Date: March 2, 2018

To: DB70237-0

Subject: Transition from PCL61x3 to PCL61x5

Nippon Pulse Motor Co., Ltd.

Thank you for your business with Nippon Pulse Motor.
We will show you the differences between the PCL61x3 series (PCL6113/6123/6143) and the 61x5 series (PCL6115/6125/6145), and some precautions to take when you intend a transition as follows:

****

1. Outline

PCL61x5 series is functionally up-graded from the PCL61x3 series, however, some functions are different, and are not complete upward compatible.

In terms of software, the software developed for PCL61x3 can control PCL61X5 in most cases.
In terms of hardware, the PCL6115 and PCL6145 share the package and the terminal assignments with the PCL6113 and PCL6143 respectively. However, the package and terminal assignment of the PCL6123 and the PCL6125 are different each other.
Also, when you mount the PCL6115 and PCL6145 on the boards for the PCL 6113 and PCL6143 respectively, check the section“2. Notes on transition”.

2. Notes on transition

Please confirm the following differences between the models on transitions.
If a problem occurs from the difference with your system, it is necessary to change software or hardware.

2-1. External dimension

The package of the PCL6115 is the same as the PCL6113.
The package of the PCL6125 is different from the PCL6123.
The package of the PCL6145 is the same as the PCL6143.

2-2. Increase in current consumption

The current consumption is different from PCL61x3.

Model	Condition	PCL61x3	PCL61x5
PCL6113,PCL6115	Drive 1-axis at 15Mpps, No load	36 mA Max	37 mA Max
PCL6123,PCL6125	Drive 2-axis at 15Mpps, No load	77 mA Max	75 mA Max
PCL6143,PCL6145	Drive 4-axis at 15Mpps, No load	180 mA Max	139 mA Max

2-3. Change the ORG sampling timing in origin return operations

In PCL61x5, the origin position can shift by one pulse from the position with PCL61x3.
In PCL61x3, the time filter and the distance filter are used together as the ORG input filter.
In the distance filter, the ORG input was sampled at the ON timing of output pulses, and if the ORG sensor is not ON for more than one pulse period, it is recognized as a noise and is ignored.
It is difficult to use a proximity sensor with a short ON distance range. Therefore, the distance filter is eliminated, and only the time filter is used in PCL61x5.

2-4. Chang of position management control method by PCS input

When RENV1.PCSM=1, PCS input terminal can be used as the CSTA input only for the own axis.

In PCL61x3, the original PCS function operates even in PCSM=1, and PCS signal is recognized as CSTA input only for own axis. At the same time, it is recognized as PCS input, and also operate as a start signal of the target position override 2.

In PCL61x5, when PCSM = 1, it is not recognized as the target position override.

2-5. Change of ERC output condition

The clear signal of the deviation counter (ERC) in a motor driver is output by the ERC output command at an origin return operation and an error stop. The output condition at an error stop was changed:

In PCL61x3, when RENV1.EROE = 1, the ERC is output only when immediate stop was selected for stop method.

In PCL61x5, when RENV1.EROE = 1, it is output only when the stop speed is higher than the FL speed, and it is not affected by the stop method.

2-6. Overwrite of start command for the next operation

Since there is only one pre-register, the start command of the next operation can be written during operation, but the start command for the one after the next operation cannot be written.

In PCL61x3, if the start command is further written when the pre-register is in the fixed state (the start command for the next operation have been already written), the start command for the next operation was overwritten.

In PCL61x5, the start command that is written when the pre-register is fixed will be ignored.

2-7. Change execution axis of CMEMG command

When an emergency stop signal is input from the CEMG terminal, the all axes stop emergently.

The CMEMG command is equivalent to the CEMG input.

In PCL61x3, only the axis on which the CMEMG command was written is stopped, so when writing CMEMG command, it was necessary to enable the all axes with COMW (11 to 8). Also, an error interrupt (REST.ESEM) that occurs when the CEMG is input while axes stop, did not occur with CMEMG command when an axis stopped.

In PCL61x5, all axes stop emergently if CMEMG command is written to any one axis without setting the all axes with COMW (11 to 8). In addition, error interrupt (REST.ESEM) will occur if CMEMG command is written while an axis stops.

2-8. Change monitoring function of PCS terminal

Status of PCS input terminal can be checked by RSTS.SPCS bit.

Also, if RENV1.PCSM = 1, PCS terminal can operate as CSTA signal only for its own axis.

In PCL61x3, when RENV1.PCSM = 1, RSTS.SSTA bit is changed based on PCS terminal state, but the RSTS.SPCS bit does not change.

In PCL61x5, regardless of RENV1.PCSM setting, RSTS.SPCS bit can check the status of PCS terminal and RSTS.SSTA bit can check the status of CSTA terminal.

Model	RENV1.PCSM=0		RENV1.PCSM=1	
	RSTS.SPCS	RSTS.SSTA	RSTS.SPCS	RSTS.SSTA
PCL61x3	PCS terminal monitor	CSTA terminal monitor	Always 0	PCS terminal monitor
PCL61x5	PCS terminal monitor	CSTA terminal monitor	PCS terminal monitor	CSTA terminal monitor

2-9. RPLS monitor value during stop

The RPLS register shall be read when you need to check the number of remaining pulses during a positioning operation.

In PCL 61×3 it was not expected to be read while an axis stopped, so if RMV register or RMD register was written while an axis stopped, it changed to a value other than the number of remaining pulses.

In PCL61x5, even if you change the RMV and RMD register while an axis stops, the RPLS value at the last stop is maintained.

