

Pre-register Function

Configurable number of axes	Four axes can be configured independently.
-----------------------------	--

Description

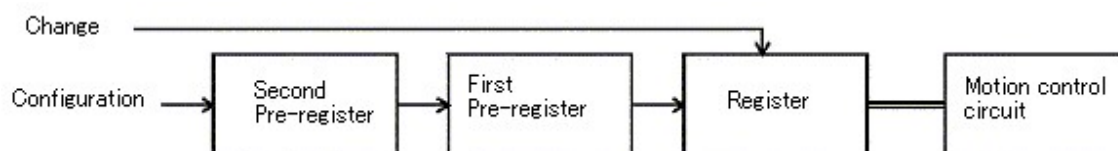
The pre-register is a register to set the subsequent motion data during operation.

The function is included in each of the following register.

Motion register such as moving quantity, velocity, and acceleration and deceleration rate,
Comparator 5 configuration register,
Start command configuration register.

Controller IC : The pre-register of the PCL6045 consist of the following two stages, and performs FIFO.

Each axis has this pre-register.



2nd pre-register -> 1st pre-register -> The register transmits (copies) data when the command is written or the previous configuration motion is completed.

Writing data to pre-register

- 1 When data are written to the second pre-register with the second and first pre-registers empty, the data is written to the second and first pre-registers and register.

Even if you write motion data to the same configured axis repeatedly in this status, you cannot configure the subsequent motion. The data in the second and first pre-registers and register are overwritten.
- 2 === Issues the start command for axis specified in procedure 1. ===

After the issue of the start command, the motion start according to the contents of the register.
-> The first pre-register becomes empty.
- 3 By writing the motion data in the operating axis specified in procedure 1 and 2, the subsequent data to the second pre-register.
(If the contents are the same as the last time, skip the procedure.)

At this time, because the first pre-register is empty, the subsequent motion data is written in the second and first pre-register.
- 4 ===Issues the start command for axis specified in procedure 3. ===

By issuing the start command for the motion specified in procedure 3,
->the first pre-register becomes stored status.
- 5 By writing the motion data in the same axis as above, the data are written to the second pre-register one after another. .
(If the contents are the same as the last time, skip the procedure.)

		<p>->Because the first pre-register is stored in procedure 4, the data are written one after another in only the second pre-register.</p>		
	6	<p>=== Issues the start command for axis specified in procedure 5. ===</p> <p>By issuing the start command for the motion specified in procedure 5, ->the second pre-register becomes stored status.</p>		
	7	<p>When the motion configured and started in procedures 1 and 2 is completed, the data in the register is transmitted in to the following registers. second pre-register -> first pre-register -> register</p> <p>-> Starts the subsequent motion specified in procedures 3 and 4. -> Copies the motion one after another specified in procedures 5 and 6.</p>		
	8	<p>Because the data in second pre-register is copied to the first pre-register, →The second pre-register becomes empty and the subsequent motion can be written.</p>		
Comments	1	<p>You can confirm in the main status that the second pre-register described above is full of data.</p> <table border="1"><tr><td>Multi function DLL</td><td>Confirm the status with the MtnGetStatus1 function.</td></tr></table>	Multi function DLL	Confirm the status with the MtnGetStatus1 function.
Multi function DLL	Confirm the status with the MtnGetStatus1 function.			
	2	<p>Interrupt can be generated when the second pre-register becomes empty after completion of the motion.</p>		
Configuration		<p>Multi-function DLL</p> <p>MtnSetMotion MtnSetMotionLine MtnSetMotionArc MtnSetSyncLine</p> <p>When the above functions are executed, data are written in the 2nd and first pre-registers, and register. (Status of description 1 above)</p> <p>The MtnStartMotion function issues the start command and starts operation according to the register. (Status of description 2 above)</p> <p>Execute the following functions for operating axes to write the next motion data to the second pre-register. (Status of description 3 above)</p> <p>MtnSetMotion MtnSetMotionLine MtnSetMotionArc MtnSetSyncLine</p> <p>Issue the the start command by the MtnStartMotion function. (Status of description 4 above)</p> <p>Execute the following functions to write data after another to the second pre-register. (Status of description 5 above)</p>		

[MtnSetMotion](#)
[MtnSetMotionLine](#)
[MtnSetMotionArc](#)
[MtnSetSyncLine](#)

Issue the start command by the [MtnStartMotion](#) function.
([Status of description 6 above](#))

© 2003, 2015 Interface Corporation. All rights reserved.

[\[Top\]](#)