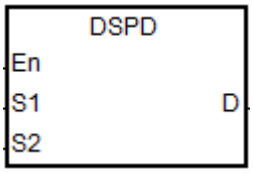
API	Instruction			Operand	Description
1008	D	SPD		S_1 · S_2 · D	Speed detection

Device	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1							○									
S_2								○	○				○	○		
D								○								

Data type	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1												●	
S_2			●				●						
D			●				●						

Pulse Instruction	16-bit instruction	32-bit instruction
-	-	AS

Graphic expression:



S_1 : Counter value

S_2 : Setting value of the cycle time

D : Detected speed value

Explanation:

1. When executing this speed detection instruciton, S_1 has to be used with the instruction DCNT to enable the high speed counter with the counter value over HC200 (including HC200).
2. S_2 is the setting value of the cycle time and in unit of millisecond (ms). Setting range is between 10~1000. When the value is out of range, the system will execute it as the minimum value or the maximum value and the PLC will not send error messages.
3. When the set value in S_2 is reached, this instruction will store the speed value in the D apporinted device. That is why the PLC will not be affected by the PLC scanning.
4. This instruction has no limitation on the editing times, but it only allows 8 sets of speed detection instructions to run simultaneously. The 9th set of the speed detection instruction or later ones will be ignored and no error messages will be sent. When executing this instruction, the set parameters of the operand will be recorded. Thus during the execution of this instruciton, editing on the parameters is not allowed.

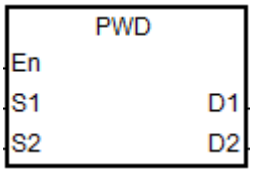
API	Instruction	Operand	Description
1009	PWD	S_1 · S_2 · D	Pulse Width Detection

Device	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1	○															
S_2								○	○				○	○		
D_1								○								
D_2		○	○	○												

Data type	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1	●												
S_2		●				●							
D_1			●				●						
D_2	●												

Pulse Instruction	16-bit instruction	32-bit instruction
-	AS	-

Graphic expression:



S_1 : Input value

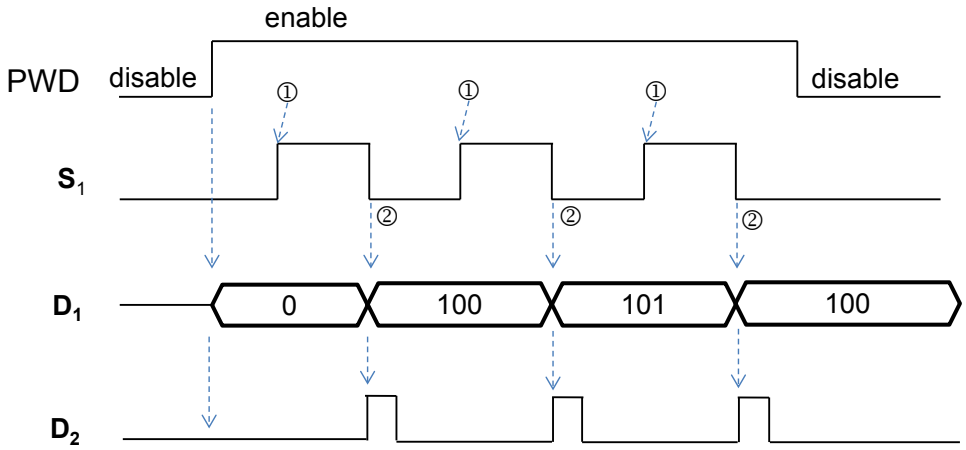
S_2 : Units of measurement

D_1 : Pulse width detection time (32-bit value)

D_2 Updated flag

Explanation:

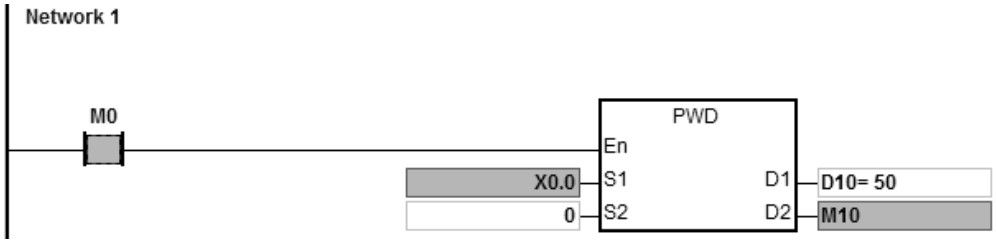
1. S_1 supports the following 6 inputs, X0.0/X0.2/X0.4/X0.6/X0.8/X0.10 but S_1 cannot share the same inputs with the high speed counter.
2. S_2 is the unit of measurement. When the value of S_2 is set to 0, it means the unit of measurement is microsecond (μs), and the acceptable input range is 1~10kHz. When the value of S_2 is set to 1, it means the unit of measurement is millisecond (ms), and the acceptable input range is 0.02~100Hz. When the value of S_2 is set to 2, it means the unit of measurement is 0.01 microsecond (μs), and the acceptable input range is 10~1MHz. When the value of S_2 is none of the above, the value will be seen as 0 and will be executed as the S_2 is set to 0.
3. D_1 is used for storing the pulse width detection time (32-bit value) and the detection range is 0~100,000,000. If the vaule is over the maximum value, it will be seen as the maximum value. If the vaule is 0, that means during the execution of this instruction, there is no input switched from ON to OFF.
4. D_2 is the updated flag. Whenever the S_1 input is switched from ON to OFF during the execution of this instruciton, the updated flag will be switched to ON in one scanning cycle time. When the D_2 is ON, users can check the value in D_1 to see the updated detection value. When execution the instruciton for the first time, the updated flag will be cleared and the value will be set to OFF.
5. See the timing diagram below for the procedures performed such as storing detection values, and updating flags during the execution of the instruction. Time to start the timer is when the S_1 input is switched from OFF to ON as it is shown in the position ① of the following diagram. Time to store the detection time is when the S_1 input is switched from ON to OFF as in the position ② of the following diagram.



5. This instruction has no limitation on the editing times, but it only allows 6 sets of pulse width detection instructions to run simultaneously. The 7th set of the pulse width detection instruction or later ones will be ignored and no error messages will be sent. When executing this instruction, the set parameters of the operand will be recorded. Thus during the execution of this instruciton, editing on the parameters is not allowed.
6. Before executing this instruction, please check the input hardware response time and the pulse time set in HWCONFIG. For example, when the value in the **S_2** is set to 0 or 2, that means the unit of measurement is microsecond (μs). And the **S_1** input value should set to be 0 to disable the Input Point Filter Time in HWCONFIG.

Example 1:

There is a pulse signal of 10kHz in the input X0.0. The instruction PWD can detect the input signal of the X0.0 and have the pulse width shown in D10/D11 (32-bit data), the time is set to 0 and the width detected from D10 is 50μs.



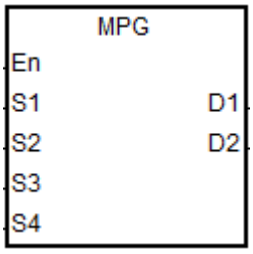
API	Instruction	Operand	Description
2722	MPG	S_1 · S_2 · S_3 · S_4 · D_1 · D_2	Manual Pulse Generator

Device	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1							○									
S_2								●								
S_3								●								
S_4								●								
D_1		○														
D_2								●								

Data type	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1												●	
S_2			●				●						
S_3		●			●	●							
S_4		●			●	●							
D_1	●												
D_2			●				●						

Pulse Instruction	16-bit instruction	32-bit instruction
-	AS	-

Graphic expression:



S_1 : The source of high speed counter

S_2 : Input speed display

S_3 : Multiplier setting for the speed ratio of the input and output

S_4 : Divisor setting for the speed ratio of the input and output

D_1 : Pulse output device

D_2 : Output speed display

Explanation:

1. S_1 high speed counter supports hardware counters with over HC200 but not counters without specified HC numbers. High speed counters should be activated by the instruction DCNT and when the input signals coming in, the instruction will start positioning output control according to the setups. The 6 sets of high speed counters are listed below (before executing the instruction, all the values in the high speed counters should be cleared.).

Input Group	1	2	3	4	5	6
Input Point (A-phase)	X0.0	X0.2	X0.4	X0.6	X0.8	X0.10
Input Point (B-phase)	X0.1	X0.3	X0.5	X0.7	X0.9	X0.11
High Speed	HC202	HC206	HC210	HC214	HC218	HC222

Counter Value						

2. **S_2** displays the input pulse speed. Users can use 32-bit variable to declare. The unit of measurement is 1Hz, and the acceptable input speed range is 1~200kHz. When the input speed is less than 1Hz, it will be regarded as 1Hz; when the input speed is over 200kHz, it will be seen as 200kHz.
3. **S_3** is the multiplier setting for the speed ratio of the input and output and **S_4** is the divisor setting for the speed ratio of the input and output. The acceptable range for **S_3** and **S_4** is 1~255. For values which are out of range will be treated as the minimum or the maximum of the range, depending on how small or how great the value is. The ratio formula for the output pulse rate is input pulse rate x **S_4** / **S_5**. For instance, the input value versus the output value is 5:3; the multiplier will be K3 and the divisor will be K5. If the ratio is 1:2 and then the multiplier will be K2 and the divisor K1.
4. The output points for **D_1** are Y0.0、Y0.2、Y0.4、Y0.6、Y0.8、Y0.10 and it will occupy 2 points in a row for output. The output sets and the output special register modes are listed below.