2-10. Definition added to the main status

By adding software limit function, not defined (read value is 0) bits 10 and 11 become SCP 3 and SCP 4, respectively.

In PCL61x3, bit 11~10 are always “00” because they are not defined.

In PCL61x5, they vary based on the software limit status.

2-11. Change the definition of upper bits of RSDC register monitor value

RSDC is a read-only register for the ramp down point value.

Since the range of ramp down point is 24 bits of 0 to 16,777,215, all of the unused bits, 31 to 24 are fixed to 0 in PCL 61x3.

When the ramp down point is set “manual” (RMD.MSDP = 1), it is equal to the RDP setting value.

When the ramp down point is set “automatic” (RMD.MSDP = 0), the offset value (-8,388,608 to + 8,388,607), which is the RDP setting value, becomes the initial value of ramp down point.

After the start, it counts up in synchronization with the output pulse and it may increase up to 16,777,215.

However, it was impossible to distinguish the values of -8,388,608 (00800000h) to -1 (00FFFFFFh) and +8388,608 (00800000h) to 16,777,215 (00FFFFFFh) as the RSDC read values.

However, inside the circuit, it is recognized and is handled as signed 25-bit numbers, so there is no problem.

Therefore, in PCL61x5, the upper bit of RSDC is changed from 0-fixed to sign extension, so that it can be read as signed 32 bit data. The numerical range is -8,388,608 (FF800000h) to +16,777,215 (00FFFFFFh).

2-12. Stop interrupt in PA / PB, + DR/-DR operations

When RENV2.IEND = 1, stop interrupt (MSTSW.SENI) was designed to occur at normal stop/emergency stop. However, when PCL61x3 was set in the following three operation modes, the INT signal was not output at the time of command stop or error stop.

1) Continuous operation by pulser (PA /PB) input (RMD.MOD = 01h)
2) Continuous operation by external signal (+DR/-DR) input (RMD.MOD = 02h)
3) Positioning operation by external signal (+DR/-DR) input (RMD.MOD = 56 h)

In PCL61x5, the INT signal is output even in the above mode.

2-13. Stop interrupt while stopped

In PCL61x3, even while an axis stops in the setting of RENV2.IEND=1, stop interrupt was generated by writing STOP command or DSTP command.

In PCL61x5, stop interrupts never occur while an axis stops.

3. Improved functions

3-1. Feed amount setting range is expanded.

In PCL61x3. setting range of feed amount (PRMV, RMV) and the comparison value (RCMP1, RCMP2) were 28 bits

In PCL61x5, it is expanded to 32 bits.

3-2. Acceleration /deceleration setting range are extended

In PCL61x3, setting ranges of acceleration rate (PRUR, RUR) and deceleration rate (PRDR, RDR) were 14 bits.

In PCL61x5, they are extended to 16 bits.

3-3. Software limit function is added

There was no software limit function in PCL61x3.

Software limit function is added in PCL61x5.

Add a setting register for the (+) software limit value (RCMP 3: 32 bits) and the (-) software limit value (RCMP 4: 32 bits), and software limit is controlled by comparison with COUNTER 1 or COUNTER 2.

3-4. Automatic reset stop function of the bit of MSTSW.SENI and MSTSW.SEOR is added.

In PCL61x3, resetting the SENI and SEOR bits of the main status was performed by automatic reset with reading operations.

In PCL61x5, beside the automatic reset at reading, a function to reset the bit by a command is added.

Then RENV2.MRST = 1, the automatic reset function stops, and the SENI bit is reset by SENIR command (2Dh), and the SEOR bit is reset by SEORR command (2Eh).

However, in the case of serial bus I/F, the automatic reset function cannot be used since the RENV2.MRST bit is fixed to 1.

3-5. Automatic reset stop function of REST and RIST registers is added

In PCL61x3, the method of resetting the error interrupt status (REST register) was only automatic reset by reading the REST register. The method of resetting the event interrupt status (RIST register) is only an automatic reset by reading RIST register.

In addition to the automatic reset function by reading, the PCL61x5 has an additional function to reset only a specified bit.

When RENV2.MRST=1, the automatic reset function stops and any specified bit can be reset by the REST register write command (WREST: B2h) and the RIST register write command (WRIST: B3h).

Please note that only the bit where the write data is set to 1 can be reset.

However, in the case of a serial bus I/F, the automatic reset function cannot be used since the RENV2.MRST bit is fixed to 1.

3-6. ID monitor function to confirm the model is added.

In PCL61x3, there was no way to confirm the model (LSI name).

In PCL61×5, the ID code check function to determine the model is added.

If the RMG register read command (RRMG: D5h) is written following the ID monitor command (IDMON: 03h), bits 31 to 16 of the read data become the ID code.

The ID code is 03E0h for PCL 6115, 03F0h for PCL 6125, 0400h for PCL6145.

In PCL61x3, bits 31 to 16 of read data are 0000h for all three models

3-7. Serial interface(I/F) for CPU connection is added.

In PCL61x3, there was only parallel I/F connection of 8 and 16 bits available.

In PCL61×5, the SPI-I/F (4-wire serial I/F) is also available

Additionally, the communication clock (SCK) frequency for serial I/F should be up to 1/1.5 or less of the reference clock frequency.

【Note】

Switching between the conventional parallel I/F and serial I/F is done with the input terminals. At reset release, it becomes serial I/F in low level, and it becomes parallel I/F in other cases. Depending on the model of the CPU, initial status of the $\overline{RD}$, $\overline{WR}$ output terminals can be set to floating (to general purpose input ports), but the $\overline{CS}$, $\overline{RD}$, $\overline{WR}$ input terminals of PCL61x5 have built-in pull-up resistors, so they are recognized as in high level.

