

## CONTENT

CHAPTER 1 GENERAL INTRODUCTION.....	3
1.1 SUMMARY.....	3
1.2 PRODUCT LIST.....	3
1.3 ENVIRONMENTAL CONDITION.....	3
CHAPTER 2 CPU MODULE INTRODUCTION.....	5
2.1 OVERVIEW.....	5
2.1.1 <i>Front view</i> .....	5
2.1.2 <i>CPU Types</i> .....	5
2.2 FUNCTIONS.....	7
2.2.1 <i>CPU Status and LEDs</i> .....	7
2.2.2 <i>USB Programming port</i> .....	8
2.2.3 <i>Serial Communication Port</i> .....	9
2.2.4 <i>DIO Channel</i> .....	9
2.2.5 <i>High Speed Counter and High Speed Pulse Output</i> .....	9
2.2.6 <i>Edge Interrupts</i> .....	10
2.2.7 <i>Data Retentive and Data Backup</i> .....	10
2.2.8 <i>Real-time Clock (RTC)</i> .....	11
2.2.9 <i>Backup Battery</i> .....	11
2.3 WIRING DIAGRAM.....	12
2.4 DIMENSION.....	13
2.5 TECHNICAL SPECIFICATION.....	13
CHAPTER 3 SOFTWARE INTRODUCTION.....	16
2.1 INSTALL DRIVER OF USB PROGRAMMING PORT.....	16

---

2.2 HIGH SPEED COUNTER.....	19
2.2.1 Operation Modes and Inputs of the High-speed Counters.....	19
2.2.2 Control Byte and Status Byte.....	20
2.2.3 Preset value (PV value) setting.....	22
2.2.4 “CV=PV” Event No.....	24
2.2.5 How to use high speed counter.....	25
2.3 HOW TO USE HIGH SPEED PULSE OUTPUT.....	28
2.3.1 High speed pulse output instruction.....	28
2.3.2 How to use PLS instruction.....	28
2.3.3 How to Use Position Control Instructions.....	36

## Chapter 1 General Introduction

### 1.1 Summary

Kinco-K2 series PLC is a small and integrated PLC. Based on high performance, high reliability and powerful functions, we optimize the hardware design to reduce the cost.

Meanwhile we provide more features such as USB programming port, transistor DIO and so on, in order to fit user's requirement.

### 1.2 Product List

Name	Order No.	Description
<b>CPU Module</b>		
CPU205	K205-16DT	DC24V power supply, DI 6*DC24V,DIO 4*DC24V,DO 6*DC24V USB2.0,2*RS485,not support extension module.
	K205-16DR	DC24V power supply, DI 6*DC24V,DIO 4*DC24V,DO 6*Relay USB2.0,2*RS485,not support extension module.

### 1.3 Environmental Condition

Kinco-K2 accords with GB/T 15969.3-2007 (idt IEC61131-2: 2007) standard and test specifications.

The following table lists the conditions and requirements for Kinco-K2 to work properly. It is the user's responsibility to ensure that the service conditions are not exceeded.

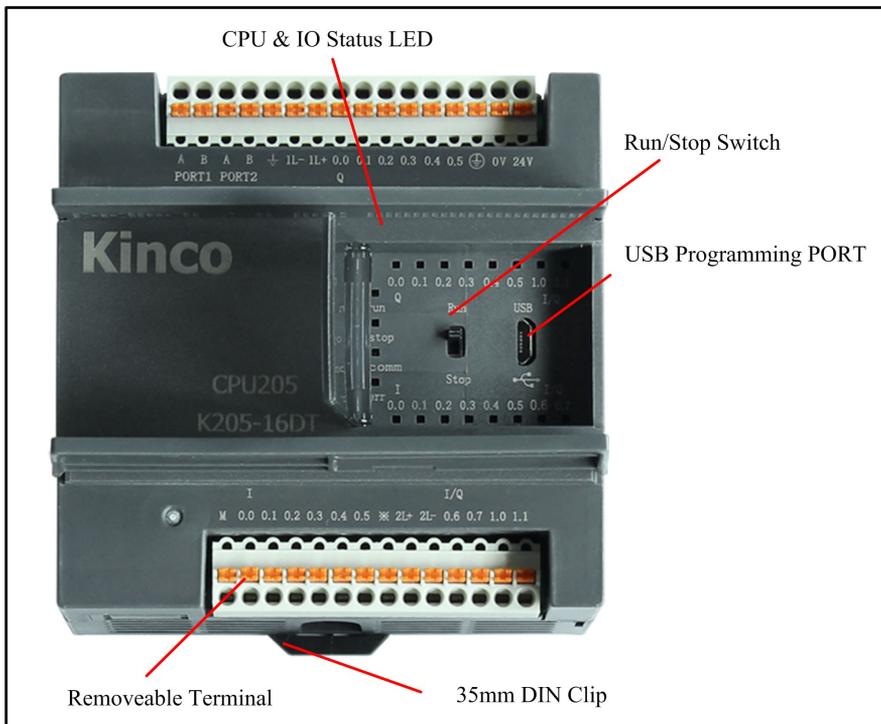
<b>Transport and storage</b>		
Ambient conditions	temperature	-40 --- +70 °C
	relative humidity	10%~95%, no condensation
	Altitude	Up to 3000 m
Mechanical conditions	Free falls	within manufacturer's original packaging, 5 falls from 1m of height.

Normal Operation		
Ambient conditions	air temperature	Open equipment : -10 --- +55°C; Enclosed equipment: -10 --- +40°C
	relative humidity	10%~95%, no condensation
	Altitude	Up to 2000 m
	Pollution degree	for use in pollution degree 2.
Mechanical conditions	Sinusoidal vibrations	5<f<8.4Hz, Occasional: 3.5mm amplitude; Continuous: 1.75mm amplitude. 8.4<f<150, Occasional: 1.0g acceleration; Continuous: 0.5g acceleration.
	Shock	occasional excursions to 15g, 11 ms, half-sine, in each of 3 mutually perpendicular axes.
Electromagnetic compatibility (EMC)	Electrostatic discharge	±4kV Contact, ±8kV Air. Performance criteria B.
	High energy surge	a.c. main power: 2KV CM, 1KV DM; d.c. main power: 0.5KV CM, 0.5KV DM; I/Os and Communication port: 1KVCM. Performance criteria B.
	Fast transient bursts	main power: 2KV, 5KHz. I/Os and Communication port: 1KV, 5KHz. Performance criteria B.
	Voltage drops and interruptions	a.c. supply: at 50Hz, 0% voltage for 1 period; 40% voltage for 10 periods; 75% voltage for 20 periods. Performance criteria A.
Ingress Protection Rating		IP20

## Chapter 2 CPU Module Introduction

### 2.1 Overview

#### 2.1.1 Front view



#### 2.1.2 CPU Types

Kinco-K2 provides different CPU models with a diversity of features and capabilities, all the CPU use DC24V power supply. The following table describes main technical data of each CPU model.

<b>Parameter</b>	<b>CPU205</b>
<b>Power Supply</b>	
Rated Power Supply	DC24V. Note : USB port can be used as power supply.

