnBUTTONStop() as below:

d2410_decel_stop(0,0.01);

See figure 5-10;

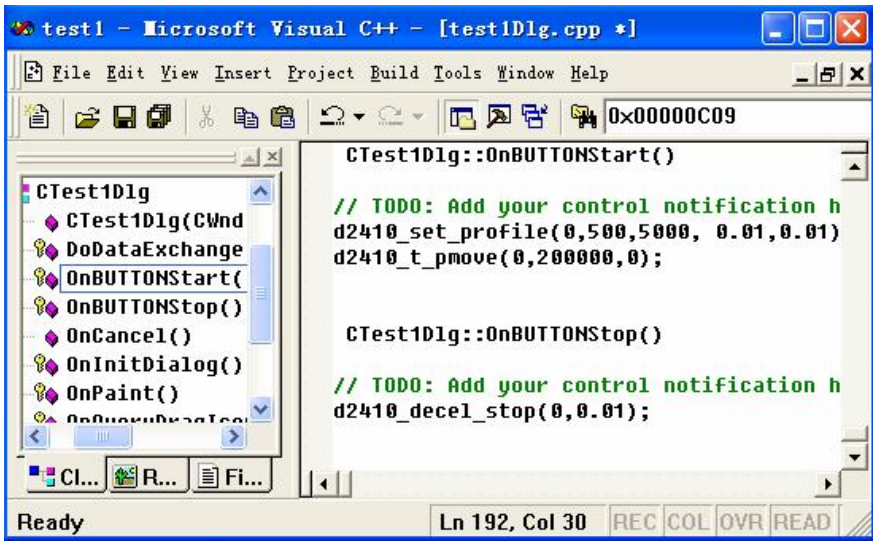


Figure 5-10 Add DMC2410 function into program

（14）Build and execute the program. Press “START” button, the axis 0 will move 200000 pulses. When the axis 0 is moving, the axis will stop after the “STOP” button is pressed. See figure 5-11.

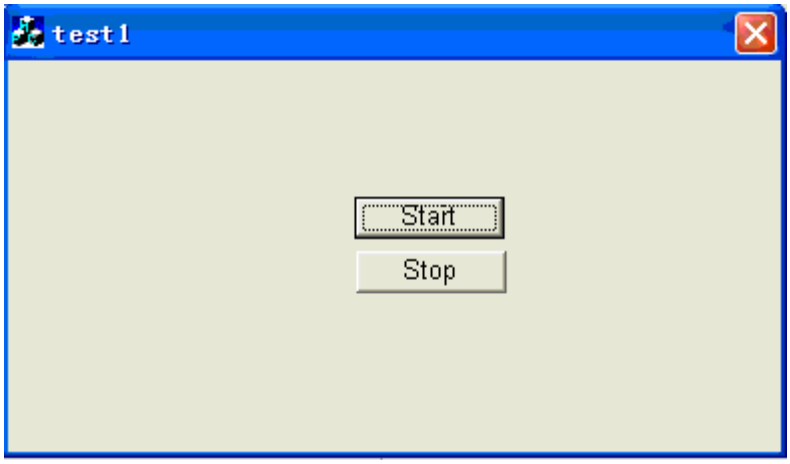


Figure 5-11 The page of the program

Chapter 6 Functions of DMC2410

DMC2410 motion control card has 15 types and 92 functions. These functions are described in detail in this chapter.

The list of DMC2410 functions is shown in table 6-1.

Table 6-1 DMC2410 functions

	Function Name	Description
Initialization functions	d2410_board_init	Card initialization
	d2410_board_close	Card Close
Setup pulse mode	d2410_set_pulse_outmode	setup command pulse output mode
Setup speed functions	d2410_set_profile	setup trapezoidal speed profile
	d2410_set_s_profile	setup S-curve speed profile with pulses
	d2410_set_st_profile	setup S-curve speed profile with time
	d2410_change_speed	Change speed on the fly
	d2410_read_current_speed	Read current speed
	d2410_variety_speed_range	Setup speed range
	d2410_decel_stop	Decelerate to stop
	d2410_imd_stop	Immediately stop
	d2410_emg_stop	Emergency stop all axes
Single axis motion control functions	d2410_t_pmove	Single axis moves a distance with trapezoidal curve speed （symmetrical acceleration and deceleration curve）
	d2410_ex_t_pmove	Single axis moves a distance with trapezoidal curve speed （ asymmetrical acceleration and deceleration curve）
	d2410_s_pmove	Single axis moves a distance with S-curve speed （ symmetrical acceleration and deceleration curve）
	d2410_ex_s_pmove	Single axis moves a distance with S-curve speed （ asymmetrical acceleration and deceleration curve）
	d2410_t_vmove	Single axis continuous moves with trapezoidal curve speed
	d2410_s_vmove	Single axis continuous moves with S-curve speed
Linear, Circle interpolation functions	d2410_set_vector_profile	setup interpolation vector speed profile
	d2410_t_line2	2-axis linear interpolation for any 2 axes with symmetric trapezoidal curve profile
	d2410_t_line3	3-axis linear interpolation for any 3 axes with symmetric trapezoidal curve profile
	d2410_t_line4	4-axis linear interpolation on one card with symmetric trapezoidal curve profile

