| *Application Note* | *THAN0196_Rev.1.30_E* |
|---|---|

# THCV231(-Q)/THCV236(-Q) Application Note

System Diagram, Register Setting and PCB Design Guideline

Contents

## Application Diagram

### RAW12bit per pixel falling edge system

GPIO4 and GPIO3 input to the THCV236(-Q) are output from the THCV231(-Q) by through GPIO function in order to send camera reset signal etc.

Set RF pin **Low** (Falling edge LVCMOS input).

BET, BETOUT and LATEN pins can be used as Bit Error Test in actual configuration in order for debug or test purpose.

**[THCV231(-Q)]**

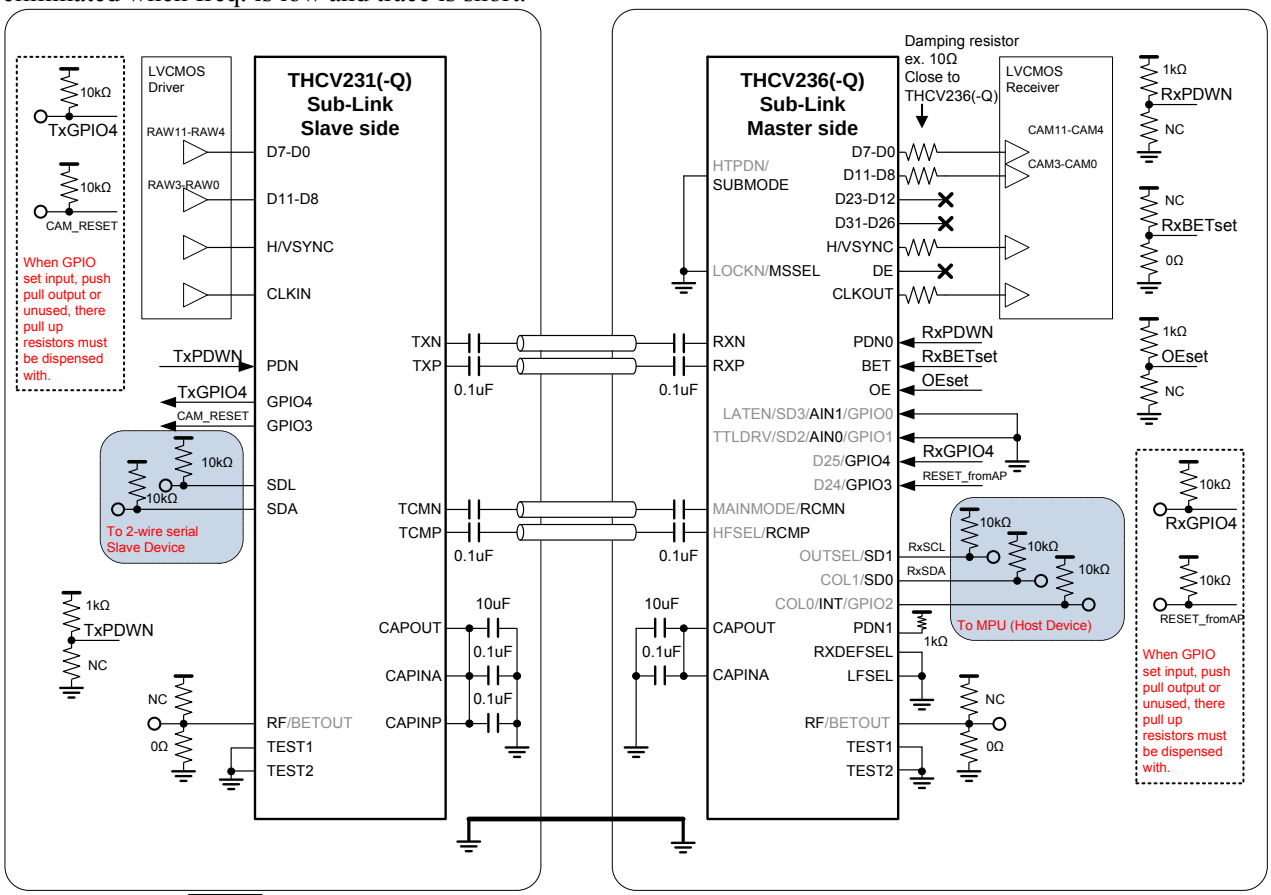
PDN can be controlled by external resistor or other driving source like MCU.

Unused LVCMOS inputs pins should be connected to **Low** (GND).

**[THCV236(-Q)]**

Set RXDEFSEL pin **Low**. Default setting is Sync Free Mode for the THCV231(-Q). Sync Free Mode is suitable to transmitting data of the system without DE sync signal like camera. PDN0 and OE can be controlled by external resistor or other driving source like MCU.

Unused LVCMOS output pins should be left **Open**. Place 10Ω resistor close to outputs, which can be eliminated when freq. is low and trace is short.




*1 indicates microstrip lines or cables with their differential characteristic impedance being 100 Ω

*2 Connect GNDs of both Tx and Rx PCB

*3 Field BET Operation. Please see the datasheet for details. (THCV231(-Q)_THCV236(-Q)_Rev.1.00_E.pdf and up)

**YCbCr8bit bus-width rising edge system**

GPIO4 and GPIO3 input to the THCV236(-Q) are output from the THCV231(-Q) by through GPIO function in order to send camera reset signal etc.

Set RF pin **High** (Rising edge LVCMOS input).

BET, BETOUT and LATEN pins can be used as Bit Error Test in actual configuration in order for debug or test purpose.

Note: This case of transmit 8bit or below, CML bit rate can be slowed if set COL1 (Register Name: refer to page 7 and page 9) to 1 and set COL0 to 0.

**[THCV231(-Q)]**

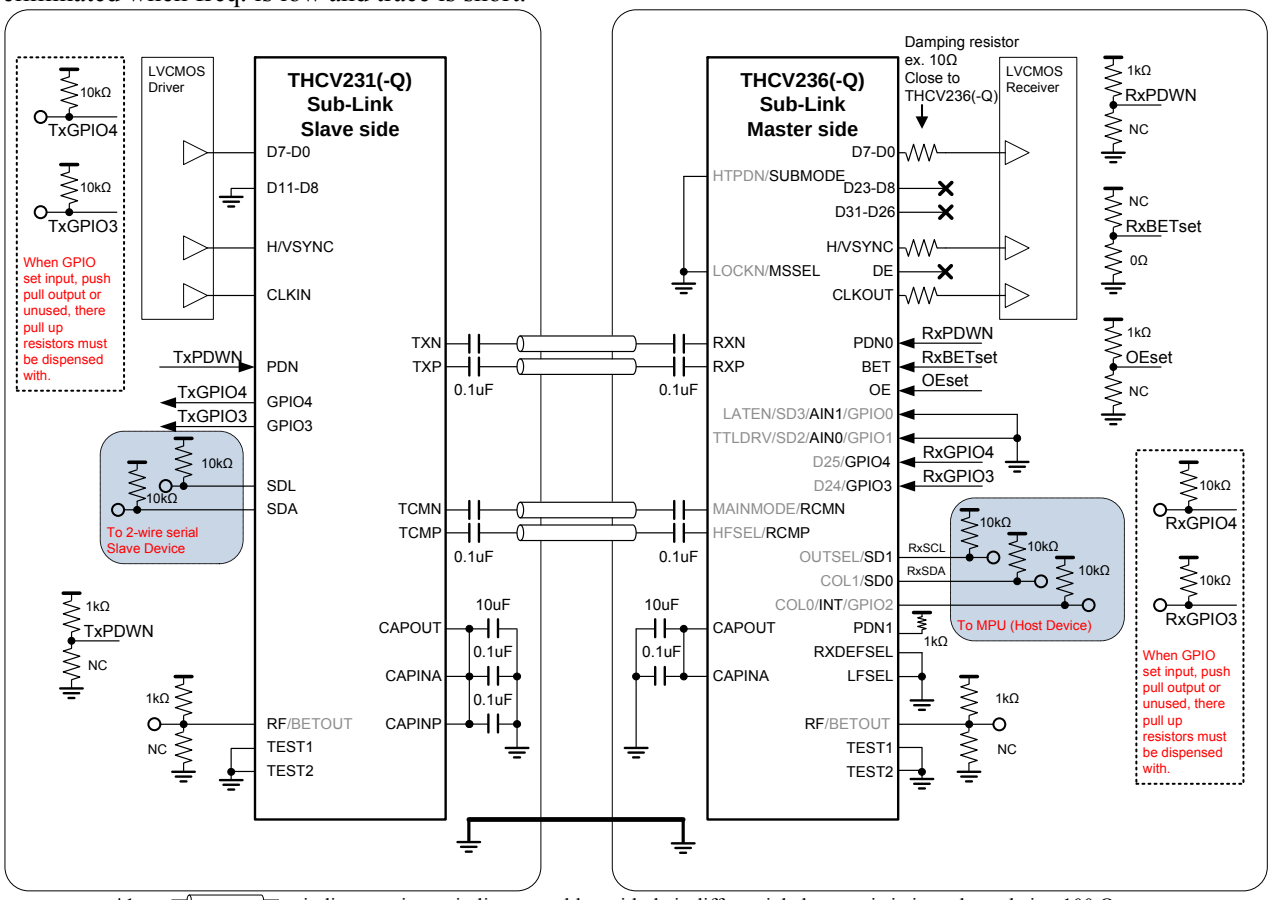
PDN can be controlled by external resistor or other driving source like MCU.

