May 2001

Getting Started EH-CU/CUE

1. Overview

CPU module reads/writes from/to EH-CU via 8 words as below. Since the access area is so limited, the parameters are transferred using “Command” in WY❒5 and “Data” in WY❒6,7.

WX ❒0	**Status register (reading out from EH-CU)**
WX ❒1	Data (Low word for CH1)
WX ❒2	Data (High word for CH1)
WX ❒3	Data (Low word for CH2)
WX ❒4	Data (High word for CH2)
WY ❒5	**Control register (writing to EH-CU)**
WY ❒6	Data (Low word)
WY ❒7	Data (High word)

❒ = slot number

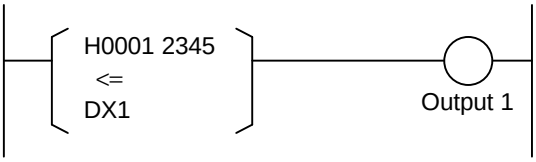
Status register WX ❒0

X15	X14	X13	X12	X11	X10	X9	X8	X7	X6	X5	X4	X3	X2	X1	X0
XHS	-	CH2	CH1	OF2	UF2	OF1	UF1	EQ2	OE2	ME2	CE2	EQ1	OE1	ME1	CE1

Control register WY ❒5

Y95	Y94	Y93	Y92	Y91	Y90	Y89	Y88	Y87	Y86	Y85	Y84	Y83	Y82	Y81	Y80
YHS	CMD	CH2	CH1	CM3	CM2	CM1	CM0	EC2	OE2	ME2	CE2	EC1	OE1	ME1	CE1
Sub command				Command				Flags for CH2				Flags for CH1			

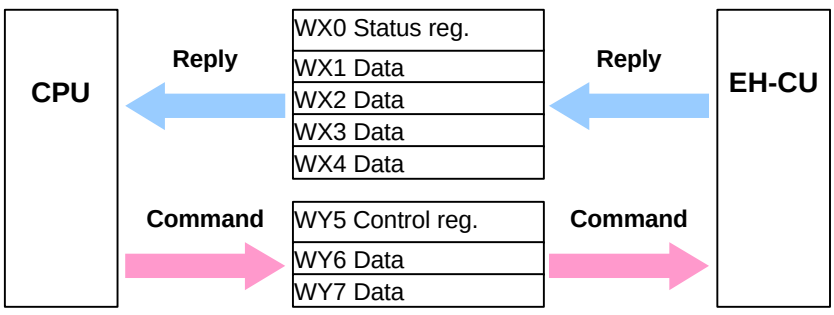
Basically, all data are exchanged by a command, but only the current value is monitored in WX1 to WX4 automatically. You can simply read it like normal analog input data as below.



The direct monitored data is read out once in a scan, which is not very fast depending on program size. In addition, the counter value is double word. This might have very small time lag in these two words. If you need more precise data, please use (1) or/and (2).

(1) “Refresh command” (FUN 80-82) for CPU

(2) The command “Read counter value” (H8100) for EH-CU (Refer to the chapter 4)



2. Normal counter and Ring counter

This table shows overview of normal and ring counter. The definition of some words are different depending on the counter type.

	Normal counter	**Ring counter**
Over/under flow	Yes	-
“>” detection	Yes	-
“=” detection	Yes	Yes
Marker input	Yes	-
Min. value	0	Assigned in “Preset value”
Max. value	HFFFF FFFF	Assigned in “Set point 1”
Preset value	The current value is changed to the preset value at M input.	(Used as the min. value.)
Set point 1	Set point for normal counter	(Used as the max. value)
Set point 2	-	Set point for Ring counter

3. Dip switch setting

No.		OFF	ON
1 & 2	OFF OFF : 2-phase counter (Max.100kHz) ON OFF : Single phase counter (CW, CCW) OFF ON : Single phase counter (CK, U/D) ON ON : 2-phase, 4-multiple counter (Max.25kHz)	-	-
3	Marker polarity for CH1	Positive logic	Negative logic
4	Marker polarity for CH2	Positive logic	Negative logic
5	Counting in CPU STOP for CH1	Disabled	Enabled
6	Counting in CPU STOP for CH2	Disabled	Enabled
7	Counter type for CH1	Normal counter	Ring counter
8	Counter type for CH2	Normal counter	Ring counter
9	-	-	-
10	-	-	-

4. Command list

		WY ❒5	WY ❒6	WY ❒7
Write counter value	CH1	H9F00	Data (Low word)	Data (High word)
	CH2	HAF00		
	Both	HBF00		
Write set point for normal counter	CH1	H9200	Data (Low word)	Data (High word)
	CH2	HA200		
	Both	HB200		
Configure outputs	-	HF100	(Refer to the next page.)	-
Write set point for Ring counter	CH1	H9300	Data (Low word)	Data (High word)
	CH2	HA300		
	Both	HB300		
Write preset value	CH1	H9E00	Data (Low word)	Data (High word)
	CH2	HAE00		
	Both	HBE00		
Clear overflow flag	CH1	HD200	-	-
	CH2	HE200		
	Both	HF200		
Clear underflow flag	CH1	HD300	-	-
	CH2	HE300		
	Both	HF300		
Enable counter (CE) (Note 1)	CH1	H8001 (Y80=1)	-	-
	CH2	H8010 (Y84 =1)		
	Both	H8011 (Y80=Y84=1)		
Enable marker (ME) (Note 1)	CH1	H8002 (Y81=1)	-	-
	CH2	H8020 (Y85=1)		
	Both	H8022 (Y81=Y85=1)		
Enable output (OE) (Note 1)	CH1	H8004 (Y82=1)	-	-
	CH2	H8040 (Y86=1)		
	Both	H8044 (Y82=Y86=1)		
Clear equal flag (EC) (Note 1)	CH1	H8008 (Y83=1)	-	-
	CH2	H8080 (Y87=1)		
	Both	H8088 (Y83=Y87=1)		
Enable data logging mode (Note 2)	CH1	H9800	-	-
	CH2	HA800		
	Both	HB800		
Read logging data	CH1	H99xx (xx=data number)	-	-
	CH2	HA9xx (xx=data number)		
	Both	HB9xx (xx=data number)		
Clear logging data	CH1	H9A00	-	-
	CH2	HAA00		
	Both	HBA00		
Read counter value (Note 3)	Both	H8100	-	-

Note 1 These commands can be combined. For example, if you need to enable both counting and output for CH1, WY❒5 should be "**H8005**"

Note 2 Data logging function is supported by EH-CU with software version 0010 or later.

> **Data logging function** : When EH-CU detects input "M" (marker), the current value will be stored in internal memory of EH-CU up to 64 times. This function enables high accuracy pulse counting if the input M is used with a sensor besides encoder.

Note 3 Without this command, WX❒1 to WX❒4 shows the current counter value always.

5. Output assignment

WX ❒6 (Use with WY❒5 = HF100)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
>	=	>	=	>	=	>	=	>	=	>	=	>	=	>	=
CH2		CH1		CH2		CH1		CH2		CH1		CH2		CH1	
Output Y3				Output Y2				Output Y1				Output Y0			

Besides this configuration, output must be enabled by another command. (See chapter 4)

One set point can be assigned to several outputs, but one output cannot be assigned by several set points information.

Ex. ">" of CH1 is for Y0 and Y1 ➔ WY❒6 = H0022

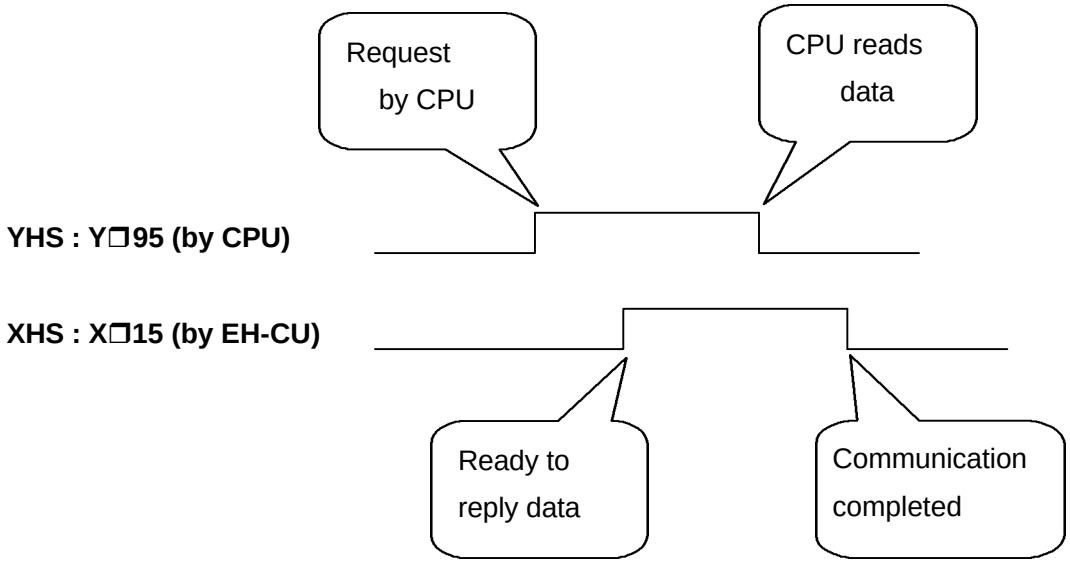
Both "=" of CH1 and ">" of CH1 are for Y0 ➔ Impossible