	d2410_arc_move	any 2 axes of 4 axes circular interpolation in absolute position
	d2410_rel_arc_move	any 2 axes of 4 axes circular interpolation in relative position
Home functions	d2410_config_HOME_pin_logic	Set home signal logic configuration
	d2410_config_home_mode	Configure home mode
	d2410_home_move	Begin a home return action
Manual pulse generator control functions	d2410_set_handwheel_inmode	Setup input mode of manual pulse generator
	d2410_handwheel_move	Manual pulse generator begins to control an axis
Motion status functions	d2410_check_done	Check status of an axis' current motion
	d2410_prebuff_status	Read the status of command's pre-buffer
	d2410_get_rsts	get status of an axis's external signals
	d2410_axis_io_status	Read an axis' motion status and its I/O
	d2410_axis_status	Read an axis' primary status
Special I/O control functions	d2410_config_SD_PIN	Configure Slow-down signal's level and mode
	d2410_config_PCS_PIN	Configure position change signal
	d2410_config_INP_PIN	Configure servo motor's in-position signal
	d2410_config_ERC_PIN	Configure servo motor's deflection counter clear signal ERC)
	d2410_config_ALM_PIN	Configure motor's alarm signal ALM
	d2410_config_LTC_PIN	Configure position capture signal LTC
	d2410_config_EL_MODE	Configure end-limit signal EL
	d2410_write_SEVON_PIN	Output servo motor on signal SEVON
	d2410_read_SEVON_PIN	Read servo motor on signal SEVON
	d2410_read_RDY_PIN	Read servo motor ready signal RDY
	d2410_write_ERC_PIN	Output servo motor's deflection counter clear signal ERC
	d2410_config_EMG_PIN	Configure emergent stop signal EMG
Command counter functions	d2410_get_position	Get an axis' current command position
	d2410_set_position	Setup an axis' command position
	d2410_reset_target_position	Change target position on the fly
General purpose I/O control functions	d2410_read_inbit	Read status of input pin
	d2410_write_outbit	Write output pin
	d2410_read_outbit	Read status of output pin
	d2410_read_inport	Read status of input port
	d2410_read_outport	Read status of output port
	d2410_write_outport	Write output port
Encoder count functions	d2410_counter_config	setup encoder's counter mode
	d2410_config_EZ_PIN	Configure encoder's EZ signal
	d2410_get_encoder	Get encoder's position
	d2410_set_encoder	Setup encoder's new position
	d2410_config_latch_mode	Configure position latch mode

	d2410_get_latch_value	Get the value of latched position
	d2410_get_latch_flag	Get flags of position latch
	d2410_reset_latch_flag	Reset flags of position latch
	d2410_get_counter_flag	Read flags of position counter
	d2410_reset_counter_flag	Reset flags of position counter
	d2410_reset_clear_flag	Reset the clear flag of position counter
	d2410_triger_chunnel	Select trigger signal channel of latching all axes position
	d2410_set_speaker_logic	Setup speaker and LED's output logic level
Position compare functions	d2410_config_CMP_PIN	Configure position compare function
	d2410_read_CMP_PIN	Read position compare signal CMP
	d2410_write_CMP_PIN	Write position compare signal CMP
	d2410_config_comparator	Configure position compare condition
	d2410_set_comparator_data	Set position compare values
Motion with scale unit, error compensate	d2410_set_equiv	Setup an axis' scale factor
	d2410_get_equiv	Read an axis' scale factor
	d2410_get_position_unitmm	Read an axis' position with scale unit
	d2410_set_position_unitmm	Set an axis' position with scale unit
	d2410_read_current_speed_unitmm	Read an axis' speed with scale unit
	d2410_set_encoder_equiv	Setup a encoder's scale factor
	d2410_get_encoder_equiv	Read a encoder's scale factor
	d2410_get_encoder_unitmm	Get an axis' encoder position with scale unit
	d2410_set_encoder_unitmm	Setup an axis' encoder position with scale unit
	d2410_arc_move_unitmm	Arc motion with scale unit in absolute position mode
	d2410_rel_arc_move_unitmm	Arc motion with scale unit in relative distance mode
	d2410_pulse_loop	Auto compensate an axis' error of command position vs. encoder position
Multi-axis synchronous motion	d2410_set_t_move_all	Setup multi-axis synchronous motion
	d2410_config_CSTA_PIN	Configure synchronous start signal mode
	d2410_start_move_all	Start multi-axis synchronous motion
	d2410_simultaneous_stop	Synchronously stop multi-axes motion
	d2410_set_sync_option	Setup multi-axis synchronous motion mode
	d2410_set_sync_stop_mode	Setup multi-axis synchronous stop mode

6.1 Initialization functions

WORD d2410_board_init(void)

Description：Allocate system resource to the card, and initialize it.

Argument：Void

Return value：Quantity of DMC2410 boards, 0～8, if no board, return 0

void d2410_board_close(void)

Description：close DMC2410 card and release the PC's resource.

Argument：Void

Return value：Null

6.2 Setup pulse mode function

void d2410_set_pulse_outmode(WORD axis, WORD outmode)

Description：Setup command pulse output mode

Argument：Axis specifies axis' number.

outmode Select command pulse output mode, 0~3 are single pulse output modes, 4~5 are dual pulse output modes. Detail as table 6-2 shown.