Power supply range	DC20.4V - 28.8V
<b>Built-in I/O and Communication</b>	
Digital channel	6*DI/6*DO/4*DIO
Analog channel	--
Expansion modules	--
Programming Port	Micro USB 2.0
Communication Port	<p>2 RS485, PORT1 and PORT2, Max.baudrate 115.2kbps.</p> <p>PORT1 supports the special programming protocol, Modbus RTU protocol (as a slave) and free-protocol communication mode.</p> <p>PORT2 supports Modbus RTU protocol (as a slave or master) and free-protocol communication mode.</p>
High-speed counters	4
Single phase	4, HSC0 and HSC1:Max.50KHz. HSC2 and HSC3:Max.20KHz.
Two phase	4, HSC0 and HSC1:Max.50KHz. HSC2 and HSC3:Max.10KHz.
High-speed pulse output	<p>3</p> <p>Q0.0 and Q0.1: Max.50KHz ( The resistor of load must less than <math>3K\Omega</math> ) .</p> <p>Q0.4:Max.10KHz.</p>
I/O Interrupts	4 rising/falling edge interrupts, I0.0-I0.3.
<b>memory area</b>	
Max. User program	4K instructions
User Data	M area: 1K bytes; V area: 4K bytes
DI image area	2 bytes
DO image area	2 bytes
AI image area	--
AO image area	--
Data Backup	E2PROM , 448 bytes
Retentive Ranges	4K bytes, lithium cell as backup power, 3 years at normal temperature.

Others	
Timers	256 1ms time-base : 4 10ms time-base : 16 100ms time-base : 236
Timed interrupts	2 with 0.1ms time-base
Counters	256
Real-time clock	Yes, deviation less than 5 min/month at 25°C

## 2.2 Functions

### 2.2.1 CPU Status and LEDs

The CPU has two modes: STOP mode and RUN mode.

In RUN mode, the CPU executes the main scan cycle and all interrupt tasks.

In STOP mode, the CPU will set all output channels (including DO and AO) to the known values which are specified in the [**Hardware Configuration**] through KincoBuilder, and only process communication requests which comes from KincoBuilder software and other Modbus RTU master device.

#### ➤ Change CPU status

Kinco-K2 provides two ways for manually changing the CPU status: Using the operation switch (RUN/STOP); Executing [Debug] -> [RUN] or [STOP] menu command in KincoBuilder.

The following table lists the combined results of these two ways.

Operation Switch Position	KincoBuilder command	Actual Operation Mode
RUN	[RUN]	RUN
	[STOP]	STOP
STOP	[RUN]	STOP
	[STOP]	STOP

Besides,if CPU detects any seriously fault,it will enter STOP status immediately.

### ➤ CPU Status LED

The CPU module provides 4 status LEDs: **RUN**, **STOP**, **Comm.** and **Err.**

**【Run】** : If CPU is in RUN status,it will turn on.

**【Stop】** : If CPU is in STOP status,it will turn on.

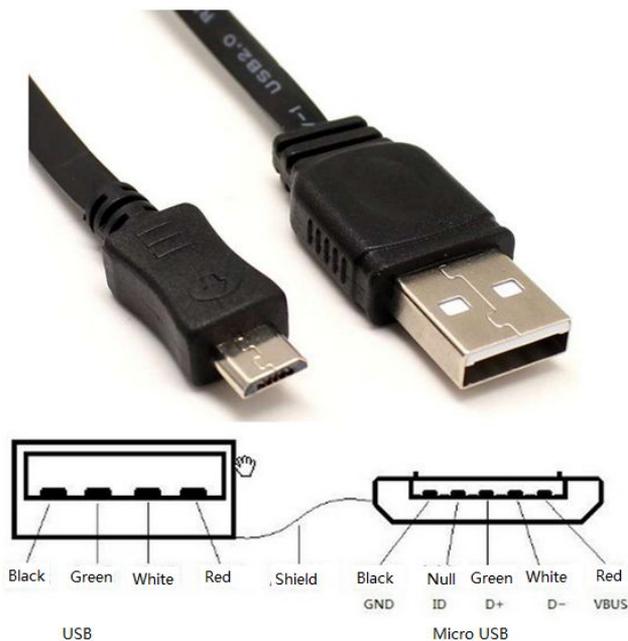
**【Comm】** : If any of communication ports (PORT0/1/2) is sending or receiving data,it will blinking.

**【Err.】** : If CPU detects error in user program or module, it will turn on.

K2 separates errors into three levels: Fatal error, Seriously error, Normal error. When CPU detects an error, it will use different way to handle according to error level and turn on Err LED, then it will save the error code in sequence for user analysis.

### 2.2.2 USB Programming port

K2 uses Micro USB(USB2.0 ) port as programming port. This connector is very popular for mobile phone, users can use same cable of mobile phone as the programming cable of K2 PLC.



In PC, the programming port of K2 will be a virtual COM port, you must install the driver for it when using

in PC first time. After finishing installing software Kincobuilder ,there will be different drivers in the path “\Kincobuilder V\*\*\*\Drivers\” for different versions of Windows system. Right now it can only support Windows XP, Windows 7 and Windows 8. When connecting programming cable to K2 and PC first time, Windows system will detect new hardware and mention installing driver, users can install the driver according to the version of Windows.

### 2.2.3 Serial Communication Port

K2 CPU provides 2 communication ports,PORT1 and PORT2.It supports baudrate up to 115.2kbps.PORT1 can be used as programming port and also support Modbus RTU slave protocol and free protocol.PORT2 supports Modbus RTU protocol (as a slave or master) and free protocol.

The number and location of the PORTs of each CPU are as following table.

CPU Type	RS485	Comment
CPU205	2	PORT1、 PORT2 are both on the left side of the upper connector.

And please refers to [2.3 Wiring Diagram](#) to know about their pin assignment.

### 2.2.4 DIO Channel

In Transistor type of K2 CPU, it provides some DIO channels. Each DIO channel can be used as DI(Source type) or DO (Source type) ,its signal is DC24V. User doesn't need to configure the DIO channels ,it will self-adapt according to the wiring.

Each DIO channel occupies two address, one is for DI(in I area) and the other one is for DO(in Q area.)

**Note: If a channel is used as DI, you shouldn't use DO address of this channel. If a channel is used as DO, you shouldn't use DI address of this channel.**

For example ,I/Q 0.6 is a DIO channel, so it occupies two address:I0.6 and Q0.6.If you need to use it as DI, then you can connect input signal to this channel directly, and use I0.6 in the program, but Q0.6 can not be used in program in this situation. If you need to use it as DO, then user can connect this channel to output device and use Q0.6 in the program, but I0.6 can not be used in program in this situation.

### 2.2.5 High Speed Counter and High Speed Pulse Output

K2 provides 4 high speed counters (HSC0~HSC3).High speed counter supports multiple modes: single phase,

CW/CCW(Up/Down),AB phase (1 multiplication and 4 multiplication).HSC0 and HSC1 can support up to 50KHz(Include single phase and AB phase).HSC2 and HSC3 can support up to 20KHz for single phase and 10 KHz for AB phase.

K2 provides 3 high speed pulse outputs(Q0.0,Q0.1 and Q0.4).All support PTO and PWM.Q0.0 and Q0.1 support up to 50KHz (The resistor of load should be less than 3K $\Omega$ ),Q0.4 supports up to 10KHz.

**Note: Relay output model of CPU module (Last number of order No. is “R”,such as K205-16AR) doesn't support high speed pulse output.**

### 2.2.6 Edge Interrupts

I0.0-I0.3 in CPU support edge interrupt function, it can execute interrupt by rising edge and falling edge of input signal. By using this function, it can capture the rising edge and falling edge of input signal quickly.For some input signal whose pulse width is less than the CPU scan time, it can respond quickly.