> **"=" detection** : Once current value reaches to set point, "Equal" flag is kept unless you set "Clear command" manually. This means, if outputs Y0-3 are configured as "=", they will keep high even though current value goes back to the below set point.
> ">" detection follows current value with real time. (No command needed to clear)

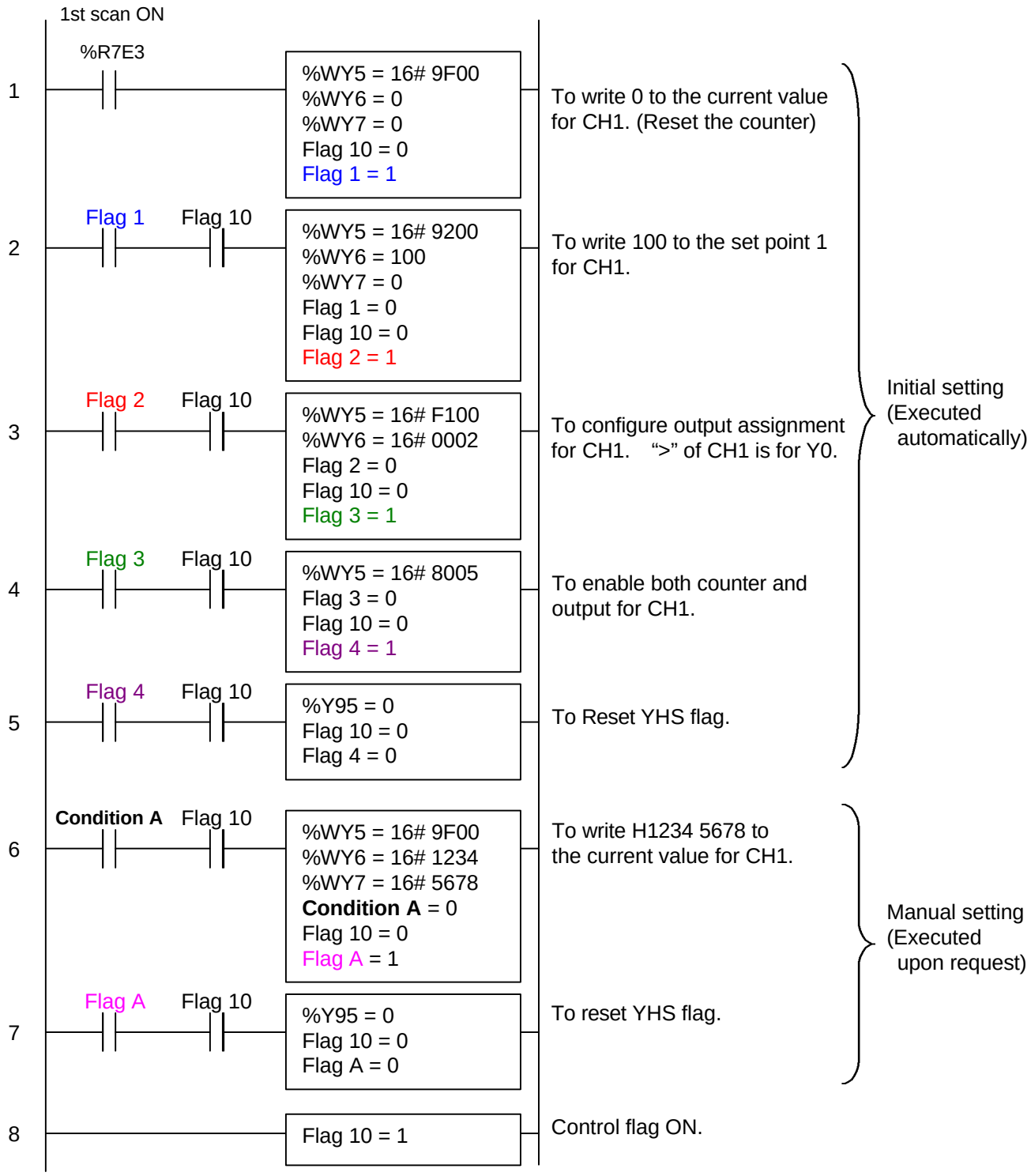
6. Handshake communication

Basically, CPU communicates with EH-CU using "Handshake communication" as below. This is the most reliable way of data communication. But from practical point of view, this communication style is not so easy to understand, and not always necessary. In most cases, data can be set by simply writing "command" and "data" in WY5-7 without the handshake communication.

