PAJ7620U2: Integrated Gesture Recognition Sensor

General Description

The PAC7620 integrates gesture recognition function with general I²C interface into a single chip forming an image analytic sensor system. It can recognize 9 human hand gesticulations such as moving up, down, left, right, forward, backward, circle-clockwise, circle-counter clockwise, and waving. It also offers built-in proximity detection in sensing approaching or departing object from the sensor. The PAC7620 is designed with great flexibility in power-saving mechanism, well suit for low power battery operated HMI devices. The PAJ7620 is packaged into module form in-built with IR LED and optics lens as a complete sensor solution

Key Features

- Gesture/Cursor/Image modes
- Built-in proximity detection
- Gesture speed is 60~600°/s in Normal Mode and 60~1200°/s in Gaming Mode
- Flexible power saving scheme
- Communication interface options
  - I²C (for Gesture/Cursor mode)
  - 4-wire SPI (for Image mode)
- I²C interface up to 400 Kbit/s
- SPI interface range from
  - 22~48 MHz (Frame subtraction mode)
  - 44~48 MHz (Raw data mode)
- Ambient light immunity
- Ambient light noise cancellation

Applications

- PAD Phone
- Tablet Personal Computer
- Automobile Application

Key Parameters

Parameter	Value
Array Size	60x60 pixels
Pixel Size	20x20 μm²
Max Frame Rate	720 fps @ 240 report rate
Input Clock	22MHz for SPI Image Out 44MHz for SPI raw data mode
Supply Voltage	VDD: 2.8~3.6V VBUS: 1.8~3.3V VLED: 3.0~4.2V
Current Consumption	Operation State:2.82 mA Standby 1 State: 2.3mA Standby 2 State: 1.5mA
Package Dimensions	5.2x3x1.88 mm³

Ordering Information

Part Number	Package Type
PAJ7620U2	13-pins LGA Module

Lead (Pb) Free
RoHS 6 fully compliant

For any additional inquiries, please contact us at http://www.pixart.com/contact.asp.

Table of Contents

List of Figures

List of Tables

1.0 Sensor Overview

1.1 Gesture Mode

For Gesture Mode, there are 9 gestures recognition being embedded in the sensor including move up, move down, move left, move right, move forward, move backward, circle-clockwise, circle-counter clockwise, and wave. These gestures information can be simply accessed by register reading via I^{2}C bus. The normal gesture detecting range from 5 to 15 cm from where PAC7620 is located through the operating view angle of diagonally 60° respectively. Additionally, The PAC7620 can be configured as Normal Mode (Gesture speed is 60°/s - 600°/s) or Gaming Mode (Gesture speed is 60°/s - 1200°/s) for different applications. The PAC7620 also offer built-in proximity detection for the purpose of sensing object approaching or departing.

Table 1. Gesture Detecting Range and View Angle

Part Number	Detecting Range	View Angle (Diagonal)
PAJ7620U2	5 to 15 cm	60°

1.2 Image Mode

For image mode, the typical report rate is 120Hz with image size equals 30x30 (Frame subtraction mode, WOI) or 30x30 (Raw data mode, WOI). The depth of pixel data depth is 9 bit and output through the SPI bus. The SCK rate of SPI bus equals to the external SPI clock input ranging from 22 to 48 MHz (Frame subtraction mode) or 44 to 48 MHz (Raw data mode). By programming the internal register set via I^{2}C serial control bus, it performs on-chip report rate, exposure time, gain adjustment, array skip mode, and array average mode.

1.3 Architecture Block Diagram

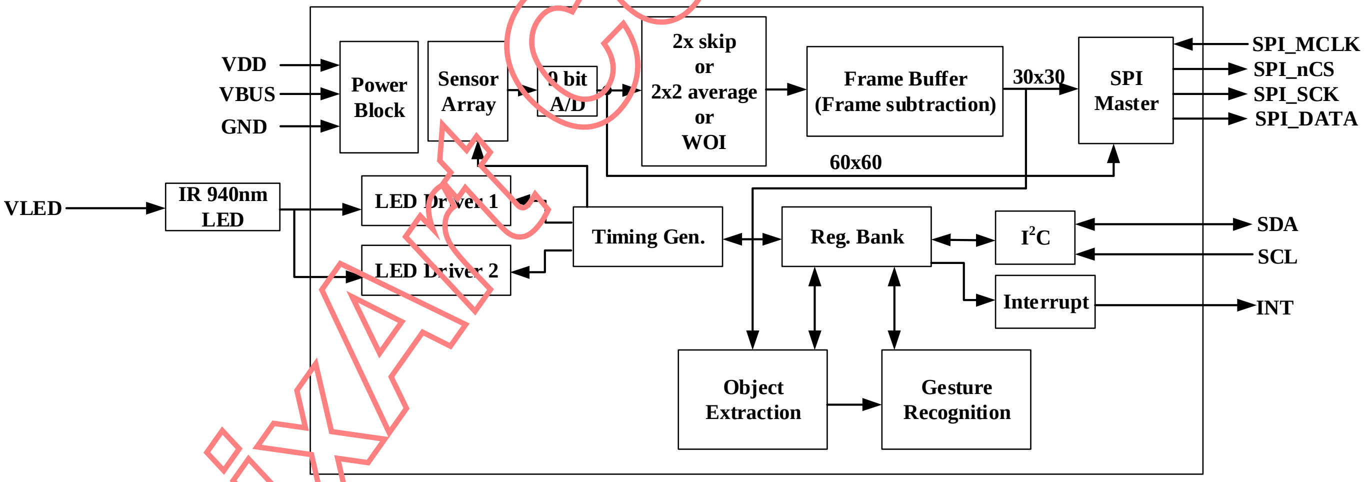


Figure 1. Architecture Block Diagram

1.4 Signal Description

Table 2. Signal Description

Signal Name	Description
SDA	I^{2}C data pin

SCL	I^2C clk pin
INT_N	Interrupt pin (Active low) for Gesture Mode.
GPIO3(SPI_DATA)	Data out of SPI master for image mode
GPIO2(SPI_SCK)	SCK signal of SPI master for image mode
GPIO1(SPI_nCS)	nCS signal of SPI master for image mode
GPIO0(SPI_MCLK)	External clock input of SPI master for image mode

1.5 Pin Configuration

Table 3. PAJ7620U2 Pin Definition

Pin No.	Symbol	Type	Function
1	V_{BUS}	POWER	BUS power supply
2	SDA	IN/OUT	I^2C data pin (Open Drain)
3	INT_N	OUT	Interrupt pin (Active low) (Open Drain)
4	TESTMD	IN	For Module Test Only
5	SCL	IN	I^2C clock pin (Open Drain)
6, 10	GND	GND	Ground
7	GPIO3 (SPI_DATA)	SPI Mode : OUT GPIO Mode : IN/OUT	SPI Mode : Data out of SPI master GPIO Mode : GPIO
8	GPIO2 (SPI_SCK)	SPI Mode : OUT GPIO Mode : IN/OUT	SPI Mode : SCK signal of SPI master GPIO Mode : GPIO
9	GPIO1 (SPI_nCS)	SPI Mode : OUT GPIO Mode : IN/OUT	SPI Mode : Chip select signal of SPI master (Active Low) GPIO Mode : GPIO
11	V_{LED}	POWER	LED power input
12	V_{DD}	POWER	Main power supply
13	GPIO0 (SPI_MCLK)	SPI Mode : IN GPIO Mode : IN/OUT	SPI Mode : External clock input for SPI GPIO Mode : GPIO

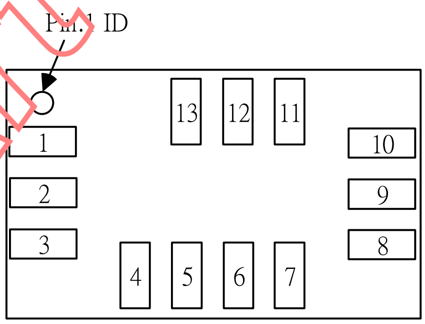


Figure 2. PAJ7620U2 Module Pin Configuration (BTM VIEW)

2.0 Mechanical Specifications

2.1 Package Dimension

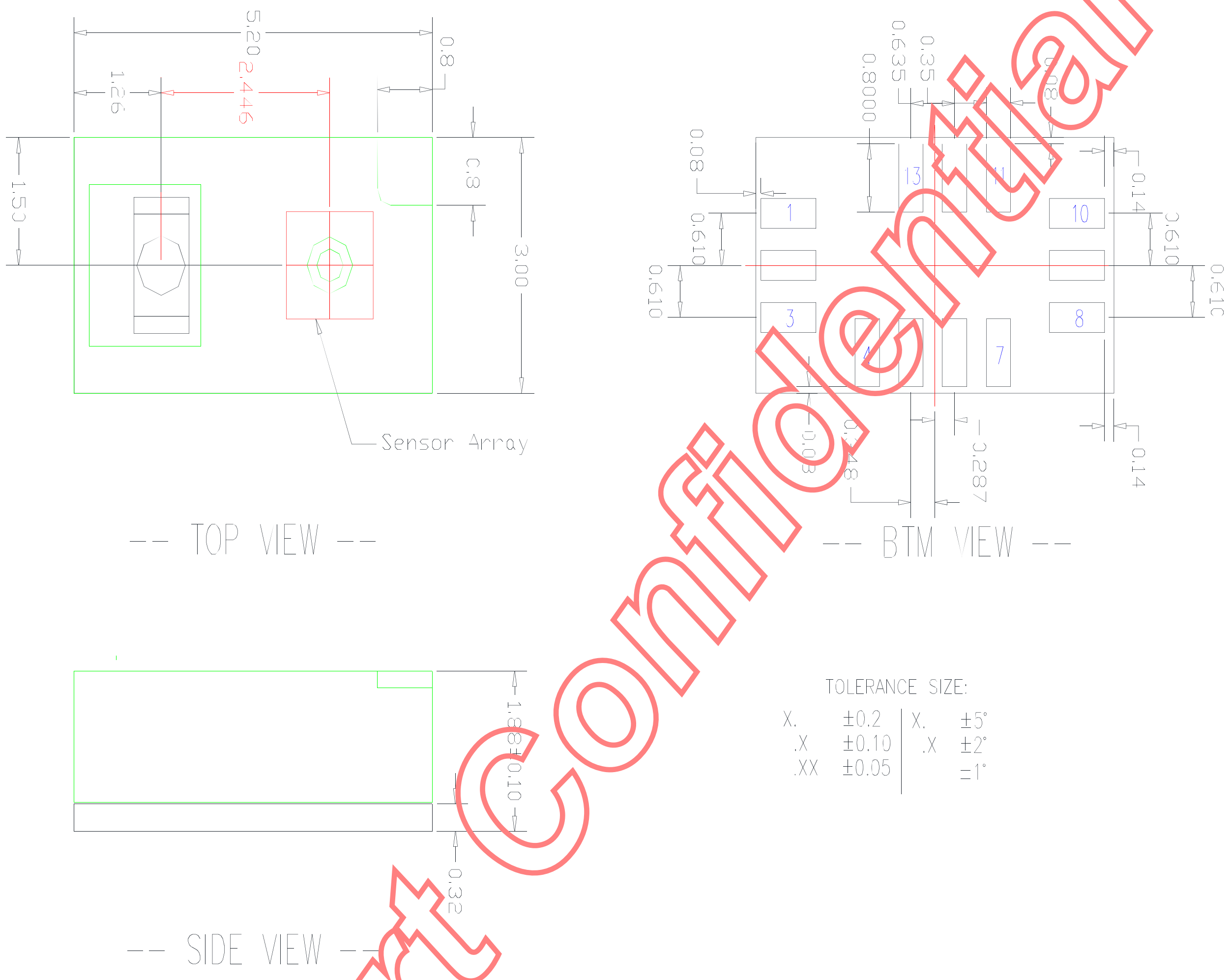


Figure 3. PAJ7620U2 Package Outline Diagram

2.2 IR Reflow Recommendation

2.2.1 Recommended Pb-free Solder Paste

- Almit LFM-48W TM-HP
- Senju M705-GRN360-K

2.2.2 IR Reflow Soldering Profile

Temperature profile is the most important control in reflow soldering. It must be fine-tuned to establish a robust process. The typical recommended IR reflow profile is showed in Figure 4 below.

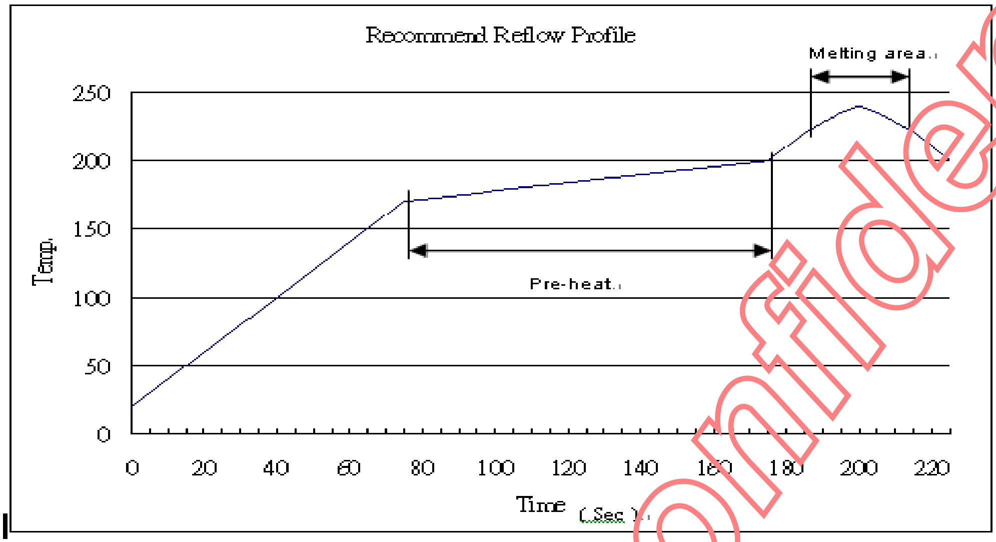


Figure 4. IR Reflow Profile

Reflow Profile:

I. Average Ramp-up Rate (30°C to preheat zone): 1.5~ 2.5 Degree C/ Sec)

II. Preheat zone:

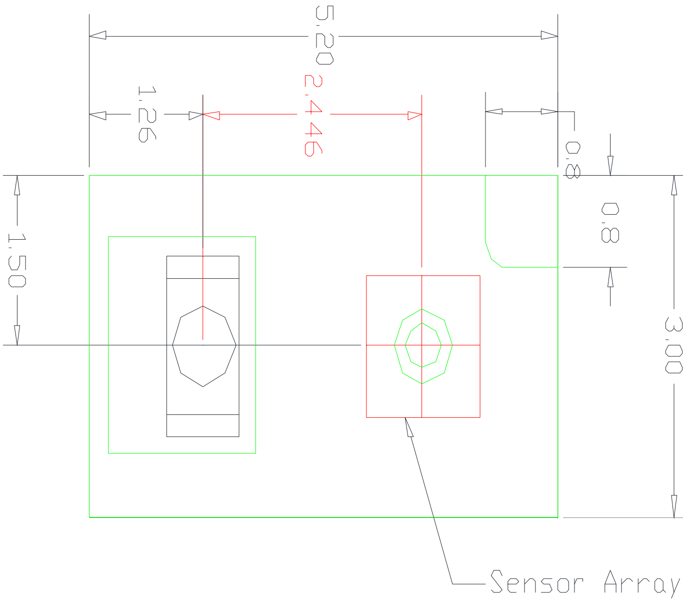
- Temp ramp from 170~ 200 degree C
- Exposure time: 90 +/- 30 sec

III. Melting zone:

- Melting area temp > 220 degree C for at least 30 ~ 50 sec
- Peak temperature : 245 degree C

2.3 Recommend Mechanical Design

PXI suggest mechanical design as below to optimize the performance.