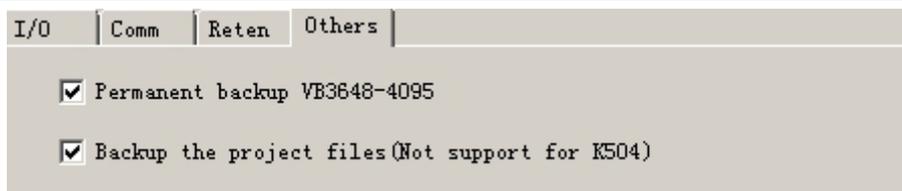
### 2.2.7 Data Retentive and Data Backup

Data retentive means the data in RAM can retain after power failure.CPU provides a lithium battery (Replaceable but un-rechargeable) for data retentive.When CPU loses power, the data in the RAM will be maintained by the lithium battery, and the retentive ranges will be left unchanged at next power on.Through **[Hardware]** configuration in KincoBuilder, user can select the type of data retentive (Such as V,C area) and the range. The life of battery is 5 years and the retaining duration is 3 years at normal temperature.

Data backup is that CPU provides an E<sup>2</sup>PROM to store data permanently. At power on, the CPU will restore the data from E<sup>2</sup>PROM into RAM to execute.

**Note: Because E<sup>2</sup>PROM has a writing limit of 1 million times, users should avoid to write data into data backup area frequently.**

There are 448 bytes in V area for data backup (VB3648--VB4095),the data in this area will save in E<sup>2</sup>PROM automatically.K2 sets VB3648--VB3902 as data backup by default,if user needs to use VB3903--VB4095 for data backup,it needs to configure in **【PLC hardware configuration】**.The configuration interface is as following figure.



### 2.2.8 Real-time Clock (RTC)

The real-time clock built in the all CPU modules can provide real-time clock/calendar indication. Users need to use KincoBuilder **【PLC】**->**【Time of Day Clock...】** to set the clock when using RTC first time. Then users can use real-time clock instructions (READ\_RTC、SET\_RTC、RTC\_W、RTC\_R) .

After CPU power off, the real-time clock can be maintained by lithium battery. The life of battery is 5 years and the retaining duration is 3 years at normal temperature.

### 2.2.9 Backup Battery

K2 can use certain specification lithium battery as backup battery. When PLC is power-off, it will use the backup battery to maintain real-time clock and RAM.

The backup battery is removable, user can replace new battery by themselves when the battery is empty.

The lithium battery is CR2032(3V) with connector. As shown in figure, user can order the battery separately.



### 2.3 Wiring diagram

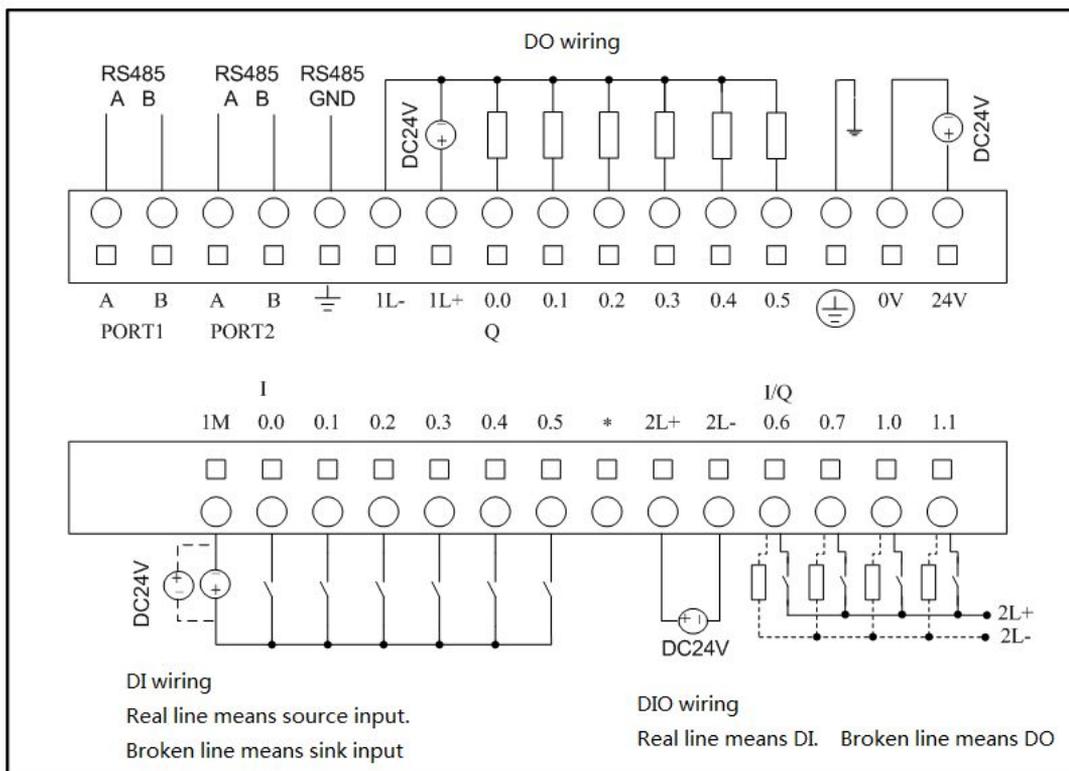
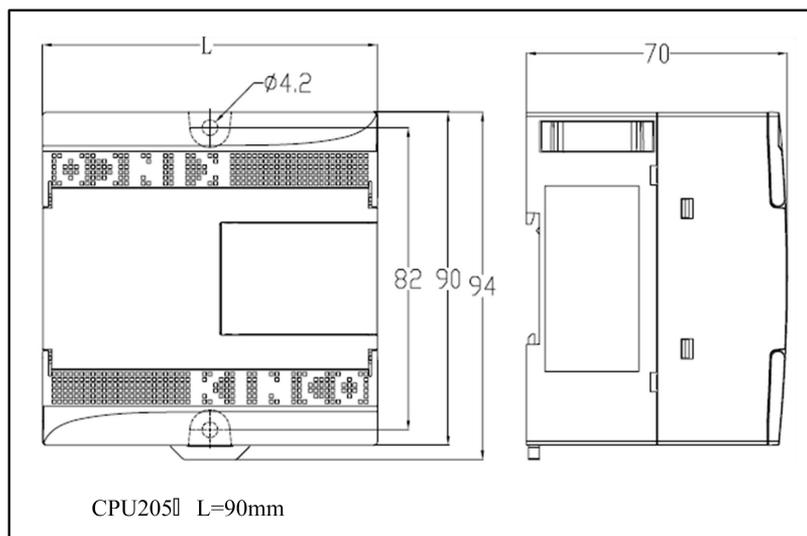


Fig.2-1 K205-16DT wiring diagram

## 2.4 Dimension



## 2.5 Technical Specification

### ➤ DI Specifications

Input type	Source/Sink
Rated input voltage	DC 24V (Max. 30V)
Rated input current	3.5mA@24VDC
Max input voltage of logic 0	5V@0.7mA
Minimum input voltage of logic 1	Common channel: 11V@2.0mA
Input filter time delay	
· off-to-on	Common channel: 15 $\mu$ s; HSC channel: 10 $\mu$ s;
· on-to-off	Common channel: 60 $\mu$ s; HSC channel: 6 $\mu$ s;
Isolation between input and internal circuit	
· Mode	Opto-electrical isolation
· Voltage	500VAC/1 min
Status indicator	“1” when Green LED turns on

➤ DO Specifications(Transistor type)