Unused LVCMOS inputs pins should be connected to **Low** (GND).

**[THCV236(-Q)]**

Set RXDEFSEL pin **Low**. Default setting is Sync Free Mode for the THCV231(-Q). Sync Free Mode is also available to transmitting data of embedded sync system without a separate sync signal (ITU656 etc.). PDN0 and OE can be controlled by external resistor or other driving source like MCU.

Unused LVCMOS output pins should be left **Open**. Place 10Ω resistor close to outputs, which can be eliminated when freq. is low and trace is short.




*1 indicates microstrip lines or cables with their differential characteristic impedance being 100 Ω

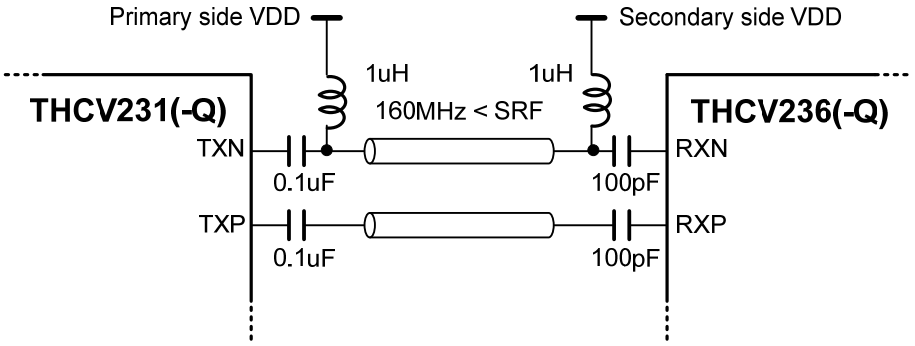
*2 Connect GNDs of both Tx and Rx PCB

*3 Field BET Operation. Please see the datasheet for details. (THCV231(-Q)_THCV236(-Q)_Rev.1.00_E.pdf and up)

## Signaling Usage

**Example Connection 1: Sharing Power Supply**

Power supply on common trace is also accomplished by another simple PCB circuit. Put 1uH inductor on both side. Inductor SRF (Self-resonant frequency) should be more than 160MHz. Inductor supply current tolerance must be more than requirement (ex. 500mA).
The THCV236(-Q) side AC coupling capacitors must be 100pF.



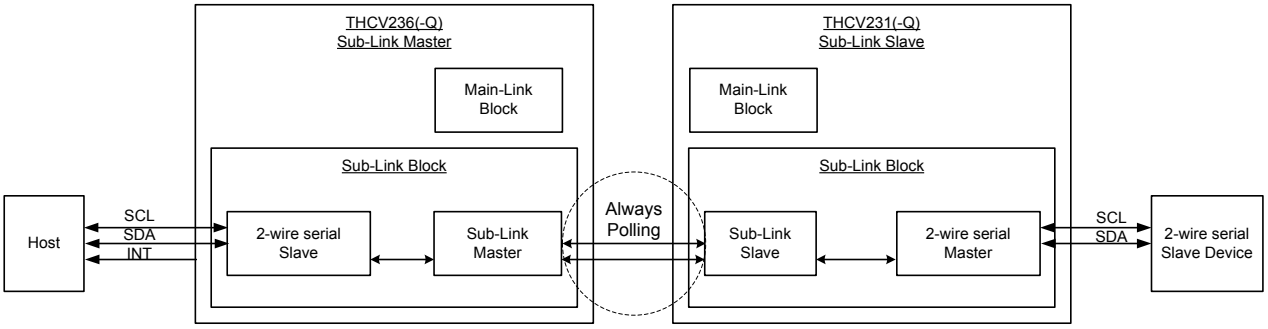
## Sub-Link Communication

Sub-Link Notification

All of Sub-Link specifications are on the premise that Sub-Link transmission is established because Sub-Link continues the polling operation even if there are no data accesses.
Please keep below conditions.

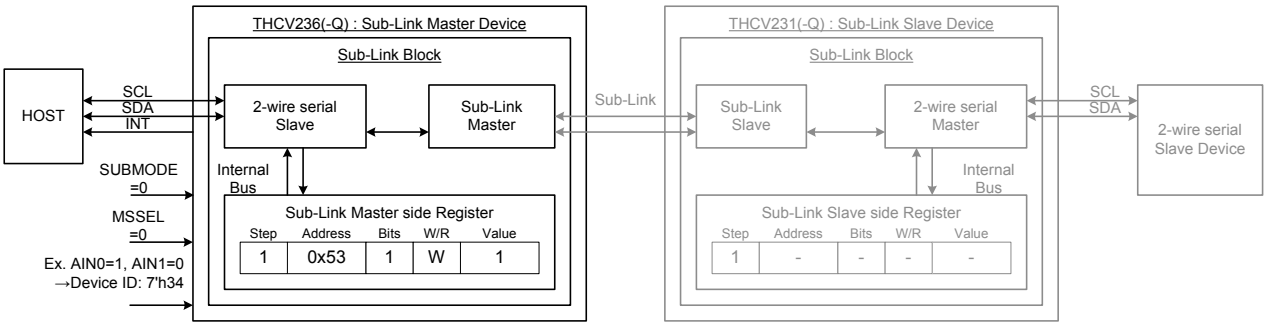
1. Sub-Link I/O is connected between the THCV231(-Q) and THCV236(-Q).
2. The THCV231(-Q) is powered on and PDN=1, the THCV236(-Q) is powered on and PDN1=1.

In order to incoming data transmit with the THCV231(-Q) and THCV236(-Q), Sub-Link must be continued communication.
The THCV231(-Q) is set Sub-Link Slave side at all times, and the THCV236(-Q) is set Sub-Link Master side at all times.



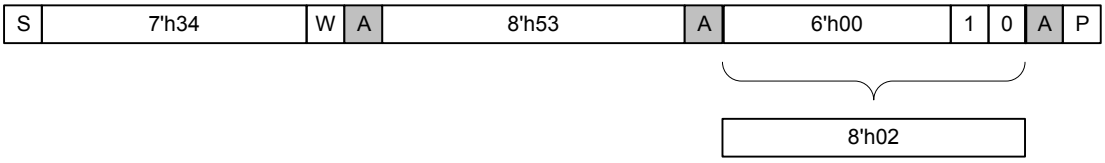
**Example Operation 1-1: Access to Sub-Link Master Device Register**

Writing BET to Sub-Link Master Device Register.



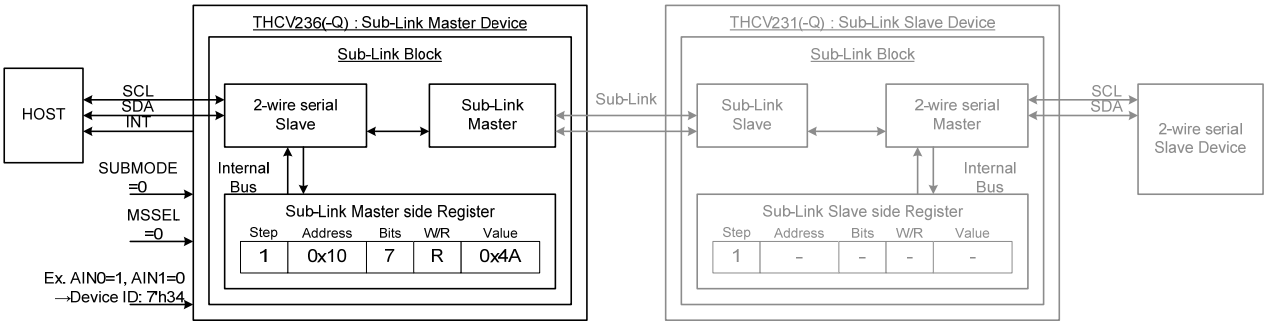

1. Write bit[1] of 0x53 in Sub-Link Master Device Register.
   0 = Normal Mode (Default)
   1 = Field BET Operation

Note: Sub-Link Master Register can set by 0x00-0x7F.




| | | | |
|---|---|---|---|
| S | Start condition | R | Read command indicator (1) |
| P | Stop condition | W | Write command indicator (0) |
| A | ACK | | Access from 2-wire serial interface Master |
| $\overline{A}$ | NACK | | Access from 2-wire serial interface Slave |
| Sr | Repeated Start condition | | |

Reading 2WIRE_DATA0 on Sub-Link Master Register.