3-8. Shared port that can be used only for serial I/F is added.

When using PCL61x5 with serial I/F, the data bus (D0 to D15) are not used. Therefore, they are designed to be used as 16-bit general purpose ports. The port is not assigned for each axis, and it is used by the all axes in common in the case of multi-axis products (PCL6125, PCL6145). For this reason, this general-purpose port is shown as "shared port", and is distinguished from the general-purpose port assigned for each axis. Additionally, input and output can be set for each bit.

3-9. Interrupt in origin return operation using EZ input is added.

When RENV2.ORM = 1, it will perform an origin return operation using the ORG input and the EZ input. At high speed start, the deceleration will start with ORG=ON, and the axis will stop immediately with the specified number of EZ inputs, however, the speed at stop is not confirmed. Therefore, it is necessary to specify the number of EZ which can surely complete deceleration.

In PCL 61×3, even if the axis stopped in the middle of deceleration due to the insufficient EZ number setting, it could not be detected.

In PCL61x5, a function to generate an event interrupt when an axis stops in the middle of deceleration without completion due to high-speed origin return operation with RENV2.ORM = 1.

3-10.Start interrupt function is added.

There will be some delay time between writing the start command and actual start due to wait for the completions of the direction change timer and the ERC control timer.

In PCL61x3, you needed to repeat checking the motion status monitor (RSTS.CND3 - 0).

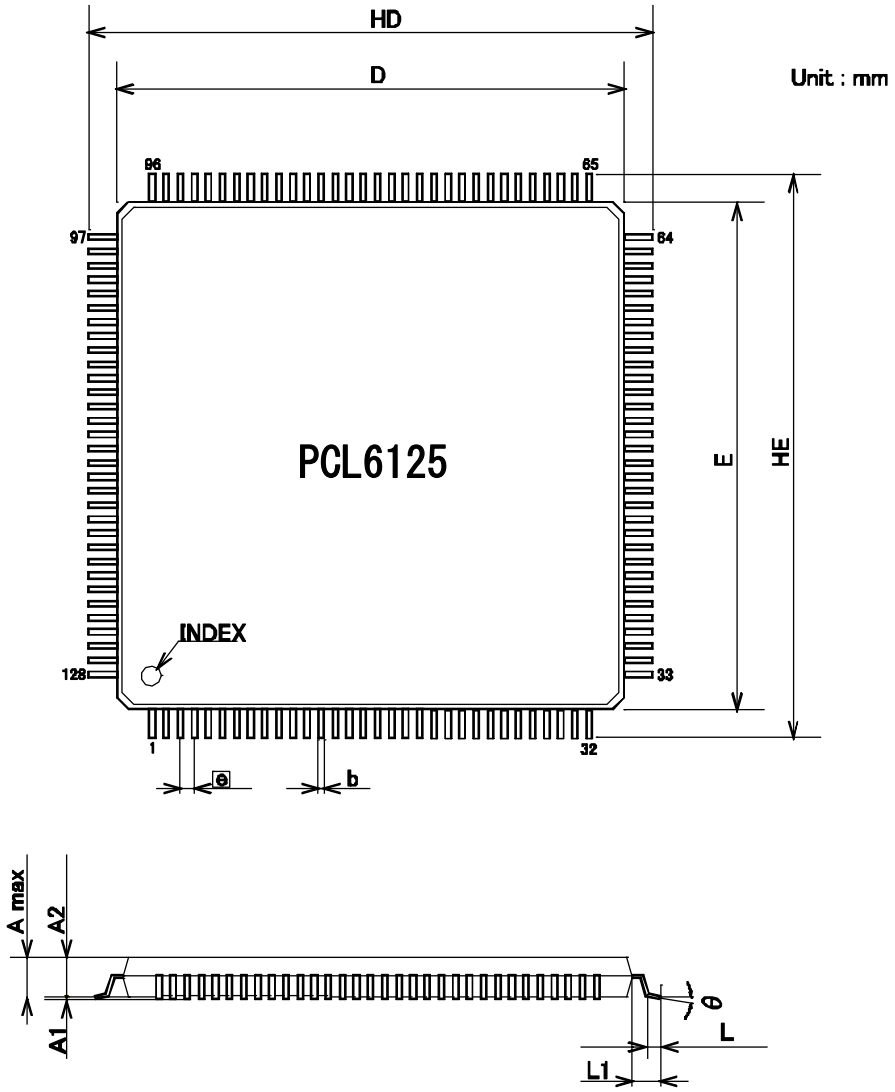
However, in PCL61x5, a function is added to generate an event interrupt at the start of actual operation.

3-11. Latch circuit of current position counter is added.

Two latch circuits for current position counters are added. (RLTC 3, RLTC 4)

4. Difference in external dimensions (PCL 6115/6145 are the same as PCL 6113/6143).

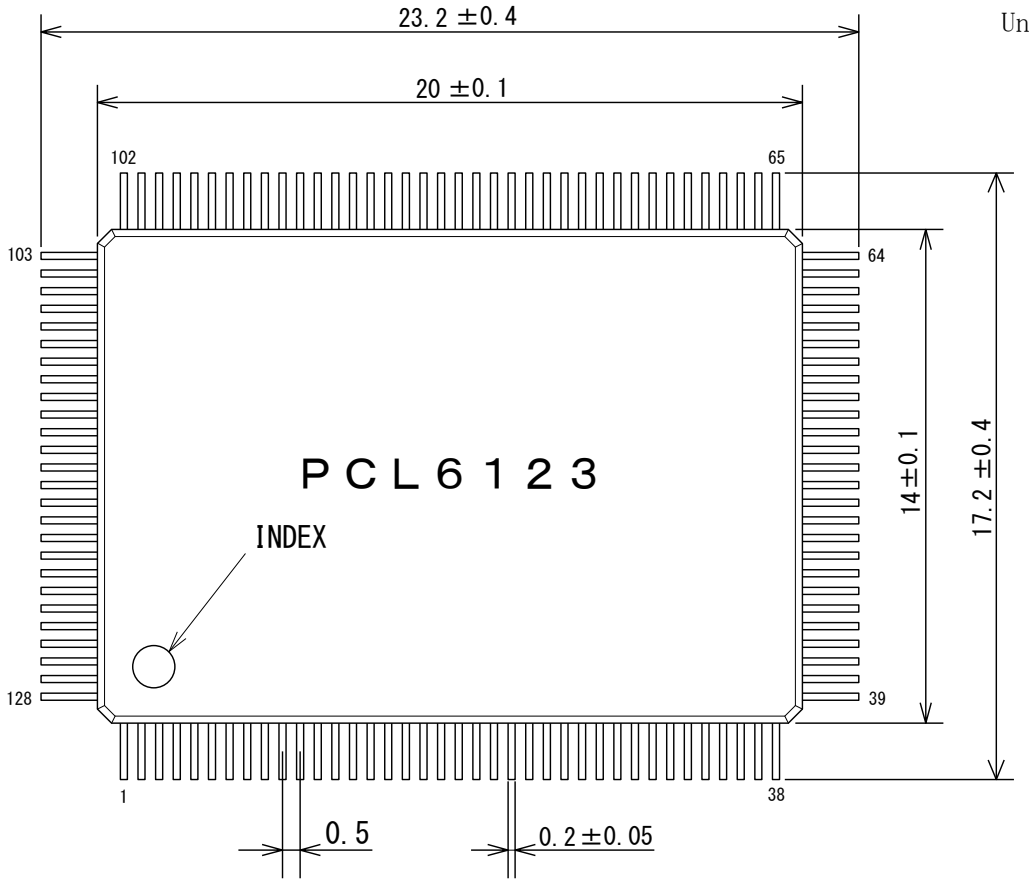
(PCL6125 dimensions)

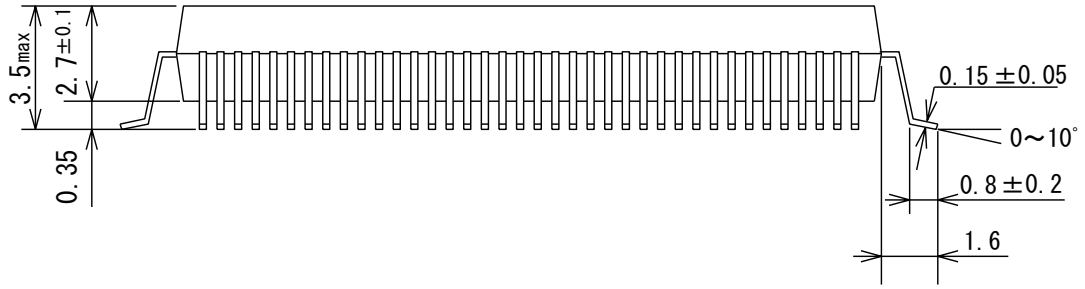


Symbol	Dimension in mm		
	Min.	Nom.	Max.
E	-	14	-
D	-	14	-
HE	-	16	-
HD	-	16	-
e		0.4	
b	0.13	-	0.23
A max	-	-	1.7
A1	-	0.1	-
A2	-	1.4	-
L	0.3	-	0.75
L1	-	1	-
θ	0°	-	10°
y	-	-	0.08

(PCL6123 dimensions)

Unit: mm





5. Added commands

5-1. ID Monitor command

COMBO	Symbol	Description
03h	IDMON	Make the ID code readable by RRMG command

If you read the RMG register immediately after writing this command, you can check the model ID code with bits (31 to 16). (ID code is 0000h in PCL61x3)

Model	Model ID code
PCL6115	03E0h
PCL6125	03F0h
PCL6145	0400h

5-2. Status reset command

COMBO	Symbol	Description
2Dh	SENIR	Reset the stop interrupt status (MSTSW.SENI)
2Eh	SEORR	Reset the target position override failure status (MSTSW.SEOR)

5-3. Register control command

Description	Bit length	Name	Read command		Write command	
			COMB0	Symbol	COMB0	Symbol
Environment setting 4	32	RENV4	DFh	RRENV4	9Fh	WRENV4
Latch data 3	32	RLTC3	EFh	RRLTC3		
Latch data 4	32	RLTC4	F0h	RRLTC4		
Obtain error interrupt cause	11	REST	F2h	RREST	B2h	WREST
Obtain event interrupt cause	18	RIST	F3h	RRIST	B3h	WRIST
Shared port (GP0 - 15) specification setting	16	RGPM	FAh	RRGPM	BAh	WRGPM
Shared port (GP0 - 15) data	16	RGPD	FBh	RRGPD	BBh	WRGPD

5-4. Latch 3 and 4 control command

COMBO	Symbol	Description
3Ch	LTC3E	Enable RLTC3 latch operation
3Dh	LTC4E	Enable RLTC4 latch operation
3Eh	LTC3D	Disable RLTC3 latch operation
3Fh	LTC4D	Disable RLTC4 latch operation

6. Added/ Revised registers

To enhance the function, the definitions of bits (shaded in pink and blue), which have not been defined in PCL61x3, are added in PCL61x5. Since PCL61x3's manual describes them to set as '0' fixed, any problems should never occur in using PCL61x3 software. However, please check if the bits are set to '0' when the operation is wrong.

6-1. RMG register （Bit definition is added）

Bit definition for ID monitor is added.

IDMON command must be written immediately before reading ID monitor value.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	RMG data											

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IDDT(ID motitor value)															

Bit	Bit synbol	Description
31-16	IDDT	The ID code can be read only when the RRMG command (D5h) is written immediately after the ID monitor command (IDMON: 03h). The read value at other time is 0000h.

6-2. RENV2 register (Bit definition is added)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
POFF	EOFF	CSPO	P7M	P6M	P5M	P4M1	P4M0	P3M1	P3M0	P2M1	P2M0	P1M1	P1M0	P0M1	P0M0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MRST	IEND	ORM	EZL	EZD3	EZD2	EZD1	EZD0	PDIR	PINF	PIM1	PIM0	EDIR	EINF	EIM1	EIM0

Bit	Bit Symbol	Descrition
31	MRST	Control the automatic reset function immediately after reading MSTSW, REST, RIST. 0: Automatic reset enabled 1: Automatic reset disabled In case of serial bus I / F, it is fixed to "1".

Timing	When MRST=0	When MRST=1
Immediately after MSTSW reading	Reset SENI,SEOR bit automatically	Not be reset automtically
Immediately after REST reading	Automatic rest of all bits	Same as above
Immediately after RIST reading	Automatic rest of all bits	Same as above

6-3. RENV3 register (Bit definition is added)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
C2S1	C2S0	C1S1	C1S0	C2RM	CU2R	LOF2	CU2L	C1RM	CU1R	LOF1	CU1L	CU2H	CU1H	CIS2	CIS1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	SLCU	SLM1	SLM0	SYI1	SYI0	SYO3	SYO2	SYO1	SYO0

Bit	Bit symbol	Description
23 - 22	SLM1 - SLM0	00：Stop the software limit function 01：Event interrupt occures at the software limit position(not stop) 10：Immediate stop at the software limit position and an error interrupt occurs. 11：Decelerate and stop at the software limit position, and an error interrupt occurs
24	SLCU	Select the counter for software limit management 0：COUNTER1,　1：COUNTER2

6-4. RENV4 register　(Bit definition is added)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
L4F1	L4F0	L4MD	L4DT	L4TL	L4T2	L4T1	L4T0	L3F1	L3F0	L3MD	L3DT	L3TL	L3T2	L3T1	L3T0
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Bit symbol	Description
2 - 0	L3T2 - L3T0	Select the trigger signal to latch in RLTC3 000: Disable　001:LTC input　010:ORG input　011:EZ input 100:P4 input　101:P5 input　110:P6 input　111:P7 input
3	L3TL	Select the enabled edge of trigger signal for RLTC3　* NOTE 0:Falling edge　1:Rising edge
4	L3DT	Select the counter to latch in RLTC3. 0:COUNTER1(RCUN1)　1:COUNTER2(RCUN2)
5	L3MD	Latch operation mode selection for RLTC3 0: Latch once between LTC3E command and LTC3D command 1: Latch every time between LTC3E command and LTC3D command
7 - 6	L3F1 - L3F0	Select the filter of trigger input for RLTC3(LTC,ORG,EZ,P4 - P7) 00: No filter (Input pulse width ＞ CLK cycle) 01: Input with a pulse width less than 3.2 us is ignored 10: Input with a pulse width less than 25 us is ignored 11: Input with a pulse width less than 200 us is ignored *NOTE. It has nothing to do with the setting of RENV1.FLTR, RENV2.EINF.
10 - 8	L4T2 - L4T0	Select the trigger signal to latch in RLTC 4 000: Disable　001:LTC input　010:ORG input　011:EZ input 100:P4 input　101:P5 input　110:P6 input　111:P7 input
11	L4TL	Select the enabled edge of trigger signal for RLTC 4　*NOTE: 0:Falling edge　1:Rising edge
12	L4DT	Select the counter to latch in RLTC4. 0:COUNTER1(RCUN1)　1:COUNTER2(RCUN2)
13	L4MD	Latch4 operation mode selection for RLTC4 0: Latch once between LTC4E command and LTC4D command 1: Latch every time between LTC4E and LTC4D command
15 - 14	L4F1- L4F0	Select the filter of trigger input for RLTC4 (LTC, ORG, EZ, P4 - P7) 00: No filter (Input pulse width ＞ CLK cycle) 01: Input with a pulse width less than 3.2 us is ignored 10: Input with a pulse width less than 25 us is ignored 11: Input with a pulse width less than 200 us is ignored *NOTE. It has nothing to do with the setting of RENV1.FLTR, RENV2.EINF.
31 - 16	Not defined	(Alway set to “0”)

* NOTE: When writing an enable command (LTC3E, LTC4E) after changing the trigger signal input specifications, make sure to wait for the filter time set by L3F (or L4F) to elapse.
If an enable command is written before the filter time elapses, an extra latch operation will occur.
To shorten waiting time, change the input specification of the trigger signal with the filter

setting to "00", and then change only the filter value again.
(When the filter setting is "00", time to write to the register is longer than filter time, so time wait processing is not necessary.)

6-5. RIRQ register (Bit definition is added)

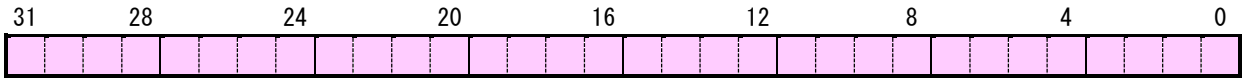
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	IRBY	IREZ	IRSA	IRDR	IRSD	IROL	IRLT	IRC2	IRC1	IRDE	IRDS	IRUE	IRUS	IRNM	IREN

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	0	0	0	0	0	IRL4	IRL3

Bit	Bit symbol	Description
13	IREZ	When stopped in the middle of deceleration in a high-speed origin return operation with RENV2.ORM = 1.
14	IRBY	When an axis becomes in operation (at start)
15	Not defined	(Always set to “0”)
16	IRL3	When latch to RLTC3
17	IRL4	When latch to RLTC4
31 - 18	Not defined	(Always set to “0”)

6-6. RLTC3 register (Register is added)

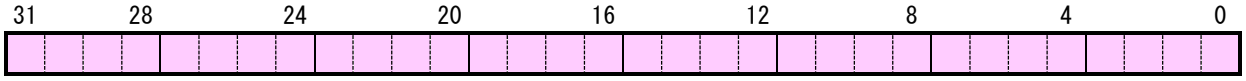
Stored data of latch 3 (Read only)



The numerical range is -2,147,483,648～+2,147,483,647.

6-7. RLTC4 register (Register is added)

Stored data of latch 4 (Read only)



The numerical range is -2,147,483,648～+2,147,483,647.

6-8. RSTS register (Bit definition is added)

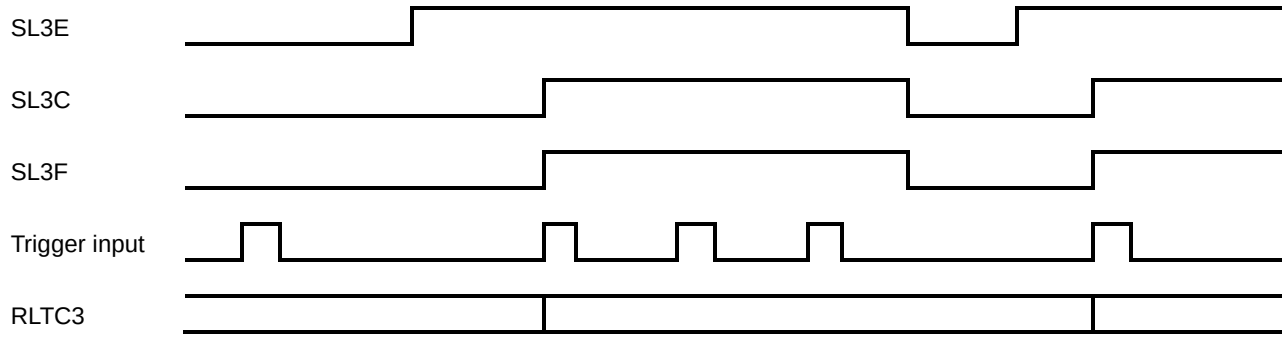
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SINP	SDIN	SLTC	SDRM	SDRP	SEZ	SERC	SPCS	SEMG	SSTP	SSTA	SCD	CND3	CND2	CND1	CND0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	SL4F	SL4C	SL4E	SL3F	SL3C	SL3E	SDIR

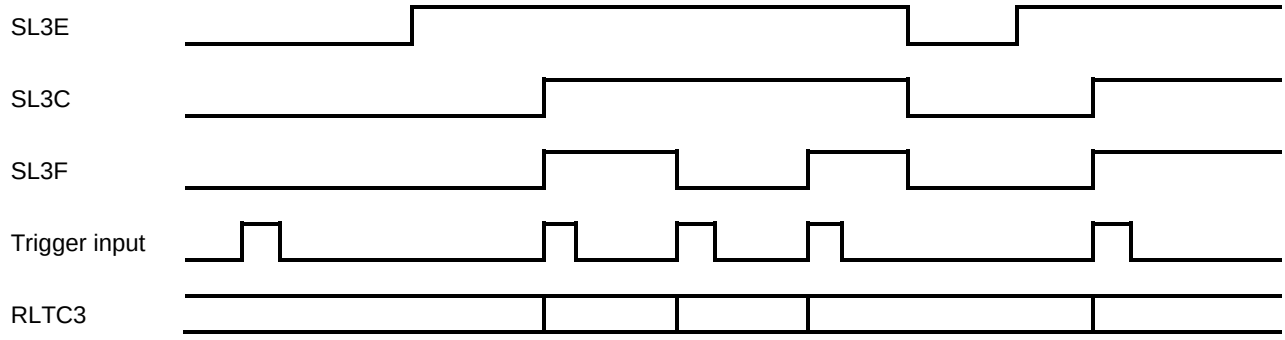
Bit	Bit symbol	Description
17	SL3E	1: RLTC3 latch enable status (LTC3E command write to LTC3D command write)
18	SL3C	1: The first latch to RLTC3 is completed (reset by LTC3D command)
19	SL3F	Toggle change (reset by LTC3D command) when value of RLTC3 changes
20	SL4E	1: RLTC4 latch enabled status (LTC4E command write to LTC4D command write)
21	SL4C	1: First latch to RLTC4 is completed (reset by LTC4D command)
22	SL4F	Toggle change (reset by LTC4D command) when value of RLTC4 changes
31 - 23	Not defined	(Always set to “0”)

Status change timing

1) When RENV4.L3MD=0, RENV4.L3TL=1



2) When RENV4.L3MD=1, RENV4.L3TL=1



6-9. REST register (Bit definition is added)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	ESMS	ESPS	ESPE	ESEE	ESP0	ESSD	ESEM	ESSP	ESAL	ESML	ESPL