Output type	Source
Rated power supply voltage	DC24V,allowance range: DC20.4V—DC28.8V.(Same as power supply)
Output current per channel	Rated current:200mA,max.300mA @24VDC
Instant impulse current per channel	1A,less than 1s
Output leakage current	Max.0.5μA
Output impedance	Max. 0.2Ω
Output delay · off-to-on · on-to-off	Common channel: 15μs; HSC channel: 10μs; Common channel: 35μs; HSC channel: 6μs;
Protection: · Reverse polarity protection of power supply · Inductive load protection · Short-circuit protection · Reverse polarity protection of output	No Yes Yes Yes, less than 10s.
Isolation between output and internal circuit · Mode · Voltage	Opto-electrical isolation 500VAC/1 min
Status indicator	“1” when Green LED turns on

➤ DIO Specification (Transistor type)

Input/output type	Source
Rated input voltage	DC 24V (Max. 30V)
Rated input current	3.5mA@24VDC
Max input voltage of logic 0	5V@0.7mA
Minimum input voltage of logic 1	Common channel:11V@2.0mA
Rated power supply voltage	DC24V,allowance range: DC20.4V—DC28.8V.(Same as power supply)
Output current per channel	Rated current:200mA,max.300mA @24VDC

Instant impulse current per channel	1A,less than 1s
Output leakage current	Max.0.5 $\mu$ A
Output impedance	Max. 0.2 $\Omega$
Input filter time delay · off-to-on · on-to-off	15 $\mu$ s 60 $\mu$ S
Output delay · off-to-on · on-to-off	15 $\mu$ s 35 $\mu$ s
Protection: ·Reverse polarity protection of power supply ·Inductive load protection ·Short-circuit protection ·Reverse polarity protection of output	No Yes Yes Yes,less than 10s.
Isolation between output and internal circuit · Mode · Voltage	Opto-electrical isolation 500VAC/1 min
Status indicator	“1” when Green LED turns on

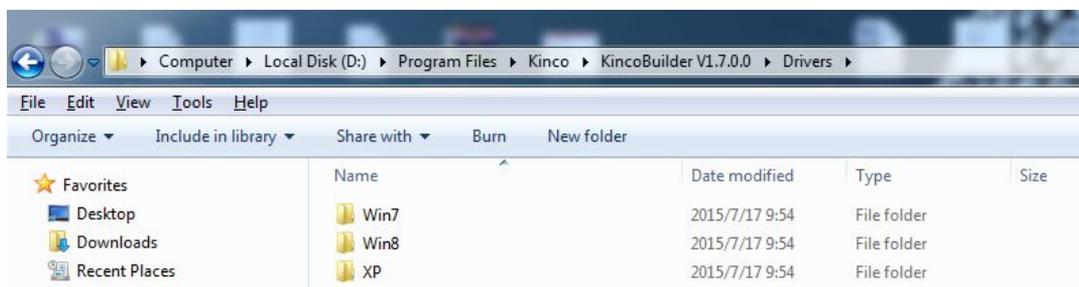
➤ DO Specification (Relay type)

Output type	Relay
Load voltage	DC30V/AC250V
Output current per channel	1A (DC30V/AC250V)
Output current per group	Max. 5A
Output off-to-on delay	Max. 10ms
Output on-to-off delay	Max. 5ms
Contact endurance · No load · Rated load	20,000,000 times (12,000 times/min) 100,000 times (6 times/min)
Isolation · Mode · Between coil and contact	Relay 2000Vrms
Status indicator	“1” when Green LED turns on

## Chapter 3 Software Introduction

### 2.1 Install driver of USB programming port

Kinco-K2 provides USB programming port. This programming port will be used as virtual serial port in PC. Its driver files are located in \drivers in KincoBuilder installation folder. Right now it supports Windows XP、Windows 7 和 Windows 8 systems, as shown in following figure:



When connecting programming cable to K2 and PC first time, Windows system will detect new hardware and mention installing driver, users can install the driver according to the version of Windows.

#### ➤ **Fail in installing driver in Windows 7?**

If it is simplified Win7 system, it will be lack of files mdmcpq.inf and usbser.sys, so it can not install the virtual serial port.

These two files are in the path ..\Win7\windows, users can copy these two files to following path and install the driver again.

- Copy mdmcpq.inf to C:\WINDOWS\INF.
- Copy usbser.sys to C:\WINDOWS\SYSTEM32\DRIVERS\.

#### ➤ **How to install driver in Windows 8?**

Windows 8 and above version require the inf file certificated by WHQL(Windows laboratory),otherwise it will forbid installation.

Solution: Execute [Advanced Startup] -> [Restart now], then select [7]Disable driver signature enforcement] in the “Startup Settings” window and restart PC. After startup, you can continue to install K2’s driver.

# Startup Settings

Press a number to choose from the options below:

Use number keys or functions keys F1-F9.

- 1) Enable debugging
- 2) Enable boot logging
- 3) Enable low-resolution video
- 4) Enable Safe Mode
- 5) Enable Safe Mode with Networking
- 6) Enable Safe Mode with Command Prompt
- 7) Disable driver signature enforcement
- 8) Disable early launch anti-malware protection
- 9) Disable automatic restart after failure

Press F10 for more options

Press Enter to return to your operating system

## 2.2 High speed counter

K2 provides 4 high speed counters HSC0-HSC3.HSC0 and HSC1 can support up to 50KHz(Both single phase and AB phase).HSC2 and HSC3 can support up to 20KHz for single phase and 10 KHz for AB phase.

High speed counter supports multiple modes: single phase, CW/CCW,AB phase (1 multiplication and 4 multiplication).

All high speed counter can support maximum 32 PV and support 32 “CV=PV” interrupts. PV can be set as relative value or absolute value. If it is relative value,

### 2.2.1 Operation Modes and Inputs of the High-speed Counters

Input signals of high-speed counter include: clock (input impulse), direction, start and reset.

In different operation modes input signals is different. Please see below:

HSC 0				
Mode	Description	I0.1	I0.0	I0.5
0	Single-phase up/down counter with internal direction control: SM37.3	Clock		
1			Reset	
2			Reset	Start
3	Single-phase up/down counter with external direction control	Clock		Direction
4			Reset	Direction
6	Two-phase counter with up/down clock inputs	Clock Down	Clock Up	
9	A/B phase quadrature counter	Clock A	Clock B	

HSC1					
Mode	Description	I0.4	I0.6	I0.3	I0.2
0	Single-phase up/down counter with internal direction control: SM47.3			Clock	
1		Reset			
2		Reset	Start		
3	Single-phase up/down counter with external direction control			Clock	Direction
4		Reset			Direction
6	Two-phase counter			Clock	Clock Up

7	with up/down clock inputs	Reset		Down	
9	A/B phase quadrature counter			Clock A	Clock B
10		Reset			

HSC 2			
Mode	Description	I0.4	I0.5
0	Single-phase up/down counter with internal direction control:SM57.3		Clock
9	A/B phase quadrature counter	Clock A	Clock B

HSC 3			
Mode	Description	I0.6	I0.7
0	Single-phase up/down counter with internal direction control:SM137.3		Clock
9	A/B phase quadrature counter	Clock A	Clock B

### 2.2.2 Control Byte and Status Byte

#### ➤ Control Byte

In SM area, each high-speed counter is assigned control byte to save its configuration data: one control word (8 bit), current value and pre-set (double-integer with 32 bit). Initial value of current assigned value. If the current value is written in the high-speed counter, it will start counting from that value. Please see below:

HSC0	HSC1	HSC2	HSC3	Description
SM37.0	SM47.0	SM57.0	SM137.0	Effective electrical level of reset signal : 0=high ; 1=low
SM37.1	SM47.1	SM57.1	SM137.1	Effective electrical level to start signal : 0=high ; 1=low
SM37.2	SM47.2	SM57.2	SM137.2	Orthogonal counter rate : 0=1x rate ; 1=4x rate <sup>§</sup>
SM37.3	SM47.3	SM57.3	SM137.3	Counting direction:0=Decrease ; 1=Increase
SM37.4	SM47.4	SM57.4	SM137.4	Write counting direction in HSC? 0= NO; 1= Yes
SM37.5	SM47.5	SM57.5	SM137.5	Write new pre-set value in HSC? 0= NO; 1= Yes
SM37.6	SM47.6	SM57.6	SM137.6	Write new current value in HSC? 0= NO; 1= Yes
SM37.7	SM47.7	SM57.7	SM137.7	Allow this high-speed counter? 0=NO; 1= YES
HSC0	HSC1	HSC2	HSC3	Description
SMD38	SMD48	SMD58	SMD138	Current value

SMD42	SMD52	SMD62	SMD142	Pre-set value
-------	-------	-------	--------	---------------

HSC0	HSC1	HSC2	HSC3	Description
SM141.0	SM151.0	SM161.0	SM171.0	Use multiple preset value:0=No. 1=Yes.
SM141.1	SM151.1	SM161.1	SM171.1	Preset value type:0=Absolute value. 1=Relative value.
SM141.2	SM151.2	SM161.2	SM171.2	Preset value comparison interrupt (“CV=PV”) cyclic execution. 0=No. 1=Yes. Note:Only valid when preset value is relative value.
SM141.3	SM151.3	SM161.3	SM171.3	Reserved
SM141.4	SM151.4	SM161.4	SM171.4	Update multiple PV segment and preset value:0=No. 1=Yes
SM141.5	SM151.5	SM161.5	SM171.5	Reset interrupt variable:0=Yes. 1=No.
SM141.6	SM151.6	SM161.6	SM171.6	Reserved
SM141.7	SM151.7	SM161.7	SM171.7	Reserved
HSC0	HSC1	HSC2	HSC2	Description
SMW142	SMW152	SMW162	SMW172	Starting value of preset value table ( It is offset corresponding to VB0),it must be odd value.

It needs to pay attention that not all the control bits of the control byte is suitable for all operation mode. For example, “Counting direction” and “Write counting direction in HSC” can be only used in mode 0,1 and 2 (Single-phase up/down counter with internal direction control),if the operation mode is with external direction control, then these two bits will be ignored.

The control byte, current value and preset value are 0 by default after power on.

#### ➤ Status Byte

In SM area, each high-speed counter has a status byte, which indicates the current status of high speed counter.

HSC0	HSC1	HSC2	HSC3	Description
SM36.0	SM46.0	SM56.0	SM136.0	Reserved
SM36.1	SM46.1	SM56.1	SM136.1	Reserved

SM36.2	SM46.2	SM56.2	SM136.2	Reserved
SM36.3	SM46.3	SM56.3	SM136.3	Fault in multiple PV value table:0=No,1=Yes
SM36.4	SM46.4	SM56.4	SM136.4	Reserved
SM36.5	SM46.5	SM56.5	SM136.5	Current counting direction: 0 = Down; 1 = Up
SM36.6	SM46.6	SM56.6	SM136.6	Current value equal to preset value: 0 = No,1 = Yes
SM36.7	SM46.7	SM56.7	SM136.7	Current value greater than preset value: 0 = No,1 = Yes
HSC0	HSC1	HSC2	HSC3	Description
SMB140	SMB150	SMB160	SMB170	Current PV segment No.(Start from 0)

### 2.2.3 Preset value (PV value) setting

K2 supports up to 32 PV value for each high speed counter, and supports setting PV value as relative value or absolute value. It supports “CV=PV” interrupt cyclic execution.

Follows take HSC0 as example to describe PV value function and setting.

#### ➤ How to select “multiple PV” mode

In the control byte of each high speed counter, there is one control bit for enable multiple preset value.

In HSC0, this control bit is SM141.0.

If SM141.0 is 0,it will use single PV value, same as K5 PLC.SMD42 is for new PV value,SM37.5 is to update this new PV value.

If SM141.0 is 1,it will use multiple PV values. In this situation,SM37.5 and SMD42 is invalid. All the PV values will be in the PV table(SMW142 is for starting address of the table),SM141.4 defines whether it use the data in PV table or not.If SM141.4 is 1,it means when HSC starts, it will get the data from PV table. If SM141.4 is 0,when HSC starts,it will ignore the data in PV table and get the data from last preset value.

#### ➤ Multiple PV table

If using PV table,all the PV value will get from PV table.

Each HSC provides one control word which is used to set the starting address of PV table.If using multiple PV,then all PV value will get from PV table.The starting address of PV table is odd address of V area,such as

301(Means VB301).

The format of PV table is as follows.

OFFSET <sup>(1)</sup>	Data type	Description
0	BYTE	Quantity of PV
1	DINT	First PV
5	DINT	Second PV
...	DINT	...

- (1) All the offset value are the offset bytes related to the table.
- (2) When it is set as relative value,then the absolute value of PV data must be greater than 1,or PLC will consider the segment of multiple PV finish and count the number of PV according to this(Higher priority than setting quantity of PV).

When it is set as absolute value,the difference between two adjacent PV's absolute value must be greater than 1,or PLC will consider the segment of multiple PV finish and count the number of PV according to this(Higher priority than setting quantity of PV).

- (3) "CV=PV" interrupts must execute in sequence,it means that after the counter reaches the first PV and executes interrupt,then it will compare with the second PV and so forth.
- (4) PV must be set reasonably.Here takes relative value as example,if it is positive counting,PV must be greater than 0,otherwise the "CV=PV"interrupt will never execute.If it is negative counting,PV must be less than 0,otherwise the "CV=PV"interrupt will also never execute.

#### ➤ Relative value and absolute value

In the control byte of each high speed counter, there is one control bit which is used to set PV as relative value or absolute value.

For HSC0,the control bit is SM141.1.

If SM141.1 is 0,it means PV is absolute value. When counting value is equal to PV,it will execute "CV=PV" interrupt. For example,if it sets 3 PV values,such as 1000,2000 and 3000,then when counting value reaches 1000,it will execute the first "CV=PV"interrupt. When the counting value reaches 2000,it will execute the second "CV=PV" interrupt and so forth.

If SM141.1 is 1,it means PV is relative value.If counter takes current counting value as reference,when

the value it continues to count is equal to PV, it will execute “CV=PV” interrupt. For example, if it sets 3 PV values, such as 10, 1000 and 1000, and the current counting value is 100 before HSC starts, then when the counting value reaches 110, 1110 and 2110, it will execute corresponding “CV=PV” interrupt.

#### ➤ “CV=PV” interrupt cyclic execution

“CV=PV” interrupt cyclic execution is only valid when PV is set as relative value.

If SM141.0 is 0, it means “CV=PV” interrupt only executes once. When all interrupts finish execution, then it will stop. If it needs to execute again, it must modify the related registers and execute HSC instruction again.

If SM141.0 is 1, it means “CV=PV” interrupt is cyclic execution. When the last PV interrupt finishes execution, PLC will take the current counting value as reference to calculate new value for PV interrupt, then it will start to compare the counting value and execute “CV=PV” interrupt and so forth. This process will execute cyclically.