Return value：Null

Table 6-2 Command pulse output mode

Mode	Positive direction pulse		Negative direction pulse	
	OUT Pin	DIR Pin	OUT Pin	DIR Pin
0		High level		Low level
1		High level		Low level
2		Low level		High level
3		Low level		High level
4		High level	High level	
5		Low level	Low level	

6.3 Setup speed functions

void d2410_set_profile(WORD axis,double Min_Vel,double Max_Vel,double Tacc,double Tdec)

Description：Setup trapezoidal speed profile

Argument：Axis axis number

Min_Vel minimum velocity, when the axis starts or stop, unit:: pps

Max_Vel maximum velocity, unit: pps

Tacc acceleration time，unit: second

Tdec deceleration time, unit: second

Return value：Null

void d2410_set_s_profile(WORD axis,double Min_Vel,double Max_Vel, double Tacc, double Tdec, long Sacc,long Sdec)

Description：Setup S-curve speed profile with pulses

Argument：axis axis number

	Min_Vel	minimum velocity, when the axis start or stop, unit:: pps
	Max_Vel	maximum velocity, unit: pps
	Tacc	acceleration time，unit: second
	Tdec	deceleration time, unit: second
	Sacc	pulse number in accelerating S-curve part, see figure 5-9.
	Sdec	pulse number in decelerating S-curve part, see figure 5-9.

Return value：Null

void d2410_set_st_profile(WORD axis,double Min_Vel, double Max_Vel,double Tacc, double Tdec, double Tsacc, double Tsdec)

Description：setup S-curve speed profile with time

Argument：axis		axis number
	Min_Vel	minimum velocity, when the axis start or stop, unit:: pps
	Max_Vel	maximum velocity, unit: pps
	Tacc	acceleration time，unit: second
	Tdec	deceleration time, unit: second
	Tsacc	time of accelerating S-curve part, it is less half of Tacc.
	Tsdec	time of decelerating S-curve part, it is less half of Tdec.

Return value：Null

void d2410_change_speed(WORD axis,double Curr_Vel)

Description：Change speed on the fly. This function only can be used at signal axis motion. Before used it, the d2410_variety_speed_range function should be used to setup the change speed function to enable, and setup the range of change speed.

Argument：	axis	axis number
	Curr_Vel	new velocity, unit: pps

Return value：Null

void d2410_variety_speed_range(WORD axis,WORD chg_enable ,double Max_Vel)

Description：Setup speed range and setup change speed function to enable

Argument：	axis	axis number
	chg_enable	0 is to disable, 1 is to enable the change speed function
	Max_Vel	the maximum velocity, unit: pps

Return value：Null

double d2410_read_current_speed(WORD axis)

Description：Read current speed

Argument： axis axis number

Return value：current speed of the axis, unit: pps

void d2410_decel_stop(WORD axis,double Tdec)

Description：As the function is executed, the axis begins to decelerate, when the speed arrives to the minimum velocity, the axie stop.

Argument： axis axis number

Tdec Time of deceleration, unit second.

Return value：Null

void d2410_imd_stop(WORD axis)

Description：Immediately stop, not have deceleration process.

Argument： axis axis number

Return value：Null

void d2410_emg_stop(void)

Description：Emergency stop all axes

Argument：Null

Return value：Null

6.4 Single axis motion control functions

void d2410_t_pmove(WORD axis, long Dist,WORD posi_mode)

Description：Single axis moves a distance with trapezoidal curve speed（symmetrical acceleration and deceleration curve）

Argument：axis axis number

Dist distance or position, unit: pulse

posi_mode coordinates mode：0 is relative, 1 is absolute

Return value：Null

void d2410_ex_t_pmove(WORD axis,long Dist,WORD posi_mode)

Description：Single axis moves a distance with trapezoidal curve speed（asymmetrical acceleration and deceleration curve）

Argument：axis axis number

Dist distance or position, unit: pulse

posi_mode coordinates mode：0 is relative, 1 is absolute

Return value：Null

void d2410_s_pmove(WORD axis,long Dist,WORD posi_mode)

Description：Single axis moves a distance with S-curve speed（symmetrical acceleration and deceleration curve）

Argument：axis axis number
Dist distance or position, unit: pulse
posi_mode coordinates mode：0 is relative, 1 is absolute
Return value：Null

void d2410_ex_s_pmove(WORD axis,long Dist,WORD posi_mode)

Description：Single axis moves a distance with S-curve speed（asymmetrical acceleration and deceleration curve）
Argument：axis axis number
Dist distance or position, unit: pulse
posi_mode coordinates mode：0 is relative, 1 is absolute
Return value：Null

void d2410_t_vmove(WORD axis,WORD dir)

Description：Single axis continuous moves with trapezoidal curve speed
Argument：axis axis number
dir direction of moving, 0 is negative, 1 is positive direction
Return value：Null

void d2410_s_vmove(WORD axis,WORD dir)

Description：Single axis continuous moves with S-curve speed
Argument：axis axis number
dir direction of moving, 0 is negative, 1 is positive direction
Return value：Null

6.5 Linear, Circle interpolation functions

void d2410_set_vector_profile(double Min_Vel,double Max_Vel,double Tacc, double Tdec)

Description：Setup interpolation vector speed profile
Argument： Min_Vel minimum velocity, when the axis start or stop, unit:: pps
Max_Vel maximum velocity, unit: pps
Tacc acceleration time，unit: second
Tdec deceleration time, unit: second
Return value：Null

void d2410_t_line2(WORD axis1,long Dist1,WORD axis2,long Dist2,WORD posi_mode)