Note:

1. Recommended Cover Glass Material: Glass or PC
2. Clear Glass Part Transparency: > 90%
3. Cover Glass Thickness ≤ 0.7mm
4. Cover Glass and PAJ7620U2 are close as much as possible. Air Gap ≤ 0.2mm

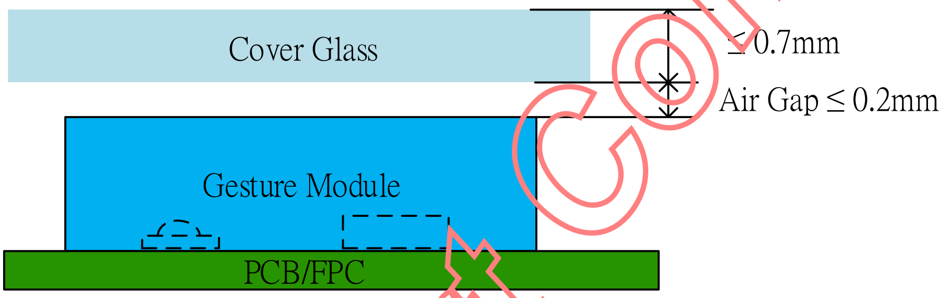


Figure 5. PAJ7620U2 Mechanical Design Guide

Note:

1. If thickness of cover glass ≤ 0.7mm, and Air Gap ≤ 0.2mm, the black Ink region is not necessary.
2. For appearance reason, Ink on cover may be necessary. Please follow the IR ink spectrum above

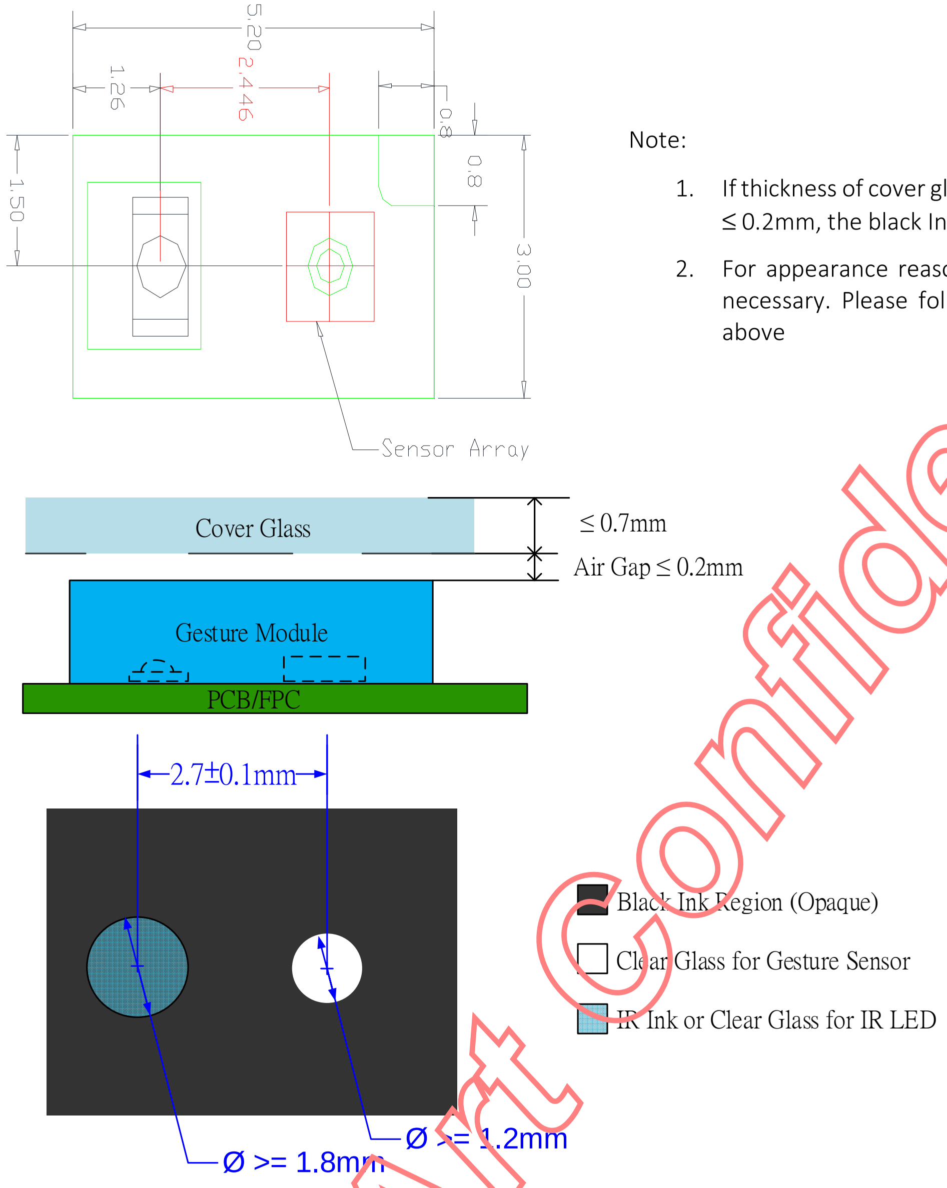


Figure 6. PAJ7620U2 Mechanical Design Guide (IR Ink Suggestion)

2.4 Recommended IR Ink Spectrum

IR Ink Vendor: Teikoku Ink

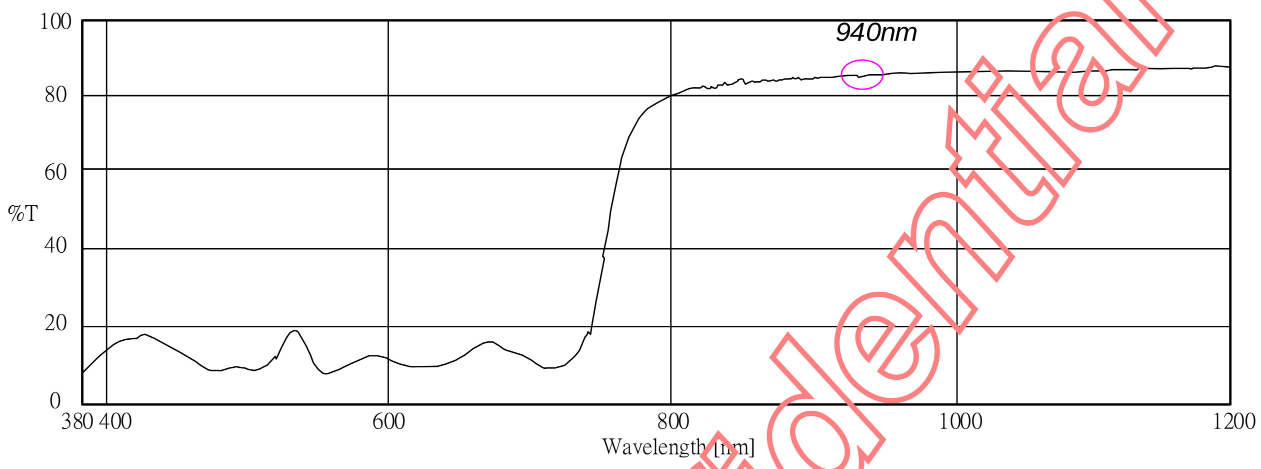
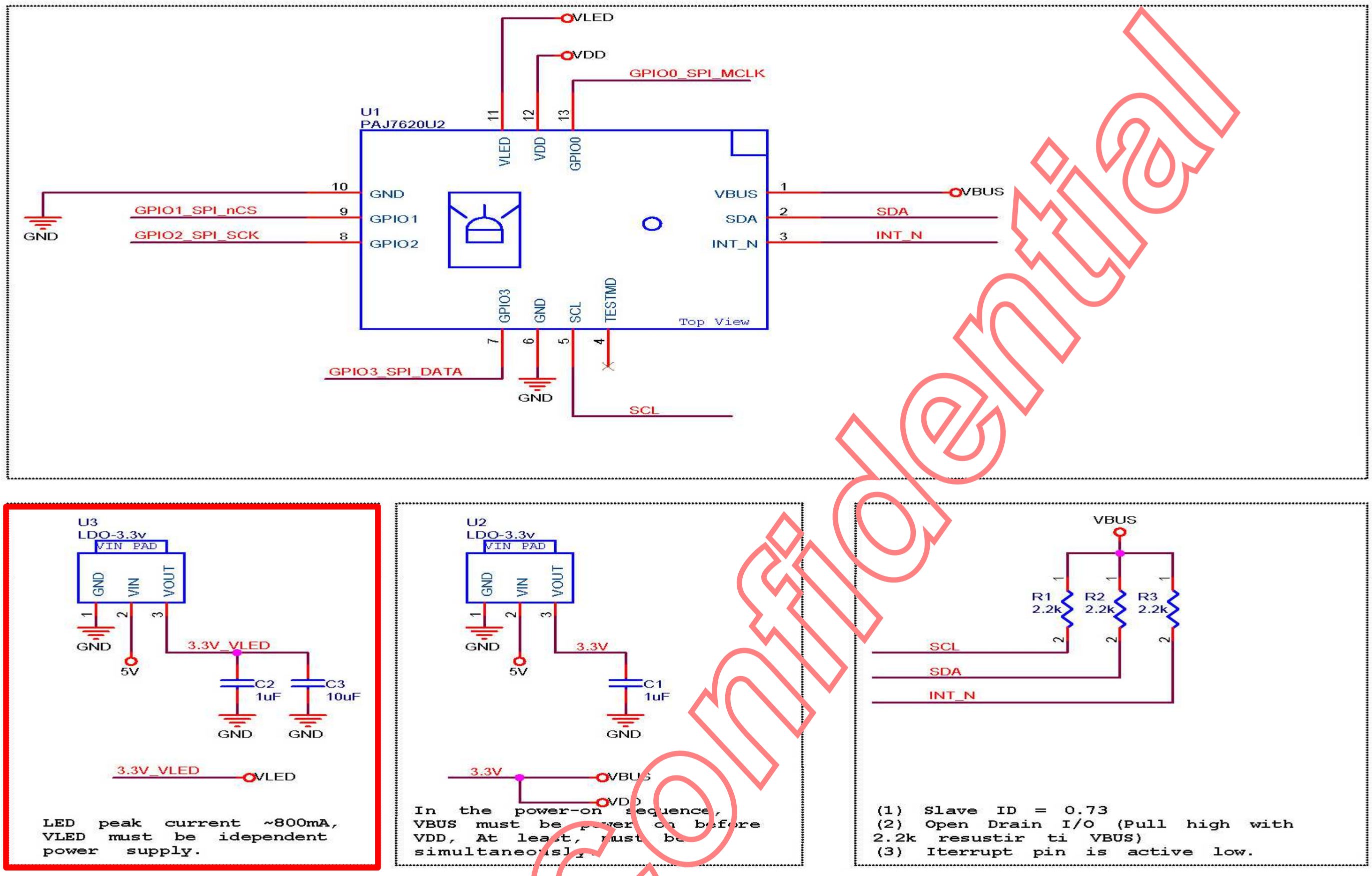


Figure 7. IR Ink Spectrum

3.0 Reference Schematics



LED peak current ~800mA,
VLED must be independent
power supply.

In the power-on sequence,
VBUS must be power on before
VDD, At least, must be
simultaneously.

(1) Slave ID = 0.73
(2) Open Drain I/O (Pull high with
2.2k resustir ti VBUS)
(3) Iterrupt pin is active low.

Figure 8. PAJ7620U2 Application Circuit

4.0 Sensor Specifications

4.1 Absolute Maximum Ratings, T_A = 27°C

Parameters	Symbol	Min.	Max.	Unit	Conditions
Supply Voltage	V_{DD}	-	4	V	
LED Supply Voltage	V_{LED}	-	4.6	V	
LED Pulse Current	I_{LED}	-	2	A	*Pulse Width < 500us, Duty Cycle < 5%*
I^2C Pin, INT_N Pin Voltage	V_{BUS}	-0.3	V_{DD}+0.3	V	SCL, SDA, INT_N
I^2C Pin, INT_N Pin Current	I_{BUS}	-	10	mA	SCL, SDA, INT_N
I/O Pin Voltage	V_{DDIO}	-0.3	V_{DD}+0.3	V	SPIM_CLK, SPI_SCK, SPI_DATA, SPI_nCS
I/O Pin Current	I_{DDIO}	-	10	mA	SPIM_CLK, SPI_SCK, SPI_DATA, SPI_nCS
ESD, human body model	ESD_{HBM}	-	2	kV	
ESD, Machine model	ESD_{MM}	-	200	V	

Notes:

1. Maximum Ratings are those values beyond which damage to the device may occur.
2. Exposure to these conditions or conditions beyond those indicated may adversely affect device reliability.
3. Functional operation under absolute maximum-rated conditions is not implied. Functional operation should be restricted to the Recommended Operating Conditions.

4.2 Recommended Operating Condition

Parameters		Symbol	Min.	Typ.	Max.	Unit	Conditions
Supply Voltage		V_{DD}	2.8	-	3.6	V	
LED Supply Voltage		V_{LED}	3	-	4.2	V	
Peak LED Driver 1/ 2 Current Pulse		I_{LED}	-	360	430	mA	*Pulse Width < 500us, Duty Cycle < 5%*
I^2C Pin, INT_N Pin Voltage		V_{BUS}	1.8	-	3.3	V	SCL, SDA, INT_N
I^2C Pin, INT_N Pin Current		I_{BUS}	-	-	5	mA	SCL, SDA, INT_N
I/O Pin Voltage		V_{DDIO}	1.8	-	3.3	V	SPIM_CLK, SPI_SCK, SPI_DATA, SPI_nCS
I/O Pin Current		I_{DDIO}	-	-	5	mA	SPIM_CLK, SPI_SCK, SPI_DATA, SPI_nCS
Frequency of external SPI clock input	Frame subtraction mode	f_{SPI}	22	-	48	MHz	
	Raw data mode		44 or 22*	-	48		*If using EXT CLK En=1 and EXT CLK IN = 4 MHz
Operating Temperature		T_{op}	-40	-	85	°C	
Storage Temperature		$T_{storage}$	-40	-	125	°C	

4.3 Electrical Specifications, V_{DD}=2.8V, T_A = 27°C

Parameters		Symbol	Min.	Typ.	Max.	Unit	Conditions
Supply Voltage		V_{DD}	2.8	-	3.6	V	
LED Supply Voltage		V_{LED}	3.0	-	4.2	V	LED Supply Voltage
I²C, INT_N Pin Pull-up Voltage		V_{BUS}	1.8	-	3.3	V	
Current Consumption for Operation Modes	PAJ7620U2	I_{DD}	-	2.82	-	mA	Under Normal Mode. Including LED current (Peak = 760mA)
	Average Mode	$I_{DD\_Avg\_Mode}$	-	1	-	mA	Excluding IR LED. 120Hz report rate. 2x2 pixel average mode
	Skip Mode	$I_{DD\_Skip\_Mode}$	-	0.8	-	mA	Excluding IR LED. 120Hz report rate. 2x skip mode
Suspend Current		I_{DD_SUS}	-	15	-	µA	
Current Consumption for Standby State 1		I_{DD_ST1}	-	2.3	-	mA	Refer to Operating Principle 1. Under Normal Mode 2. $S_{1,\text{ Response Factor}}$ =0.5 3. $S_{2,\text{ Response Factor}}$ =0.25 4. Including LED current @ Peak = 760mA
Current Consumption for Standby State 2		I_{DD_ST2}	-	1.5	-	mA	
Current Consumption for Proximity Detection		I_{PS}		0.2		mA	1. Detecting Rate = 10Hz 2. LED peak current = 600mA 3. LED on time = 6.8µs
I²C Bus Input High Voltage		$V_{IH\_ I2C}$	0.7* V_{BUS}	-	V_{BUS} +0.3	V	
I²C Bus Input Low Voltage		$V_{IL\_ I2C}$	-0.3	-	0.3* V_{BUS}	V	
Output Low Voltage		V_{OL_SDA} $V_{OL\_INT\_N}$	-	-	0.1* V_{BUS}	V	For INT_N, SDA
I/O Input High Voltage		V_{IH}	0.7* V_{DDIO}	-	V_{DDIO} +0.3	V	
I/O Input Low Voltage		V_{IL}	-0.3	-	0.2* V_{DDIO}	V	
I/O Output High Voltage		V_{OH}	V_{DDIO} - 0.3	-	-	V	
I/O Output Low Voltage		V_{OL}	-	-	0.3	V	

4.4 Gesture Functional Specifications

Parameters	Symbol	Min.	Typ.	Max.	Unit	Condition
Gesture Detecting Range	d_{OP}	5	-	15	cm	Calculated from PAJ7620U2 sensor center
Gesture Detecting View Angle	θ_{OP}	-	60	-	degree	Calculated diagonally
Gesture Speed Response	ω	60		600	degree/s	Angular velocity under Normal Mode
		60		1200		Angular velocity under Gamming Mode
Gesture Update Rate	f_{Update}	-	120	240	Hz	120 Hz for Normal Mode

						240 Hz for Gaming Mode
LED View Angle	$2\theta_{1/2}$		60		degree	
LED Peak Wavelength	λ		940		nm	

4.5 Interface AC Specifications

Parameters	Symbol	Min.	Typ.	Max.	Unit	Conditions
I^2C Speed	S_{I2C}	-	100	400	Kbit/s	
Frequency of External SPI Clock Input		22	-	48	MHz	Frame subtraction mode
		44	-	48	MHz	Raw data mode

5.0 Serial Interface Communication

5.1 I²C Interface

SDA (serial data) and SCL (serial clock) form a two-wire serial interface compatible with I²C. The PAC7620 is implemented as a slave-only device so it never drives SCL. It drives SDA during (host) read cycles and transmission of the Acknowledge bit. PAC7620 uses 7-bit addressing and does not support clock stretching. The SDA and SCL pins are open-drain structure requiring external pull-up resistors.

- Start and stop condition: SDA high to low transition while SCL is high defines a Start condition. SDA low to high transition while SCL is high defines a Stop condition. (Refer to Figure 99)
- Valid data: The data on SDA line must be stable during high period of SCL. MSB is always transferred first for each byte. LSB of the first byte is Read / Write control bit. (Refer to Figure 1010)
- Both master and slave can transmit and receive data from the bus.
- Acknowledge: The Receiving device should pull down SDA during high period of the 9th clock (SCL) after a complete byte has been received from the transmitter. In the case of the master receiving data from the slave, the master does not generate an Acknowledge bit after the last byte to indicate the end of a master read cycle.

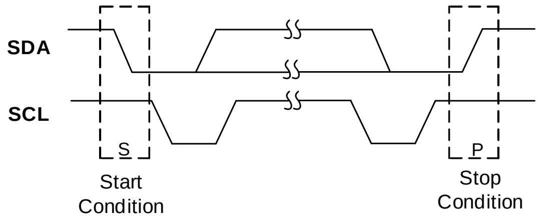


Figure 9. Start and Stop Conditions

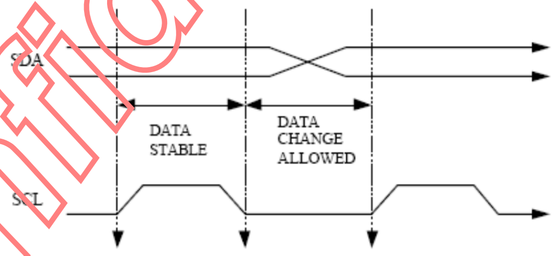


Figure 10. Valid Data

5.1.1 I²C Protocol

The I²C Slave ID is using 7 bit addressing protocol: 0x13, 0x1B, 0x23, 0x2B, 0x5B, 0x63, 0x6B, 0x73 and default is 0x73.

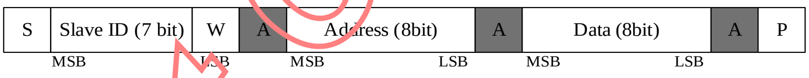


Figure 11. Single Write Protocol

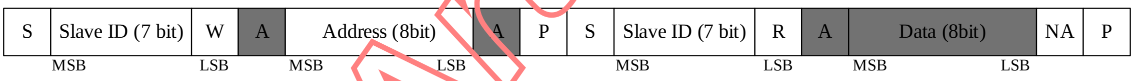


Figure 12. Single Read Protocol

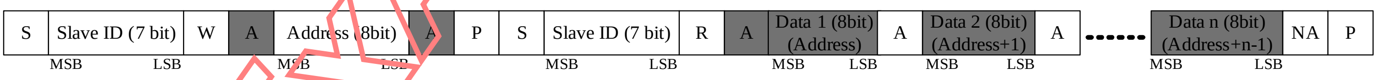


Figure 13. Burst Read Protocol

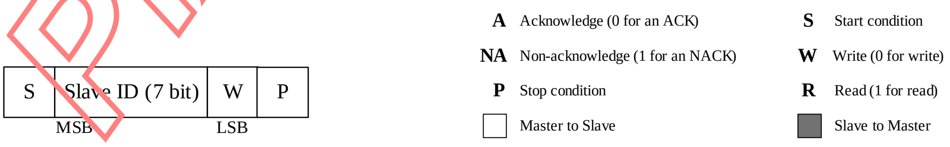


Figure 14. I²C Wake-up command Protocol

5.1.2 I²C Timing Parameter

Parameter	Symbol	STANDARD MODE		FAST MODE		Unit
		Min.	Max.	Min.	Max.	
SCL clock frequency.	f_{scl}	10	100	10	400	kHz
Hold time for Start/Repeat Start. After this period, the first clock pulse is generated.	$t_{HD.STA}$	4		0.6		µs
Set-up time for a repeated Start.	$t_{SU.STA}$	4.7		0.6		µs
Low period of SCL clock.	t_{LOW}	4.7		1.3		µs
High period of SCL clock.	t_{HIGH}	4		0.6		µs
Data hold time.	$t_{HD.DAT}$	0		0		µs
Data set-up time.	$t_{SU.DAT}$	250		100		ns
Rise time of both SDA and SCL signals.	t_r		1000	-	300	ns
Fall time of both SDA and SCL signals.	t_f		300	-	300	ns
Set-up time for STOP condition.	$t_{SU.STO}$	4		0.6		µs
Bus free time between a STOP and START.	t_{BUF}	4.7		1.3		µs

Note: Maximum current is 5mA and capacitance load spec. =100pF

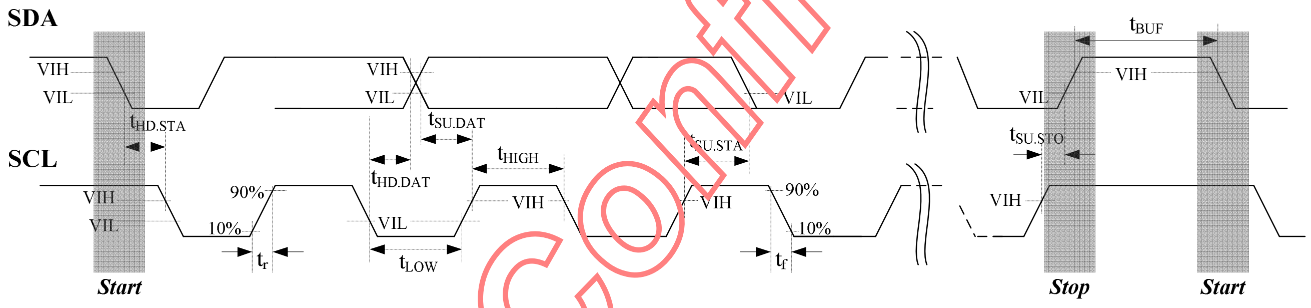
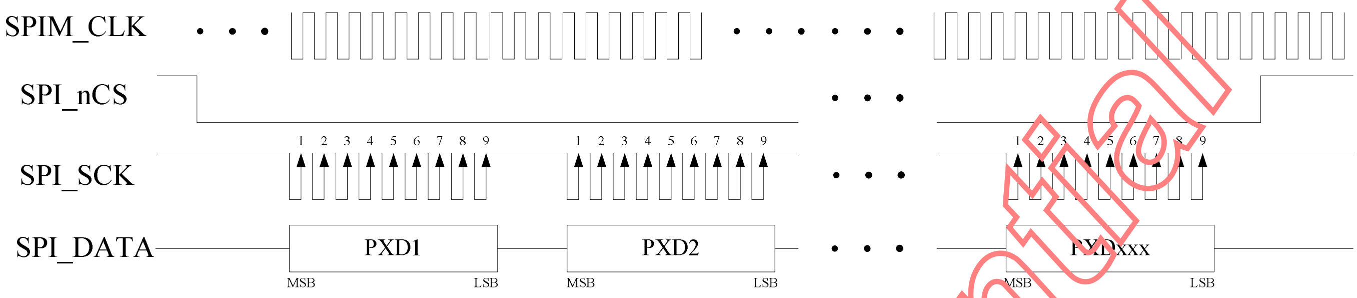


Figure 15. I²C Timing Diagram

5.2 Four-Wire SPI Interface (For Image Mode Only)

5.2.1 SPI Master Protocol



Notes: xxx = 900 for 30x30 (Frame subtraction mode), =3600 for 60x60 (Raw data mode)

Figure 16. SPI Master Protocol

5.2.2 SPI Timing Parameter

Parameter	Symbol	Typ. (measured)	Unit
SCK clock frequency. ($f_{sck} = 1/t_p$)	f_{sck}	24	MHz
Low period of SCK clock.	t_{LOW}	15.5	ns
High period of SCL clock.	t_{HIGH}	12.5	
Data output valid time.	$t_{V(MO)}$	20.3	
Data output hold time.	$t_{H(MO)}$	20.8	
Rise time of SCK clock	t_r	8	
Fall time of SCK clock	t_f	8	

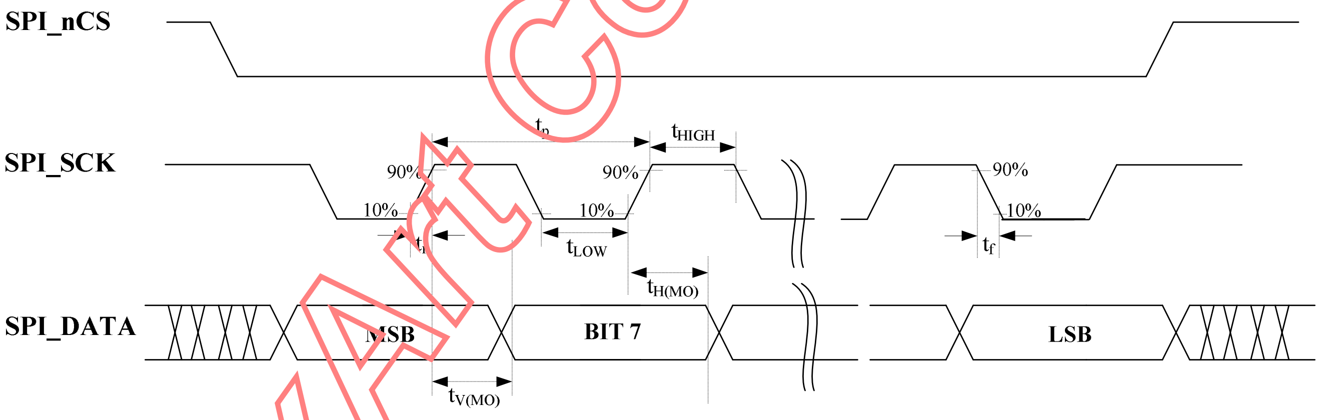


Figure 17. SPI Timing Diagram

6.0 Operation Principles

6.1 Gesture mode

6.1.1 Module Orientation

The sensor module package should be placed in the correct orientation to have the gesture direction as per 錯誤! 找不到參照來源。18.

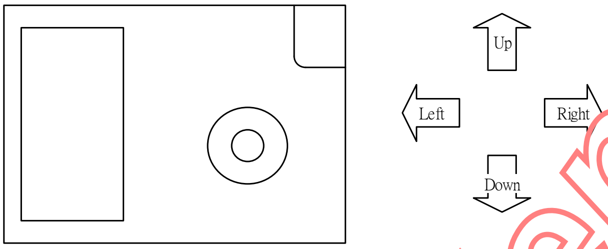


Figure 18. Module Orientation of PAJ7620U2 (Front View)

If the sensor module package is rotated to a different angle, the direction of gesture detection interrupt flag mask (Register Bank 0, ADDR 0x41) and gesture detection interrupt flag (Register Bank 0, ADDR 0x43) needs to be re-mapped.

Bank	Address	Register Name	Default Value	R/W	Description
0	0x41	R_Int_1_En	0xFF	R/W	Interrupt flag mask control. Bit[7] Counter-Clockwise Mask Bit[6]Clockwise Mask Bit[5]Backward Mask Bit[4]Forward Mask Bit[3]Up Mask Bit[2]Down Mask Bit[1]Right Mask Bit[0]Left Mask
0	0x43	IntFlag_1	0x00	R/W	Interrupt flag. Bit[7] Counter-Clockwise Bit[6]Clockwise Bit[5]Backward Bit[4]Forward Bit[3]Up Bit[2]Down Bit[1]Right Bit[0]Left

6.1.2 Power-On Sequence

In the power-on sequence, The V_{BUS} Must be power on before V_{DD}. After power on, wait T_1 µs for PAJ7620U2 to stabilize and then write slave ID (0x73) to process I^2C wake-up (Refer to topic "I^2C Interface"). After T_2 µs, write the initial settings and the different modes settings to PAJ7620U2. Lastly, enable PAJ7620U2 by writing Register Bank1, Addr0x72 with 0x01. The gesture data can now be accessed through the I^2C bus.

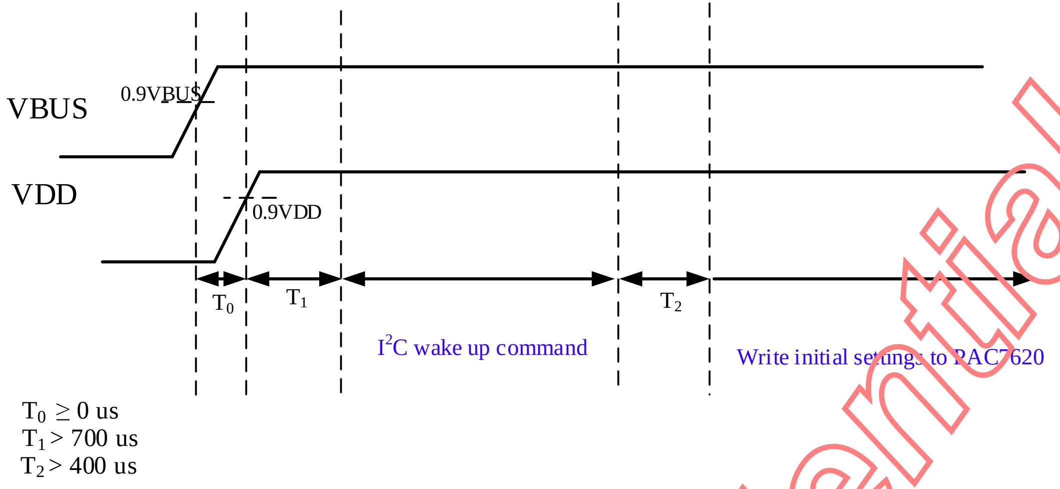


Figure 19. PAJ7620U2 Power-On Timing Diagram

6.1.3 Gesture Detection Operating State

6.1.3.1 Operation State (OP state)

When in operation state, the gesture update rate is 120Hz for Normal Mode and 240Hz for Gaming Mode respectively. The gesture result can be accessed by interrupt mechanism or continuous polling the gesture detection interrupt flag (Register Bank 0, Addr0x43).

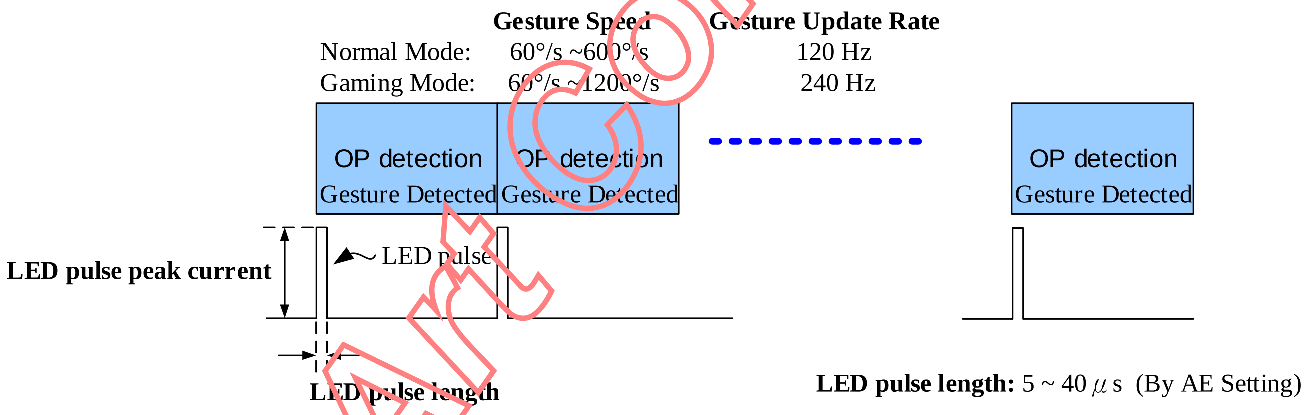


Figure 20. Operation State (OP state) Diagram

6.1.3.2 Standby 1 State (S1 state)

When in Standby1 state, the object detection rate equals $S_{1,\ \text{Response Factor}}$ multiply the gesture update rate of Normal Mode or Gaming Mode.

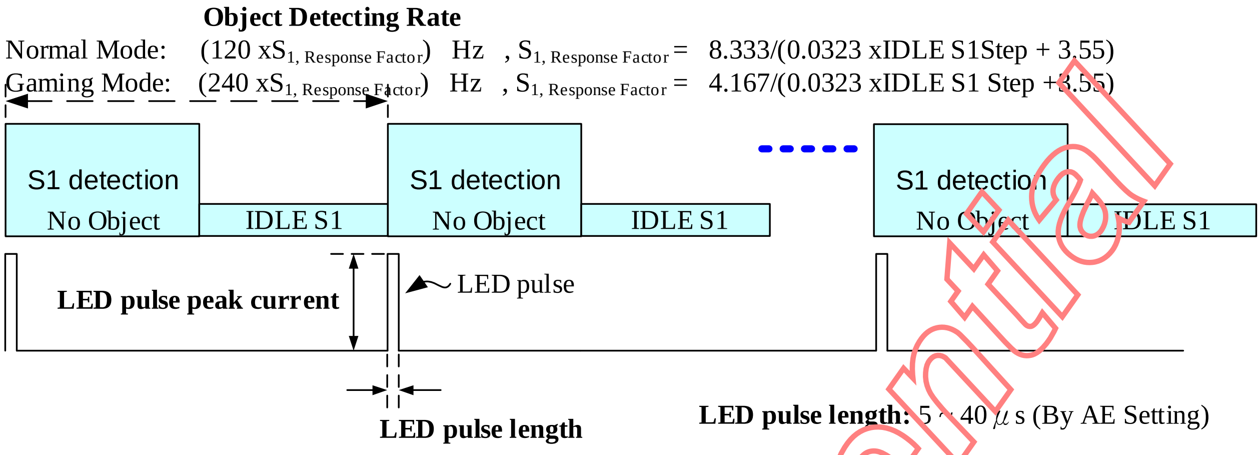

Object Detecting Rate
Normal Mode: (120 xS1, Response Factor) Hz , S1, Response Factor = 8.333/(0.0323 xIDLE S1Step + 3.55)
Gaming Mode: (240 xS1, Response Factor) Hz , S1, Response Factor = 4.167/(0.0323 xIDLE S1 Step +3.55)
S1 detection
No Object
IDLE S1
S1 detection
No Object
IDLE S1
S1 detection
No Object
IDLE S1
LED pulse peak current
LED pulse
LED pulse length
LED pulse length: 5 ~ 40 μs (By AE Setting)

Figure 21. Standby 1 State (S1 state) Diagram

6.1.3.3 *Standby 2 State (S2 state)*

When in Standby 2 state, the object detection rate equals $S_{2,\ Response\ Factor}$ multiply the gesture update rate of Normal Mode or Gaming Mode.

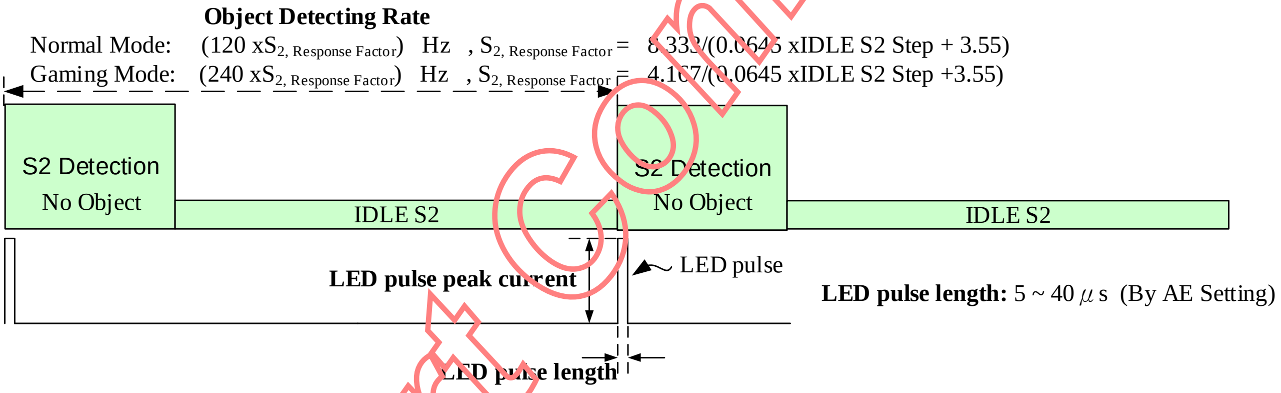

Object Detecting Rate
Normal Mode: (120 xS2, Response Factor) Hz , S2, Response Factor = 8.333/(0.0645 xIDLE S2 Step + 3.55)
Gaming Mode: (240 xS2, Response Factor) Hz , S2, Response Factor = 4.167/(0.0645 xIDLE S2 Step +3.55)
S2 Detection
No Object
IDLE S2
S2 Detection
No Object
IDLE S2
LED pulse peak current
LED pulse
LED pulse length: 5 ~ 40 μs (By AE Setting)
LED pulse length

Figure 22. Standby 2 State (S2 state) Diagram

6.1.3.4 *Suspend State (SUS state)*

To enter the suspend state, first disable the PAC7620 by writing Register Bank 1, ADDR 0x72 with 0x00 then process the I^{2}C suspend command by writing Register Bank 0, ADDR 0x03 with 0x01.

To exit the suspend state, first process the I^{2}C wake-up command by writing the slave ID (Refer to topic "I^{2}C Bus Timing Characteristics and Protocol") then enable the PAC7620 by writing Register Bank 1, ADDR 0x72 with 0x01.

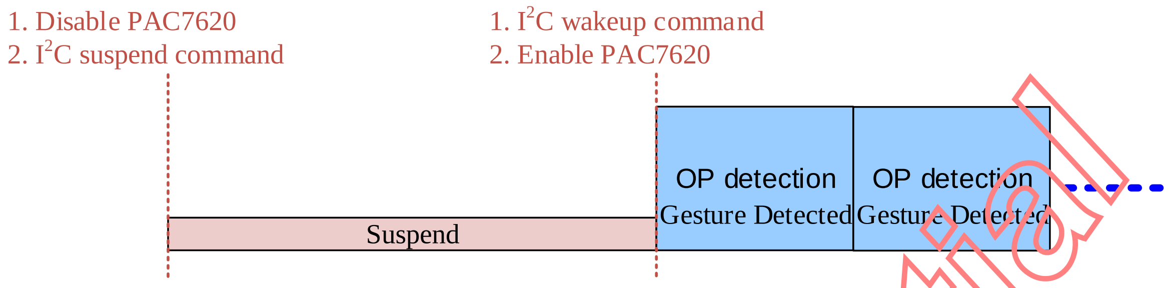
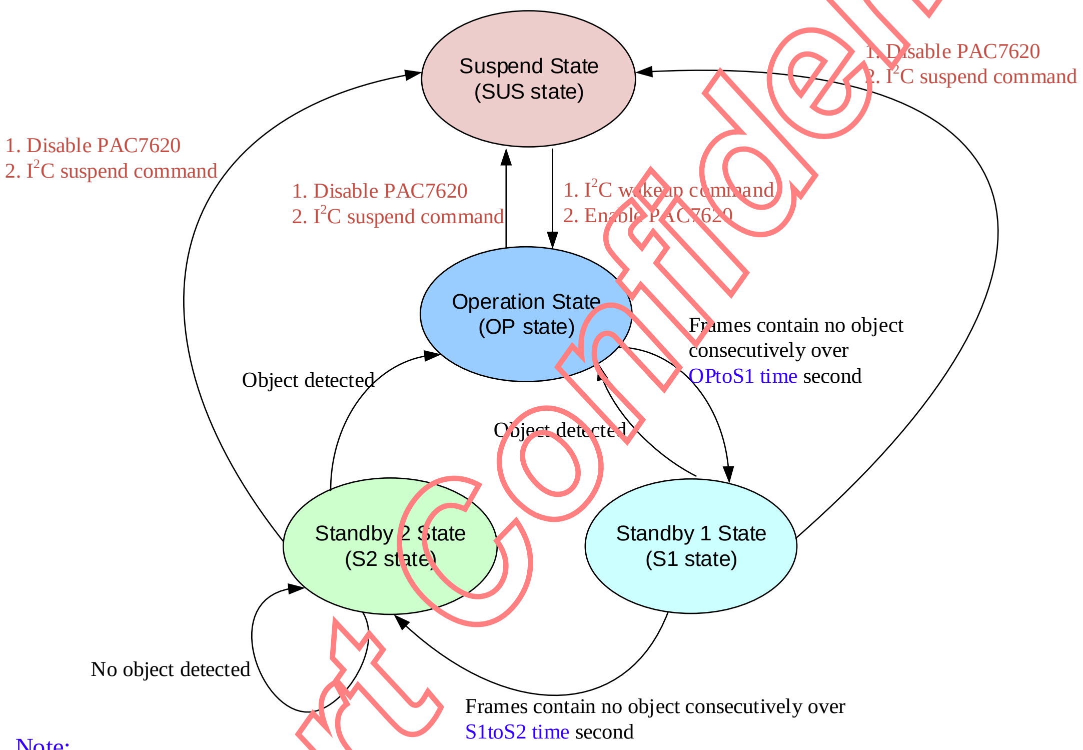


Figure 23. Suspend State (SUS state) Diagram

6.1.3.5 State Machine



Note:

OPtoS1 time = OPtoS1 step/120 @ Normal Mode
= OPtoS1 step/240 @ Gaming Mode

S1toS2 time = S1toS2 step/(60 x$S_{1, \text{Response Factor}}$) @ Normal Mode
= S1toS2 step/(120 x$S_{1, \text{Response Factor}}$) @ Gaming Mode

Figure 24. State Machine of Gesture Detection

6.2 Image mode

6.2.1 Output Image Timing

6.2.1.1 30x30, Frame Subtraction Mode

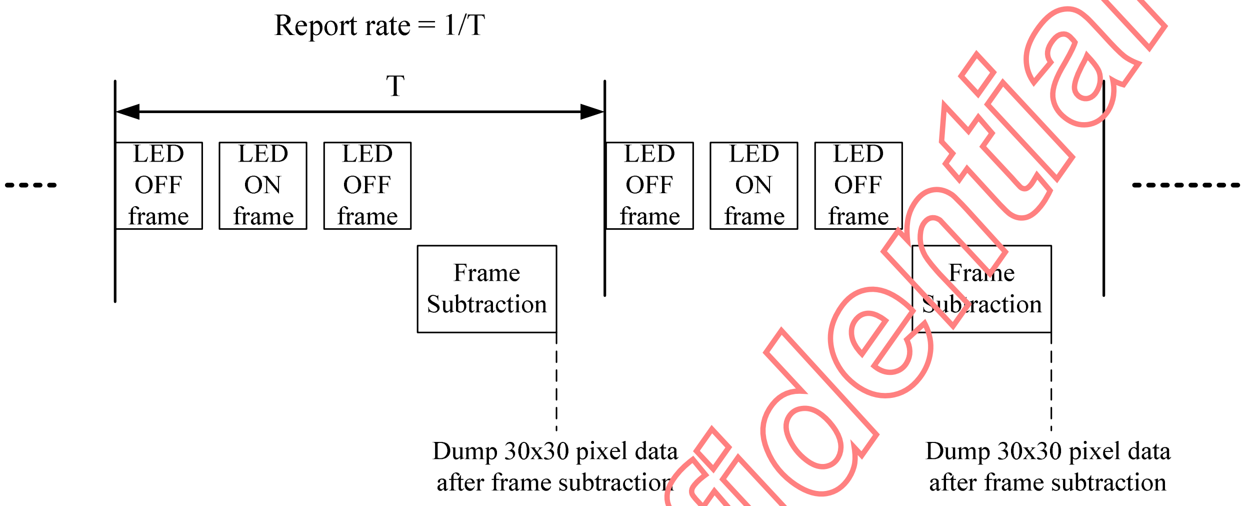
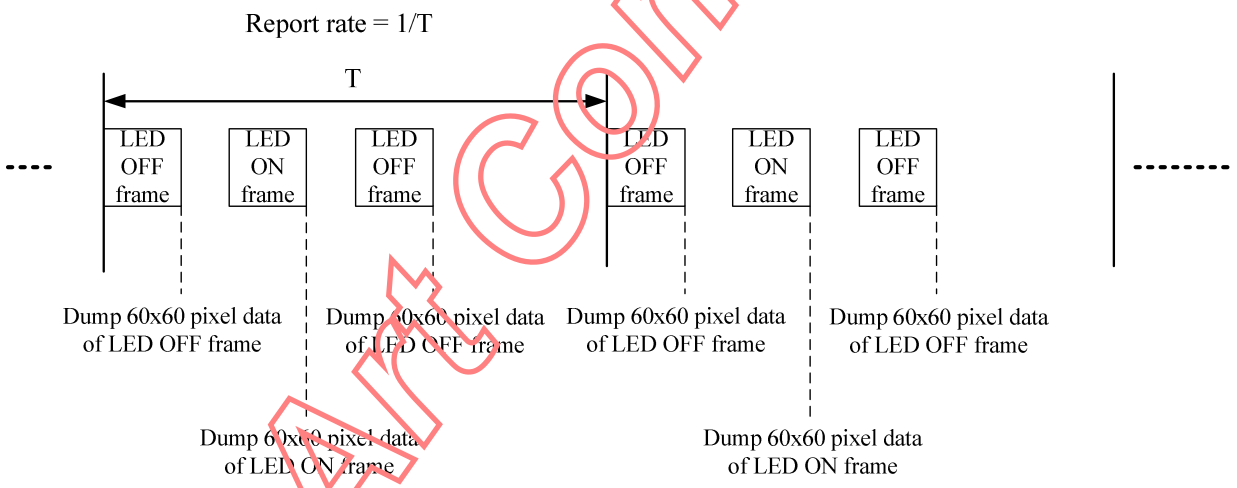


Figure 25. Timing of Output Image for 30x30 pixels

6.2.1.2 60x60, Raw Data Mode


Report rate = 1/T
T

LED OFF frame
LED ON frame
LED OFF frame
LED OFF frame
LED ON frame
LED OFF frame

Dump 60x60 pixel data of LED OFF frame
Dump 60x60 pixel data of LED OFF frame
Dump 60x60 pixel data of LED OFF frame
Dump 60x60 pixel data of LED OFF frame
Dump 60x60 pixel data of LED ON frame
Dump 60x60 pixel data of LED ON frame

Figure 26. Timing of Output Image for 60x60 pixels

6.2.2 Suspend State

To enter the suspend state, first disable the SPI output by writing Register Bank 1, ADDR 0x7E with 0x00. Second, disable the PAC7620 by writing Register Bank 1, ADDR 0x72 with 0x00 then process the I^{2}C suspend command by writing Register Bank 0, ADDR 0x03 with 0x00.

To exit the suspend state, first process the I^{2}C wake-up command by writing the slave ID (Refer to topic “I^{2}C Interface”). Second, enable the PAC7620 by writing Register Bank 1, ADDR 0x72 with 0x01 then enable the SPI output by writing Register Bank 1, ADDR 0x7E with 0x01.

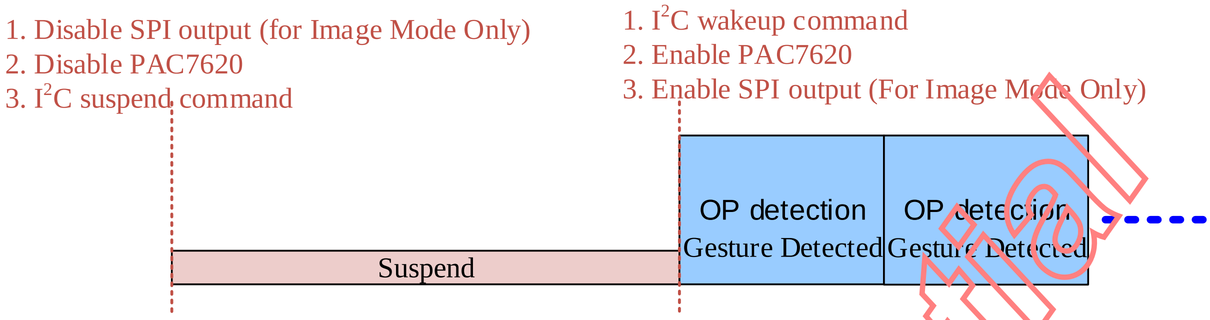


Figure 27. Suspend State (SUS state) Diagram

6.3 Proximity Detection Mode

Refer to Figure 2828 on how to set the Proximity Sensing (PS) hysteresis window and the interrupt mechanism of proximity detection.

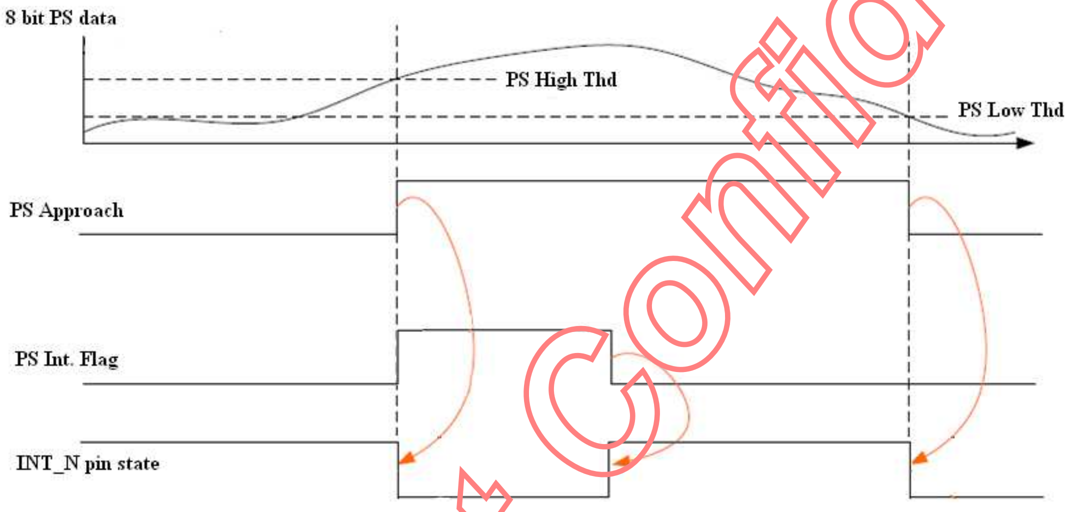


Figure 28. Proximity Sensing Functional Diagram

6.3.1 Proximity Detection Operating State

6.3.1.1 Proximity Operation State (PS OP state)

When in operation state, the update rate is 10Hz and the LED on time is 8µs. The LED peak current is 760 mA.

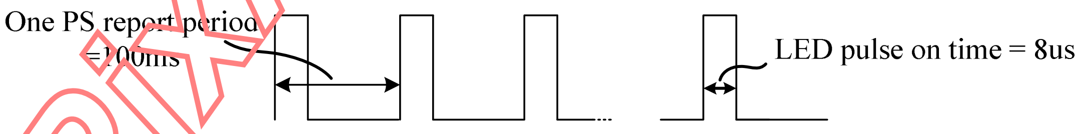


Figure 29. Proximity Operation State (PS OP state) Diagram

6.3.1.2 Suspend State (SUS state)

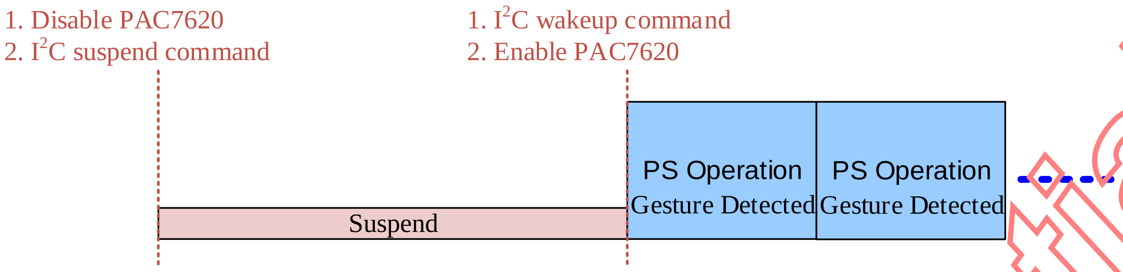


Figure 30. Suspend State (SUS state) Diagram

6.3.1.3 State Machine

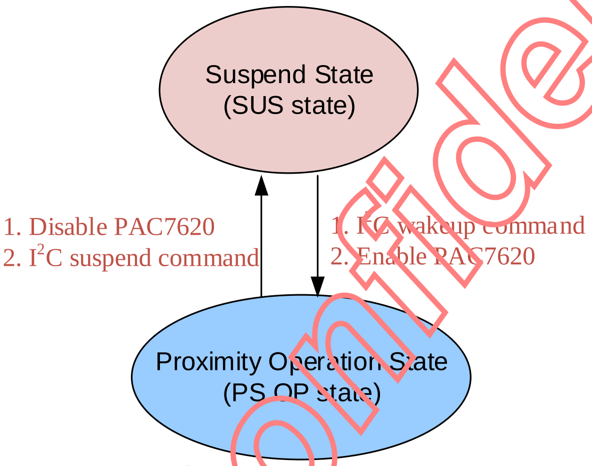


Figure 31. State Machine of Proximity Detection

7.0 Register Tables

7.1 Register Bank Select

Bank	Address	Register Name	Default Value	R/W	Description
0/1	0xEF	R_RegBankSel[0]	0x00	R/W	Register Bank Select. 0 : Register Bank 0 1 : Register Bank 1
1	0x7F	R_RegBankSel[0]	0x00	R/W	Register Bank Select. 0 : Register Bank 0 1 : Register Bank 1

7.2 Image Size Setting

Bank	Address	Register Name	Default Value	R/W	Description
0	0xAA	R_ImageHeight[5:0]	0x1E	R/W	DSP image vertical size
0	0xAB	R_ImageWidth[5:0]	0x1E	R/W	DSP image horizontal size
1	0x00	Cmd_HSize[5:0]	0x1E	R/W	Horizontal size
1	0x01	Cmd_VSize[5:0]	0x1E	R/W	Vertical size
1	0x02	Cmd_HStart[5:0]	0x00	R/W	Horizontal start point
1	0x03	Cmd_VStart[5:0]	0x00	R/W	Vertical start point
1	0x04	Cmd_ASkip_V[5]	0x01	R/W	Analog vertical skip
1	0x04	Cmd_ASkip_H[4]	0x01	R/W	Analog horizontal skip
1	0x04	Cmd_DAvg_V[3]	0x00	R/W	Digital vertical average
1	0x04	Cmd_VFlip[1]	0x00	R/W	Vertical flip
1	0x04	Cmd_HFlip[0]	0x00	R/W	Horizontal flip

7.3 Setting 30x30 Pixels Image Output Mode

Bank	Address	Bit	2x skip mode	2x2 average mode	WOI mode
0	0xAA	5:0	30	30	30
0	0xAB	5:0	30	30	30
1	0x00	5:0	30	30	30
1	0x01	5:0	30	60	30
1	0x02	5:0	0	0	15
1	0x03	5:0	0	0	15
1	0x04	7	0	1	0
1		6	0	0	0
1		5	1	0	0
1		4	1	1	0
1		3	0	1	0
1		2	0	0	0

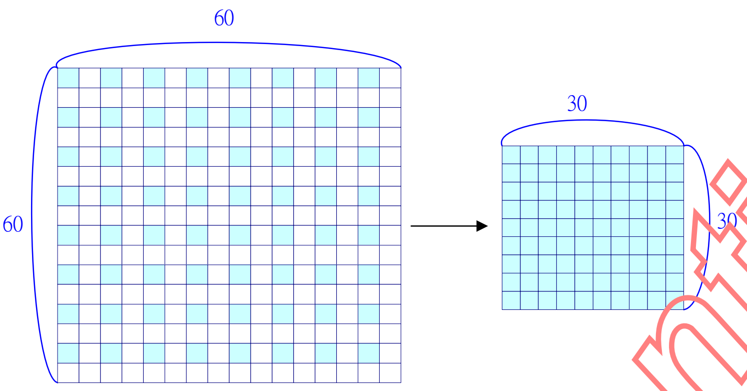


Figure 32. 2x Skip Mode

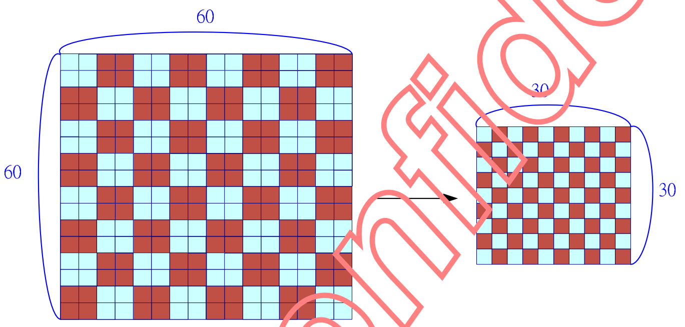


Figure 33. 2x2 Average Mode

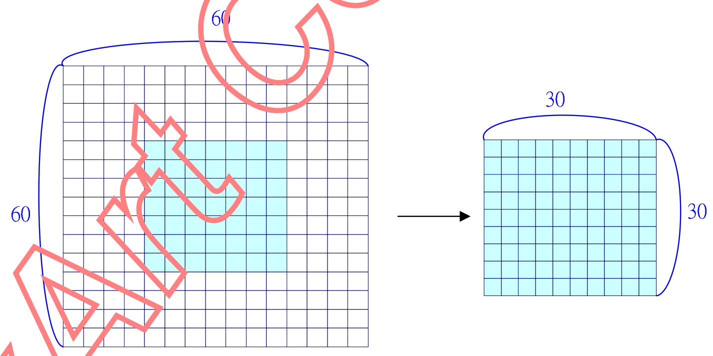


Figure 34. WOI Mode

7.4 AE/AG Controls

Bank	Address	Register Name	Default Value	R/W	Description
0	0x46	R_AELedOff_UB[7:0]	0x60	R/W	Decrease exposure time, if AE_LED_Off_YAvg (Bank 0, Reg 0x58) > R_AELedOff_UB
0	0x47	R_AELedOff_LB[7:0]	0x20	R/W	Increase exposure time, if AE_LED_Off_YAvg (Bank 0, Reg 0x58) < R_AELedOff_LB,
0	0x48	R_AE_Exposure_UB[7:0]	0x20	R/W	Auto exposure time up bound, Exposure time (us) = R_AE_Exposure_UB / 4
0	0x49	R_AE_Exposure_UB[15:8]	0x03	R/W	
0	0x4A	R_AE_Exposure_LB[7:0]	0xC8	R/W	Auto exposure time low bound, R_AE_Exposure_LB = R_AE_Exposure_UB / 2 (important)
0	0x4B	R_AE_Exposure_LB[15:8]	0x00	R/W	
0	0x4C	R_AE_Gain_UB[7:0]	0x14	R/W	Auto gain up bound, Gain = 1+ R_AE_Gain_UB/16
0	0x4D	R_AE_Gain_LB[7:0]	0x00	R/W	Auto gain low bound, Gain = 1+ R_AE_Gain_LB/16
0	0x4E	R_AE_Gain_Step[3:0]	0x0A	R/W	Gain stage adjust step, new gain stage = current gain stage - this
0	0x4E	R_SleepAEAG_AutoDisable [4]	0x01	R/W	Under sleeping, if Gain Stage is at UB, disable wakeup AE mode. Wake up AE mode: when sleeping, AE/AG will be set to higher value.
0	0x4F	R_AE_Gain_Default[6:0]	0x14	R/W	Gain stage default value at AE turning on.
0	0x50	R_Exp_Sel[2:0]	0x00	R/W	At R_Manual_Exposure_Default=1, Exp = R_AE_Exposure_UP/2^(R_Exp_Sel)
0	0x51	R_Manual_GG[0]	0x01	R/W	1: Gain Manual mode, ggh gain= R_ggh, global gain=R_glogal. 0: Gain Auto mode
0	0x51	R_Manual_Exposure[1]	0x00	R/W	1: Exposure Manual mode, exposure time = R_AE_Exposure_UB. 0: Auto Exposure mode
0	0x51	R_Manual_Exposure_Default[2]	0x01	R/W	Exp Manual Mode, Exp = R_AE_Exposure_UP/2^(R_Exp_Sel)
0	0x51	R_AE_EnH[4]	0x00	R/W	1: AE Enable 0: AE Disable
0	0x54	AG_stage_GG[7:0]	-	R	Gain Stage, gain for analog = 2^AG_stage_GG[7:4]
0	0x55	Reg_ExposureNum[7:0]	-	R	Currently exposure time, Exposure time (us) = Reg_ExposureNum/4
0	0x56	Reg_ExposureNum[15:8]	-	R	
0	0x57	Reg_ggh[1:0]	-	R	ggh gain 0:1x, 2:2x, 3:4x
0	0x57	Reg_global[3:0]	-	R	Bit[7:4] : global gain 0:1x, 8:2x Total gain = ggh*global = 2^AG_stage_GG[7:4]

Bank	Address	Register Name	Default Value	R/W	Description
0	0x58	AE_LED_Off_YAvg[8:1]	-	R	Bit[7:0]: OFF Frame average brightness
0	0x59	AE_Dec[0]	-	R	AE decrease identifier
0	0x59	AE_Inc[1]	-	R	AE increase identifier
0	0x5A	AE_Normal_Factor[2:0]	-	R	AE normalize factor, DSP data = sensor image data * 2^AE_Normal_Factor
1	0x42	R_global[3:0]	0x08	R/W	Bit[7:4]: PGA global gain in gain manual mode
1	0x44	R_ggh[1:0]	0x00	R/W	Bit[7:6]: PGA ggh gain in gain manual mode

7.5 GPIO Setting

Bank	Address	Register Name	Default Value	R/W	Description
0	0x80	Im_GPIO0	-	R	Bit[0] : GPIO0 Stats at input mode
0	0x80	Tm_GPIO0_OEL	0x01	R/W	Bit[1] : 0: set GPIO0 as output
0	0x80	Tm_GPIO0_IEB	0x01	R/W	Bit[2] : 0: set GPIO0 as input
0	0x80	R_GPIO0	0x01	W	Bit[3] : GPIO0 default value at output mode. To set GPIOX as Input, X=0~3: Set: Tm_GPIOX_OEL=1, Tm_GPIOX_IEB=0. At this time, Im_GPIOX = GPIO Stats. To set GPIOX as Output, X=0~3: Set: Tm_GPIOX_OEL=0, Tm_GPIOX_IEB=1. At this time, Tm_GPIOX = R_GPIOX
0	0x80	Im_GPIO1	-	R	Bit[4] : GPIO1 Stats at input mode
0	0x80	Tm_GPIO1_OEL	0x01	R/W	Bit[5] : 0: set GPIO1 as output
0	0x80	Tm_GPIO1_IEB	0x01	R/W	Bit[6] : 0: set GPIO1 as input
0	0x80	R_GPIO1	0x01	W	Bit[7] : GPIO1 default value at output mode.
0	0x80	Im_GPIO2	-	R	Bit[0] : GPIO2 Stats at input mode
0	0x81	Tm_GPIO2_OEL	0x01	R/W	Bit[1] : 0: set GPIO2 as output
0	0x81	Tm_GPIO2_IEB	0x01	R/W	Bit[2] : 0: set GPIO2 as input
0	0x81	R_GPIO2	0x01	W	Bit[3] : GPIO2 default value at output mode.
0	0x80	Im_GPIO3	-	R	Bit[4] : GPIO3 Stats at input mode
0	0x81	Tm_GPIO3_OEL	0x01	R/W	Bit[5] : 0: set GPIO3 as output
0	0x81	Tm_GPIO3_IEB	0x01	R/W	Bit[6] : 0: set GPIO3 as input
0	0x81	R_GPIO3	0x01	W	Bit[7] : GPIO3 default value at output mode.
0	0x82	Im_INT	-	R	Bit[0] : INT pin status
0	0x82	Tm_INT_OEL	0x00	R/W	Bit[1] 0: set INT as output
0	0x82	Tm_INT_IEB	0x01	R/W	Bit[2] : 0: set INT as input
0	0x82	Tm_INT	0x01	W	Bit[3] : INT default value at output mode. To set INT as Input: Set Tm_INT_OEL=1, Tm_INT_IEB=0. At this time, Im_INT = INT_N Stats. To set INT as Output: Set: Tm_INT_OEL=0, Tm_INT_IEB=1. At this time, Tm_INT = INT_N

7.6 Interrupt Controls

Bank	Address	Register Name	Default Value	R/W	Description
0	0x40	R_MCU_IntFlagGClr	0x01	R/W	Bit[1] : 1: Auto clean intflag_1/Intflag_2 after I2C readout.
0	0x40	R_MCU_IntFlagInv	0x00	R/W	Bit[4] 1: INT pin high active, 0:INT pin low active
0	0x41	R_Int_1_En[7:0]	0xFF	R/W	If the corresponding bit is 1: the corresponding interrupt event is enabled Bit[0] Up Bit[1] Down Bit[2] Left Bit[3] Right Bit[4] Forward Bit[5] Backward Bit[6] Clockwise Bit[7] Counterclockwise
0	0x42	R_Int_2_En[7:0]	0xFF	R/W	If the corresponding bit is 1: the corresponding interrupt event is enabled Bit[0] Wave, wave mode use only Bit[1] Proximity, proximity mode use only Bit[2] Has Object, cursor mode use only Bit[3] Wake up trigger, trigger mode use only Bit[4] N/A Bit[5] N/A Bit[6] N/A Bit[7] No Object, cursor mode use only
0	0x43	IntFlag_1[7:0]	-	R	When interrupt event happens, the corresponding bit is set to 1 Bit[0] Up Bit[1] Down Bit[2] Left Bit[3] Right Bit[4] Forward Bit[5] Backward Bit[6] Clockwise Bit[7] Counterclockwise
0	0x44	IntFlag_2[7:0]	-	R	When interrupt event happens, the corresponding bit is set to 1 Bit[0] Wave, wave mode use only Bit[1] Proximity, proximity mode use only Bit[2] Has Object, cursor mode use only Bit[3] Wake up trigger, trigger mode use only Bit[4] N/A Bit[5] N/A Bit[6] N/A Bit[7]No Object, cursor mode use only

7.7 Gesture Mode Controls

Bank	Address	Register Name	Default Value	R/W	Description
0	0x83	R_LightThd[7:0]	0x20	R/W	Only if the pixel > this, it would be taken as the part of object. Ex: Pixel array: [10 10 40 40 40 10], R_LightThd=32 => Object array: [0 0 40 40 40 0]
0	0x84	R_GestureStartTh[7:0]	0x20	R/W	When the objet size is larger than this, state machine goes to has object state
0	0x85	R_GestureStartTh[9:8]	0x00	R/W	Bit[1:0]
0	0x86	R_GestureEndTh[7:0]	0x10	R/W	When the objet size is less than this, state machine goes out of has object state
0	0x87	R_GestureEndTh[9:8]	0x00	R/W	Bit[1:0]
0	0x88	R_ObjectMinZ[4:0]	0x05	R/W	Z direction minimum threshold
0	0x89	R_ObjectMaxZ[5:0]	0x18	R/W	Z direction maximum threshold
0	0x8C	R_ProcessResolution[1:0]	0x03	R/W	Bit[5:4] Object of gesture detection resolution
0	0x8D	R_TimeDelayNum[7:0]	0x00	R/W	The detection gap between this gesture and the next gesture
0	0x8E	R_Disable45Degree	0x00	R/W	Bit[0] 45 degree gesture detection 1 : Disable 45 degree gesture detection
0	0x8E	R_45DegreeRatio[3:0]	0xF0	R/W	Bit[7:4] The ratio to define 45 degree
0	0x8F	R_XtoYGain	0x01	R/W	Bit[0] X and Y direction gain enable
0	0x8F	R_XYGainRatio[3:0]	0x08	R/W	Bit7:4] X and Y direction gain ratio.
0	0x90	R_NoMotionCountThd[6:0]	0x0C	R/W	No motion counter threshold to quit has motion state
0	0x91	R_NoObjectCountThd[6:0]	0x06	R/W	No object counter threshold to quit has object state
0	0x92	R_NormalizedImageWidth [4:0]	0x1E	R/W	Image normalized factor
0	0x93	R_XDirectionThd[4:0]	0x0D	R/W	Gesture detection horizontal threshold
0	0x94	R_YDirectionThd[4:0]	0x0A	R/W	Gesture detection vertical threshold
0	0x95	R_ZDirectionThd[4:0]	0x0C	R/W	Gesture detection z direction threshold
0	0x96	R_ZDirectionXYThd[4:0]	0x0A	R/W	Gesture detection x and y threshold to detect forward or backward
0	0x97	R_ZDirectionAngleThd[3:0]	0x04	R/W	Gesture detection angle threshold to detect forward or backward
0	0x98	R_RotateAngleThd[5:0]	0x0A	R/W	Gesture detection angle threshold to detect rotation
0	0x99	R_RotateContiEnh	0x01	R/W	Bit[0]Continuous rotation gesture detection enable
0	0x99	R_RotateContiThd[5:0]	0x00	R/W	Bit[5:1] Continuous rotation gesture detection angle threshold

Bank	Address	Register Name	Default Value	R/W	Description
0	0x9A	R_RotateXYThd[4:0]	0x0A	R/W	Gesture detection x and y threshold to detect rotation
0	0x9B	R_RotateZThd[4:0]	0x0A	R/W	Gesture detection z threshold to detect rotation
0	0x9C	R_FilterWeight[1:0]	0x03	R/W	Bit[1:0] IIR filter weight between frame position distance
0	0x9C	R_FilterDistThd[4:0]	0x0A	R/W	Bit[6:2] IIR filter frame position distance threshold
0	0x9D	R_StartDistThd[3:0]	0x03	R/W	Bit[3:0]Object position difference between frames threshold to enter the process state
0	0x9D	R_EndDistThd[2:0]	0x03	R/W	Bit[6:4]Object position difference between frames threshold to quit the process state
0	0x9F	R_RotateEnh	0x01	R/W	Bit[4] Rotate gesture detection enable
0	0x9F	R_ZDirectionEnh	0x01	R/W	Bit[5] Backward and Forward gesture detection enable
0	0x9F	R_YDirectionEnh	0x01	R/W	Bit[6] Up and down gesture detection enable
0	0x9F	R_XDirectionEnh	0x01	R/W	Bit[7] Left and right gesture detection enable
0	0xA5	R_FilterImage	0x01	R/W	Bit[0] ref to R_FilterAverage_Mode
0	0xA5	R_FilterAverage_Mode	0x00	R/W	Bit[3:2] Image filter mode: 0: weak average, 1: strong average, 2: 3 out of 9 median average.
0	0xA5	R_UseLightWeight	0x01	R/W	Bit[4] Use pixel brightness as weight to calculate center enable
0	0xA9	R_DiffAngleThd[3:0]	0x04	R/W	Frame angle accumulation threshold
0	0xAC	ObjectCenterX[7:0]	-	R	Horizontal Object Center
0	0xAD	ObjectCenterX[12:8]	-	R	Bit[4:0] Horizontal Object Center
0	0xAE	ObjectCenterY[7:0]	-	R	Vertical Object Center
0	0xAF	ObjectCenterY[12:8]	-	R	Bit[4:0] Vertical Object Center
0	0xB0	ObjectAvgY[8:1]	-	R	Object brightness, Max 255.
0	0xB1	ObjectSize[7:0]	-	R	Object size, Max 900.
0	0xB2	ObjectSize[11:8]	-	R	Bit[3:0] Object size, Max 900.
0	0xB3	Gx[5:0]	-	R	Gesture x direction movement
0	0xB4	Gy[5:0]	-	R	Gesture y direction movement
0	0xB5	Gz[6:0]	-	R	Gesture z direction movement
0	0xB6	GestureResult[3:0]	-	R	Bit[3:0]Gesture result: 1:up, 2:down, 3:left, 4:right, 5:forward, 6:backword, 7:clockwise 8:counterclockwise, 9:wave, 10:abort, result no keep, debug use only