For example, it sets 3 PV values, such as 10, 1000 and 1000. And the current counting value is 100 before HSC starts, then the value for every interrupt is as following table.

Current counting value	Interrupt times	First value	Second value	Third value
100	1st time	110	1110	2110
2110	2nd time	2120	3120	4120
4120	3rd time	4130	5130	6130
...	N time	...	...	...

#### 2.2.4 “CV=PV” Event No.

When it uses single PV mode, the HSC will be fully compatible with K5 (Include “CP=PV” event No.).

When it uses multiple PV mode, the HSC will assign a new event No. for 32 PV, as shown in following table.

High speed counter	Interrupt No.	Description
HSC0	64	“CV=PV” interrupt of 1st PV
	65	“CV=PV” interrupt of 2nd PV
	...	... (Plus 1)
	95	“CV=PV” interrupt of 32nd PV
HSC1	96	“CV=PV” interrupt of 1st PV

	97	“CV=Pv”interrupt of 2nd PV
	...	... (Plus 1)
	127	“CV=Pv”interrupt of 32nd PV
HSC2	128	“CV=Pv”interrupt of 1st PV
	129	“CV=Pv”interrupt of 2nd PV
	...	... (Plus 1)
	159	“CV=Pv”interrupt of 32nd PV
HSC3	160	“CV=Pv”interrupt of 1st PV
	161	“CV=Pv”interrupt of 2nd PV
	...	... (Plus 1)
	191	“CV=Pv”interrupt of 32nd PV

## 2.2.5 How to use high speed counter

### ➤ Method 1: Use instructions for programming

- 1) Configure the control byte of HSC and define the current value (i.e. starting value) and the preset value.
- 2) Use HDEF instruction to define the counter and its operation mode.
- 3) (Optional) Use ATCH instruction to define the interrupt routines.
- 4) Use HSC instruction to start the high-speed counter.

### ➤ Method 2: Use wizard of HSC

In K2 PLC, it provides configuration wizard for high speed counter. Users can use the wizard to configure all high speed counters and don't need to program. The wizard is as following figure:

After using wizard to configure HSC, user also can use “**Method 1**” to modify the parameters of HSC.

**HSC Wizard**

HSC: **HSCO** Mode: **Mode 0**  Enable HSC Start method: **Run directly at PLC startup**

Quadrature rate: **1x** Reset signal level: **High** Start signal level: **High**

Signal Input: Pulse: IO.1;

Update direction New direction: **Up**  
 Update count value New count value: **0**

Enable external reset interrupt  
Interrupt routine: \_\_\_\_\_  
 Enable external direction-changed interrupt  
Interrupt routine: \_\_\_\_\_

PV and corresponding interrupts

Enable multiple PVs Relationship between PVs: **Absolute**  Cyclic "CV=PV" interrupts

Multiple PVs settings

Update PV and quantity Quantity: **3** Starting location of PV table (VB): **3009**

I...	Address	Value	Event...	Interrupt routine
1	%VD3010	100	64	(INT00) INT_0
2	%VD3014	200	65	(INT01) INT_1
3	%VD3018	300	66	(INT02) INT_2

Up  
Down  
Delete

Single PV settings (compatible with K5)

Update preset value (PV) New PV: **2**  Enable "CV=PV" interrupt  
Interrupt routine: \_\_\_\_\_

Apply OK Cancel Help

How to use HSC wizard:

- 1) Select the counter in **【HSC】** .
- 2) Check **【Enable HSC】** , and then continue following configuration.
- 3) Select counter mode in **【Mode】** .
- 4) Select the starting mode in **【Start method】** .

There are two starting method:

“Using HSC instruction”: If selecting this method, then it needs to execute HSC instruction to start the HSC. Before executing HSC instruction, it doesn't need to configure the registers and execute HDEF

instruction.

“Run directly at PLC startup”: If selecting this method, then the HSC will start automatically after PLC power on without executing any instructions.

- 5) If user needs to use multiple PV mode, then check **【Enable multiple PVs】** and continue to configure all PV values and related ‘Value’ and ‘Interrupt subroutine’. If checking **【Update PV and quantity】**, then it can adjust the value in **【Quantity】** to modify the number of PV.
- 6) If user needs to use single PV mode, then check **【Update preset value(PV)】** in ‘Single PV settings’ and modify the PV value and related interrupt subroutine.
- 7) For other options, please refer to the descriptions to HSC.

## 2.3 How to use high speed pulse output

Kinco-K2 provides 3 channels for high speed pulse output, they are Q0.0,Q0.1 and Q0.4. Q0.0 and Q0.1 support maximum 50KHz, and Q0.4 supports maximum 10KHz.

### 2.3.1 High speed pulse output instruction

K2 provides 3types of instructions for high speed pulse output.

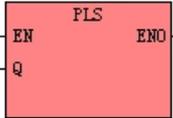
- 1) PLS: it is used to output PTO(Single segment or multiple segments) and PWM.
- 2) Position control: There are 5 instructions, include PREL(Relative positioning), PABS(Absolute positioning) ,PHOME(Homing), PJOG(Jogging) and PSTOP(Emergency stop). User can use these instructions to achieve positioning control easily .**Note: When using position control instructions, the frequency of output pulse must be not less than 125Hz.**
- 3) Following instruction PFLO\_F: There are parameters such as input frequency( $F$ ),electronic gear ratio( $NUME$ 、 $DENOM$ ), pulse number( $COUNT$ ) and so on, these parameters can be used as variable. The frequency of pulse output is equal to  $F$  multiple by electronic gear ratio. When the pulse number reaches the value  $COUNT$ , then it will stop output and set  $DONE$  bit. **Note: When using following instruction, the frequency of output pulse must be not less than 30Hz.**

### 2.3.2 How to use PLS instruction

PLS instruction can implement PTO and PWM output function.

- PTO: Pulse Train Output.
- PWM: Pulse-Width Modulation.

#### ➤ Descriptions

	Name	Usage	Group	Suitable for
LD	PLS			K2 K5

<b>IL</b>	PLS	PLS <i>Q</i>	U	
-----------	-----	--------------	---	--

<b>Operands</b>	<b>Input/Output</b>	<b>Data Type</b>	<b>Description</b>
<i>Q</i>	Input	INT	Constant (0, 1 or 2)

The *PLS* instruction is used to load the corresponding configurations of the PTO/PWM from the specified SM registers and then start outputting pulse until it finish outputting pulse. The pulse output channel is specified by parameter *Q*, 0 means Q0.0,1 means Q0.1,2 means Q0.4.

Note: In user program, it only needs to execute *PLS* instruction once when it is required. It is suggested to use edge instruction to execute *PLS* instruction. If executing *PLS* executing all the time, then it can't output normally.

- **LD**

If *EN* is 1, then *PLS* is executed.

- **IL**

If *CR* is 1, then *PLS* is executed. It won't influence the value of *CR*.

### 2.3.2.1 High-speed Pulse Output Function of Kinco-K2

The Kinco-K2 provides 3 PTO/PWM pulse generators that can be used to output PTO/PWM. Thereof, one generator is assigned to Q0.0, called PWM0 or PTO0; the second one is assigned to Q0.1, called PWM1 or PTO1, and the third one is assigned to Q0.4, called PWM2 or PTO2.

The PTO/PWM pulse generators and the DO mapping area share the memory address Q0.0, Q0.1 and Q0.4.

When the user program executes the high speed pulse output instructions, then the PTO/PWM generator controls the output and prohibits the normal use of this output channel.