Description：2-axis linear interpolation for any 2 axes with symmetric trapezoidal curve profile

Argument：axis1 first axis' number
axis2 second axis' number
Dist1 distance or position of first axis, unit: pulse
Dist2 distance or position of second axis, unit: pulse
posi_mode coordinates mode：0 is relative, 1 is absolute
Return value：Null

void d2410_t_line3(WORD *axis,long Dist1,long Dist2,long Dist3,WORD posi_mode)

Description：3-axis linear interpolation for any 3 axes with symmetric trapezoidal curve profile
Argument：axis the pointer of axes' number list
Dist1 distance or position of first axis, unit: pulse
Dist2 distance or position of second axis, unit: pulse
Dist3 distance or position of third axis, unit: pulse
posi_mode coordinates mode：0 is relative, 1 is absolute
Return value：Null

void d2410_t_line4(WORD cardno, long Dist1, long Dist2, long Dist3, long Dist4,WORD posi_mode)

Description：4-axis linear interpolation on one card with symmetric trapezoidal curve profile
Argument：cardno specify card's number, the range is 0~7.
Dist1 distance or position of first axis, unit: pulse
Dist2 distance or position of second axis, unit: pulse
Dist3 distance or position of third axis, unit: pulse
Dist4 distance or position of fourth axis, unit: pulse
posi_mode coordinates mode：0 is relative, 1 is absolute
Return value：Null

void d2410_arc_move(WORD *axis,long *target_pos,long *cen_pos, WORD arc_dir)

Description：any 2 axes of 4 axes circular interpolation in absolute position mode
Argument：axis the pointer of axes' number list
target_pos the pointer of target position's list, unit: pulse
cen_pos the pointer of position's list of center of the circle, unit: pulse
arc_dir arc direction: 0 is clockwise, 1 is anti-clockwise
Return value：Null

void d2410_rel_arc_move(WORD *axis,long *rel_pos,long *rel_cen, WORD arc_dir)

Description：any 2 axes of 4 axes circular interpolation in relative distance mode
Argument： axis the pointer of axes' number list

	rel_pos	the pointer of target distance's list, unit: pulse
	rel_cen	the pointer of distance's list of center of the circle, unit: pulse
	arc_dir	arc direction: 0 is clockwise, 1 is anti-clockwise

Return value：Null

6.6 Home functions

void d2410_set_HOME_pin_logic(WORD axis,WORD org_logic,WORD filter)

Description：Set home signal logic configuration

Argument：	axis	axis number
	org_logic	0—low level of Home signal is active, 1—high level is active
	filter	0 is the filter disable, 1 is the filter enable

Return value：Null

void d2410_config_home_mode(WORD axis,WORD mode,WORD EZ_count)

Description：Configure home mode

Argument：	axis	axis number
	mode	0 is only use home signal, 1 is use home and EZ signals
	EZ_count	the times that EZ signal occurs, the range is 1~16. It is used in mode=1. After home signal is active, when EZ signal occur EZ_count times, then axis stop.

Return value：Null

void d2410_home_move(WORD axis,WORD home_mode,WORD vel_mode)

Description：Begin a home return action

Argument：	axis	axis number
	home_mode	1 is positive direction home, 2 is negative direction home
	vel_mode	0 is home with min velocity, 1 is home with max velocity

Return value：Null

6.7 Manual pulse generator control functions

void d2410_set_handwheel_inmode(WORD axis, WORD inmode, WORD count_dir)

Description：Setup input mode of manual pulse generator

Argument：	axis	specify axis number
	inmode	input mode：
		0—A、B phase signals
		1—A、B phase signals, 2 times frequency to counting
		2—A、B phase signals, 4 times frequency to counting
		3—difference pulse singal

count_dir counter's direction：
0—default direction
1—inverse direction of default

Return value：Null

void d2410_handwheel_move(WORD axis,double vh)

Description：startup Manual pulse generator to control axis moving

Argument：axis axis number

vh setup maximum input frequecy, unit: pps

Return value：Null

6.8 Motion status functions

WORD d2410_check_done(WORD axis)

Description：Check status of an axis' current motion

Argument：axis axis number

Return value：0—the axis is moving, 1—the axis is stop

WORD d2410_prebuff_status(WORD axis)

Description：Read the status of command's pre-buffer

Argument：axis axis number

Return value：0—pre-buffer is empty, 1—pre-buffer is full

DWORD d2410_get_rsts(WORD axis)

Description：get status of an axis's external signals

Argument：axis specify axis number

Return value：see table 6-3

Table 6-3 Return value of d2410_get_rsts()

Bit of Return value	Name	Return value description
0~3	reserve	
4	CSD	1 means multi-axis simultaneous deceleration signal (CSD) is ON
5	STA	1 means multi-axis simultaneous start signal (STA) is ON
6	STP	1 means multi-axis simultaneous stop signal (STP) is ON
7	EMG	1 means emergency stop signal (EMG) is ON
8	PCS	1 means position change signal (PCS) is ON
9	ERC	1 means servo motor's deflection counter clear signal (ERC) is ON
10	EZ	1 means encoder's index signal (EZ) is ON
11	PA	1 means external pulse control signal PA is ON
12	PB	1 means external pulse control signal PB is ON
13	reserve	