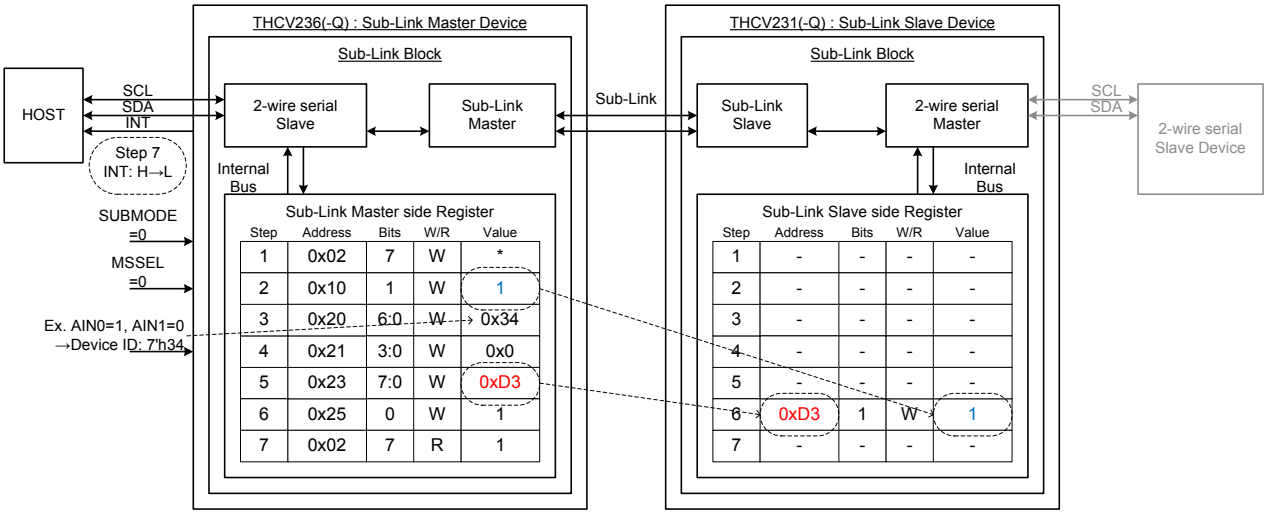

1. Read 0x10 of Sub-Link Master Device Register.
   Bit[7:0] 2-wire serial I/F Write/Read Data #0

   Note: Sub-Link Master Register can set by 0x00-0x7F.

| S | 7'h34 | W | A | 8'h10 | A | Sr | 7'h34 | R | A | 8'h4A | $\overline{A}$ | P |
|---|---|---|---|---|---|---|---|---|---|---|---|---|

**Example Operation 1-2: Access to Sub-Link Slave Device Register**

Writing BET to Sub-Link Slave Device.




1. Write any value to bit[7] of 0x02 for clear access status register. This bit is into 0 after any write action.

| S | 7'h34 | W | A | 8'h02 | A | 1 | 7'h00 | A | P |
|---|---|---|---|---|---|---|---|---|---|

8'h80

2. Set 1 to bit[1] of 0x10 for BET change.
   0 = Normal Mode (Default)
   1 = Field BET Operation

| S | 7'h34 | W | A | 8'h10 | A | 6'h00 | 1 | 0 | A | P |
|---|---|---|---|---|---|---|---|---|---|---|

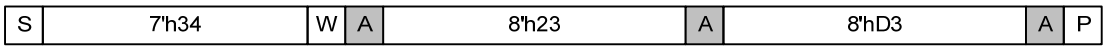
3. Set target Device ID (0x34) to bit[6:0] of 0x20.

| S | 7'h34 | W | A | 8'h20 | A | 0 | 7'h34 | A | P |
|---|---|---|---|---|---|---|---|---|---|

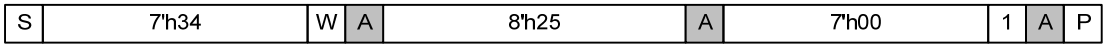
4. Set 0x00 to bit[3:0] of 0x21 for the amount of data byte intended to be sent.
   Note 1: The actual number of sent byte is register value +1.
   Note 2: The maximum data size is 16byte.

| S | 7'h34 | W | A | 8'h21 | A | 4'h0 | 4'h0 | A | P |
|---|---|---|---|---|---|---|---|---|---|

5. Set start address (0xD3) to bit[7:0] of 0x23.

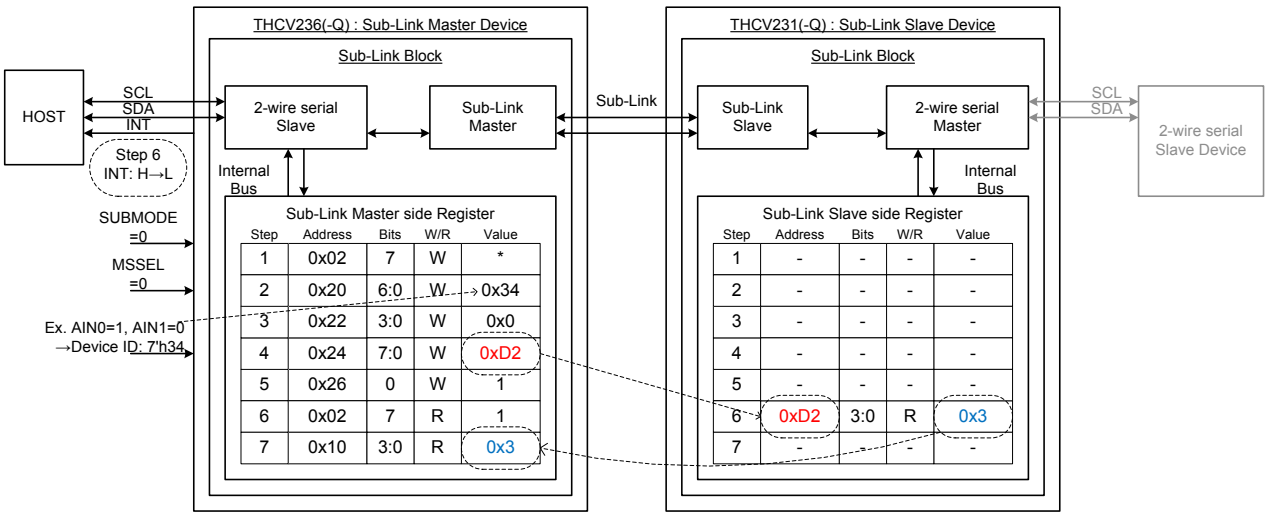



6. Write 1 to bit[0] of 0x25 to start write access to Sub-Link Slave register.




7. When write access is complete, bit[7] of 0x02 value becomes “1” and INT pin H→L. HOST MPU then receives interruption read access status register, confirming that write access is complete. If normally ended, the read result should be 0x80.

Reading FMOD on Sub-Link Slave Register.




1. Write any value to bit[7] of 0x02 for clear access status register. This bit is into 0 after any write action.

| S | 7'h34 | W | A | 8'h02 | A | 8'h80 | A | P |
|---|---|---|---|---|---|---|---|---|

2. Set target Device ID (0x34) in 0x20.

| S | 7'h34 | W | A | 8'h20 | A | 8'h34 | A | P |
|---|---|---|---|---|---|---|---|---|

3. Set “the number of reading register -1” (0x0) to bit[3:0] of 0x22.
   Note: The maximum data size is 16byte.

| S | 7'h34 | W | A | 8'h22 | A | 8'h00 | A | P |
|---|---|---|---|---|---|---|---|---|

4. Set start address (0xD2) in 0x24 for setting Slave Device Address.

| S | 7'h34 | W | A | 8'h24 | A | 8'hD2 | A | P |
|---|---|---|---|---|---|---|---|---|

5. Write 1 to bit[0] of 0x26, and start read access to Sub-Link Slave register.

| S | 7'h34 | W | A | 8'h26 | A | 8'h01 | A | P |
|---|---|---|---|---|---|---|---|---|

6. When read access is complete, read data is stored bit[7:0] in 0x10 (FMOD value is bit[3:0]), bit[7] of 0x02 value becomes “1” and INT pin H→L. If normally ended, the read result should be 0x80.

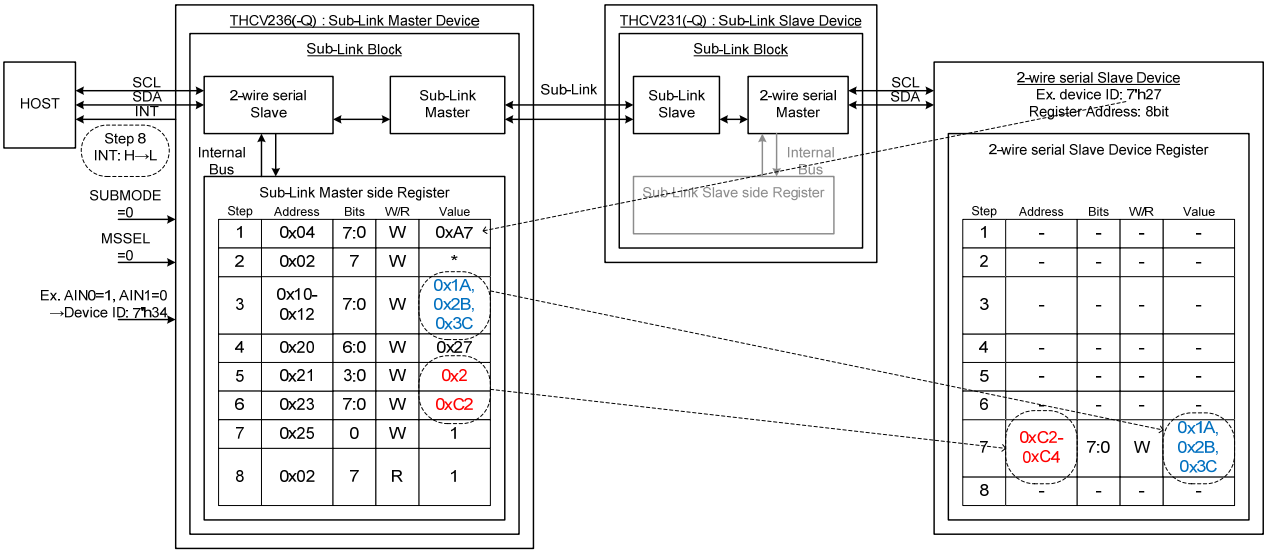
| S | 7'h34 | W | A | 8'h02 | A | Sr | 7'h34 | R | A | 8'h80 | $\overline{A}$ | P |
|---|---|---|---|---|---|---|---|---|---|---|---|---|

7. Read FMOD setting in 0x10.

| S | 7'h34 | W | A | 8'h10 | A | Sr | 7'h34 | R | A | 8'h03 | $\overline{A}$ | P |
|---|---|---|---|---|---|---|---|---|---|---|---|---|

## Example Operation 1-3: Access to 2-wire serial Slave Device with 8bit Register Address

Writing 3byte data to 2-wire serial Slave Device with 8bit Register Address.




1. Set 2-wire serial Slave Device ID (ex. 0x27) to bit[6:0] in 0x04 and set 1 to bit[7]. This step is not supposed to be repeated.
   Bit[6:0] Device ID
   Bit[7] 0=Disable / 1=Enable

| S | 7'h34 | W | A | 8'h04 | A | 8'hA7 | A | P |
|---|---|---|---|---|---|---|---|---|

Note: The maximum entry is 8devices, 0x04-0x0B.

2. Write any value to bit[7] of 0x02 for clear access status register. This bit is into 0 after any write action.

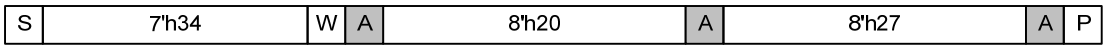
| S | 7'h34 | W | A | 8'h02 | A | 8'h80 | A | P |
|---|---|---|---|---|---|---|---|---|

3. Set 3byte data (ex. 0x1A, 0x2B, 0x3C) into 0x10-0x12 for writing into 2-wire serial Slave Device.

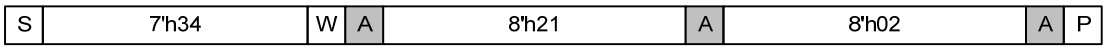
| S | 7'h34 | W | A | 8'h10 | A | 8'h1A | A |
|---|---|---|---|---|---|---|---|
| 8'h2B | A | 8'h3C | A | P | | | |

Note: The maximum data size is 16byte.

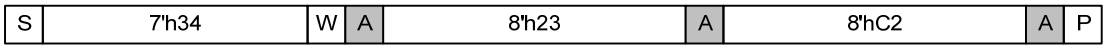
4. Set target 2-wire serial Slave Device ID (0x27) bit[6:0] of 0x20 and set 0 to bit[7].
   Bit[7] 0 = 8bit Register Address (Default)
   1 = 16bit Register Address
   Bit[6:0] Target Device ID



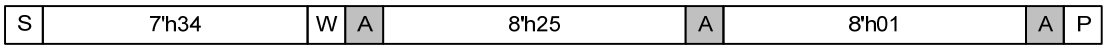

5. Set “the number of write byte -1” (0x02) in 0x21.




6. Set start address of 2-wire serial Slave Device (ex. 0xC2) in 0x23.



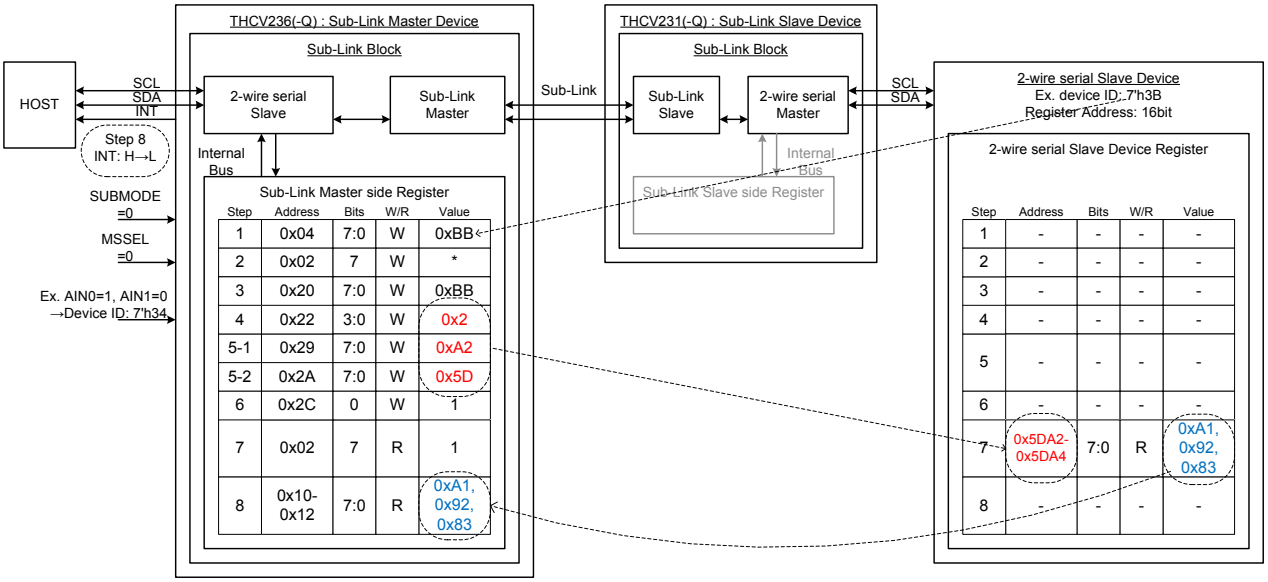

7. Write 1 to bit[0] of 0x25 to start write access to 2-wire serial Slave Device register.




8. When write access is complete, bit[7] of 0x02 value becomes “1” and INT pin H→L. HOST MPU then receives interruption read access status register, confirming that write access is complete. If normally ended, the read result should be 0x80.

**Example Operation 1-4: Access to 2-wire serial Slave Device with 16bit Register Address**

Reading 3byte data to 2-wire serial Slave Device with 16bit Register Address.




1. Set 2-wire serial Slave Device ID (ex. 0x3B) to bit[6:0] of 0x04 and set 1 to bit[7]. This step is not supposed to be repeated.
   Bit[7] 0=Disable (Default) / 1=Enable
   Bit[6:0] Device ID

| S | 7'h34 | W | A | 8'h04 | A | 8'hBB | A | P |
|---|---|---|---|---|---|---|---|---|

Note: The maximum entry is 8devices, 0x04-0x0B.

2. Write any value to bit[7] of 0x02 for clear access status register. This bit is into 0 after any write action.

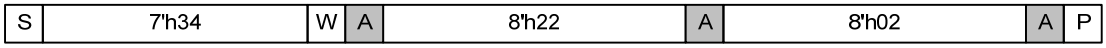
| S | 7'h34 | W | A | 8'h02 | A | 8'h80 | A | P |
|---|---|---|---|---|---|---|---|---|

3. Set target 2-wire serial Slave Device ID (0x3B) bit[6:0] of 0x20 for reading register access and set 1 to bit[7] for 16bit Register Address.
   Bit[7] 0 = 8bit Register Address (Default)
   1 = 16bit Register Address
   Bit[6:0] Target Device ID

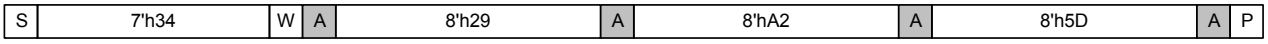
| S | 7'h34 | W | A | 8'h20 | A | 8'hBB | A | P |
|---|---|---|---|---|---|---|---|---|

4. Set “the number of read byte -1” (0x02) in 0x22.

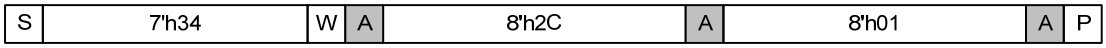
Note: The maximum data size is 16byte.



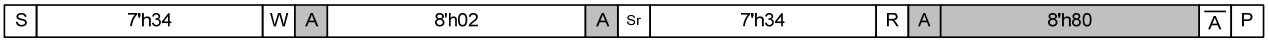

5. Set start address of 2-wire serial Slave Device ID (ex. 0x5DA2) in 0x29 and 0x2A.
   0x29: start address [7:0]
   0x2A: start address [15:8]



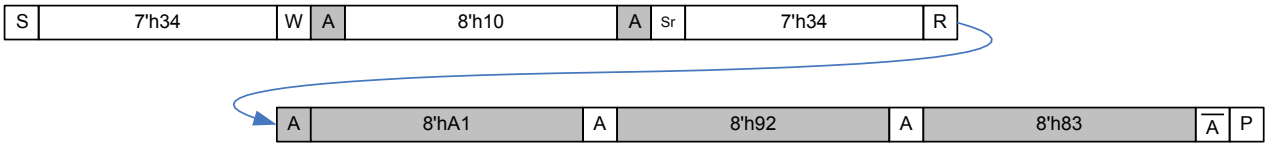

6. Write 1 to bit[0] of 0x2C to start read access to 2-wire serial Slave Device register.




7. When read access is complete, reading data is stored into 0x10-0x12, bit[7] of 0x02 becomes “1”, and INT pin H→L. If normally ended, the read result should be 0x80.

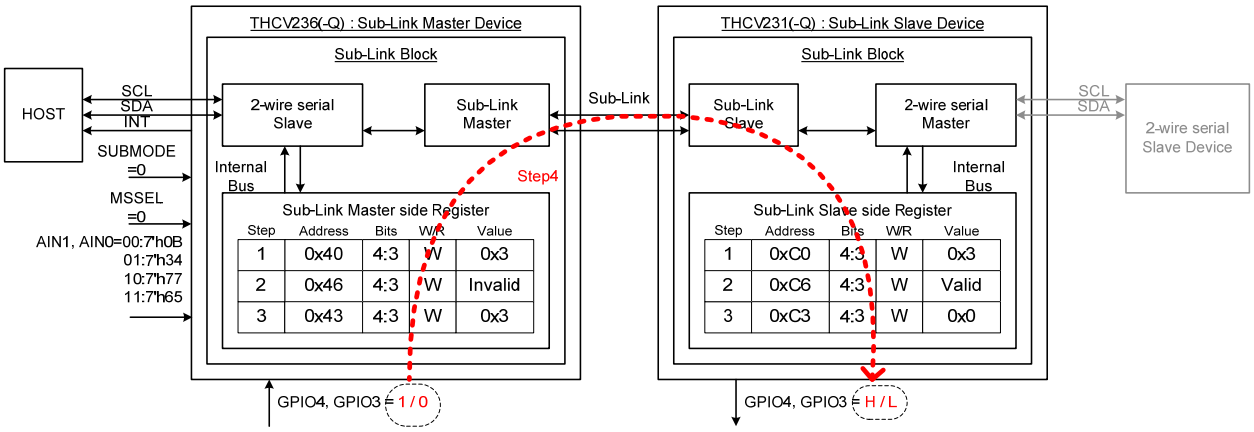



8. Read 3 byte data in 0x10-0x12.



**Example Operation 2-1: Through GPIO Sub-Link Master to Sub-Link Slave**

Control through GPIO of the THCV236(-Q) (Sub-Link Master) and through GPIO of the THCV231(-Q) (Sub-Link Slave Device).




1. GPIO transmission type setting with bit[4:3] of 0x40 (Master side Register) and bit[4:3] of 0xC0 (Slave side Register, refer to page 9).
   0 = Programmable GPIO enable
   1 = Through GPIO enable (Default)

2. GPIO output buffer type setting with bit[4:3] of 0xC6 (Slave side Register).
   1=Push-pull output
   0=Open-Drain output (Default)

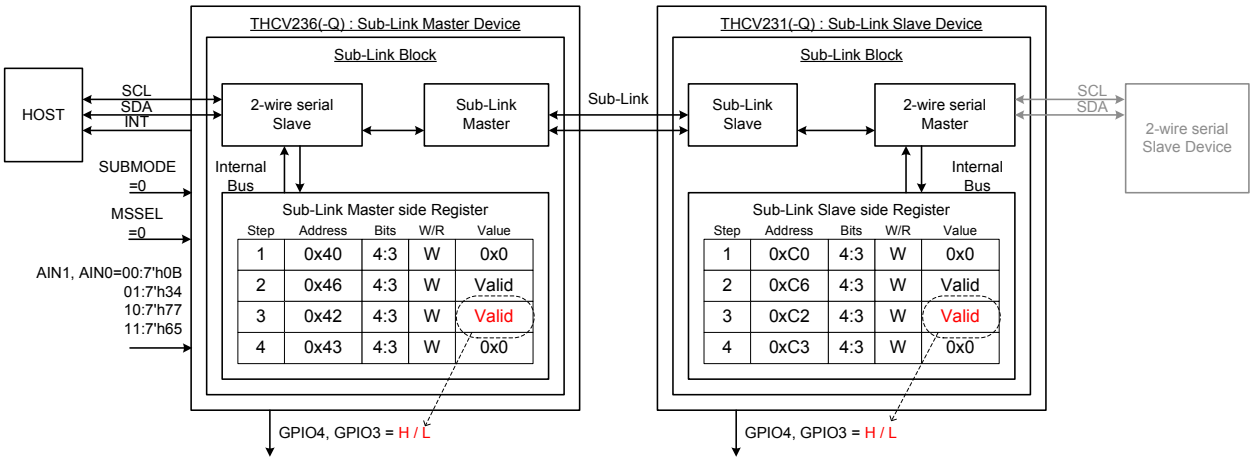
3. GPIO polarity setting with bit[4:3] of 0x43 (Master side Register) and bit[4:3] of 0xC3 (Slave side Register).
   1=Input
   0=Output

4. GPIO4 and GPIO3 input of Master side is updated automatically to GPIO4 and GPIO3 output of Slave side.

Note: This through GPIO function is able to use with default register setting.

**Example Operation 2-2: Programmable GPIO output control**

Control output GPIO of the THCV236(-Q) (Sub-Link Master side) and GPIO of the THCV231(-Q) (Sub-Link Slave side).




1. GPIO transmission type setting with bit[4:3] of 0x40 (Master side Register) and bit[4:3] of 0xC0 (Slave side Register, refer to page9).
   0 = Programmable GPIO enable
   1 = Through GPIO enable (Default)

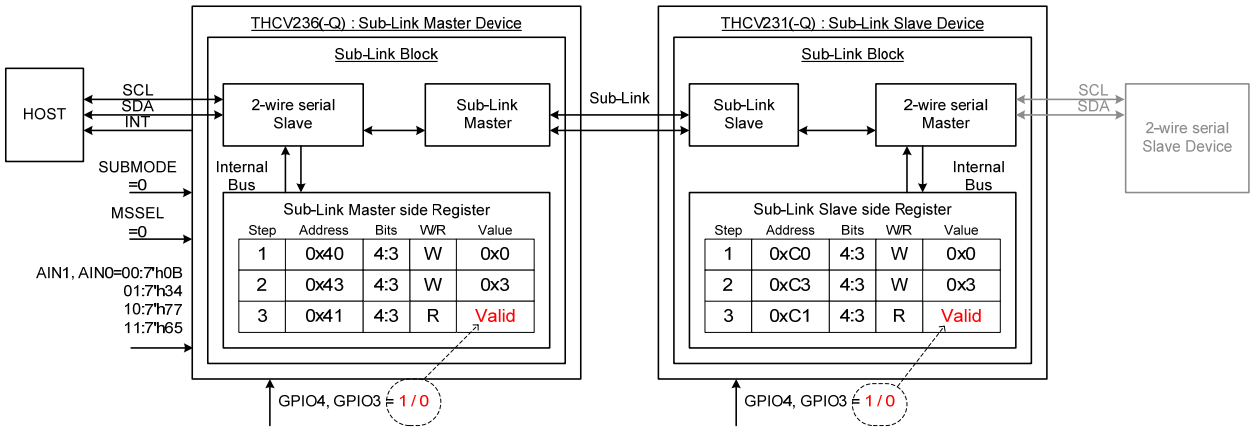
2. GPIO output buffer type setting with bit[4:3] of 0x46 (Master side Register) and bit[4:3] of 0xC6 (Slave side Register).
   1=Push-pull output
   0=Open-Drain output (Default)

3. GPIO output value setting with bit[4:3] of 0x42 (Master side Register) and bit[4:3] of 0xC2 (Slave side Register).
   1=H (Push-pull output: H / Open-Drain output: Hi-Z)
   0=L (Default)

4. GPIO polarity setting with bit[4:3] of 0x43 (Master side Register) and bit[4:3] of 0xC3 (Slave side Register).
   1=Input
   0=Output

**Example Operation 2-3: Programmable GPIO input control**

Control input GPIO of the THCV236(-Q) (Sub-Link Master side) and GPIO of the THCV231(-Q) (Sub-Link Slave side).



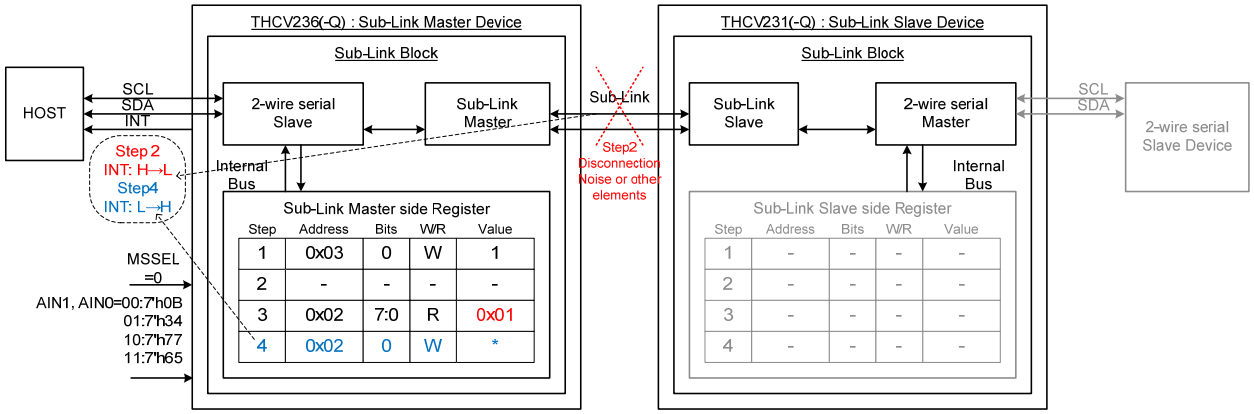

1. GPIO transmission type setting with bit[4:3] of 0x40 (Master side Register) and bit[4:3] of 0xC0 (Slave side Register, refer to page9).
   0 = Programmable GPIO enable
   1 = Through GPIO enable (Default)

2. GPIO polarity setting with bit[4:3] of 0x43 (Master side Register) and bit[4:3] of 0xC3 (Slave side Register).
   1=Input
   0=Output

3. Read bit[4:3] of 0x41 (Master side Register) and bit[4:3] of 0xC1 (Slave side Register, refer to page 11).

**Example Operation 3-1: Interrupt feedback as internal factor**

Cause of interrupt is internal factor (ex. Sub-Link time out)




1. Set interrupt permission status about Sub-Link time out with bit[0] of 0x03 (Master side Register).
   1=Interrupt allowed
   0=Interrupt blocked (Default)

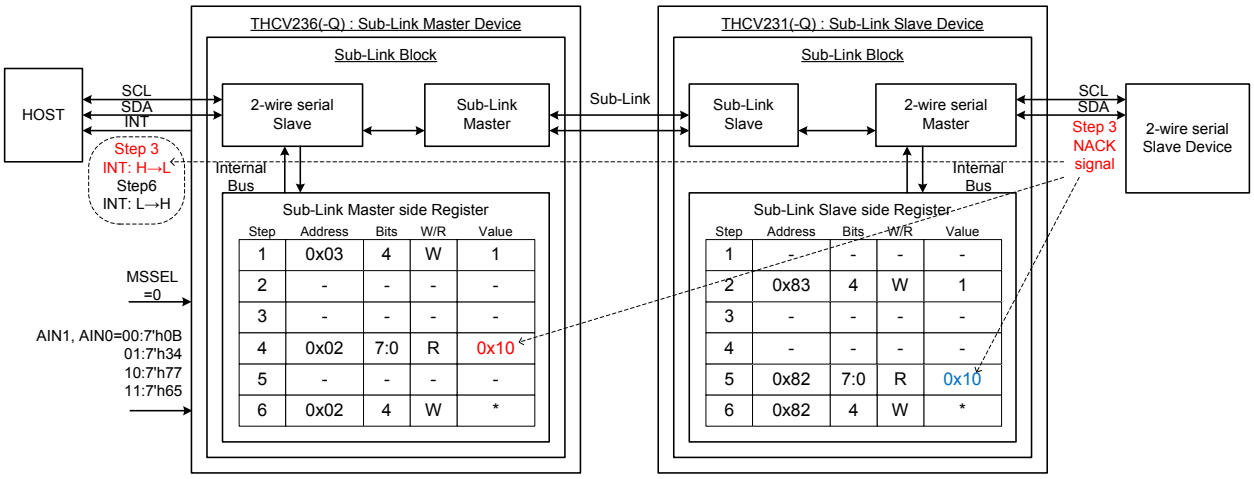
2. When disconnection occurs or Sub-Link access time out is occurred by Noise or other elements, an interruption occurs. (INT=H → L)

3. Read the cause of interruption register with bit[7:0] in 0x02 (Master side Register).

4. Write any value to the cause of interruption register on bit[0] in 0x02 (Master side Register).
   The cause of interruption register is cleared.
   Interruption released. (INT=L → H)

**Example Operation 3-2: Interrupt output as external factor by Slave side**

Cause of interrupt is external factor (ex. NACK signal from 2-wire serial Slave Device).