Some registers are provided in SM area for each PTO/PWM generator. When user needs to use high speed pulse output function, it needs to configure these memories, and then executes *PLS* instruction to implement desired operation of PTO/PWM.



*Notice: Make sure not to use the PTO and PWM functions if Q0.0 ,Q0.1 and Q0.4 are relay-output!*

### ➤ PWM

PWM provides a continuous pulse output with a variable duty cycle, and you can control the cycle time and the pulse width.

The unit of cycle time and pulse width time is microsecond(us) or millisecond(ms). The maximum value of cycle time is 65535. If the pulse width time is greater than the cycle time value, the duty cycle is set to be 100% automatically and the output is on continuously. If the pulse width time is 0, the duty cycle is set to be 0% and the output is off.

### ➤ PTO

PTO provides a square wave (50% duty cycle) output, and you can control the cycle time and the number of the output pulses. The unit of cycle time is microsecond(us) or millisecond(ms).The maximum value of cycle time is 65535.The range of pulse number is 2~4,294,967,295.If the specified pulse number is less than 2, then Kinco-K2 will set related error bit and prohibit the output.

PTO function provides single segment of pulse and multiple segment of pulse.

- **Single segment pulse**

In single segment pulse mode, it only executes pulse train output once after executing PLS instruction.

- **Multiple segment pulse**

In multi-segment pulse mode, CPU automatically reads the configurations of each PTO segment from a profile table located in V area and executes the related PTO segment.

The length of each segment is 8 bytes, including a cycle time value (16-bit, WORD), a reserved value (It is not used now,16-bit, INT), and a pulse number value (32-bit, DWORD).Thereof, all the pulse output frequency are the same in same segment. It uses PLS instruction to start multiple segment pulse.

In this mode, the starting address of the profile table is stored in SMW168 (corresponding to PTO0) ,SMW178 (corresponding to PTO1) and SMW268(corresponding to PTO2).Time base is configured by SM67.3 (corresponding to PTO0) ,SM77.3 (corresponding to PTO1) and SM87.3 (corresponding to PTO2). The time

base can be in either microsecond or millisecond. All cycle values in the profile table must use same time base, and cannot be modified when the profile is executing.

The following table describes the format of the profile table.

Byte offset <sup>1</sup>	Length	Segment	Description
0	8-bit		The number of segments (1 to 64)
1	16-bit	1	Initial cycle time (2 to 65535 times of the time base)
3	16-bit		Reserved
5	32-bit		Pulse number(1 to 4,294,967,295)
9	16-bit	2	Initial cycle time (2 to 65535 times of the time base)
11	16-bit		Reserved
13	32-bit		Pulse number(1 to 4,294,967,295)
...		...	...

**1** All the offsets in this column are relative to the starting position of the profile table.



*Notice: the starting position of the profile table must be an odd address in V area, e.g. VB3001.*

### 2.3.2.2 PTO/PWM Register

Each PTO/PWM generator is provided with some registers in SM area to store its configurations, as shown in following table.

Q0.0	Q0.1	Q0.4	Description
SM67.0	SM77.0	SM87.0	PTO/PWM Whether to update the cycle time: 0 = No; 1 = Yes
SM67.1	SM77.1	SM87.1	PWM Whether to update pulse width time: 0=No; 1=Yes
SM67.2	SM77.2	SM87.2	PTO Whether to update the pulse number: 0=No; 1=Yes
SM67.3	SM77.3	SM87.3	PTO/PWM Time base: 0=1 $\mu$ s; 1=1ms
SM67.4	SM77.4	SM87.4	PWM Update method: 0 = asynchronous update; 1 = synchronous update

SM67.5	SM77.5	SM87.5	PTO Operation mode: 0 = single segment; 1 = multiple segment
SM67.6	SM77.6	SM87.6	Function selection: 0= PTO; 1=PWM
SM67.7	SM77.7	SM87.7	PTO/PWM Enable/disable: 0=disable; 1= enable
<b>Q0.0</b>	<b>Q0.1</b>	<b>Q0.4</b>	<b>Description</b>
SMW68	SMW78	SMW88	PTO/PWM Cycle time , Range:2~65535
SMW70	SMW80	SMW90	PWM Pulse width, Range: 0~65535
SMD72	SMD82	SMD92	PTO Pulse number, Range:1~4,294,967,295
SMW168	SMW178	SMW268	The starting location of the profile table (byte offset from V0)For multi-segment PTO operation only

All the default value for control byte, cycle time and pulse number are 0. The way to modify configuration of PTO/PWM is that configure related control registers first, if it is PTO multiple segment pulse, it also needs to configure profile table, and then execute PLS instruction.

Each PTO/PWM generator also provides a status bytes in SM area, user can get the status information of PTO/PWM generator from the status bytes, as shown in following table.

Q0.0	Q0.1	Q0.4	Description
SM66.0	SM76.0	SM86.0	Reserved
SM66.1	SM76.1	SM86.1	Reserved
SM66.2	SM76.2	SM86.2	Reserved
SM66.3	SM76.3	SM86.3	PWM idle: 0=No, 1=Yes
SM66.4	SM76.4	SM76.4	Whether the cycle time or pulse number of PTO is wrong: 0=No, 1=Yes <b>Note: Cycle time and pulse number must be greater than 1.</b>
SM66.5	SM76.5	SM76.5	PTO profile terminated due to user command: 0=No, 1=Yes
SM66.6	SM76.6	SM76.6	Reserved
SM66.7	SM76.7	SM76.7	PTO idle: 0=No, 1=Yes

The PTO idle bit or PWM idle bit indicate the completion of the PTO or PWM output.

### 2.3.3.3 PTO Operations

The following takes PTO0 as an example to introduce how to configure and operate the PTO/PWM generator

in the user program.

There are two procedures for using PTO: Configure related control registers and initialize PTO. Execute PLS instruction.

Use SM0.1 (the first scan memory bit) to call a subroutine that contains the initialization instructions. Since SM0.1 is used, the subroutine shall be executed only once, and this reduces CPU scan time and provides a better program structure.

➤ **Execute the PTO (Single-Segment Operation)**

- 1) Set control byte SMB67 according to the desired operation.

For example, SMB67 = B#16#85 indicates:

- Enable the PTO/PWM function
  - Select PTO operation
  - Select 1 $\mu$ s as the time base
  - Allow updating the pulse number and cycling time.
- 2) Set SMW68 according to desired cycle time.
  - 3) Set SMD72 according to desired pulse number.
  - 4) (Optional) use ATCH to attach the PTO0-complete event (event 28) to an interrupt routine to respond in real time to a PTO0-complete event.
  - 5) Execute the *PLS* instruction to configure PTO0 and start it.

➤ **Changing the PTO Cycle Time (Single-Segment Operation)**

Follow these steps to change the PTO cycle time.

- 1) Set control byte SMB67 according to the desired operation.

For example, SMB67 = B#16#81 indicates:

- Enable the PTO/PWM function
- Select PTO operation
- Select 1 $\mu$ s as the time base
- Allow updating the cycle time value.

- 2) Set SMW68 according to desired cycle time.
- 3) Execute the *PLS* instruction to configure PTO0 and start it, then a new PTO with the updated cycle time shall be generated.