The following sample program shows the simple way without handshake communication.



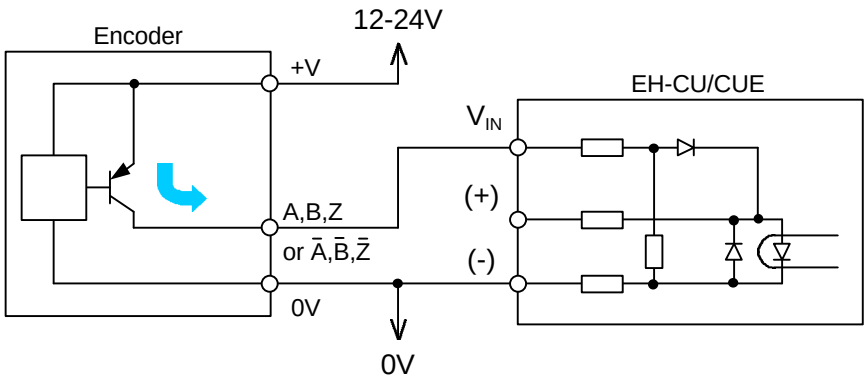
7. Sample program



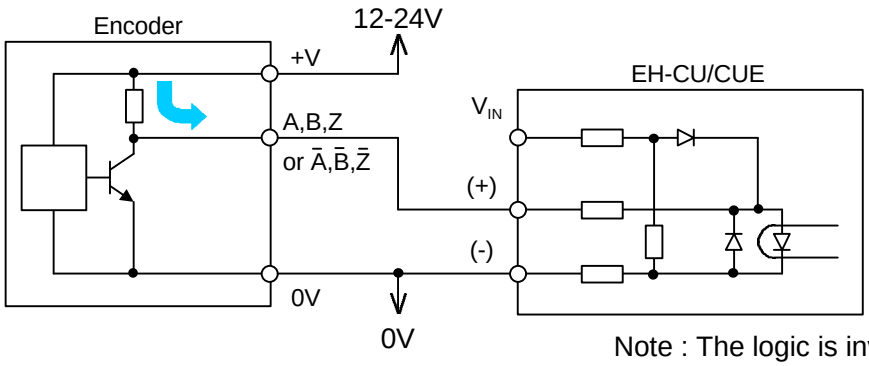
The meaning of "Flag 10" : Since EH-CPU is refresh processing, external data WY and WX are refreshed once in one scan. But internal registers R/WR/M/WM are refreshed immediately. The "Flag 10" is to execute in-line box once in one scan.

8. Wiring

PNP transistor open collector

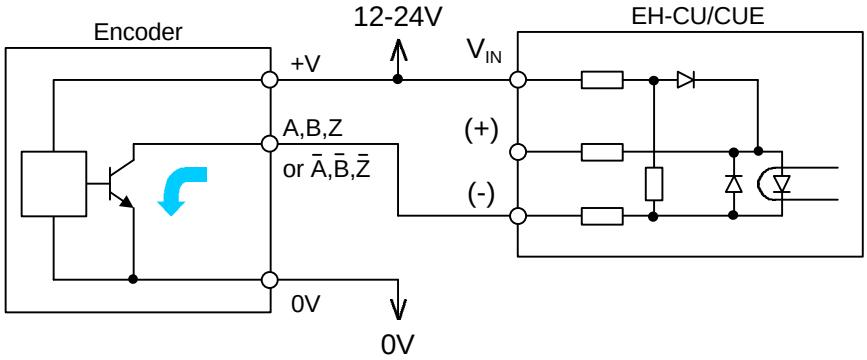


NPN transistor



Note : The logic is inverted

NPN transistor open collector



Line driver