14	SD	1 means slow-down signal (SD) is ON
15	INP	1 means servo motor in-position signal (INP) is ON
16	DIR	1 means pulse direction is negative, 0 is positive
17~31	reserve	

WORD d2410_axis_io_status(WORD axis)

Description：Read an axis' motion status and its I/O

Argument：axis specify axis number

Return value：see table 6-4

Table 6-4 Return value of d2410_axis_io_status( )

Bit of Return value	Name	Return value description
0~7	reserve	
8	FU	1 means the axis is accelerating
9	FD	1 means the axis is decelerating
10	FC	1 means the axis is moving with minimum speed
11	ALM	1 means the motor alarm signal (ALM) is ON
12	+EL	1 means end-limit in positive direction signal (+EL) is ON
13	-EL	1 means end-limit in negative direction signal (-EL) is ON
14	ORG	1 means home signal (ORG) is ON
15	SD	1 means slow-down signal (SD) is ON

WORD d2410_axis_status(WORD axis)

Description：Read an axis' primary status

Argument：axis specify axis number

Return value：see table 6-5

Table 6-5 Return value of d2410_axis_status( )

Bit of Return value	Name	Return value description
0	SSCM	1 means a start command is written; when the operation is stop, the return value is 0
1	SRUN	1 means pulse output is start, 0 means pulse output is stop
2	SENI	1 means motion stopped and a interrupt flag is ON
3	SEND	1 means motion is stop, a start command is written, the return value is 0
4	SERR	1 means an error interrupt occurs
5	SINT	1 means a event interrupt occurs
6~12	reserve	
13	SEOR	When a position override can't be executed, this signal changes to 1. After the main status is read, it changed to 0
14	SPRF	1 means the pre-register for the subsequent operation data is full
15	reserve	

6.9 Special I/O control functions

void d2410_config_SD_PIN(WORD axis,WORD enable, WORD sd_logic,WORD sd_mode)

Description：Configure Slow-down signal's level and mode

Argument： axis axis number

enable 0—disable, 1—enable

sd_logic 0—low level of SD signal is active, 1—high level is active

sd_mode Setup slow-down mode:

0—The axis decelerate to minimum velocity, if SD signal turns OFF, the axis will accelerate to maximum velocity

1—The axis decelerate to minimum velocity and stop, if SD signal turns OFF before axis stopped, the axis will accelerate to maximum velocity

2—The axis decelerate to minimum velocity, even if the SD signal is turned OFF before the velocity is not changed to minimum velocity

3—The axis decelerate to minimum velocity and stopped, even if the SD signal is turned OFF before the axis stopped

void d2410_config_PCS_PIN(WORD axis,WORD enable,WORD pcs_logic)

Description：Configure position change signal PCS

Argument：axis axis number

enable 0—disable, 1—enable

pcs_logic 0—low level of PCS signal is active, 1—high level is active

Return value：Null

void d2410_config_INP_PIN(WORD axis,WORD enable,WORD inp_logic)

Description：Configure servo motor's in-position signal INP

Argument：axis axis number

enable 0—disable, 1—enable

inp_logic 0—low level of INP signal is active, 1—high level is active

Return value：Null

void d2410_config_ERC_PIN(WORD axis,WORD enable,WORD erc_logic,WORD erc_width,WORD erc_off_time)

Description：Configure servo motor's deflection counter clear signal (ERC) and its mode

Argument：axis axis number

enable setup enable and its mode：

0—Do not output ERC signal

1—After received EL, ALM or EMG signal, output ERC signal

automatically.
2—After received ORG signal, output ERC signal automatically
3—After received EL, ALM, EMG or ORG signal, output ERC signal automatically.

erc_logic 0—low level of ERC signal is active, 1—high level is active
erc_width ERC signal width.
0—12 us, 1—102 us, 2—409 us, 3—1.6 ms
4—13 ms, 5—52 ms, 6—104 ms, 7—level output
erc_off_time The time of ERC off before next ERC come.
0—0 us, 1—12 us, 2—1.6 ms, 3—104 ms

Return value：Null

void d2410_config_ALM_PIN(WORD axis,WORD alm_logic,WORD alm_action)

Description：Configure motor's alarm signal (ALM) and its mode
Argument：axis axis number
alm_logic ALM signal input level. 0—low level is active, 1—high level is active
alm_action ALM signal action：0—axes stop immediately, 1—slow-down first, then stop
Return value：Null

void d2410_config_LTC_PIN(WORD axis,WORD ltc_logic, WORD ltc_mode)

Description：Configure position capture signal (LTC)
Argument：axis axis number
ltc_logic 0—low level of LTC signal is active, 1—high level is active
ltc_mode reserve
Return value：Null

void d2410_config_EL_MODE(WORD axis,WORD el_mode)

Description：Configure end-limit (EL) signal and its mode
Argument：axis axis number
el_mode set the level and mode of the EL signal
0—low level active, axis stop immediately
1—low level active, axis slow-down and stop
2—high level active, axis stop immediately
3—high level active, axis slow-down and stop
Return value：Null

void d2410_write_SEVON_PIN(WORD axis, WORD on_off);

Description：Output servo motor on signal (SEVON)

Argument：axis axis number

on_off set the level of output signal：0—low, 1—high

Return value：Null

Note: Switch S1 of DMC2410 can setup the output port logic level of SEVON1～SEVON4. When S1 is selected at ”ON” position, write “0” to a output bit, the port output low level; write “1” to a output bit, the port output high level. When S1 is selected at “OFF” position, write “0” to an output bit, the port output high level; write “1” to an output bit, the port output low level.

int d2410_read_SEVON_PIN(WORD axis)

Description：Read servo motor on signal (SEVON)

Argument：axis axis number

Return value：0—low level, 1—high level

int d2410_read_RDY_PIN(WORD axis)

Description：Read servo motor ready signal (RDY)

Argument：axis axis number

Return value：0—low level, 1—high level

void d2410_write_ERC_PIN(WORD axis, WORD sel)

Description：Output servo motor‘s deflection counter clear signal (ERC)

Argument：axis axis number

sel 0—reset ERC signal, 1—set ERC signal

Return value：Null

void d2410_config_EMG_PIN(WORD cardno, WORD enable,WORD emg_logic)

Description：Configure emergent stop signal (EMG) to stop all axes in single card

Argument：cardno card number

enable 0—disable, 1—enable

emg_logic 0—low level of EMG signal is active, 1—high level is active

Return value：Null

6.10 Command counter functions

long d2410_get_position(WORD axis)

Description：Get an axis’ current command position

Argument：axis axis number

Return value：Current command position, unit: pulse

void d2410_set_position(WORD axis,long current_position)

Description：Setup an axis' command position

Argument：axis axis number

current_position absolute position, unit:: pulse

Return value：Null

void d2410_reset_target_position(WORD axis,long dist)

Description：Change target position on the fly in relative distance mode

Argument：axis axis number

dist relative distance, unit: pulse

Return value：Null

6.11 General purpose I/O control functions

int d2410_read_inbit(WORD cardno, WORD bitno)

Description：Read status of input pin

Argument：cardno card number

bitno pin number, range: 1~20

Return value：0—low level; 1—high level

void d2410_write_outbit (WORD cardno, WORD bitno,WORD on_off)

Description：Write output pin

Argument：cardno card number

bitno pin number, range: 1~20

on_off 0—output low level, 1—output high level

Return value：Null

Note: Switch S1 of DMC2410 can setup the output port logic level of OUT1～OUT12. When S1 is selected at "ON" position, write "0" to a output bit, the port output low level; write "1" to a output bit, the port output high level. When S1 is selected at "OFF" position, write "0" to an output bit, the port output high level; write "1" to an output bit, the port output low level.

int d2410_read_outbit(WORD cardno, WORD bitno)

Description：Read status of output pin

Argument：cardno card number

bitno pin number, range: 1~20

Return value：0—output low level, 1—output high level

long d2410_read_inport(WORD cardno)

Description：Read status of input port of a DMC2410

Argument：cardno card number

Return value：bit0~bit19 of the return value show the value of input pin IN1~20

long d2410_read_outport(WORD cardno)

Description：Read status of output port of a DMC2410

Argument：cardno card number

Return value：bit0~bit19 of the return value show the value of output pin OUT1~20

void d2410_write_outport(WORD cardno, DWORD port_value)

Description：Write output port

Argument：cardno card number

port_value bit0~bit19 of the written value is the value of output pin OUT1~20. 0—output low level, 1—output high level

Return value：Null

6.12 Encoder count functions

void d2410_counter_config(WORD axis,WORD mode)

Description：setup encoder's counter mode

Argument：axis axis number

mode counter mode：

0 Pulse/Direction signals

1 A/B signals

2 two frequency doubling of A/B signals

3 four frequency doubling of A/B signals

Return value：Null

void d2410_config_EZ_PIN(WORD axis,WORD ez_logic, WORD ez_mode)

Description：Configure encoder's EZ signal level and mode

Argument：axis axis number

ez_logic setup EZ signal's level

0—low level is active, 1—high level is active

ez_mode using EZ signal's mode

0—EZ signal disable

1—EZ signal is a reset signal of counter

2—EZ signal is home signal, but do not reset counter

3—EZ signal is home signal, and reset counter

Return value：Null

long d2410_get_encoder(WORD axis)

Description：Get encoder counter's position value

Argument：axis axis number

Return value：position, unit: pulse

void d2410_set_encoder(WORD axis,long encoder_value)

Description：Setup encoder counter's new position

Argument：axis axis number

encoder_value new position value

Return value：Null

void d2410_config_latch_mode(WORD cardno, WORD all_enable)

Description：Configure position latch mode

Argument：cardno axis number

all_enable 0—latch one axis position, 1—latch 4 axes position

Return value：Null

long d2410_get_latch_value(WORD axis)

Description：Get the value of latched position

Argument：axis axis number

Return value：Latched position value, unit: pulse

long d2410_get_latch_flag(WORD cardno)

Description：Get flags of position latch

Argument：cardno card number

Return value：see table 6-6

Table 6-6 Return value of d2410_get_latch_flag ( )

Bit of Return value	Return value description
0	1 means axis 0 has a latch signal
1	1 means axis 1 has a latch signal
2	1 means axis 2 has a latch signal
3	1 means axis 3 has a latch signal
4	1 means axis 0 has a reset signal
5	1 means axis 1 has a reset signal
6	1 means axis 2 has a reset signal
7	1 means axis 3 has a reset signal
8	1 means axis 0's position has latched
9	1 means axis 1's position has latched
10	1 means axis 2's position has latched
11	1 means axis 3's position has latched
12	1 means axis 0's position has reset
13	1 means axis1's position has reset

14	1 means axis 2's position has reset
15	1 means axis 3's position has reset
16～31	reserve

void d2410_reset_latch_flag(WORD cardno)

Description：Reset flags of a position latch

Argument：cardno card number

Return value：Null

long d2410_get_counter_flag(WORD cardno)

Description：Read flags of position counter

Argument：cardno card number

Return value：see table 6-7

Table 6-7 Return value of d2410_get_counter_flag ( )

Bit of Return value	Return value description
0	Borrow bit flag. when axis 0's counter underflow, the flag is triggered
1	Carry flag. when axis 0's counter overflow, the flag is triggered
2	Sign flag. when axis 0's counter overflow, flag is 0; when underflow, flag is 1
3	Counter add/subtract flag. when axis 0's counter add, the flag is1,when counter subtract, the flag is 0
4～7	reserve
8	Borrow bit flag. when axis 1's counter underflow, the flag is triggered
9	Carry flag. when axis 1's counter overflow, the flag is triggered
10	Sign flag. when axis 1's counter overflow, flag is 0; when underflow, flag is 1
11	Counter add/subtract flag. when axis 1's counter add, the flag is1,when counter subtract, the flag is 0
12～15	reserve
16	Borrow bit flag. when axis 2's counter underflow, the flag is triggered
17	Carry flag. when axis 2's counter overflow, the flag is triggered
18	Sign flag. when axis 2's counter overflow, flag is 0; when underflow, flag is 1
19	Counter add/subtract flag. when axis 2's counter add, the flag is1,when counter subtract, the flag is 0
20～23	reserve
24	Borrow bit flag. when axis 3's counter underflow, the flag is triggered
25	Carry flag. when axis 3's counter overflow, the flag is triggered
26	Sign flag. when axis 3's counter overflow, flag is 0; when underflow, flag is 1
27	Counter add/subtract flag. when axis 3's counter add, the flag is1,when counter subtract, the flag is 0
28～31	reserve

void d2410_reset_counter_flag(WORD cardno)

Description：Reset flags of position counter
Argument：cardno card number
Return value：Null

void d2410_reset_clear_flag(WORD cardno)

Description：Reset the clear flag of position counter
Argument：cardno card number
Return value：Null

void d2410_triger_chunnel(WORD cardno, WORD num)

Description：Select trigger signal channel of latching all axes position
Argument：cardno card number
Num the trigger signal channel：
1 — using LTC1 to latch. (Default)
2 — using LTC2 to latch.
Return value：Null

void d2410_set_speaker_logic(WORD cardno, WORD logic)

Description：Setup output logic level of speaker and LED. This output signal is an indication signal of the position capture signal LTC
Argument： cardno card number
logic logic level：0 is low level active, (default)
1 is high level active
Return value：Null

6.13 Position compare functions

void d2410_config_CMP_PIN(WORD axis, WORD cmp1_enable, WORD cmp2_enable WORD CMP_logic)

Description：Configure position compare function
Argument： axis axis number
cmp1_enable 0—position compare is disable. CMP1, CMP2, CMP3, CMP4 can be used as general digital output port
1—position compare is enable. CMP1, CMP2, CMP3, CMP4 are CMP output signals
cmp2_enable 0—position compare is disable. CMP1', CMP2', CMP3', CMP4' can be used as general digital output port
1—position compare is enable. CMP1', CMP2', CMP3', CMP4' are CMP output signals

CMP_logic 0 is low level active, 1 is high level active

Return value：Null

int d2410_read_CMP_PIN(WORD axis)

Description：Read position compare signal CMP

Argument： axis axis number

Return value：1—high level; 0—low level

void d2410_write_CMP_PIN(WORD axis, WORD on_off)

Description：Write position compare signal CMP

Argument： axis axis number

on_off 1—high level; 0—low level

Return value：Null

void d2410_config_comparator(WORD axis, WORD cmp1_condition, WORD cmp2_condition, WORD source_sel, WORD SL_action)

Description：Configure position compare condition of 2 CMP signal

Argument：axis axis number

cmp1_condition trigger condition of CMP

0：CMP disable to position compare

1： value of counter is equal to value of comparator of CMP

2： value of counter is less than value of comparator of CMP

3：value of counter is bigger than value of comparator of CMP

cmp2_condition trigger condition of CMP'

0：CMP' disable to position compare

1： value of counter is equal to value of comparator of CMP'

2： value of counter is less than value of comparator of CMP'

3：value of counter is bigger than value of comparator of CMP'

source_sel configure counter type

0：2 comparators compare to command counter

1： comparator of CMP compare to command counter, comparator of CMP' compare to

encoder counter

SL_action 0：reserve

Return value：Null

void d2410_set_comparator_data(WORD axis,long cmp1_data,long cmp2_data)

Description：Set position compares values

Argument： axis axis number

cmp1_data value of comparator of CMP

cmp2_data value of comparator of CMP'

Return value：Null

6.14 Motion with scale unit, error compensate

int d2410_set_equiv(WORD axis, double new_equiv)

Description：Setup an axis' scale factor

Argument： axis axis number

new_equiv Scale factor, unit: pulse/mm

Return value：reserve

int d2410_get_equiv(WORD axis, double *equiv)

Description：Read an axis' scale factor

Argument： axis axis number

equiv the pointer of the scale factor

Return value：reserve

int d2410_get_position_unitmm(WORD axis, double * pos_by_mm)

Description：Read an axis' position with scale unit

Argument： axis axis number

pos_by_mm the pointer of the axis' position, the position unit:: mm

Return value：reserve

int d2410_set_position_unitmm(WORD axis, double pos_by_mm)

Description：Set an axis' position with scale unit

Argument： axis axis number

pos_by_mm new axis position, unit: mm

Return value：reserve

int d2410_read_current_speed_unitmm(WORD axis, double *current_speed)

Description：Read an axis' speed with scale unit

Argument： axis axis number

current_speed the pointer of the axis' speed, the speed unit:: mm/s

Return value：reserve

int d2410_set_encoder_equiv(WORD axis, double new_equiv)

Description：Setup an encoder's scale factor

Argument： axis axis number

new_equiv new scale factor, unit:: count/mm

Return value：reserve

int d2410_get_encoder_equiv(WORD axis, double *equiv)

Description：Read a encoder's scale factor

Argument： axis axis number

equiv the pointer of the encoder's scale factor, unit:: count/mm

Return value：reserve

int d2410_get_encoder_unitmm(WORD axis, double *encoder_pos_by_mm)

Description：Get an axis' encoder position with scale unit

Argument：axis axis number

encoder_pos_by_mm the pointer of the encoder's position, the position unit:: mm

Return value：reserve

int d2410_set_encoder_unitmm(WORD axis, double encoder_pos_by_mm)

Description：Setup an axis' encoder position with scale unit

Argument：axis axis number

encoder_pos_by_mm the encoder's position, unit:: mm

Return value：reserve

void d2410_arc_move_unitmm (WORD *axis,double *target_pos, double *cen_pos, WORD arc_dir)

Description：Arc, circle or ellipse motions with scale unit in absolute position mode

Argument：axis the pointer of axes' number list

target_pos the pointer of target position's list, unit: mm

cen_pos the pointer of position's list of center of the circle, unit: mm

arc_dir arc direction: 0 is clockwise, 1 is anti-clockwise

Return value：Null

void d2410_rel_arc_move_unitmm (WORD *axis, double *rel_pos, double *rel_cen, WORD arc_dir)

Description：Arc, circle or ellipse motions with scale unit in relative distance mode

Argument：axis the pointer of axes' number list

rel_pos the pointer of target distance's list, unit: mm

rel_cen the pointer of distance's list of center of the circle, unit: mm
arc_dir arc direction: 0 is clockwise, 1 is anti-clockwise
Return value：Null

int d2410_pulse_loop(WORD axis)
Description：Automatic compensate position error between command position and actual position which encoder read.
Argument：axis axis number
Return value：reserve

6.15 Multi-axis synchronous motion

int d2410_set_t_move_all (WORD TotalAxes, WORD *pAxis, long *pDist, WORD posi_mode)
Description：Setup multi-axis synchronous motion
Argument：TotalAxes the amount of synchronous motion axes
pAxis the pointer of axes' number list
pDist the pointer of target distance's list, unit: pulse
posi_mode coordinates mode：0 is relative, 1 is absolute
Return value：1 is OK; -1 is error of parameters

int d2410_config_CSTA_PIN(WORD axis, WORD edge_mode)
Description：Configure the mode of synchronous start signal, which is low level active
Argument：axis axis number
edge_mode trigger mode：0 means that use level;
1 means that use edge
Return value：1 is OK; -1 is error of parameters

int d2410_start_move_all(WORD FirstAxis)
Description：Start multi-axis synchronous motion
Argument：FirstAxis the number of first axis
Return value：1 is OK; -1 is error of parameters

void d2410_simultaneous_stop(WORD axis)
Description：Output a stop signal from the specify axis' CSTP pin. If some axes' CSTP pins are connected, these axes will synchronous stop when a CSTP signal comes.
Argument：axis axis number
Return value：Null

int d2410_set_sync_option (WORD axis, WORD sync_stop_on, WORD

cstop_output_on, WORD sync_option1, WORD sync_option2)

Description：Setup multi-axis synchronous motion mode

Argument：axis axis number

sync_stop_on 1 enable CSTP signal to control multi-axis synchronous stop;
0 disable CSTP

cstop_output_on 1 means that output a CSTP signal when multi-axis stop abnormal;
0 means that don't output CSTP

sync_option1 0 means this function is effective immediately.
1 means this function is effective after CSTA signal come

sync_option2 0, reserve

Return value：1 is OK; -1 is error of parameters

int d2410_set_sync_stop_mode(WORD axis, WORD stop_mode)

Description：Setup multi-axis synchronous stop mode

Argument：axis axis number

stop_mode 0 means stop immediately; 1 means stop after deceleration

Return value：1 is OK; -1 is error of parameters

深圳市雷泰控制技术有限公司
SHENZHEN LEADTECH CONTROL TECHNOLOGY CO., LTD

Shenzhen leadtech control technology company, Limited
Address： 4th floor, 6 building, Nanyou Tian'an industrial park, Shenzhen
Post cord： 518054
Tel. ： 86-755-26417575
Fax： 86-755-26417609
Email： info@szleadtech.com.cn
Web site ： http://www.szleadtech.com.cn