➤ **Changing the PTO Pulse Number(Single-Segment Operation)**

Follow these steps to change the PTO pulse count:

- 1) Set control byte SMB67 according to the desired operation.  
For example, SMB67 = B#16#84 indicates:
  - Enable the PTO/PWM function
  - Select PTO operation
  - Select 1 $\mu$ s as the time base
  - Allow updating the pulse number
- 2) Set SMD72 according to desired pulse number.
- 3) Execute the *PLS* instruction to configure PTO0 and start it, then a new PTO with the updated pulse number shall be generated.

➤ **Execute the PTO (Multiple-Segment Operation)**

- 1) Set control byte SMB67 according to the desired operation.  
For example, SMB67 = B#16#A0 indicates:
  - Enable the PTO/PWM function
  - Select PTO operation
  - Select multi-segment operation
  - Select 1 $\mu$ s as the time base
- 2) Set an odd number as the starting position of the profile table into SMW168.
- 3) Use V area to configure the profile table.
- 4) (Optional) Use ATCH to attach the PTO0-complete event (event 28) to an interrupt routine to respond in

real time to a PTO0-complete event.

- 5) Execute the *PLS* instruction to configure PTO0 and start it.

### 2.3.3.3 PWM Operations

Following takes PWM0 as an example to introduce how to configure and operate the PTO/PWM generator in the user program.

There are two procedures for using PWM: Configure related control registers and initialize PTO. Execute *PLS* instruction.

Use SM0.1 (the first scan memory bit) to call a subroutine that contains the initialization instructions. Since SM0.1 is used, the subroutine shall be executed only once, and this reduces CPU scan time and provides a better program structure.

#### ➤ Execute PWM

- 1) Set control byte SMB67 according to the desired operation.

For example, SMB67 = B#16#D3 indicates:

- Enable the PTO/PWM function
  - Select PWM operation
  - Select 1 $\mu$ s as the time base
  - Allow updating the pulse width value and cycle time value
- 2) Set SMW68 according to desired cycle time.
  - 3) Set SMW70 according to desired pulse width.
  - 4) Execute the *PLS* instruction to configure PWM0 and start it.

#### ➤ Changing the Pulse Width for the PWM Output

The following steps describes how to change PWM output pulse width.

- 1) Set control byte SMB67 according to the desired operation.

For example, SMB67 = B#16#D2 indicates:

- Enable the PTO/PWM function
  - Select PWM operation
  - Select 1 $\mu$ s as the time base
  - Allow updating the pulse width value and cycle time value
- 2) Set SMW70 according to desired pulse width.
  - 3) Execute the *PLS* instruction to configure PWM0 and start it.

### 2.3.3 How to Use Position Control Instructions

#### 2.3.3.1 How to Modify the Current Value of Position Control Instructions

##### ➤ Control Registers and Status Registers

For the Position Control instructions, Kinco-K2 specifies a control byte for each high-speed output channel to store its configurations. Besides, it assigns a current value register(DINT) to store the pulse number which has outputted currently (This value will increase when run forward and decrease when run reverse).The following table describes the control byte and the current value.

Q0.0	Q0.1	Q0.4	Description
SMD212	SMD242	SMD262	Read only. Current value (Increase when run forward, decrease when run reverse).It indicates the pulse number which has already outputted.
SMD208	SMD238	SDM258	Read/Write. New current value. Use to modify the current value together with specific control bit.
Q0.0	Q0.1	Q0.4	Description
SM201.7	SM231.7	SM251.7	Read/Write. Emergency-Stop bit. If this bit is 1, no position control instructions can be executed. When executing the PSTOP instruction, this bit is set to 1 automatically, and it must be reset in the program.。
SM201.6	SM231.6	SM251.6	Read/Write. Reset the current value or not

			1 --- Clear the current value. 0 --- Maintain the current value.。
SM201.5	SM231.5	SM251.5	Reserved
SM201.4	SM231.4	SM251.4	Read/Write. Use to modify current value. 1 - Modify current value. 0 - Maintain the current value.
SM201.3	SM231.3	SM251.3	Read/Write. Direction control bit. 1 --- Disable the direction output channel, it will be used as normal output. 0 --- Enable the direction output channel.
SM201.0 ~ SM201.2	SM231.0~ SM231.2	SM251.0~ SM251.2	Reserved

### ➤ How to modify current value

Each high speed output channel has one register for current value, they are SMD212,SMD242 and SMD262.The outputted pulse number are stored in these registers. Current value registers are read only, if user needs to modify the current value, it can use following methods.

#### • Method 1

User reset bit to clear current value.

The reset bits for 3 output channels are SM201.6、 SM231.6 and SM251.6.

When the reset bit is 1, PLC will set the current value as 0.Therefore, t only needs one scan time for the reset bit to activate. When it needs to use this bit, try to avoid to keep this bit always 1 and also avoid to set this bit while the Position Control instruction (Include PHOME, PREL, PABS, JOG and PFLO\_F) is executing, otherwise the counting value may be wrong.

Following takes channel 0 as example to describe how to reset current value.

(\* Network 0 \*)

(\*Based on homing signal, when it moves to homing, it requires to clear current value\*)

LD        %SM0.0

PHOME   0, %M0.0, %M0.1, %M0.2, %VW0, %VW2, %VW4, %VD6, %VW10, %M0.4, %M0.5, %MB1

(\* Network 1 \*)

(\*After PHOME finishing, it uses finishing bit “DONE” to clear current value\*)

```
LD      %M0.4
R_TRIG
ST      %SM201.6
```

- **Method 2**

Modify current value by using following registers.

Q0.0	Q0.1	Q0.4	Description
SMD208	SMD238	SDM258	Read/Write. New current value. Use to modify the current value together with specific control bit.
SM201.4	SM231.4	SM251.4	Read/Write. Use to modify current value. 1 - Modify current value. 0 - Maintain the current value. 。

Here takes channel 0 as example to describe the method: If SM201.4 is 0, then it will maintain the current value SMD212. If SM201.4 is 1, then it will move the value of SMD208 to SMD212. When it needs to use this bit, avoid to keep this bit always 1 and also avoid to set this bit while the Position Control instruction (Include PHOME, PREL, PABS, JOG and PFLO\_F) is executing, otherwise the counting value may be wrong.

Following takes channel 0 as example to describe how to modify current value:

(\* Network 0 \*)

(\*Based on homing signal, hen it moves to homing, t requires to set current value as 100.\*)

```
LD      %SM0.0
PHOME  0, %M0.0, %M0.1, %M0.2, %VW0, %VW2, %VW4, %VD6, %VW10, %M0.4, %M0.5, %MB1
```

(\* Network 1 \*)

(\*When PHOME instruction finishing, it uses finishing bit DONE to modify current value.\*)

```
LD      %M0.4
R_TRIG
MOVE   DI#100, %SMD208
ST      %SM201.4
```

**2.3.3.2 Can it change maximum output frequency when position control instruction is executing?**

PREL (Relative position) and PABS (Absolute position) will not change maximum output frequency when it is executing. It will read the parameters minimum frequency, maximum frequency and acceleration/deceleration time parameters when it starts, and calculates suitable acceleration/deceleration segments according to the value of these parameters, then it will start outputting pulse. During pulse outputting, PREL and PABS will not read the parameters above again, therefore, changing these parameters will not affect the pulse output.

PJOG (Jogging) will read pulse input frequency(MAXF) all the time when it is executing, and adjust the pulse output frequency according to new setting frequency.

PHOME (Homing) will read the maximum frequency (MAXF) all the time when it is running at maximum frequency but hasn't found homing signal, and calculate acceleration or deceleration segment automatically according the new setting frequency, then it will accelerate or decelerate to new frequency to output pulse.