1. Set interrupt permission status about Slave side interrupt action with bit[4] of 0x03 (Master side Register).
   1=Interrupt allowed
   0=Interrupt blocked (Default)

2. Set interrupt permission status about 2-wire serial Slave Device NACK action with bit[4] of 0x83 (Slave side Register, refer to page9).
   1=Allowed to be reported to Master side
   0=Blocked to be reported to Master side (Default)

3. Slave side device receive NACK signal from 2-wire serial Slave Device. An interruption occurs. (INT=H → L)

4. Read the cause of interruption register (SLAVESIDE_INT) with bit[4] of 0x02 (Master side Register).

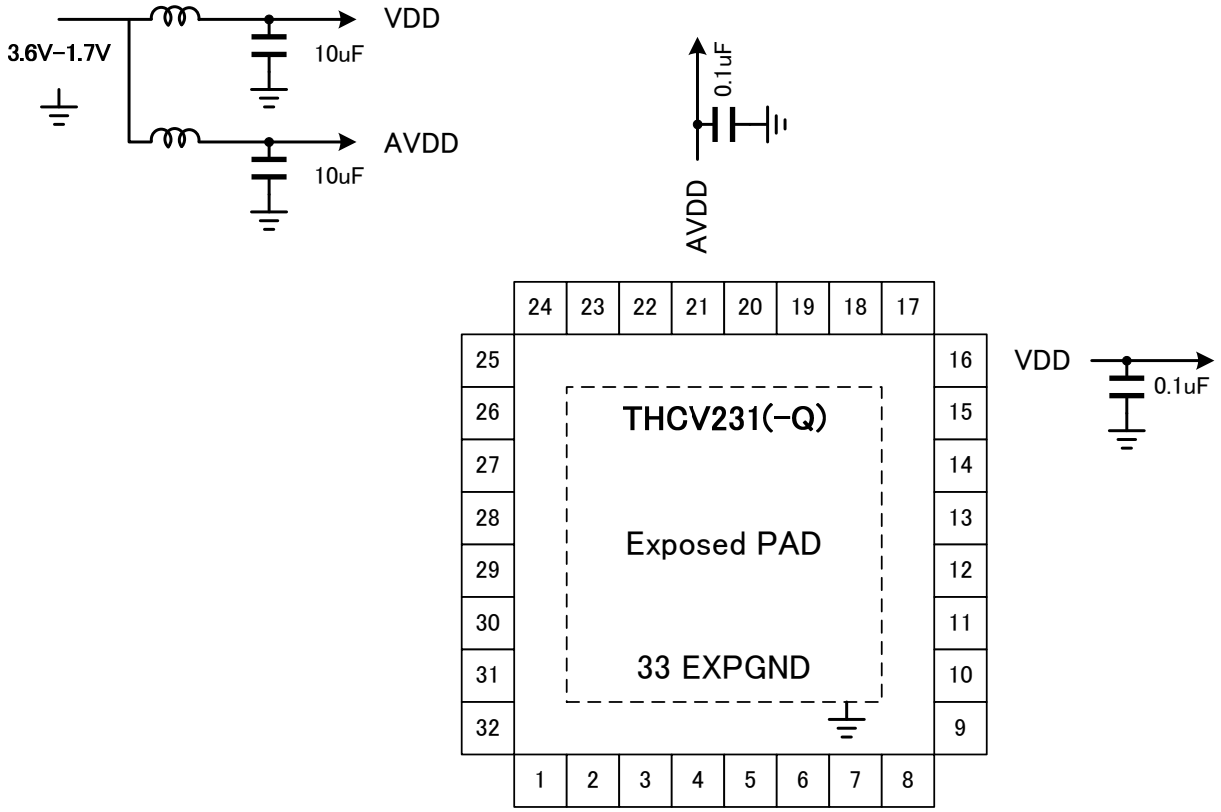
5. Read the cause of interruption register (2WIRE_NACK_INT) with bit[4] of 0x82 (Slave side Register, refer to page 11).

6. Write any value to a cause of interruption register with bit4 of 0x02 (Master side Register) and bit[4] of 0x82 (Slave side register).
   Cause of interruption register is cleared.
   Interruption released. (INT=L → H)

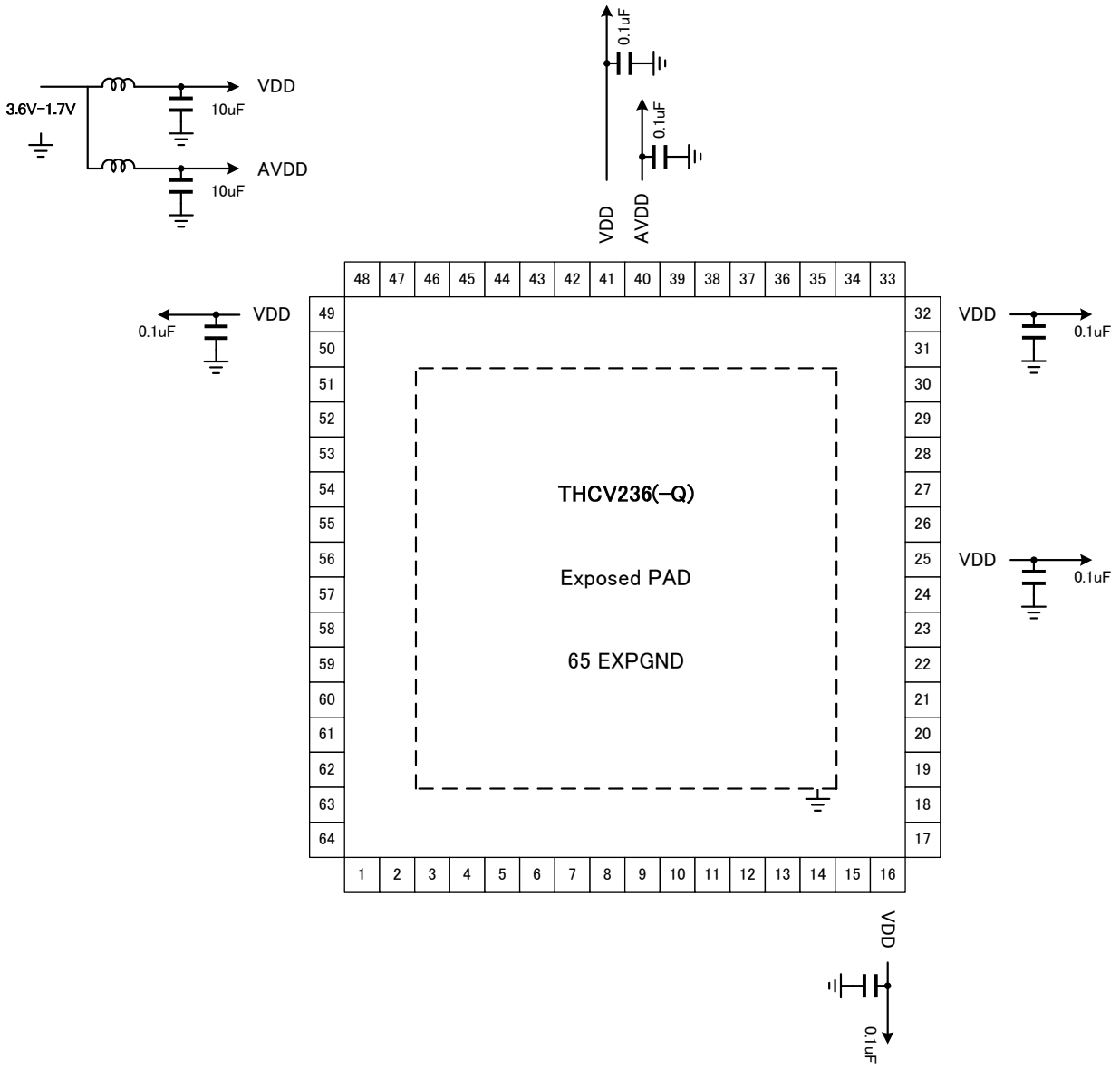
## Power Supply Usage

- Separate all power domains in order to avoid unwanted noise coupling between noisy digital and sensitive analog domains. In particular, PCB layout guide between AVDD (40 Pin) and VDD (41 Pin) are shown page 23 for the THCV236(-Q).
- Use 0.1uF high frequency ceramic capacitors as bypass capacitors between power and ground pins. Place the capacitors as close to each power pin as possible.

**Power Supply for THCV231(-Q)**



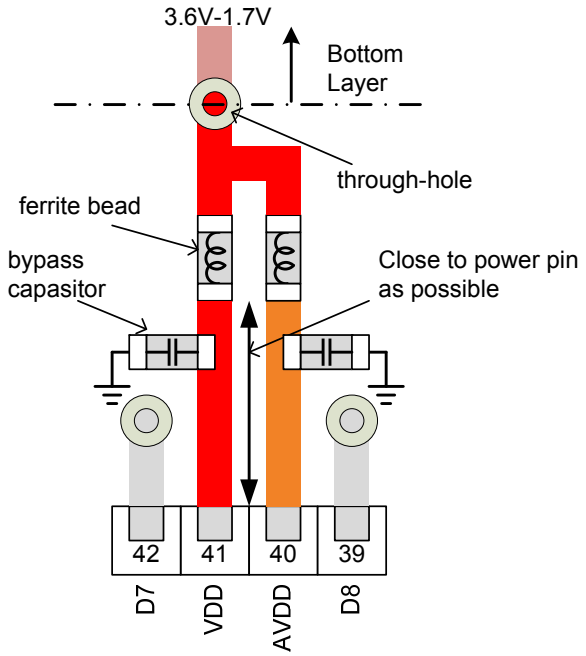
**Power Supply for THCV236(-Q)**



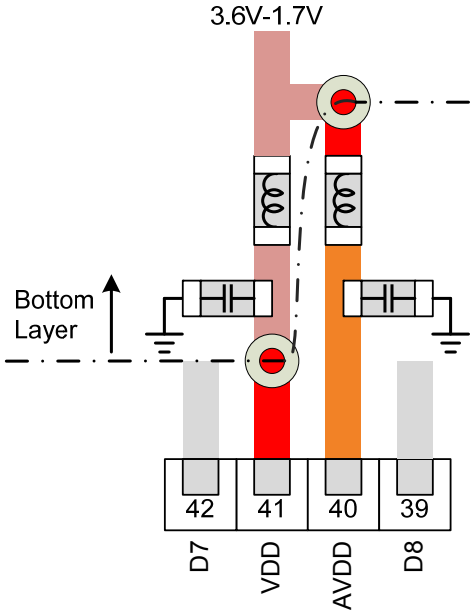
**PCB layout guideline between VDD (41Pin) and AVDD (40Pin) for THCV236(-Q)**

When power is supplied from reverse side layer to AVDD, please place ferrite bead behind through-hole (Good Example1, 2). If it is needed to set ferrite beads on reverse side, please set GND-through-hole between AVDD and VDD, and separate the distance as possible (Example). Do not set through-holes next to each other behind ferrite beads (Bad Example).

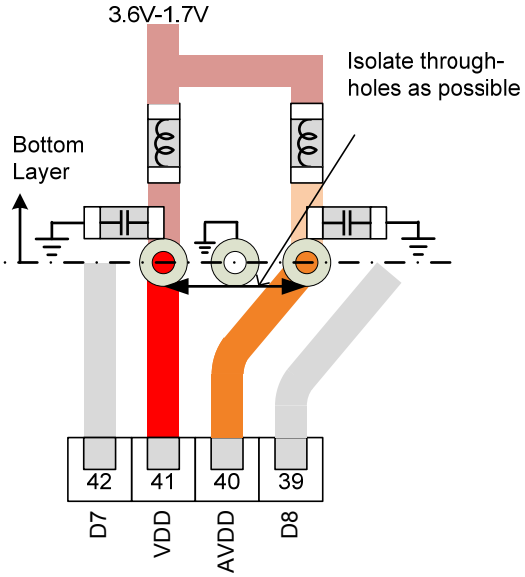
**Good Example 1**



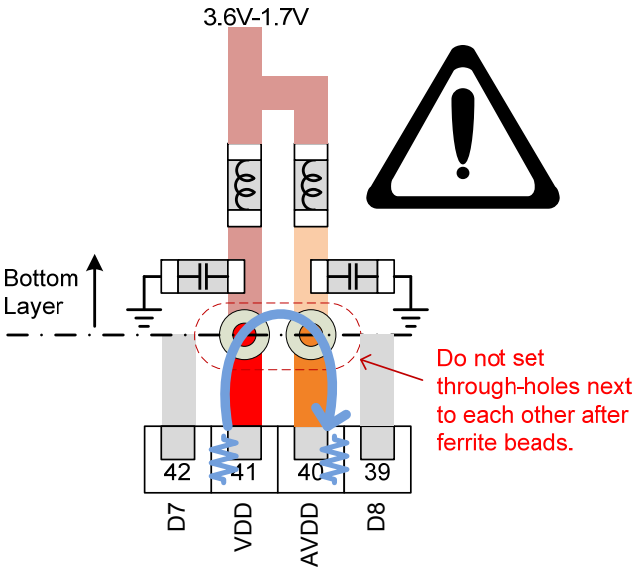

**Good Example 2**




**Example**




**Bad Example**



Note

**1) Power On Sequence**

Do not input clock or data before the THCV231(-Q) on in order to keep absolute maximum ratings.

**2) Cable Connection and Disconnection**

Do not connect and disconnect the LVCMOS and CML cable/connector, when the power is supplied to the system.

**3) GND Connection**

Connect the each GND of the PCB where the THCV231(-Q) and THCV236(-Q) are on it. It is better for EMI reduction to place GND cable as close to LVCMOS and CML cable as possible.

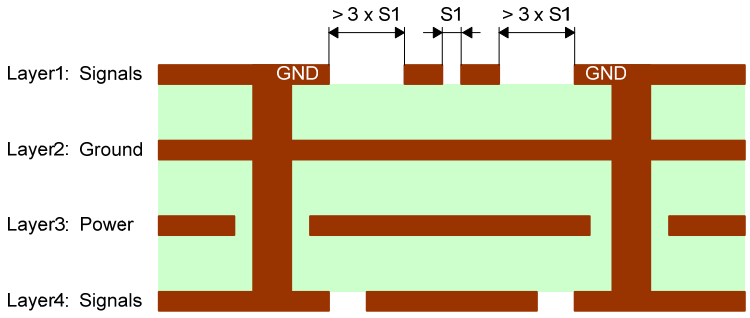
**4) Low Input Pulse into PDN, PDN1 and PDN0 Period Requirement**

Do not input Low Pulse within 1msec into PDN (THCV231(-Q)), PDN1 and PDN0 (THCV236(-Q)).

## PCB Layout Considerations

- Use at least four-layer PCBs with signals, ground, power, and signals assigned for each layer. (Refer to figure below.)
- PCB traces for high-speed signals must be single-ended microstirp lines or coupled microstrip lines whose differential characteristic impedance is 100Ω.
- Minimize the distance between traces of a differential pair (S1) to maximize common mode rejection and coupling effect which works to reduce EMI (Electro-Magnetic Interference).
- Route differential signal traces symmetrically.
- Avoid right-angle turns or minimize the number of vias on the high speed traces because they usually cause impedance discontinuity in the transmission lines and degrade the signal integrity.
- Mismatch among impedances of PCB traces, connectors, or cables also caused reflection, limiting the bandwidth of the high-speed channels.
- Using common-mode filter on differential traces is desirable to reduce EMI. Pay attention on data-rate driven noise. For example, if data-rate is 1.5Gbps, common mode choke coil of 1.5GHz common mode impedance is desired to be high, while 1.5GHz differential impedance is low.

Differential signal traces (Microstrip Lines)



Notices and Requests

1. The product specifications described in this material are subject to change without prior notice.
2. The circuit diagrams described in this material are examples of the application which may not always apply to the customer's design. We are not responsible for possible errors and omissions in this material. Please note if errors or omissions should be found in this material, we may not be able to correct them immediately.
3. This material contains our copyright, know-how or other proprietary. Copying or disclosing to third parties the contents of this material without our prior permission is prohibited.
4. Note that if infringement of any third party's industrial ownership should occur by using this product, we will be exempted from the responsibility unless it directly relates to the production process or functions of the product.
5. Product Application

5.1. Application of this product is intended for and limited to the following applications: audio-video device, office automation device, communication device, consumer electronics, smartphone, feature phone, and amusement machine device. This product must not be used for applications that require extremely high-reliability/safety such as aerospace device, traffic device, transportation device, nuclear power control device, combustion chamber device, medical device related to critical care, or any kind of safety device.

5.2. This product is not intended to be used as an automotive part, unless the product is specified as a product conforming to the demands and specifications of ISO/TS16949 ("the Specified Product") in this data sheet. THine Electronics, Inc. (“THine”) accepts no liability whatsoever for any product other than the Specified Product for it not conforming to the aforementioned demands and specifications.

5.3. THine accepts liability for demands and specifications of the Specified Product only to the extent that the user and THine have been previously and explicitly agreed to each other.

6. Despite our utmost efforts to improve the quality and reliability of the product, faults will occur with a certain small probability, which is inevitable to a semi-conductor product. Therefore, you are encouraged to have sufficiently redundant or error preventive design applied to the use of the product so as not to have our product cause any social or public damage.
7. Please note that this product is not designed to be radiation-proof.
8. Testing and other quality control techniques are used to this product to the extent THine deems necessary to support warranty for performance of this product. Except where mandated by applicable law or deemed necessary by THine based on the user’s request, testing of all functions and performance of the product is not necessarily performed.
9. Customers are asked, if required, to judge by themselves if this product falls under the category of strategic goods under the Foreign Exchange and Foreign Trade Control Law.
10. The product or peripheral parts may be damaged by a surge in voltage over the absolute maximum ratings or malfunction, if pins of the product are shorted by such as foreign substance. The damages may cause a smoking and ignition. Therefore, you are encouraged to implement safety measures by adding protection devices, such as fuses.

***THine Electronics, Inc.***

sales@thine.co.jp