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Bit symbol	Description
9	ESPS	When stopped by (+) software limit detection (enabled only when RENV3.SLM1 = "1")
10	ESMS	When stopped by(-) software limit detection (enabled only when RENV3.SLM1 = "1")
31 - 11	Not defined	(Always set to “0”)

6-10. RIST register (Bit definition is added)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ISMS	ISPS	ISSA	ISMD	ISPD	ISSD	ISOL	ISLT	ISC2	ISC1	ISDE	ISDS	ISUE	ISUS	ISNM	ISEN

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	0	0	0	ISL4	ISL3	ISBY	ISEZ

Bit	Bit symbol	Description
14	ISPS	(＋) software limit is detected (enabled only when RENV3.SLM1,0="01")
15	ISMS	(－) software limit is detected (enabled only when RENV3.SLM1,0="01")
16	ISEZ	When stopped in the middle of deceleration in a high-speed origin return operation with RENV2.ORM = 1.
17	ISBY	When it becomes in operation (at start)
18	ISL3	When latch to RLTC3
19	ISL4	When latch to RLTC4
31 - 20	Not defined	(Always set to “0”)

6-11. RSDC register (Bit definition is added)

31	28	24	20	16	12	8	4	0
# # # #	# # # #							

When automatic ramp down point setting (RMD.MSDP = 0) is selected, a negative number can also be set as an offset value in the RDP register, and the RDP setting range at that time is -8,388,608 (800000h) to +8,388,607 (7FFFFFh).

In a high-speed start, it counts up from the offset value, but bits 31 to 24 were always 00h in PCL61x3, so it was not recognized as signed 32-bit number.

In PCL61x5, bits 31 to 24 are FFh when an offset value is negative, and it can be recognized as a signed 32-bit number. Additionally, when RMD.MSDP = 1, it can be recognized as a 32-bit positive number from 0 to +16,777,215 (00FFFFFFh).

7. Serial I/F

In general, it is called SPI bus system.

When using the serial I/F, set the $\overline{RD}$ and $\overline{WR}$ input terminals to low level (GND connection).

The $\overline{CS}$, $\overline{RD}$ and $\overline{WR}$ terminals are pulled up inside the PCL61x5

The following terminals are used for the serial I/F operation.

Terminal name	Direction	Serial signal symbol	Description
CS	CPU→PCL	SS	Slave select signal
IF0	CPU→PCL	SCK	Serial clock
IF1	CPU→PCL	MOSI	Writing data
WRQ	CPU←PCL	MISO	Reading data
A0	→PCL	DS0	Device selection 0
A1	→PCL	DS1	Device selection 1

Note. The DS0 and DS1 signals are used to connect multiple PCL61x5s with one set of serial communication line. When only one PCL61×5 is connected, connect both terminals to GND.

7-1. Communication format

There are four types of communication format as below according to the type of axis selection code.

1) Format to write commands

(1)Operation command

MOSI：	Axis selection conde(S7-S0)	Command(C7-C0)

There is one axis selection code and one command per one communication.

The general-purpose output bit control command and the control command are the same as the operation command.

(2)Register writing command

MOSI：	Axis selection code(S7-S0)	Command (C7-C0)	Data x	Data y	Data z	Data u

There is one axis selection code and one command per one communication.

The number of data to write is the number of axes selected by the axis selection code.

The size of the data is 32 bits, and the arrangement order will be "data [7-0] + data [15-8] + data [23-16] + data [31-24]".

(3) Register read out command

MOSI:	Axis selection code(S7-S0)	Command(C7-C0)		
MISO :		Data x	Data y	Data z	Data u

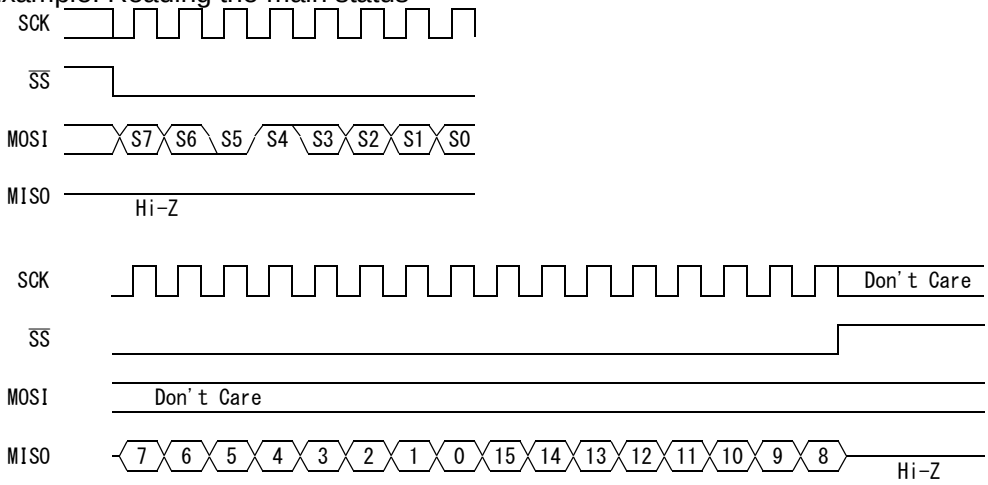
There is one axis selection code and one command per one communication.
The number of data to be read out is the number of axes selected by the axis selection code.
The size of the data is 32 bits, and the arrangement order will be "data [7-0] + data [15-8] + data [23-16] + data [31-24]".

2) Format to read the main status

MOSI :	axis selection code (S7-S0)			
MISO :	Data x	Data y	Data z	Data u

There is one axis selection code per one communication.
The number of data to be read out is the number of axes selected by the axis selection code.
The size of the data is 16 bits, and the arrangement order should be "data [7-0] + data[15-8].

Example. Reading the main status



S7-S0 : Axis selection code (Format to read main status is S5="0", S4="1")
15-0 : Main status read out data

3) Format to write general-purpose port

MOSI :	Axis selection code (S7-S0)	Data x	Data y	Data z	Data u

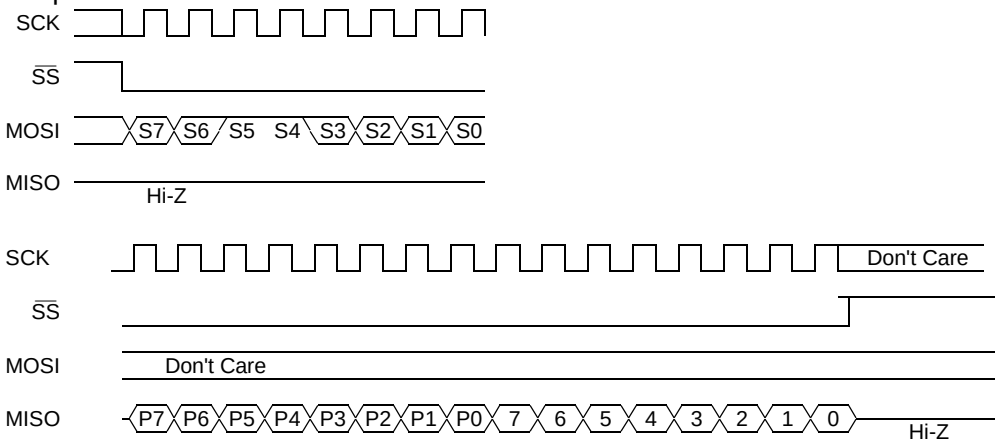
There is one axis selection code per one communication.
The number of data to write is the number of axes selected by the axis selection code.
The size of the data is 8 bits, and the arrangement order should be "data [7-0].

4) Format to read out sub status

MOSI :	axis selection code (S7-S0)			
MISO :	Data x	Data y	Data z	Data u

There is one axis selection code per one communication.
The number of data to write is the number of axes selected by the axis selection code.
The size of the data is 16 bits, and the arrangement order should be "data [7-0].
"Data [7-0]" is general-purpose port data, and "Data [15-8]" is the sub status data.

Example. Read out the sub status



S7-S0 : Axis selection code (Format to read the sub status is S5="1", S4="1")
P7-P0 : General-purpose port read data
7-0 : Sub status read out data

Note:

1. If multiple axes are selected with the axis selection code, access can be made to multiple axes by one serial communication.
   In that case, the data order will be X> Y> Z> U among the selected axes.
2. Commands (register) writing and writing in general-purpose ports for multiple axes are conducted at the rising edge of the SS signal.
3. The register reading for multiple axes should be sequentially conducted after latching the status when "C0" bit was written.
4. Reading the main status and the sub status for multiple axes should also be sequentially conducted after latching the status when "S0" bit was written.
5. Please note that unexpected data will be written if control is aborted without writing the number of bits according to the format ($\overline{SS}$ signal is raised during writing).
6. Please note that the remaining data will be discarded if control is aborted without reading the number of bits according to the format ($\overline{SS}$ signal is raised during reading).

7-2. Axis selection code

Axis selection codes are 1byte. They consist of the following 8 bits.

Bit	PCL6114	PCL6124	PCL6144
0	Select X axis	Select X axis	Select X axis
1	(Fixed to "0")	Select Y axis	Select Y axis
2	(Fixed to "0")	(Fixed to "0")	Select Z axis
3	(Fixed to "0")	(Fixed to "0")	Select U axis
4	Type selection A		
5	Type selection B		
6	Device selection 0		
7	Device selection 1		

1) Axis selection (Bits 3 - 0)

Select a target axis to write or read.
When the corresponding axis select bit is "1", they shows an axis selected.
When the all axis select bits are "0", only X axis is selected (same as "0001" b).

2) Type selection (Bits 5, 4)

Select among 4 types of communication formats

Type selection		Communication format
B	A	
0	0	Write a command
0	1	Read out the main status
1	0	Write the general-purpose port
1	1	Read out the sub status, general-purpose port

3) Device selection (Bits 7, 6)

Normally, only one LSI is connected with one slave select signal ($\overline{SS}$), but this LSI can be expanded up to four (4) LSIs at the maximum.

The bits for device selection 1 and device selection 0 correspond to the device select terminals (DS1, DS0).

Multiple LSIs with different device select number terminal settings cannot be accessed simultaneously.

Device selection		Device selection n umber terminal	
1	0	DS1	DS0
0	0	L	L
0	1	L	H
1	0	H	L
1	1	H	H

7-3. Command

Command code is 1byte. It is the same as the command in parallel I/F.

7-4. Data

Data is an integer multiple of 1 byte, and each byte is arranged in the order of Bit 7 (MSB) to Bit 0 (LSB) in the order from the lower byte to the upper byte.

Even if the data is less than 8 bits, substitute "0" for missing bits and set it in 1 byte unit.

Also if the register writing data is less than 4 bytes, substitute 00h for the missing bytes and set it in 4 bytes unit. In the case of multiple axis register lump writing as well, set them in units of 4 bytes per axis.

- End of the document -