Bank	Address	Register Name	Default Value	R/W	Description
0	0xB6	State[1:0]	-	R	Bit[5:4] DSP FSM state: 0:Initial ,1: Process, 2:End
0	0xB7	WaveCount[3:0]	-	R	Bit[3:0] Wave gesture counter
0	0xB7	AbortCount[2:0]	-	R	Bit[6:4] Abort gesture counter
0	0xB8	NoObjectCount[7:0]	-	R	No object counter
0	0xB9	NoMotionCount[7:0]	-	R	No motion counter
0	0xBA	LightCount[5:0]	-	R	Bright object counter
0	0xBB	LightAcc[7:0]	-	R	Object brightness accumulation
0	0xBC	LightAcc[9:8]	-	R	Bit[1:0] Object brightness accumulation
0	0xBD	TimeAcc[7:0]	-	R	Gesture time period
0	0xBE	TimeAcc[12:8]	-	R	Bit[4:0] Gesture time period
0	0xBF	GxAcc[7:0]	-	R	Gesture x direction movement accumulation, debug use only
0	0xC0	GxAcc[9:8]	-	R	Bit[1:0] Gesture x direction movement accumulation, debug use only
0	0xC1	GyAcc[7:0]	-	R	Gesture y direction movement accumulation, debug use only
0	0xC2	GyAcc[9:8]	-	R	Bit[1:0] Gesture y direction movement accumulation, debug use only
0	0xC3	VelX[7:0]	-	R	Gesture x direction velocity, debug use only
0	0xC4	VelX[11:8]	-	R	Bit[3:0] Gesture x direction velocity, debug use only
0	0xC5	VelY[7:0]	-	R	Gesture y direction velocity, debug use only
0	0xC6	VelY[11:8]	-	R	Bit[3:0] Gesture y direction velocity, debug use only
0	0xC7	AngleAcc[7:0]	-	R	Gesture angle accumulation
0	0xC8	AngleAcc[10:8]	-	R	Bit[2:0] Gesture angle accumulation
0	0xC9	CurAngle[4:0]	-	R	Current gesture angle, debug use only
0	0xCA	XGainValue[7:0]	-	R	45 degree gesture detection x direction parameter
0	0xCB	YGainValue[7:0]	-	R	45 degree gesture detection y direction parameter
0	0xCC	R_YtoZSum[5:0]	0x1A	R/W	Z direction mapping parameter
0	0xCD	R_YtoZFactor[5:0]	0x0D	R/W	Z direction mapping parameter
0	0xCE	R_PositionFilterLength[2:0]	0x03	R/W	IIR Filter length for cursor object center
0	0xCE	R_ProcessFilterLength[2:0]	0x00	R/W	Bit[6:4] IIR Filter length for gesture object center
0	0xCF	R_WaveCountThd[3:0]	0x03	R/W	Wave gesture counter threshold
0	0xCF	R_WaveAngleThd[3:0]	0x06	R/W	Bit[7:4] Wave gesture angle threshold
0	0xD0	R_AbortCountThd[2:0]	0x22	R/W	Bit[2:0] Abort gesture counter threshold
0	0xD0	R_AbortXYRatio[4:0]		R/W	Bit[7:3] Abort gesture X and Y direction ratio
0	0xD1	R_AbortLength[6:0]	0x0F	R/W	Abort gesture movement distance threshold
0	0xD2	R_AbortIntervalCountThd [5:0]	0x88	R/W	Abort gesture gap duration threshold

Bank	Address	Register Name	Default Value	R/W	Description
0	0xD2	R_ConfirmMode		R/W	Bit[6] Confirm mode enable
0	0xD2	R_WaveEnh		R/W	Bit[7] Wave gesture detection enable
0	0xD3	PositionFilterCenterX[7:0]	-	R	Horizontal Object Center after IIR filter for cursor mode
0	0xD4	PositionFilterCenterX[11:8]	-	R	Bit[3:4] Horizontal Object Center after IIR filter for cursor mode
0	0xD4	PositionFilterCenterY[11:8]	-	R	Bit[7:4] Vertical Object Center after IIR filter for cursor mode
0	0xD5	PositionFilterCenterY[7:0]	-	R	Vertical Object Center after IIR filter for cursor mode
0	0xD6	PositionFilterAvgY[7:0]	-	R	Object brightness after IIR filter for cursor mode
0	0xD7	PositionFilterAvgY[8]	-	R	Bit[0] Object brightness after IIR filter for cursor mode
0	0xD7	PositionFilterSize[9:8]	-	R	Bit[5:4] Object size after IIR filter for cursor mode
0	0xD8	PositionFilterSize[7:0]	-	R	Object size after IIR filter for cursor mode
0	0xD9	ProcessFilterSize[9:8]	-	R	Bit[1:0] Object size after IIR filter for cursor mode
0	0xD9	ProcessFilterAvgY[8]	-	R	Bit[2] Object brightness before IIR filter for cursor mode
0	0xD9	AbortIntervalCount[9:8]	-	R	Bit[5:4]
0	0xDA	ProcessFilterCenterX[7:0]	-	R	Horizontal Object Center after IIR filter for gesture detection
0	0xDB	ProcessFilterCenterX[11:8]	-	R	Bit[3:0] Horizontal Object Center after IIR filter for gesture detection
0	0xDB	ProcessFilterCenterY[11:8]	-	R	Bit[7:4] Vertical Object Center after IIR filter for gesture detection
0	0xDC	ProcessFilterCenterY[7:0]	-	R	Vertical Object Center after IIR filter for gesture detection
0	0xDD	ProcessFilterSize[7:0]	-	R	Object size after IIR filter for gesture detection
0	0xDe	ProcessFilterAvgY[7:0]	-	R	Object brightness after IIR filter for gesture detection
0	0xDF	AbortIntervalCount[7:0]	-	R	Abort gesture gap duration

7.8 Cursor Mode Controls

Bank	Address	Register Name	Default Value	R/W	Description
0	0x32	R_CursorUseTop	0x01	R/W	Bit[0] 1: Enable cursor center function, When enable this, the cursor object information would involve R_CursorTopRatio in calculation. Ref to R_CursorTopRatio.
0	0x32	R_CursorUseBGModel	0x00	R/W	Bit[1] 1: If cursor center function is enable, involve background model in calculation
0	0x32	R_CursorInvertY	0x00	R/W	Bit[2] 1: Horizontal cursor center inverse
0	0x32	R_CursorInvertX	0x01	R/W	Bit[3] 1: Vertical cursor center inverse
0	0x32	R_CursorTopRatio[1:0]	0x02	R/W	Bit[5:4] Use the first R_CursorTopRatio/8 part of the object to calculate center. Ex: If original object size is 100, then the cursor object size would be 25 at R_CursorTopRatio = 2. Also, cursor center would move from center of the object to upper 1/4 part.
0	0x33	R_PositionFilterStartSizeTh [7:0]	0x01	R/W	As object size>this threshold, use the object information after IIR filter for cursor
0	0x34	R_PositionFilterStartSizeTh[8]	0x00	R/W	Bit[0]
0	0x35	R_ProcessFilterStartSizeTh [7:0]	0x01	R/W	As object size>this threshold, use the object information after IIR filter for gesture detection
0	0x36	R_ProcessFilterStartSizeTh[8]	0x00	R/W	Bit[0]
0	0x37	R_CursorClampLeft[4:0]	0x09	R/W	Minimum cursor horizontal center value, if the data is less than this, clamp at 0. Otherwise, clamp at data-(R_CursorClampLeft<<R_PositionResolution) Ex: R_CursorClampLeft=9, R_CursorClampRight=21, R_PositionResolution=0 X=5 => X'=0 X=10 => X'=10-9=1 X=25 => X'=20-1-9=10 Ex: If R_CursorInvertY=0 R_CursorClampRight=20 R_PositionResolution=2 and Y=50 Then when setting R_CursorInvertY=1, R_CursorClampRight=20 R_PositionResolution=2 => Y' = 20*(2^2)-50=30
0	0x38	R_CursorClampRight[4:0]	0x15	R/W	Maximum cursor horizontal center value, if the data is larger than this, clamp at this -1

Bank	Address	Register Name	Default Value	R/W	Description
0	0x39	R_CursorClampUp[4:0]	0x0A	R/W	Minimum cursor vertical center value, if the data is less than this, clamp at 0 Otherwise, clamp at data-(R_CursorClampUp<<R_PositionResolution)
0	0x3A	R_CursorClampDown[4:0]	0x12	R/W	Maximum cursor vertical center value, if the data is larger than this, clamp at this -1
0	0x3B	CursorClampCenterX[7:0]	0x00	R	Clamping Center X[7:0]
0	0x3C	CursorClampCenterX[11:8]	0x7D	R	Bit[3:0] Clamping Center X[11:8]
0	0x3D	CursorClampCenterY[7:0]	0x0F	R	Clamping Center Y[7:0]
0	0x3E	CursorClampCenterY[11:8]	0xA0	R	Bit[3:0] Clamping Center Y[11:8]
0	0x8B	R_Cursor_ObjectSizeTh[7:0]	0x10	R/W	The object size threshold for cursor mode. If cursor object size > this, trigger cursor has object interrupt
0	0x8C	R_PositionResolution[2:0]	0x07	R/W	Object of cursor mode resolution, Ex:(x,y)=(1000,1000) at R_PositionResolution=7 means real (x',y') = (1000/2^7,1000/2^7) = (7.8125,7.8125) which maps to 30x30 scale

7.9 Proximity Mode Controls

Bank	Address	Register Name	Default Value	R/W	Description
0	0x69	R_Pox_UB[7:0]	0xC8	R/W	Proximity Up Bound
0	0x6A	R_Pox_LB[7:0]	0x40	R/W	Proximity Low Bound
0	0x6B	S_State	-	R	PS approach state, S_State. Approach = 1 , (S_AvgY >= R_Pox_UB) Not Approach = 0 , (S_AvgY <= R_Pox_LB) (Only functional at proximity detection mode)
0	0x6C	S_AvgY[8:1]	-	R	Proximity object average brightness

7.10 EFuse Program Controls

Bank	Address	Register Name	Default Value	R/W	Description
0	0x70	R_EFuse_A[5:0]	0x00	R/W	Bit[5:0] EFuse Address
0	0x70	R_EFuse_READ	0x00	R/W	Bit[6] EFuse Read
0	0x70	R_EFuse_ENB	0x01	R/W	Bit[7] EFuse Enable
0	0x71	R_EFuse_SEL[2:0]	0x00	R/W	Bit[2:0] EFuse Select
0	0x71	R_PROGRAM_BitsCnt[3:0]	0x00	R/W	Bit[7:4] Program Bit Count
0	0x72	R_EFuse_PROGRAM_En	0x00	R/W	Bit[0] EFuse Program Enable
0	0x72	EFuse_PROGRAM	0x00	R	Bit[7] EFuse Program Identifier
0	0x73	R_EFuse_PROGRAM_CLKCnt[14:8]	0x00	R/W	Bit[7:0] EFuse Program Clock Count
0	0x74	R_EFuse_PROGRAM_CLKCnt[7:0]	0xF0	R/W	Bit[7:0] EFuse Program Clock Count
0	0x75	R_EFuse_PROGRAM_Data[7:0]	0x00	R/W	EFuse Program Data
0	0x76	EFuse_Dout[63:56]	-	R	Bit[7:0] EFuse Data
0	0x77	EFuse_Dout[55:48]	-	R	Bit[7:0] EFuse Data
0	0x78	EFuse_Dout[47:40]	-	R	Bit[7:0] EFuse Data
0	0x79	EFuse_Dout[39:32]	-	R	Bit[7:0] EFuse Data
0	0x7A	EFuse_Dout[31:24]	-	R	Bit[7:0] EFuse Data
0	0x7B	EFuse_Dout[23:16]	-	R	Bit[7:0] EFuse Data
0	0x7C	EFuse_Dout[15:8]	-	R	Bit[7:0] EFuse Data
0	0x7D	EFuse_Dout[7:0]	-	R	Bit[7:0] EFuse Data

7.11 Background Controls

Bank	Address	Register Name	Default Value	R/W	Description
0	0x9F	R_UseBGModel	0x01	R/W	Bit[0] Background model enable
0	0x9F	R_BGUseDiffWeight	0x00	R/W	Bit[1] During calculating object center, use the weight between background and pixel.
0	0x9F	R_BGUpdateAtProcess	0x00	R/W	Bit[2] Update background at process state
0	0x9F	R_BGUpdateMaxIntensity_En	0x01	R/W	Bit[3] Background up bound threshold enable
0	0xA0	R_BGUpdateMaxIntensity[7:0]	0x03	R/W	Background up bound threshold, if pixel data is > this*2, update this pixel to BG SRAM
0	0xA1	R_BGFilterLengthUp[1:0]	0x01	R/W	Bit[1:0]The IIR filter weight of updating background not at process state IIR Filter Out (X: previous data, X':current date) = {X*2^(R_BGFilterLength-1)+X'}/2^R_BGFilterLength
0	0xA1	R_BGFilterLengthDown[1:0]	0x10	R/W	Bit[5:4] The IIR filter weight of updating background at process state
0	0xA2	R_BGDiffThd[5:0]	0x08	R/W	Only if the difference between BG and pixel > this, it would be taken as the part of object
0	0xA3	R_BGUpdateFreq[7:0]	0x00	R/W	Update background per this number frames
0	0xA4	R_BGUpdateFreq[9:8]	0x03	R/W	Bit[1:0] Update background per this number frames
0	0xA4	R_BGResolution[1:0]	0x03	R/W	Bit[5:4] Background resolution, BG data = pixel data<<R_BGResolution

7.12 Lens Shading Compensation

Bank	Address	Register Name	Default Value	R/W	Description
1	0x04	R_LS_Comp_DAvg_V	0x00	R/W	Bit[7] Lens Shading for digital vertical average
1	0x25	R_LensShadingComp_EnH	0x00	R/W	Lens Shading compensation enable, active high
1	0x26	R_OffsetX[6:0]	0x00	R/W	horizontal offset of lens (s+6, -63~63)
1	0x27	R_OffsetY[6:0]	0x00	R/W	vertical offset of lens (s+6, -63~63)
1	0x28	R_LSC[6:0]	0x40	R/W	A, R2 coeff, (un-signed, 0~127)
1	0x29	R_LSFT[3:0]	0x0A	R/W	shift amount of A*R2(un-signed, 0~15)

7.13 LED Controls

Bank	Address	Register Name	Default Value	R/W	Description
1	0x30	R_LED_SoftStart_time[7:0]	0x03	R/W	LED soft start time before TS_ab falling
1	0x31	R_LED2_DAC_EnL	0x00	R/W	Bit[5] 0=turn on the LED_DAC2 1=turn off the LED_DAC2
1	0x31	R_LED1_DAC_EnL	0x00	R/W	Bit[4] 0=turn on the LED_DAC1 1=turn off the LED_DAC1
1	0x31	R_LED2_DAC_manual	0x00	R/W	Bit[1] LED2 manual enable, active high 0:TS_LED2_EnL/Tm_LEDC2_DAC controlled by digital 1:TS_LED2_EnL/Tm_LEDC2_DAC controlled by R_LED2_EnL/R_LED2_DAC_UB
1	0x31	R_LED1_DAC_manual	0x00	R/W	Bit[0] LED1 manual enable, active high 0:TS_LED1_EnL/Tm_LEDC1_DAC are controlled automatically by digital 1:TS_LED1_EnL/Tm_LEDC1_DAC are controlled manually by R_LED1_EnL/R_LED1_DAC_UB
1	0x32	R_LED1_DAC_UB[4:0]	0x14	R/W	LED1 upper bound
1	0x33	R_LED2_DAC_UB[4:0]	0x14	R/W	LED2 upper bound
1	0x34	R_LEDC_Step_Up[4:0]	0x07	R/W	step up : B_LEDC_Step_Up[4:0]=0, let TS_LED1_thermo[30:0] & TS_LED2_thermo[30:0] all current DAC off

					B_LEDC_Step_Up[4:0]=1, let TS_LED1_thermo[30:0] & TS_LED2_thermo[30:0] starts from thermo<0> sequentially
1	0x35	R_LEDC_Step_Down[4:0]	0x07	R/W	step down : B_LEDC_Step_Up[4:0]=0, let TS_LED1_thermo[30:0] & TS_LED2_thermo[30:0] all current DAC off B_LEDC_Step_Up[4:0]=1, let TS_LED1_thermo[30:0] & TS_LED2_thermo[30:0] starts from thermo<31> sequentially

7.14 Sleep Mode

Bank	Address	Register Name	Default Value	R/W	Description
0	0x45	SleepMode_Status[1:0]	-	R	Indicate operation status, 0:normal, 1:weak sleep, 2:deep sleep. normal operation: use R_IDLE_TIME for IDLE weak sleep: use R_IDLE_TIME_SLEEP_1 for IDLE deep sleep: use R_IDLE_TIME_SLEEP_2*2 for IDLE See R_IDLE_TIME
0	0x52	R_SleepAE1[1:0]	0x00	R/W	Bit[7:6] At weak sleep, 1:Exp'=Exp*2 , 2:Exp'=Exp*4, other: Exp'=Exp Wake up AE mode setting. Only active under sleep status
0	0x52	R_SleepAE2[1:0]	0x00	R/W	Bit[5:4] At deep sleep, 1:Exp'=Exp*2 , 2:Exp'=Exp*4, other: Exp'=Exp Wake up AE mode setting. Only active under sleep status
0	0x52	R_SleepAG1[1:0]	0x00	R/W	Bit[3:2] At weak sleep, 1:Gain'=Gain*2, 2:Gain'=Gain*4, other: Gain'=Gain Wake up AE mode setting. Only active under sleep status
0	0x52	R_SleepAG2[1:0]	0x00	R/W	Bit[1:0] At deep sleep, Gain'=Gain*2 , 2:Gain' Gain*4, Gain' = Gain Wake up AE mode setting. Only active under sleep status
1	0x65	R_IDLE_TIME[7:0]	0xB4	R/W	idle time for normal operation

Bank	Address	Register Name	Default Value	R/W	Description
1	0x66	R_IDLE_TIME[15:8]			Unit: T= 256/System CLK = 32us Ex: Far Mode: 1 report time = (77+R_IDLE_TIME)T Report rate 120 fps: R_IDLE_TIME=1/(120*T)-77=183 Report rate 240 fps: R_IDLE_TIME=1/(240*T)-77=53 Near Mode: 1 report time = (112+R_IDLE_TIME)T Report rate 120 fps: R_IDLE_TIME=1/(120*T)-112=148 Report rate 240 fps: R_IDLE_TIME=1/(240*T)-112=18
1	0x67	R_IDLE_TIME_SLEEP_1[7:0]	0x168	R/W	idle time for weak sleep In weak sleep status, using this idle time. Report Rate=120*8.333/(0.0323*this+X), X=2.41 for Far Mode, 3.55 for Near Mode
1	0x68	R_IDLE_TIME_SLEEP_1[15:8]			
1	0x69	R_IDLE_TIME_SLEEP_2[7:0]	0x2D0	R/W	idle time for deep sleep, 2X In deep sleep status, using this idle time*2. Report Rate=120*8.333/(0.0646*this+X), X=2.41 for Far Mode, 3.55 for Near Mode
1	0x6A	R_IDLE_TIME_SLEEP_2[15:8]			
1	0x6B	R_Obj_TIME_1[7:0]	0x4B0	R/W	Weak sleep enter time, unit: one report frame time.
1	0x6C	R_Obj_TIME_1[15:8]			
1	0x6D	R_Obj_TIME_2[7:0]	0x960	R/W	Deep sleep enter time, unit: two report frame time.
1	0x6E	R_Obj_TIME_2[15:8]			
1	0x6F	R_TG_INIT_TIME[7:0]	0x32	R/W	fine tune of idle time for normal operation, unit: 4us
1	0x71	R_TG_POWERON_WAKEUP_TIME	0x00	R/W	wakeup time after TG_START for Analog CDS power on stable, unit: 4us
1	0x72	R_TG_EnH	0x00	R/W	TG enable, After setting all parameters, enable this at last.
1	0x73	R_AUTO_SLEEP_Mode1	0x00	R/W	Bit[1:0] The way goes to weak sleep, 0: No motion sleep, 1: No object sleep
1	0x73	R_AUTO_SLEEP_Mode2	0x00	R/W	Bit[3:2] The way goes to deep sleep, 1: No object sleep

Bank	Address	Register Name	Default Value	R/W	Description
1	0x73	R_AUTO_WakeUP_Mode	0x00	R/W	Bit[5:4] The way to wake up, bit[4]: 0: motion wake up at weak sleep 1: object wake up at weak sleep bit[5]: 0: motion wake up at deep sleep, 1: object wake up at deep sleep
1	0x74	R_WakeUpSig_Sel[1:0]	0x00	R/W	Bit[5:4]1: GPIO0 output TG_Start signal, 2: GPIO0 output TG_Finish signal

7.15 Clock Controls

Bank	Address	Register Name	Default Value	R/W	Description
0	0x3F	R_oscrcpd	0x00	R/W	Internal Osc. Control
0	0x5C	R_SenClkPrd[5:0]	0x02	R/W	Analog or TG CLK = System CLK/R_SenClkPrd. System CLK=8MHz
0	0x5D	R_SENCLK_Control	0x00	R/W	Sensor Clock Control
0	0x5E	R_Other_CLK_manual	0x00	R/W	Clock Manual Mode Controls
0	0x5F	R_CLK_manual	0x7F	R/W	Clock Manual Mode Controls
0	0x60	TS_osc_code[6:0]	-	R	OSC code in use
0	0x61	R_osc_code[6:0]	0x34	R/W	OSC code
0	0x63	R_Ref_CLK_Cnt_UB[15:8]	0x03	R/W	OSC bist clock count up bound
0	0x64	R_Ref_CLK_Cnt_UB[7:0]	0xF7	R/W	OSC bist clock count up bound
0	0x65	R_Ref_CLK_Cnt_LB[15:8]	0x03	R/W	OSC bist clock count low bound
0	0x66	R_Ref_CLK_Cnt_LB[7:0]	0xD9	R/W	OSC bist clock count low bound
0	0x67	R_OSC_Wait[7:0]	0x03	R/W	Reference clock sample cycle
0	0x68	T_osc_range[1:0]	0x01	R/W	For Analog Use

7.16 Chip/Version ID

Bank	Address	Register Name	Default Value	R/W	Description
0	0x00	PartID[7:0]	0x20	R	Part ID
0	0x01	PartID[15:8]	0x76	R	
0	0x02	VersionID[7:0]	0x01	R	Version ID

7.17 Suspend/Reset

Bank	Address	Register Name	Default Value	R/W	Description
0	0x03	SW_Suspend_EnL	0x01	W	0: software suspend
0	0xEE	R_RegBank Reset	0x01	W	0: Register Bank 0 reset
0	0xEE	R_TGRegBank_SWRstn	0x01	R/W	0: Register Bank 1 reset
0	0xEE	R_DMSRegBank_SWRstn	0x01	R/W	0: DSP reset

7.18 Test Mode

Bank	Address	Register Name	Default Value	R/W	Description
0	0xE0	R_DSP_TestObj_Size[3:0]	0x01	R/W	Bit[3:0] Test Pattern object size. The square object size is R_DSP_TestObj_Size^2.
0	0xE0	R_DSP_Test_Mode[3:0]	0x00	R/W	Bit[7:4] 0: disable test mode 1: reset test mode 2: Object move from LU corner to RU corner 3: Object move from RU corner to LU corner 4: Object move from LU corner to LD corner 5: Object move from LD corner to LU corner(L,R,U,D corner ref to Addr228~230 note) 6: Object move from LU=>RU=>RD=>LD=>LU 7: Object move from RU=>LU=>LD=>RD=>RU 8: Fix Pattern, object number = R_DSP_TestObj_Num^2 Object Position is starting from LU corner. 9: Fix Pattern, object number = R_DSP_TestObj_Num Object Position is starting from LU corner.
0	0xE1	R_DSP_TestObj_Hi[7:0]	0x04	R/W	Test Pattern object brightness
0	0xE2	R_DSP_TestObj_Lo[7:0]	0x41	R/W	Test Pattern background

Bank	Address	Register Name	Default Value	R/W	Description
0	0xE3	R_DSP_TestObj_Dist[3:0]	0x06	R/W	Bit[3:0] Test Pattern distance between objects. If object one center is (5,5) and R_DSP_TestObj_Dist=10. Then, object two center is (5,15) for horizontal movement. or object two center is (15,5) for vertical movement
0	0xE3	R_DSP_TestObj_Speed[3:0]	0x0D	R/W	Bit[7:4] Test Pattern object move speed. If object center is (5,5) at frame one, and R_DSP_TestObj_Speed = 6, movement is L to R, then, object center is (11,5) at frame two.
0	0xE4	R_DSP_TestObj_PosL[5:0]	0x00	R/W	Test Pattern object left position parameter. L corner starting point. Ex: R_DSP_TestObj_PosL=5 R_DSP_TestObj_PosU=6 R_DSP_TestObj_Size = 10 In LU condition, there is an square object from (5,6) to (14,15)
0	0xE5	R_DSP_TestObj_PosR[5:0]	0x8C	Test Pattern object right position parameter	Test Pattern object right position parameter. R corner starting point. Ex: R_DSP_TestObj_PosR=15 R_DSP_TestObj_PosU=6 R_DSP_TestObj_Size = 10 In RU condition, there is an square object from (15,6) to (24,15)
0	0xE6	R_DSP_TestObj_PosU[5:0]	0x4A	R/W	Test Pattern object up position parameter
0	0xE7	R_DSP_TestObj_PosD[5:0]	0x00	R/W	Test Pattern object down position parameter
0	0xE8	R_DSP_TestObj_FrameCnt[5:0]	0x00	R/W	Bit[5:0] Test Pattern gesture frame counter. The active frame number of object movement, which needs to be long enough to cover the movement event.

Bank	Address	Register Name	Default Value	R/W	Description
					Ex: The object movement: (1,1)=>(1,2)=>(1,3) This frame count should be larger than 3.
0	0xE8	R_DSP_TestObj_Num[1:0]	0x00	R/W	Bit[7:6] Test Pattern object number. In mode 8, R_DSP_TestObj_Num^2 is the number of objects showing on the screen. Otherwise, R_DSP_TestObj_Num is the number of objects showing on the screen.
0	0xE9	R_DSP_Test_Noise_EnH	0x00	R/W	Test Pattern random noise enable. The test pattern would involve random noise

7.19 Reserved Registers List

Bank	Address	Register Name	Default Value	R/W	Description
0	0x5B	Trigger	0x00	R/W	Trigger mode use only, as setting to one, trigger IC to report one frame
0	0x6D	R_SRAM_DS[3:0]	0x04	R/W	SRAM Bist Test
0	0x6E	R_BIST_mode[3:0]	0x00	R/W	
0	0x6F	R_BIST_mode_1[3:0]	0x00	R/W	
0	0x8A	R_LockReg	0x00	R/W	Bit[0] Manually lock the value of specified register
0	0x8A	R_AutoLock	0x10	R/W	Bit[4] Auto lock the value of specified register as read the register
1	0x05	Cmd_DebugMode	0x00	R/W	TG debug mode 0: disable (normal mode) 1: ramp 0~255 2: ramp 255~0 3: fixed pattern, Cmd_DebugPattern[8:0]
1	0x06	Cmd_CDS_Timing_Control	0x4B0	R/W	CDS Timing Control
1	0x07	Cmd_CDS_Timing_Control		R/W	CDS Timing Control
1	0x08	Cmd_CDS_Timing_Control	0x0D	R/W	CDS Timing Control
1	0x09	Cmd_CDS_Timing_Control	0x0E	R/W	CDS Timing Control
1	0x0A	Cmd_CDS_Timing_Control	0x708	R/W	CDS Timing Control
1	0x0B	Cmd_CDS_Timing_Control		R/W	CDS Timing Control
1	0x0C	Cmd_CDS_Timing_Control	0x05	R/W	CDS Timing Control
1	0x0D	Cmd_CDS_Timing_Control	0x0F	R/W	CDS Timing Control
1	0x0E	Cmd_CDS_Timing_Control	0x02	R/W	CDS Timing Control

1	0x0F	Cmd_CDS_Timing_Control	0x12	R/W	CDS Timing Control
1	0x10	Cmd_CDS_Timing_Control	0x02	R/W	CDS Timing Control
1	0x11	Cmd_CDS_Timing_Control	0x02	R/W	CDS Timing Control
1	0x12	Cmd_CDS_Timing_Control	0x00	R/W	CDS Timing Control
1	0x13	Cmd_CDS_Timing_Control	0x01	R/W	CDS Timing Control
1	0x14	Cmd_CDS_Timing_Control	0x05	R/W	CDS Timing Control
1	0x15	Cmd_CDS_Timing_Control	0x07	R/W	CDS Timing Control
1	0x16	Cmd_CDS_Timing_Control	0x05	R/W	CDS Timing Control
1	0x17	Cmd_CDS_Timing_Control	0x07	R/W	CDS Timing Control
1	0x18	Cmd_CDS_Timing_Control	0x01	R/W	CDS Timing Control
1	0x19	Cmd_CDS_Timing_Control	0x04	R/W	CDS Timing Control
1	0x1A	Cmd_CDS_Timing_Control	0x05	R/W	CDS Timing Control
1	0x1B	Cmd_CDS_Timing_Control	0x0C	R/W	CDS Timing Control
1	0x1C	Cmd_CDS_Timing_Control	0x2A	R/W	CDS Timing Control
1	0x1D	Cmd_CDS_Timing_Control	0x01	R/W	CDS Timing Control
1	0x1E	R_TG_Mode	0x00	R/W	Bit[1:0] TG operating mode 0: TG_START followed by 3 frames (OFF -> ON -> OFF) on-(off1+off2)/2 1: TG_START followed by 1 frame (ON/OFF switches) on-off 2: TG_START followed by 1 frame (always ON) on 3: TG_START followed by 1 frame (ON/OFF switches)
1	0x21	R_OnOffNegComp[7:0]	0x00	R/W	pre-compensation for negative of (ON - OFF_avg)
1	0x22	R_OnOffNegComp[8]	0x00	R/W	Bit[0] pre-compensation for negative of (ON - OFF_avg)
1	0x23	R_RowDummyTrailing0_EnH	0x00	R/W	Designed for DSP to cut off object at end of row 0: disable row dummy trailing 2 bytes (16'h0000) => output 40x40 1: enable row dummy trailing 2 bytes (16'h0000) => output 42x40
1	0x36	R_I2C_AutoResync_EnH	0x01	R/W	I2C auto re-sync enable, active high
1	0x37	R_I2C_AutoResync_Time[7:0]	0xFF	R/W	time that I2C slave hold SDA after clock in -> re-sync, unit: us, min = 100, max = 255
1	0x38	R_adc_control	0x03	R/W	R_ADC_Control
1	0x39	R_pwrsv_control	0x01	R/W	Power saving mode control
1	0x3A	R_I2CID_Sel[2:0]	0x00	R/W	0: 7'h73 1: 7'h13 2: 7'h1b 3: 7'h23

					4: 7'h2b 5: 7'h5b 6: 7'h63 7: 7'h6b
1	0x3E	Cmd_DebugPattern[7:0]	0x00	R/W	debug pattern for TG output
1	0x3F	Cmd_DebugPattern[8]	0x00	R/W	Bit[0] debug pattern for TG output
1	0x40	R_cds_mode	0x00	R/W	CDS mode setting
1	0x41	R_dac_control	0x00	R/W	DAC mode setting
1	0x43	R_pga_test	0x00	R/W	Bit[6] PGA test mode
1	0x45	R_adc_test	0x00	R/W	ADC Test mode
1	0x46	R_ScanDAC_Control	0x00	R/W	SCANDAC Test
1	0x47	R_dac_control	0x00	R/W	DAC Control
1	0x48	R_PDN_Test	0x01	R/W	PDN Test Mode
1	0x4A	R_CP_WOI_HSize[5:0]	0x1E	R/W	size of CP image width
1	0x4B	R_CP_WOI_VSize[5:0]	0x1E	R/W	size of CP image height
1	0x4C	R_CP_WOI_HStart[5:0]	0x00	R/W	CP image start column address
1	0x4D	R_CP_WOI_VStart[5:0]	0x00	R/W	CP image start row address
1	0x4E	R_Clamp_control	0xA0	R/W	Clamp Control
1	0x4F	R_WAKE_ivreg_EnL[7:0]	0x80	R/W	power on sequence control timing
1	0x50	R_bgp_control	0x00	R/W	Bgp Control
1	0x51	R_TS_Test	0x00	R/W	TS Mode
1	0x52	R_ref_gen_EnL_Manual	0x00	R/W	1:TS_ref_gen_EnL = R_ref_gen_EnL
1	0x53	R_ref_gen_EnL	0x00	R/W	manual value for TS_ref_gen_EnL
1	0x54	R_ref_gen_forceNoPwrsv_EnH	0x00	R/W	TS_ref_gen_EnL force no power saving
1	0x57	R_WAKE_tg_EnL[7:0]	0x80	R/W	power on sequence control timing
1	0x59	T_ab_blk_EnH	0x01	R/W	ab block function enable
1	0x5A	T_cds_asout_EnH	0x00	R/W	probe out R_gcout through I/O
1	0x5B	T_pixbias[2:0]	0x04	R/W	select 1st source follow bias
1	0x5C	T_pxoset_EnH	0x01	R/W	enable pxoset
1	0x5D	T_tg_allow_EnH	0x00	R/W	Bit[7]all tg turn off
1	0x5E	T_clamp_drv_ctl[1:0]	0x01	R/W	Bit[7:6]Constant gm clamp circuit driver NMOS number select
1	0x5F	T_vdda28_lvl[2:0]	0x04	R/W	VDDA28 voltage select: 2.5, 2.58, 2.65, 2.72, 2.82, 2.92, 3.02, 3.18
1	0x60	T_vrtb_shift[1:0]	0x01	R/W	"Vrt-Vrb" voltage com selection: 0.85V, 0.8V, 0.75V, 0.7V
1	0x61	T_flush_lvl[2:0]	0x04	R/W	flush voltage selection : 0.8v, 0.9v, 1.0v, 1.1v, 1.2v, 1.3v, 1.4v, 1.5v
1	0x63	T_adc_vcmtest_EnH	0x00	R/W	ADC vcm test signal control
1	0x64	T_GPIO_OPDRV[1:0]	0x00	R/W	Bit[3:2] GPIO driver capability setting.
1	0x74	R_Control_Mode[2:0]	0x00	R/W	Bit[2:0]

					0:normal mode 1:wake up mode. 2:Slave gesture mode. 3:Master Cursor Mode 4:Slave Cursor Mode 5:PS Mode
1	0x74	R_GPIO_Probe_En	0x00	R/W	Bit[7:6] GPIO debug probe select, to use this probe, set GPIO as output.
1	0x75	T_ckt_test0	0x00	R/W	Test Circuit
1	0x76	T_I2C_OPDRV[1:0]	0x01	R/W	I2C driver capability setting. T_I2C_OPDRV<1:0>=1
1	0x77	R_SRAM_Read_EnH	0x00	R/W	SRAM read enable, active high
1	0x7C	R_SPIOUT_PXDNUM[7:0]	0x384	R/W	SPI output pixel amount
1	0x7D	R_SPIOUT_PXDNUM[15:8]		R/W	
1	0x7E	R_SPIOUT_CSN_Mode[1:0]	0x00	R/W	Bit[3:2] 0: follow TG_VsyncO 1: follow IDLE time 2: follow LED 3: reserved
1	0x7E	R_SPIOUT_EnH	0x00	R/W	Bit[0] SPI output enable, active high

8.0 Firmware Guides

This chapter describes how to implement firmware for PAJ7620U2 and the functional applications.

8.1 Initialization of PAJ7620U2

4. Power on the PAJ7620U2 sensor chip. The V_{Bus} must be powered on before V_{DD}.
5. Wait 700µs for PAJ7620U2 to stabilize.
6. Write slave ID or I^2C read command to process I^2C wake-up. It is recommended to read Reg_0x00 for return value of "0x20" to indicate that the wake-up process is completed. There is no-ack from PAJ7620U2 before wake-up process is ready.
7. Write initialization setting to sensor

```
unsigned char initial_register_array[][2] = {
    {0xEF,0x00},
    {0x37,0x07},
    {0x38,0x17},
    {0x39,0x06},
    {0x42,0x01},
    {0x46,0x2D},
    {0x47,0x0F},
    {0x48,0x3C},
    {0x49,0x00},
    {0x4A,0x1E},
    {0x4C,0x20},
    {0x51,0x10},
    {0x5E,0x10},
    {0x60,0x27},
    {0x80,0x42},
    {0x81,0x44},
    {0x82,0x04},
    {0x8B,0x01},
    {0x90,0x06},
    {0x95,0x0A},
    {0x96,0x0C},
    {0x97,0x05},
    {0x9A,0x14},
    {0x9C,0x3F},
    {0xA5,0x19},
    {0xCC,0x19},
    {0xCD,0x0B},
    {0xCE,0x13},
    {0xCF,0x64},
    {0xD0,0x21},
    {0xEF,0x01}
    {0x02,0x0F},
    {0x03,0x10},
    {0x04,0x02},
    {0x25,0x01},
    {0x27,0x39},
    {0x28,0x7F},
    {0x29,0x08},
    {0x3E,0xFF},
    {0x5E,0x3D},
    {0x65,0x96},
    {0x67,0x97},
    {0x69,0xCD},
    {0x6A,0x01},
    {0x6D,0x2C},
    {0x6E,0x01},
    {0x72,0x01},
    {0x73,0x35},
    {0x74,0x00},
    {0x77,0x01},
};
```

8.2 Get Gesture Result

1. Set up the Interrupt or I^{2}C polling timer.
2. Read Bank_0_Reg_0x43/0x44 for gesture result if interrupt or timer is triggered.

Note: Gesture result will be cleared after I^{2}C read access.

8.3 Change to PS Mode

1. Write PS mode setting.

```
unsigned char change_to_proximity_register_array[][2] = {
        {0xEF,0x00},
        {0x41,0x00},
        {0x42,0x02},
        {0x48,0x20},
        {0x49,0x00},
        {0x51,0x13},
        {0x83,0x00},
        {0x9F,0xF8},
        {0x69,0x96},
        {0x6A,0x02},
        {0xEF,0x01},
        {0x01,0x1E},
        {0x02,0x0F},
        {0x03,0x10},
        {0x04,0x02},
        {0x41,0x50},
        {0x43,0x34},
        {0x65,0xCE},
        {0x66,0x0B},
        {0x67,0xCE},
        {0x68,0x0B},
        {0x69,0xE9},
        {0x6A,0x05},
        {0x6B,0x50},
        {0x6C,0xC3},
        {0x6D,0x50},
        {0x6E,0xC3},
        {0x74,0x05},
    };
```

8.4 Get PS Approach Status

Read Bank_0_Reg_0x6B for PS approach status or read Bank_0_Reg_0x6C for PS raw data.

8.5 Change to Gesture mode

1. Write Gesture mode setting to gesture.

```
unsigned char change_to_gesture_register_array[][2] = {
    {0xEF,0x00},
    {0x41,0x00},
    {0x42,0x00},
    {0xEF,0x00},
    {0x48,0x3C},
    {0x49,0x00},
    {0x51,0x10},
    {0x83,0x20},
    {0x9f,0xf9},
    {0xEF,0x01},
    {0x01,0x1E},
    {0x02,0x0F},
    {0x03,0x10},
    {0x04,0x02},
    {0x41,0x40},
    {0x43,0x30},
    {0x65,0x96},
    {0x66,0x00},
    {0x67,0x97},
    {0x68,0x01},
    {0x69,0xCD},
    {0x6A,0x01},
    {0x6b,0xb0},
    {0x6c,0x04},
    {0x6D,0x2C},
    {0x6E,0x01},
    {0x74,0x00},
    {0xEF,0x00},
    {0x41,0xFF},
    {0x42,0x01},
};
```

8.6 Enter Suspend mode

1. Write Suspend mode setting to gesture.

```
unsigned char suspend_register_array[][2] = {
        {0xEF,0x01},
        {0x72,0x00},
        {0xEF,0x00},
        {0x03,0x01},
};
```

8.7 Resume to Gesture Mode

1. Write slave ID or I2C read command to process I^2C wake-up.

   It’s recommend to read Reg_0x00. It will return “0x20” when wake-up finish

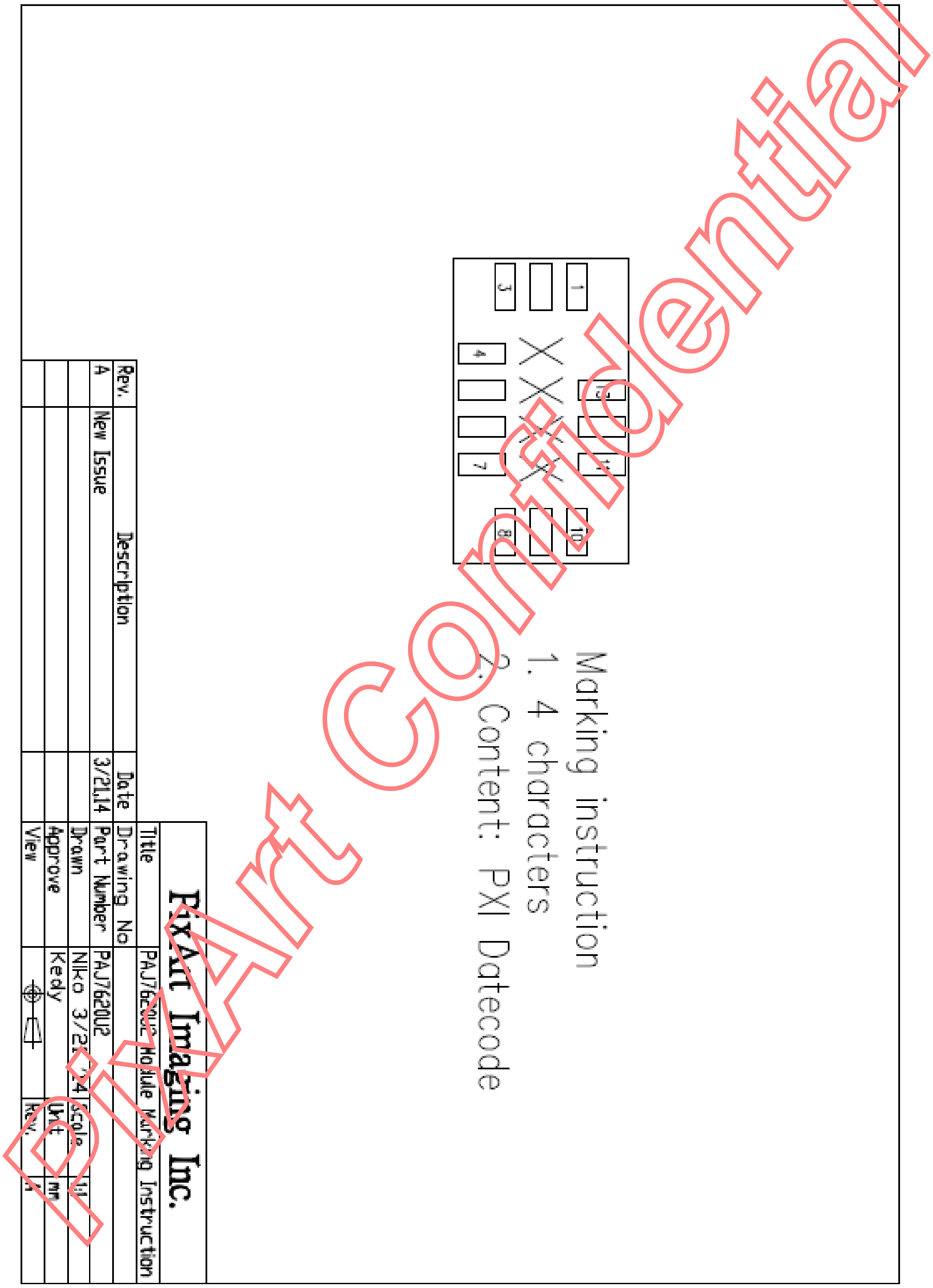
   By the way, There is no-ack from PAJ7620U2 before wake-up finish.

2. Write Resume setting to gesture.

```
unsigned char resume_register_array[][2] = {
        {0xEF,0x01},
        {0x72,0x01},
        {0xEF,0x00},
        {0xEE,0x03},
        {0xEE,0x07},
};
```

9.0 Handling Information

9.1 Marking Information



9.2 Packing Information

9.2.1 Packing method of using 2 inch tray, Module orientation should follow Figure.35 definition. Stack 10 trays with one cover tray in a bunch. (see Figure.36)

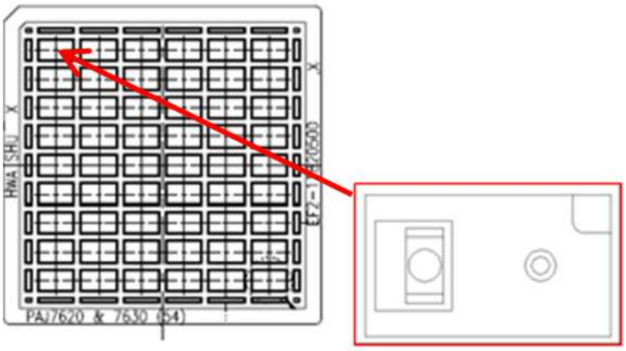


Figure. 35 Module orientation

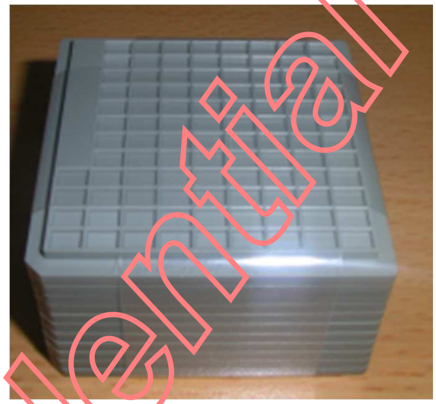


Figure. 36 Stack 10 +1 Tray

9.2.2 Pack 2 bunches of tray into one aluminum laminate moisture proof bag. (see Figure.37)

9.2.3 Pack 5 aluminum bags into a packing box (see Figure.38)

Figure.37 Al packing bag

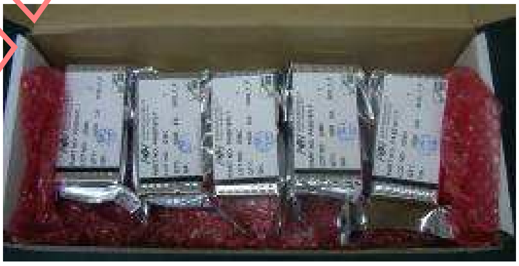


Figure.38 Packing Box Recommended PCB Layout

9.2.4 The maximum capacity of one packing box using tray of PAJ7620U2 :

One packing box	**5400 units**
Remark	(54ea per tray * 100 tray per box)

9.3 Notes

9.3.1 Please tap gently the cover before open the 2”tray

提醒您 Remind
Tray 盤打開前請先輕敲上蓋
Please tap gently the cover before open

9.3.2 When the units is out of dry packing, should be operation at :

Temperature = ↓30℃ , Humidity = ↓60% RH

9.3.3 If units is out of dry packing over 168hrs, before go through infrared the reflow process must be bake with 125℃ ±5℃ @24hrs, to remove moisture.

9.3.4 2 inch tray is not heat resistant. To dry packages at high temperatures, therefore remove products from tray and place them in suitable metal carrier before baking

10.0 Recommended Guideline for PCB Assembly

Recommended vender and type for Pb-free solder paste

1. Almit LFM-48W TM-HP
2. Senju M705-GRN360-K

IR Reflow Soldering Profile:

Temperature profile is the most important control in reflow soldering. It must be fine tuned to establish a robust process. The typical recommended IR reflow profile is showed in figure.39 below.

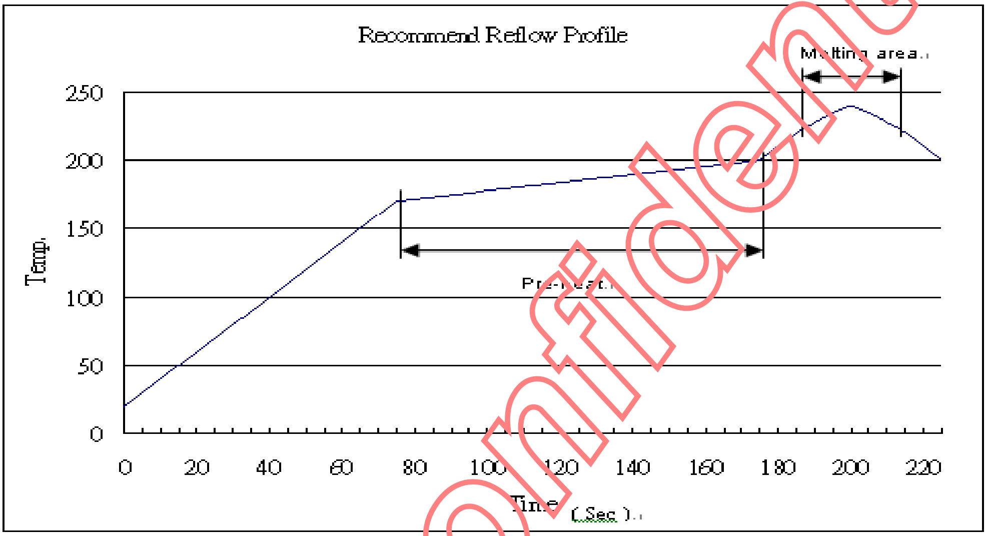


Figure.39 IR Reflow Profile

Reflow Profile :

1. Average Ramp-up Rate (30°C to preheat zone): 1.5~ 2.5 Degree C/ Sec
2. Preheat zone:
   - 2.1 Temp ramp from 170~200 degree C
   - 2.2 Exposure time: 90 +/- 30 sec
3. Melting zone:
   - 3.1 Melting area temp > 220 degree C for at least 30 ~ 50 sec

3.2 Peak temperature : 245 degree C.

11.0 Handling precaution for the prevention of ESD PCB

Explained below are procedures that must be taken in fabrication to prevent the electrostatic destruction of semiconductor devices.

The following basic rules must be obeyed.

1. Equalize potentials of terminals when transporting or storing.

2. Equalize the potentials of the electric device, work table, and operator's body that may come in contact with the IC's.

3. Prepare an environment that does not generate static electricity.

One method is keeping relative humidity in the work room to about 50%.

Operator

1.The operator should wear wrist straps.

(Must maintain electric contact with bare skin)

2. Wear cotton or antistatic-treated materials clothes and gloves.

3. When a conductive mat will be used, must be ware conductive shoes.

4. Do not touch the IC's leads. Touch the body of IC's when holding.

Equipment and tools

1. Any electrical equipments and tools located on the work table surface must be isolated from The work table surface, and ground the equipments and tools that are to be used.

2. Work table surface must be use conductive material or conductive mat. (Should be ground through a 1MΩ resistor)

Transporting, storing and packing

1. Use conductive IC's tray, and conductive or shielding bag to store module.

Soldering operation

1. Use a soldering iron with a grounding wire.

2. When perform manual soldering operation, the operator should wear wrist straps.

3. Do not use the desoldering pump when removing the module from the PCB board. Use a solder-wick or equivalent.

Document Revision History

Revision No.	Date	Description of Change(s)
1.0	2015/02/13	Released version
1.1	2016/01/15	Updated to new template format
1.2	2016/02/29	Add Handling Information