Output axis number	1	2	3	4	5	6
D_1+0 output points	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
D_1+1 output points	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
Output modes	SR462	SR482	SR502	SR522	SR542	SR562

5. **D_2** displays the output pulse speed. Users can use 32-bit variable to declare. The unit of measurement is 1Hz, and the acceptable output speed range is 1~200kHz. For values which are out of range will be treated as the minimum or the maximum of the range, depending on how small or how great the value is.
6. This instruction has no limitation on the execution times. But during execution, the appointed high speed axis cannot be occupied by other instructions. Otherwise it will not be executed.

*Note:

1. Input pulse speed is calculated according to the input pulse width of the positive half cycle (ON). When the pulse width ON versus pulse width OFF is less than 1:1, it will detect the pulse speed according to the pulse width ON.
2. Input pulse ON mean the input point light is ON. When using with the manual pulse generator (MPG) users need to make sure when the generator stops, the input light should be OFF.
3. Before executing this instruction, users have to clear the values of the high speed counter and the value should be 32-bit signed numbers.

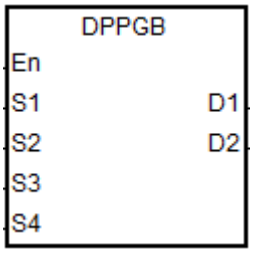
API	Instruction			Operand	Description
2723	D	PPGB		S_1 · S_2 · S_3 · S_4 · D_1 · D_2	Point to point go back

Device	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	"$"	F
S_1								●								
S_2								●								
S_3								●								
S_4								●								
D_1		○														
D_2			●													

Data type	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						
S_4									●				
D_1	●												
D_2	●												

Pulse Instruction	16-bit instruction	32-bit instruction
-	-	AS

Graphic expression:



S_1 : Relative target position A

S_2 : Relative target position B

S_3 : Target speed

S_4 : Target speed ratio adjusted value (floating point value)

D_1 : Pulse output device

D_2 : Changeable target speed indicator

Explanation:

1. This high speed output instruction is used for a movement going back and forth between 2 target positions, applicable for warping machines in the textile industry and winding & binding machines in the cable industry, and many more.
2. Before the first execution of the instruction, the target positions of S_1 and S_2 should be appointed in the first place so that the instruction can operate beforehand and can easily switch to the next output. After the outputting is done, users can modify the next target position. For the target position that is being outputting, the position cannot be changed. For example, A is outputting in the target position of S_1; the target position of S_1 can be modified. But it will only execute the new target position in the next run. The same rules apply to S_2.
3. S_3 is the target speed (32-bit integer), and S_4 is the target speed ratio adjusted value (floating point). The actual speed is the result of multiplying the values of S_3 and S_4 and then has the result rounded down to the 32-bit integer. The acceptable input speed range is 1~200kHz. For values which are out of range will be treated as the minimum or the maximum of the range, depending on how small or how great the

value is. The ratio formula for the actual target speed is **S_3** x **S_4**. For instance, target speed is 1kHz, the adjusted floating point is 1.2345 and the actual will be 1234Hz.

4. When the outputting is ongoing, the target speed and the adjusted ratio can be modified, and the result will be updated to the actual outputting speed once the instruction is being executed. But it is suggested that do not change the target speed too greatly otherwise the calculated deceleration will be affected.

5. The output points for **D_1** are Y0.0、Y0.2、Y0.4、Y0.6、Y0.8、Y0.10 and it will occupy 2 points in a row for output. The output sets and the output special register modes are listed below.

Output axis number	1	2	3	4	5	6
D_1+0 output points	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
D_1+1 output points	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
Output modes	SR462	SR482	SR502	SR522	SR542	SR562

6. **D_2** is the flag indicating the changeable target speed and the adjusted ratio. When this flag is ON that means the executing target speed can be modified. When the flag is switching from ON to OFF that means it is now decelerating and the current target speed will be seen as the next output target speed.

7. This instruction has no limitation on the execution times. But during execution, the appointed high speed axis cannot be occupied by other instructions. Otherwise it will not be executed.

Output timing diagram: