I2C and I3C Serial Interface Temperature Sensor Compatible with JEDEC JESD302-1A for DDR5

1 Features

- Exceeds JEDEC JESD302-1 temperature accuracy specification
- 11-bit resolution: 0.25°C (1 LSB)
- Low power consumption
- High sampling frequency (max 125ms)
- Two wire bus serial interface (I^2C and I3C basic operation modes)
- Up to 12.5 MHz transfer rate
- 1.8 V and1.0 V power supply input
- Packet error check function
- Parity error check function
- Bus reset function
- Two unique addresses selected by SA pin
- In band interrupt (IBI)
- 6 ball WLCSP package device functional diagram
- Operating Temperature Range: -40°C to 125°C

2 Applications

- DDR5 DIMM module
- Solid state drive (SSD)
- Server, Mobile device, Motherboards

3 Description

The GD30TP139A devices incorporate high accuracy thermal sensing capability which is controlled and read over two-wire bus. The GD30TP139A device operates from a 1.8 V nominal power supply (VDDSPD) and a 1.0 V nominal power supply (VDDIO). The GD30TP139A device is compatible with JEDEC JESD302-1A standard and is intended to operate up to 12.5 MHz on a 1.0V I3C Basic bus or up to 1 MHz on a 1.0 V to 3.3 V I2C bus. The GD30TP139A device is intended to interface to I2C or I3C Basic bus which has multiple devices on a shared bus, and must be uniquely addressed with fixed addressing on the same bus. All GD30TP139A devices respond to specific pre-defined device select codes on the two-wire bus.

The GD30TP139A is an important component of the DDR5 server DIMMs. The current mainstream DIMMs are expected to be equipped with two GD30TP139A chips per DIMM.

Device Information[1]

PART NUMBER	PACKAGE	BODY SIZE (NOM)
GD30TP139ASHTR-I	WLCSP	1.328 mm x 0.828 mm.

1. For packaging details, see *Package Information* section.

Simplified Application Schematic

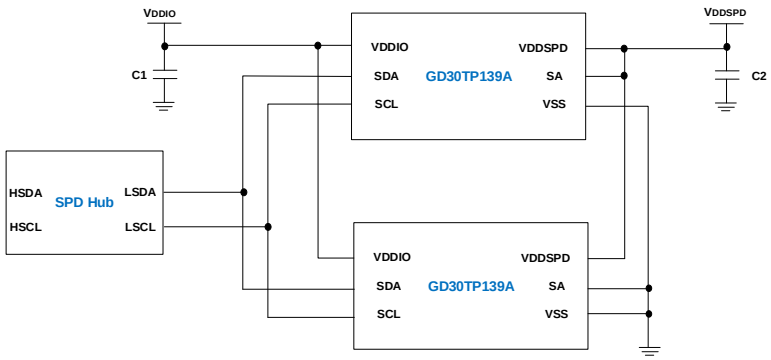
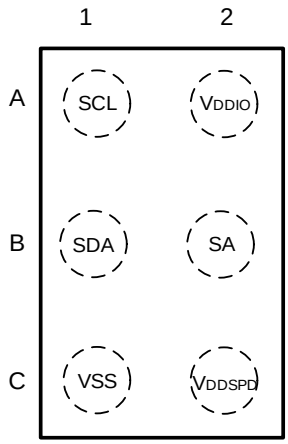


Table of Contents

4 Device Overview

4.1 Pinout and Pin Assignment



4.2 Pin Description

PIN NAME	PIN #	PIN TYPE[1]	FUNCTION
SCL	A1	I	I²C or I3C Basic Input Clock.
SDA	B1	IO	I²C or I3C Basic Data.
VSS	C1	G	Ground pin.
V_{DDIO}	A2	P	1.0 V Input Power Supply. Connect minimum of 0.1μF capacitor to VSS.
SA	B2	I	I²C or I3C Basic Address Pin. This pin is tied to either VSS or V_{DDSPD} to establish the 4-bit LID.
V_{DDSPD}	C2	P	1.8 V Input Power Supply. Connect minimum of 0.1μF capacitor to VSS.

1. P = power, G = Ground, I = input, IO=input and output.

5 Parameter Information

5.1 Absolute Maximum Ratings

Exceeding the operating temperature range (unless otherwise noted)[1]

SYMBOL	PARAMETER	MIN	MAX	UNIT
T_{STG}	Storage temperature	-65	150	°C
V_{DDIO}	Input Supply voltage	-0.5	2.1	V
V_{DDSPD}	Input Supply voltage	-0.5	2.1	V
SA	SA Pin	-0.5	2.1	V
SCL, SDA	SCL, SDA Pins	-0.5	3.6	V
T_J	Junction temperature	–55	150	°C

1. The maximum ratings are the limits to which the device can be subjected without permanently damaging the device. Note that the device is not guaranteed to operate properly at the maximum ratings. Exposure to the absolute maximum rating conditions for extended periods may affect device reliability.

5.2 Recommended Operation Conditions

SYMBOL	PARAMETER	MIN	TYP	MAX	UNIT
V_{DDSPD}	Input Supply Voltage[1]	1.7	1.8	1.98	V
V_{DDIO}	Input Supply Voltage[2]	0.95	1.0	1.05	V
T_A	Operating free-air temperature	-40		125	°C

1 For DDR5 DIMM application, the DDR5 PMIC VOUT_1.8V setting should be selected such that absolute Min and Max values for SPD Hub specification are not violated.

2 For DDR5 DIMM application, the DDR5 PMIC VOUT_1.0V setting should be selected such that absolute Min and Max values for SPD Hub specification are not violated.

5.3 Electrical Sensitivity

SYMBOL	CONDITIONS	VALUE	UNIT
$V_{ESD(HBM)}$	Human-body model (HBM), ANSI/ESDA/JEDEC JS-001-2017[1]	±2000	V
$V_{ESD(CDM)}$	Charged device model (CDM), per JEDEC specification JESD22-C101[2]	±1000	V

1. JEDEC document JEP155 states that 500-V HBM allows safe manufacturing with a standard ESD control process.
2. JEDEC document JEP157 states that 250-V CDM allows safe manufacturing with a standard ESD control process.

5.4 AC Measurement Conditions

SYMBOL	PARAMETER	MIN	MAX	UNIT
C_L	Load capacitance	40		pF
	Input rise and fall times - Open Drain	-	TBD	ns
	Input rise and fall times - Push Pull	-	TBD	ns
	Input signal swing levels	0.2	0.8	V
	Input levels for timing reference	0.3	0.7	V

1 This AC measurement condition(Figure 11) is only for the test purpose in lab.

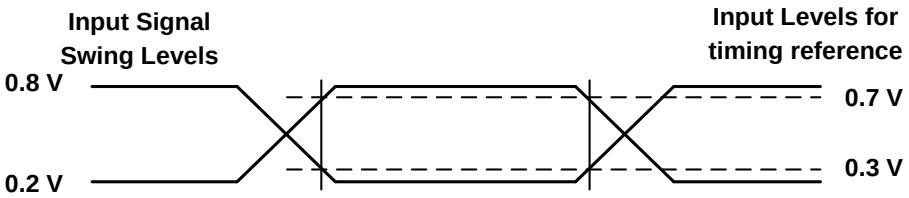


Figure 1 AC Measurement Waveform

Table 1 Input Parameters

SYMBOL	PARAMETER[1,2]	TEST CONDITION	MIN	MAX.	UNIT
C_{IN}	Input capacitance (SDA, SCL)		-	4	pF
t_{SP}	Pulse width of spikes which must be suppressed by the input filter in I^2C mode	Single glitch, f ≤ 100 KHz	-	-	ns
		Single glitch, f > 100 KHz	0	50	ns

1 T_A = 25 °C, f = 400 kHz

2 Verified by design and characterization, not necessarily tested on all devices.

Table 2 Output Ron Specification

SYMBOL	PARAMETER[1]	MIN	MAX	UNIT
R_{on}	SDA Output Pullup and Pulldown Driver Impedance	40	100	Ohm

1 Pulldown $R_{on} = V_{out} / I_{out}$; Pullup $R_{on} = (V_{DDIO} - V_{out}) / I_{out}$

5.5 DC Characteristics

SYMBOL	PARAMETER	TEST CONDITION	MIN	MAX	UNIT
I_{LI}	Input Leakage Current		-	± 5	μA
I_{LO}	Output Leakage Current		-	± 5	μA
I_{DDR}	Supply Current, Read Operation[1]	V_{DDSPD} = 1.8V, f_C = 12.5 MHz	-	2	mA
I_{DDW}	Supply Current, Write Operation[1]	V_{DDSPD} = 1.8V, f_C = 12.5 MHz	-	3	mA
I_{DD1}	Standby Supply Current	$V_{IN} = V_{DDSPD}$ = 1.8V	-	TBD	μA
V_{IL}	Input Low Voltage		-0.35	0.3	V
V_{IH}	Input High Voltage		0.7	3.6	V
V_{OL}	Output Low Voltage	3 mA sink current	-	0.3	V
V_{OH}	Output High Voltage	3 mA source current	0.75	-	V
I_{OL}	Output Low Current (SDA)	V_{OL} = 0.3V	-3	-	mA
I_{OH}	Output High Current (SDA)	$V_{OH} = V_{DDIO}$ - 0.3V	-	-3	mA
Slew_Rate	Rising Output Slew Rate (SDA)[2]		0.1	1.0	V/ns
	Falling Output Slew Rate (SDA)[2]		0.1	1.0	V/ns
V_{PON}	Power On Reset Threshold	Monotonic rise between V_{PON} and $V_{DDSPD(min)}$ without ringback	1.6	-	V
V_{POFF}	Power Off Threshold for Warm Power On Cycle	No ringback above VP_{OFF}	-	0.3	V

1 Thermal sensor is active.

2 Output slew rate is guaranteed by design and / or characterization. The output slew rate reference load is shown in Figure 5 and Figure 6 shows the timing measurement points. For slew rate measurements, the V_{OH} level shown in Figure 6 is a function of R_{on} value; V_{OH} = {1.0 / (R_{on} + 50)} * 50.

5.6 AC Characteristics

Symbol	Parameter	I²C Mode - Open Drain		I3C Basic - Push-Pull[1]		
		Min	Max	Min	Max	Unit
f_{SCL}	Clock Frequency	0.01	1	0	12.5	MHz
t_{HIGH}	Clock Pulse Width High Time	260	-	35	-	ns
t_{LOW}	Clock Pulse Width Low Time	500	-	35	-	ns
$t_{TIMEOUT}$	Detect Clock Low Timeout	10	50	10	50	ms
t_R	SDA Rise Time[2,3]	-	120	-	5	ns
t_F	SDA Fall Time[2,3]	-	120	-	5	ns
$t_{SU:DAT}$	Data In Set-up Time[2]	50	-	8	-	ns
$t_{HD:DI}$	Data In Hold Time[2]	0	-	3	-	ns
$t_{SU:STA}$	Start Condition Setup Time[2]	260	-	12	-	ns
$t_{HD:STA}$	Start Condition Hold Time[2]	260	-	30	-	ns
$t_{SU:STO}$	Stop Condition Setup Time[2]	260	-	12	-	ns
t_{BUF}	Time between Stop Condition and next Start Condition[2,4]	500	-	500	-	ns
t_{POFF}	Warm Power Cycle Off Time	1	-	1	-	ms
t_{Sense_SA}	Time from Valid 1.8V Supply to Sense SA Pin for LID Code Assignment	-	5	-	5	ms
t_{INIT}	Time from Power On to First Command	10	-	10	-	ms
t_{RST}	Device Re-initialization Time	-	-	TBD	TDB	ms
t_{AVAL}	Bus Available Time (No Edges Seen on SDA and SCL)	-	-	1	-	µs
t_{IBI_ISSUE}	Time to Issue IBI after an Event is Detected when Bus is Available	-	-	-	15	µs
$t_{CLR\_I3C\_CMD\_Delay}$	Time from Clear Register Status to any I3C Operation with Start Condition to Avoid IBI Generation; PEC Disabled	-	-	2.5	-	µs
	Time from Clear Register Status to any I3C Operation with Start Condition to Avoid IBI Generation; PEC Enabled	-	-	15	-	µs
$t_{HD:DAT}$	SCL Falling Clock In to HSDA Data Out Hold Time[5]	0.5	350	N/A	N/A	ns
t_{DOUT}	SCL Falling Clock In to HSDA Valid Data Out Time[6]	N/A	N/A	0.5	12	ns
t_{DOFFT}	SCL Rising Clock In to SDA Output Off[7]	N/A	N/A	0.5	12	ns
t_{DOFFC}	SCL Rising Clock In to Controller SDA Output Off[8]	N/A	N/A	0.5	t_{HIGH}	ns

Symbol	Parameter	I²C Mode - Open Drain		I3C Basic - Push-Pull[1]		
		Min	Max	Min	Max	Unit
$t_{CL\_r\_DAT\_f}$	SCL Rising Clock In to Controller Driving SDA Signal Low[9]	N/A	N/A	40	-	ns
$t_{DEVCTRLCCC\_DELAY\_PEC\_DIS}$	DEVCTRL CCC Followed by DEVCTRL CCC or Register Read/Write Command Delay[10,11,12]	3	-	3	-	µs
$t_{WR\_RD\_DELAY\_PEC\_EN}$	Register Write Command Followed by Register Read Command Delay in PEC Enabled Mode[13,14,15]	N/A	N/A	8	-	µs
$t_{I2C\_CCC\_Update\_Delay}$	SETHID CCC or SETAASA CCC to any other CCC or Read/Write Command Delay	2.5	-	-	-	µs
$t_{I3C\_CCC\_Update\_Delay}$	RSTDAA CCC or ENEC CCC or DISEC CCC to any other CCC or Read/Write Command Delay	-	-	2.5	-	µs
t_{CCC_Delay}	Any CCC to RSTDAA CCC delay	N/A	N/A	2.5	-	µs

1. I3C mode with Open Drain operation follows timing values as shown in *I²C Mode - Open Drain* column.
2. See Figure 2 for input timing parameter definition.
3. See Figure 7 for voltage threshold definition for rise and fall times.
4. If PEC is enabled, $t_{WR\_RD\_DELAY\_PEC\_EN}$ timing parameter also applies.
5. See Figure 3 *I3C Basic Bus AC Data Output Timing Parameter Definition*
6. for output timing parameter definition.
7. The device must be in configured in I3C Basic mode to guarantee t_{DOUT} value. See Figure 2 *I2C or I3C Basic Bus AC Input Timing Parameter Definition*
8. for output timing parameter definition. See Figure 5 for output timing parameter measurement reference load.
9. The GD30TP139A device must be configured in I3C Basic mode to guarantee t_{DOFFT} value. See Figure 11. See Figure 5 for output timing parameter measurement reference load. Also refer to MIPI Alliance Specification for *I3C Basic Version 1.0-19 July 2018, section 5.1.2.3.2, Transition from Address ACK to Mandatory Byte during IBI*.
10. The GD30TP139A device must be configured in I3C Basic mode. The Host guarantees t_{DOFFC} value. See Figure 12. See Figure 5 for output timing parameter measurement reference load.
11. See Figure 14.
12. From STOP condition of DEVCTRL CCC to START condition for Register Read or Register Write Command Data Packet delay.
13. The GD30TP139A device may send ACK or NACK if Host does not satisfy $t_{DEVCTRLCCC\_DELAY\_PEC\_DIS}$ timing parameter.
14. This timing parameter restriction is only applicable when PEC function is disabled in GD30TP139A. If PEC is enabled, this timing parameter does not apply.
15. From STOP condition for Register Write Command Data Packet to START condition for Register Read Command Data Packet delay.
16. This timing parameter restriction is only applicable when PEC function is enabled in GD30TP139A. If PEC is disabled, this timing parameter does not apply.
17. The GD30TP139A device may send ACK or NACK if Host does not satisfy $t_{WR\_RD\_DELAY\_PEC\_EN}$ timing parameter.

5.7 AC Timing Definition

5.7.1 I²C or I3C Basic Bus Timing

The GD30TP139A device follows the I²C or I3C Basic bus timing requirements. The Figure 2 and Figure 2 I2C or I3C Basic Bus AC Input Timing Parameter Definition
show the timing diagram for Data Bus Input and Data Output parameters.

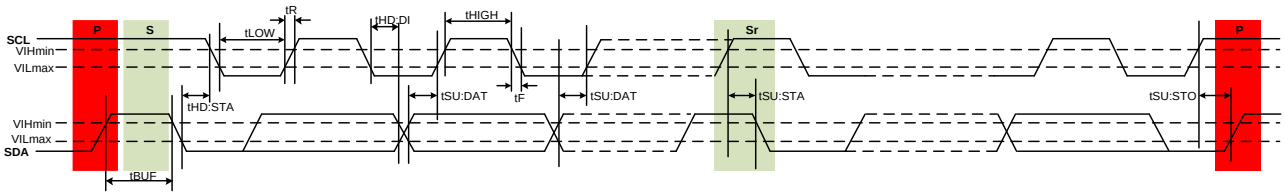


Figure 2 I²C or I3C Basic Bus AC Input Timing Parameter Definition

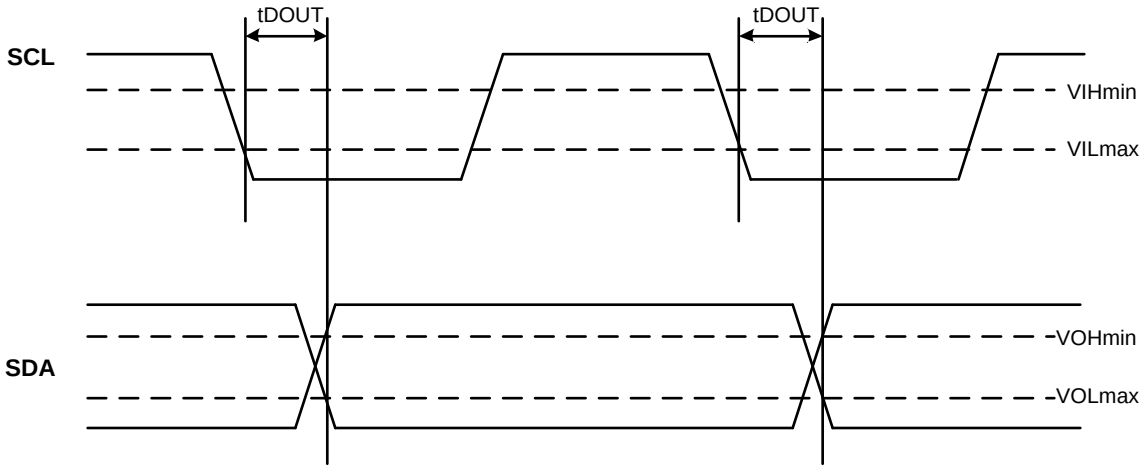


Figure 3 I3C Basic Bus AC Data Output Timing Parameter Definition

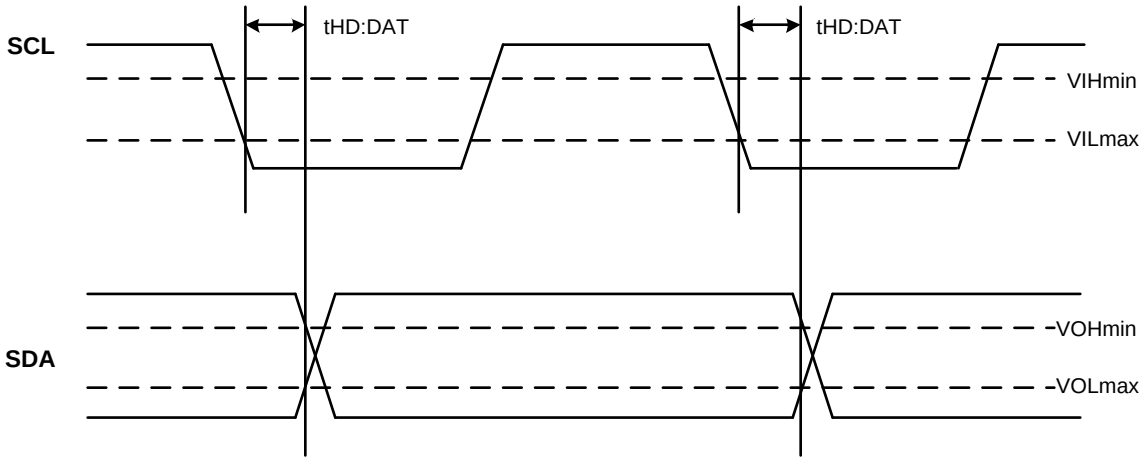


Figure 4 I²C Bus AC Data Output Timing Parameter Definition

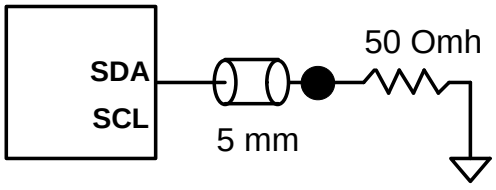


Figure 5 Output Slew Rate and Output Timing Reference Load

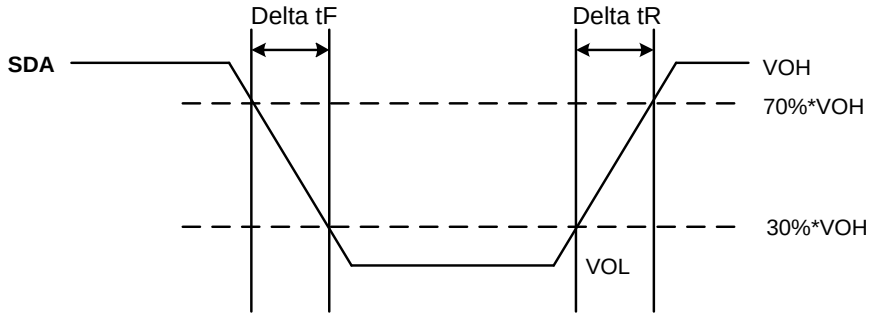


Figure 6 Output Slew Rate Measurement Points

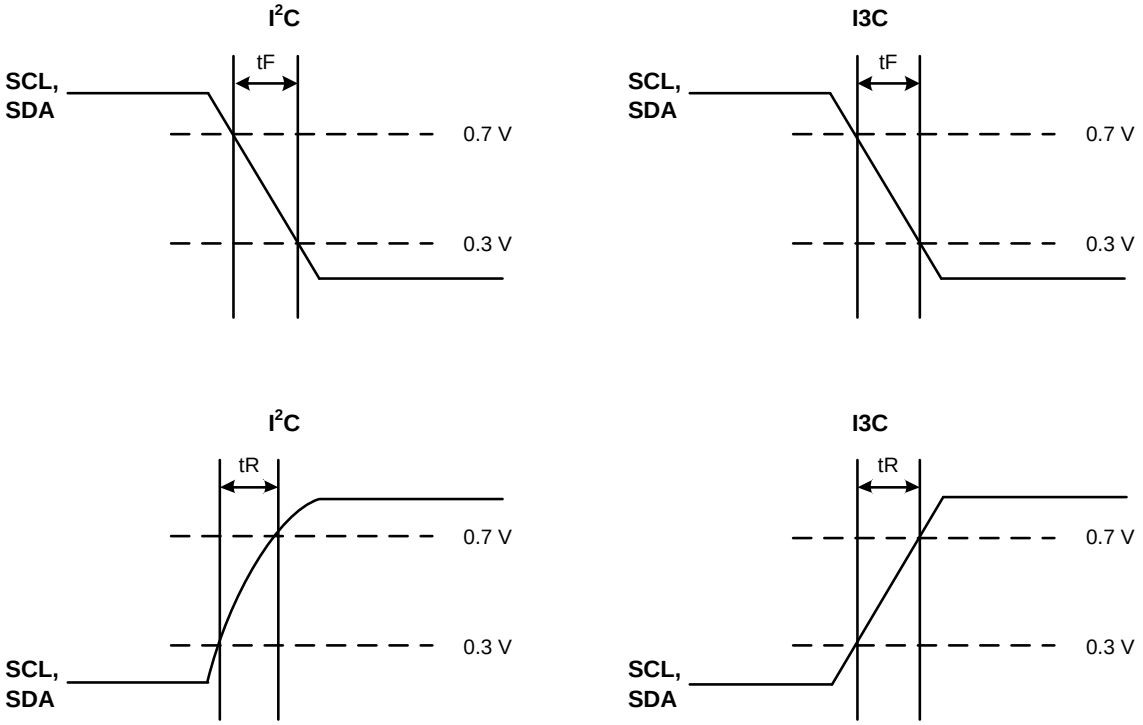


Figure 7 Rise and Fall Timing Parameter Definition

5.8 Temperature Sensor Performance

SYMBOL	PARAMETER	TEST CONDITION	MIN	TYP	MAX	UNIT
T_{ACC}	Temperature Sensor Accuracy (Active Range)	+75°C ≤ T_A ≤ +95°C	-	±0.5	±1.0	°C
	Temperature Sensor Accuracy (Monitor Range)	+40°C ≤ T_A ≤ +125°C	-	±1.0	±2.0	°C
	Temperature Sensor Accuracy (Industrial Temperature Range)	-40°C ≤ T_A ≤ +125°C	-	±2.0	±3.0	°C
R_{TS}	Temperature Sensor Resolution			0.25		°C
t_{CONV}	Conversion Time	Assumes 0.25 °C accuracy			68	ms
T_{HYST}	Hysteresis between temperature events		1	-	-	°C

6 Detailed Description

6.1 Block Diagram

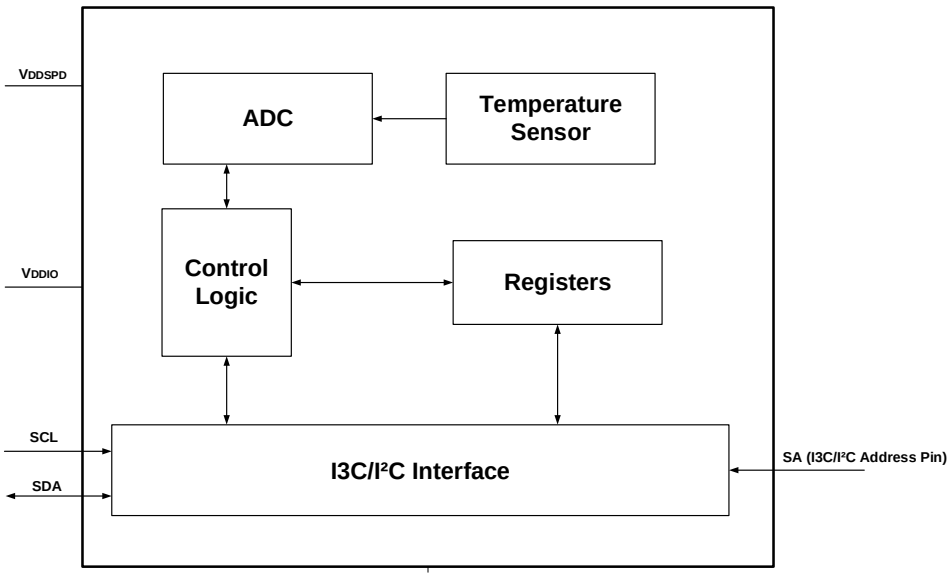


Figure 8. GD30TP139A Functional Block Diagram

6.2 Device Power Up, Reset and Initializatio-n

The GD30TP139A device has one V_{DDSPD} supply input and one V_{DDIO} supply input.

In order to prevent inadvertent operations during power up, a Power-On Reset (POR) circuit is included. On cold power on, V_{DDSPD} input supply must rise monotonically between VPON and $V_{DDSPDmin}$ without ring back to ensure proper startup.

The GD30TP139A device uses V_{DDIO} supply for its IO levels and it must reach $V_{DDIOmin}$ to ensure proper operation of I^2C or I3C Basic bus interface.

Once the V_{DDSPD} and V_{DDIO} supply is valid and stable, the GD30TP139A device shall:

1. Once V_{DDSPD} supply is valid and stable, within t_{Sense_SA} time, sense its SA pin to automatically configure the LID code based on what is detected on SA pin.
2. Enable I^2C interface within t_{INIT} time and be ready to receive the command from the Host. The GD30TP139A device is ready for operation after t_{INIT} time.

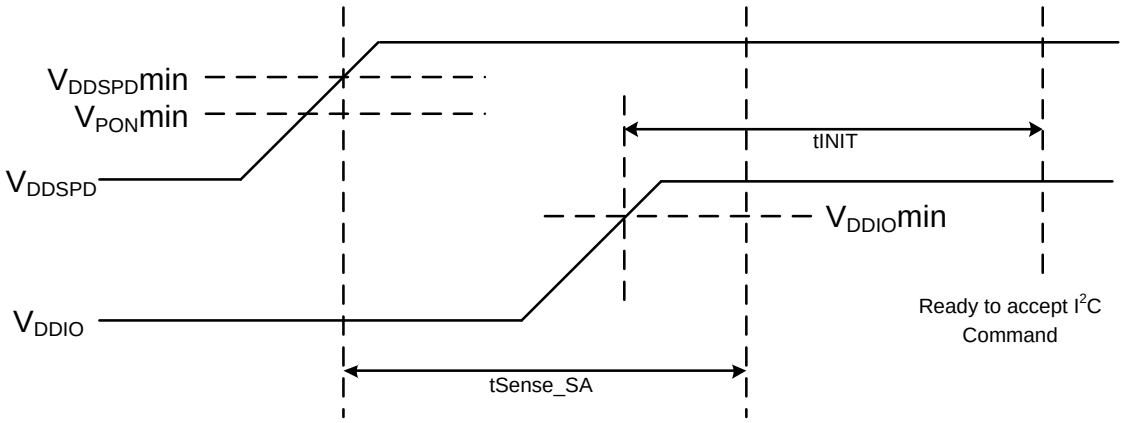


Figure 9 Device Power Up Sequence

6.2.1 Device Reset and Initialization

At power down (phase during which V_{DDSPD} input supply decreases continuously), as soon as V_{DDSPD} input supply drops below the $V_{DDSPDmin}$, the GD30TP139A device does not guarantee the operation.

On warm power cycling, the V_{DDSPD} and V_{DDIO} input supply must remain below V_{POFF} for t_{POFF} and must meet cold power on reset timing when restoring the power.

6.2.2 Bus Clear

The GD30TP139A device supports the following described Bus Clear feature in I²C mode only. Any attempt by Host to perform I²C Bus Clear on a Target device in I3C mode may result in an active drive bus contention on the SDA data line.

There may be abnormal circumstances when the host abruptly stops clocking SCL while the Target device is in the middle of outputting data for read operation. For this type of events, the SDA data line may appear as stuck low as the device is expecting to receive more clock pulses from the Host. Eventually when the Host has control of the SCL clock, the Host may optionally clear the device that is stuck low on the SDA data line by sending continuous 18 clock pulses without driving the SDA data line followed by STOP operation. The device floats the SDA line within 18 clock pulses and returns to the Idle state. The device is ready for normal new transaction with Start condition.

6.2.3 Bus Reset

To prevent a malfunctioning device from locking up the I²C bus or I3C Basic bus, a bus reset mechanism is defined. It uses a timeout mechanism on SCL as shown in Figure 10 to force a device bus reset. All devices on an I²C or I3C Basic bus reset simultaneously. Bus reset operation works in the same way regardless of whether the device is operating in I²C or I3C Basic mode.

To guarantee the device resets I²C bus or I3C Basic bus, the HSCL clock input Low time has to be greater than or equal to $t_{TIMEOUT(Max)}$.

The GD30TP139A device does not reset I²C bus or I3C Basic bus if the SCL clock input Low time is less than $t_{TIMEOUT(Min)}$.

If the SCL clock input Low time is between $t_{TIMEOUT(Min)}$ and $t_{TIMEOUT(Max)}$, the GD30TP139A device does not

guarantee and it may or may not reset the I²C bus or I3C Basic bus.

When RESET, the GD30TP139A device takes the following actions:

1. Interface and any pending command or transactions are cleared.
2. All internal register values are preserved unless noted otherwise in item # 3 below.
3. Device returns to I²C mode of operation; *MR7*[3:1] resets to '111'; *MR18*[7:5] resets to '000'; *MR27*[4] resets to '0'; *MR52*[1:0] resets to '00'.
4. Device does not re-sample SA pin.
5. Device floats the SDA pin such that it gets pulled High by external/other device pullup.
6. Device treats bus resets as STOP operation.

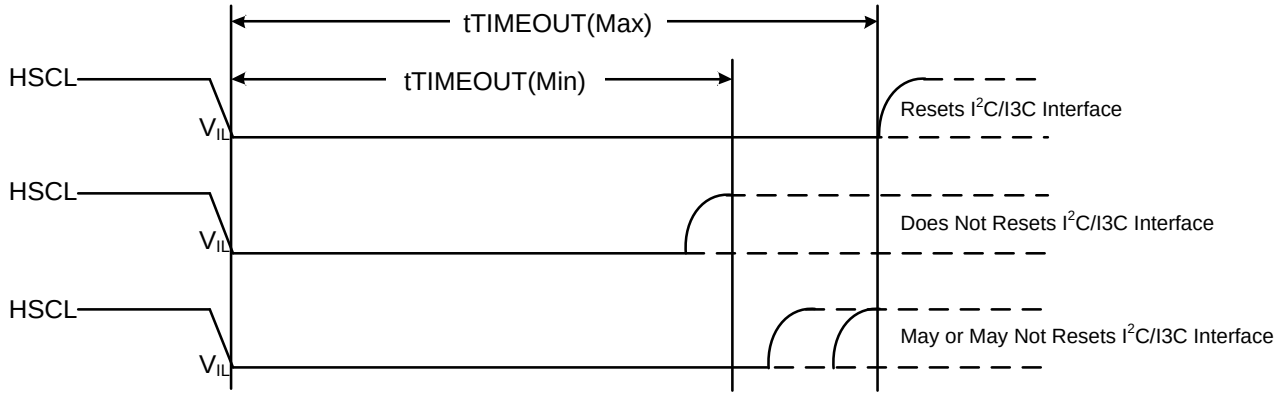


Figure 10 I2C or I3C Basic Bus Reset - GD30TP139A Device

6.3 IO Operation

At power on, by default, the GD30TP139A device comes up in legacy I²C mode of operation with Open Drain IO for its interface. The maximum speed is limited to 1 MHz and supported IO voltage levels are from 1.0V to 3.3V for Grade A.

After power on, the Host may put the GD30TP139A device in I3C mode of operation. In I3C Basic mode of operation, the maximum speed is limited to 12.5 MHz and supported IO voltage levels are from 1.0 V to 1.2 V.

In I3C Basic mode, the Host may drive the SCL clock input of the GD30TP139A device using either push-pull output driver or using the open-drain output driver. It is expected that for all DDR5 DIMM family environment, the Host may always drive the SCL clock input using a push-pull output driver.

To support in band interrupt, the GD30TP139A device supports dynamic switching between Open Drain mode and Push Pull mode on its SDA bus for various events. The Table 3 below describes the different mode of operation by the GD30TP139A device for each cycle.

Table 3 GD30TP139A Device Dynamic IO Operation Mode Switching

Features	Open Drain Mode	Push Pull Mode
Start + Device Select Code	Yes	No
Start + 7'H7e Ibi Header Byte	Yes	No
Repeat Start + Device Select Code	No	Yes
Repeat Start + 7'H7e Header Byte	No	Yes
CCC Bytes (I.E After 7'H7e+W=0+ACK)	No	Yes
Stop	No	Yes
Ack/Nack Responses	Yes	No
Command, Block Address, Address Operation	No	Yes
Interrupt Request By Target + Device Select Code	Yes	No
IBI Payload	No	Yes
Write Data, T-Bit Sequence	No	Yes
Read Data, T-Bit Sequence	No	Yes
PEC, T-Bit Sequence	No	Yes

6.4 Device Interface - Protocol

6.4.1 I²C and I3C Basic Operation

At power on, by default, the GD30TP139A device comes up in I²C mode of operation. Following applies in I²C mode:

1. The maximum operation speed is limited to 1MHz.
2. In-band interrupts are not supported.
3. Bus reset is supported.
4. Parity check is not supported except for supported CCCs.
5. Packet Error check is not supported.

The GD30TP139A device shall operate in the I²C mode until put into I³C Basic mode via command.

The Host may put the GD30TP139A device in I3C Basic mode by issuing SETAASA CCC.

Following applies in I3C mode:

1. The maximum operation speed is up to 12.5MHz.
2. In-band interrupts are supported.
3. Bus reset is supported.
4. Parity check is always enabled by default.
5. Packet error check is supported and by default is disabled.

6.4.2 Serial Address of the GD30TP139A Device

The 7-bit serial address of the GD30TP139A device applies to both I²C and I3C Basic mode of operation identically. The GD30TP139A device 4-bit binary value is 0010b or 0110b depending on sample status of SA pin on power up.

The GD30TP139A device samples the status of the SA pin on power up. The sampled status of the SA pin is used to select one of the two possible unique LID code for the device. The selected LID code either 0010b or 0110b is merged with a 3-bit HID code *MR7*[3:1] to establish the 7-bit address code for the device. With the default setting in *MR7*[3:1] = '111'; if the SA pin is connected to VSS, the device address shall be 0010 111b and if the SA pin is connected to VDDSPD, the device address shall be 0110 111b.

Table 4 7-bit Address of GD30TP139A Device

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	SA	1	0	1	1	1	R/W
GD30TP139A Device Type ID (LID)				Host ID (HID)			Read/Write

6.4.3 Switch from I²C Mode to I3C Basic Mode

By default when GD30TP139A first powers on, it operates in I²C mode. The GD30TP139A device shall operate in I²C mode until put into I3C Basic mode via command.

In I²C mode, the Host is allowed to issue only 3 CCCs (DEVCTRL, SETHID, SETAASA). All other CCCs are not supported and the GD30TP139A device may simply ignore it. The Host must issue DEVCTRL and SETHID CCC first (if required) followed by SETAASA CCC.

The Host puts the GD30TP139A device in I3C Basic mode by issuing SETAASA CCC. When SETAASA CCC is registered by the GD30TP139A device, it updates the *MR18*[5] to '1'. When SETHID CCC is registered by the GD30TP139A device, it updates the *MR7*[3:1].

6.4.4 Switch from I3C Basic Mode to I²C Mode

The Host can put the GD30TP139A device back in I²C mode from I3C mode at any time by issuing RSTDAA CCC. When RSTDAA CCC is registered by the GD30TP139A device, it updates the *MR18*[5] to '0'.

6.5 I2C Target Protocol

The GD30TP139A devices operate on a standard I²C serial interface. Transactions where the GD30TP139A device is the targeted Target device begin with the Host issuing a START condition followed by a 7-bit GD30TP139A device address then a read or write bit, RW. All data are transmitted with the most significant bit MSB first. During the address followed by R/W bit transmission, the GD30TP139A device typically replies with an ACK unless there are conditions when it may passively assert a NACK.

The GD30TP139A device accepts 1 byte of address which covers 256 bytes of registers. The GD30TP139A device volatile register space does not require page selection process as all registers are within first 256 bytes.

6.5.1 Write Operation - Data Packet

Table 5 Write Command Data Packet

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N	Stop
S or Sr[1]	0	X	1	0	HID			W=0	A	
	Address [7:0]								A	
	Data								A	
	Data								A	
	…								A	
	Data								A	Sr or P

1. In I²C mode, Start or Repeat Start operation followed by 7'h7E with W=0 is only allowed for the purpose of issuing CCCs that are allowed in I²C mode. Any other operation including another Repeat Start is considered an illegal operation.

6.5.2 Read Operation - Data Packet

Table 6 Read Command Data Packet

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N	Stop
S or Sr[1]	0	0	1	0	HID			W=0	A	
	Address [7:0]								A	
Sr	0	0	1	0	HID			R=1	A[2]	
	Data								A	
	Data								A	
	…								A	
	Data								N	Sr or P

1. In I2C mode, Start or Repeat Start operation followed by 7'h7E with W=0 is only allowed for the purpose of issuing CCCs that are allowed in I2C mode. Any other operation including another Repeat Start is considered an illegal operation.
2. If Target device NACKs during Repeat Start for any reason, the Host may re-try Repeat Start again. The Host can do the Repeat Start as many times it may desire.

6.5.3 Default Read Address Pointer Mode

During normal operation of the DDR5 DIMM, the Host periodically may poll critical information from the same location. An example may be the GD30TP139A device's temperature readout. To help improve the efficiency of the I²C bus protocol, the GD30TP139A offers a default read address pointer mode so that whenever the GD30TP139A device sees the STOP operation on its SCL and SDA bus, its read address pointer always resets to default address. The default read pointer address mode is enabled through register *MR18*[4] and default starting address for read operation is selectable through register *MR18*[3:2]. This allows Host to read the read command data packet as shown in Table 7. The default read address pointer reduces the packet overhead by 2 bytes. The Host typically enables this mode when the normal operation of the DDR5 DIMM begins.

Table 7 Read Command Data Packet w/ Default Address Pointer Mode

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N	Stop
S or Sr	0	X	1	0	HID			R=1	A	
	Data								A	
	…								A	
	Data								N	Sr or P

6.6 I3C Basic Target Protocol - Host to GD30TP139A Device

6.6.1 Write Operation Data Packet

The GD30TP139A devices operate on a standard I3C serial interface. Transactions where the GD30TP139A device is the targeted Target device begin with the Host issuing a START condition followed by a 7-bit GD30TP139A device address then a write bit, RW. All data are transmitted with the most significant bit MSB first. During the address followed by R/W bit transmission, the GD30TP139A device typically replies with an ACK unless there are conditions when it may passively assert a NACK. The ‘T’ bit carries Parity information from the Host for each byte.

The Packet Error Code (PEC) function is disabled by default when the GD30TP139A device is put in I3C Basic mode.

The Host may enable this function through DEVCTRL CCC (RegMod = ‘0’). If enabled, the PEC is appended at the end of all transactions. If PEC is enabled, the Host must complete the burst length as indicated in CMD field. In other words, the Host must not interrupt the burst length pre-maturely for Write operation.

Table 8 Write Command Data Packet; PEC Disabled

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	0	X	1	0	HID			W=0	A[1,2,3]	
	Address [7:0]								T	
	Data								T	
	…								T	
	Data								T	Sr[4] or P

1. See Figure 11 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Address, bit[7]).
2. The GD30TP139A NACKs if there is a parity error in a previous transaction when Host performs consecutive transactions with Repeat Start.
3. The GD30TP139A device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the Host does not match with its own device code. The GD30TP139A device ignores the entire packet until STOP or next Repeat Start operation.
4. Repeat Start or Repeat Start with 7’h7E.

Table 9 Write Command Data Packet; PEC Enabled

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	0	X	1	0	HID			W=0	A[1,2,3]	
	Address [7:0]								T	
	CMD			W=0	0	0	0	0	T	
	Data								T	
	…								T	
	Data									
	PEC								T	Sr[4] or P

1. See Figure 11 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Address, bit[7]).
2. The GD30TP139A NACKs if there is a parity or PEC error in a previous transaction when host performs consecutive transactions with Repeat Start.
3. The GD30TP139A device does not check for parity or PEC error in subsequent bytes when it determines 7-bit device select code issued by the host does not match with its own device code. The GD30TP139A device ignores the entire packet until STOP or next Repeat Start operation.
4. Repeat Start or Repeat Start with 7'h7E.

The Host may optionally allow GD30TP139A device to request IBI. For this case, the transactions to the GD30TP139A device begin with the I3C Basic Host issuing a START condition followed by 7'h7E and then write bit. If GD30TP139A device has a pending IBI, it transmits its 7-bit device select code followed by R=1. If GD30TP139A device has no pending IBI, there is no action taken by GD30TP139A. The Table 10 and Table 11 show the I3C Basic bus write command data packet with optional IBI header for PEC disabled and PEC enabled cases respectively. Note that in Table 11, PEC calculation does not include IBI header byte (7'h7E followed by W=0).

Table 10 Write Command Data Packet w/ IBI Header; PEC Disabled

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S	1	1	1	1	1	1	0	W=0	A[1,3]	
Sr	0	X	1	0	HID			W=0	A[2,3,4]	
	Address [7:0]								T	
	Data								T	
	…								T	
	Data								T	Sr[5] or P

1. See Figure 11 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (Repeat Start).
1. See Figure 13 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK) and Figure 11 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Address, bit[7]).
2. The GD30TP139A NACKs if there is a parity error in a previous transaction when Host performs consecutive transactions with Repeat Start.
3. The GD30TP139A device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the Host does not match with its own device code. The GD30TP139A device ignores the entire packet until STOP or next Repeat Start operation.

4. Repeat Start or Repeat Start with 7'h7E.

Table 11 Write Command Data Packet w/ IBI Header; PEC Enabled

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S	1	1	1	1	1	1	0	W=0	A[1,3]	
Sr	0	X	1	0	HID			W=0	A[2,3,4]	
	Address [7:0]								T	
	CMD			W=0	0000				T	
	Data								T	
	…								T	
	Data								T	
	PEC								T	Sr[5] or P

1. See Figure 11 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (Repeat Start).
2. See Figure 13 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK) and Figure 11 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Address, bit[7]).
3. The GD30TP139A NACKs if there is a parity error in a previous transaction when Host performs consecutive transactions with Repeat Start.
4. The GD30TP139A device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the Host does not match with its own device code. The GD30TP139A device ignores the entire packet until STOP or next Repeat Start operation.
5. Repeat Start or Repeat Start with 7'h7E.

6.6.2 Read Operation Data Packet

The transactions to GD30TP139A Target device begin with the I3C Basic Host issuing a START condition followed by a 7-bit GD30TP139A device type identifier then a write bit. All I3C Basic bus data are transmitted with the most significant bit MSB first. During select code transmission, the GD30TP139A device typically replies with an ACK unless there are exceptional conditions when it may passively assert a NACK. See Table 12. The 'T' bit carries Parity information from the Host prior to Repeated START. After Repeated START, 'T' bit carries information from GD30TP139A device to Host indicating Continuous ('1') or Stop ('0') whether it is transmitting the last byte or not.

The Packet Error Code (PEC) function is disabled by default when GD30TP139A device is put in I3C Basic mode. The Host may enable this function through DEVCTRL CCC (RegMod = '0'). If enabled, the PEC is appended as shown in Table 13. If PEC is enabled, the Host must complete the burst length as indicated in CMD field. In other words, the Host must not interrupt the burst length pre-maturely for Read operation.

The Host may optionally allow GD30TP139A device to request IBI. For this case, the transactions to the GD30TP139A device begin with the I3C Basic Host issuing a START condition followed by 7'h7E and then write bit. If GD30TP139A device has a pending IBI, it transmits its 7-bit device select code followed by R=1. If GD30TP139A device has no pending IBI, there is no action taken by GD30TP139A. The and Table 15 show the I3C Basic bus read command data packet with optional IBI header for PEC disabled and PEC enabled cases respectively. Note that in Table 15, PEC calculation (from Host to GD30TP139A) does not include IBI header byte (7'h7E followed by W=0).

Table 12 Read Command Data Packet; PEC Disabled

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	0	X	1	0	HID			W=0	A[1,2,3]	
	Address [7:0]								T	
S or Sr	0	X	1	0	HID			R=1	A/N[4,5]	
	Data								T=1	
	…								T=1	
	Data								T=1[6,7]	Sr[8] or P

1. See Figure 11 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Address, bit[7]).
2. The GD30TP139A NACKs if there is a parity error in a previous transaction when Host performs consecutive transactions with Repeat Start.
3. The GD30TP139A device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the Host does not match with its own device code. The GD30TP139A device ignores the entire packet until STOP or next Repeat Start operation.
4. If Target device NACKs during Repeat Start for any reason, the Host may re-try Repeat Start again. The Host can do the Repeat Start as many times it may desire. If Target device NACKs due to parity error in previous bytes, it will always NACK regardless of how many times the Host tries Repeat Start. If there were no parity errors the GD30TP139A may eventually ACK.
5. See Figure 13 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK).
6. See Figure 14 to see how Host ends Target device operation.
7. When last byte (i.e. MR255) is reached (extreme rare case), the Target device sends T = ‘0’. See Figure 15 to see how Target device ends the operation followed by Host STOP operation.
8. Repeat Start or Repeat Start with 7’h7E.

Table 13 Read Command Data Packet; PEC Enabled

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	0	X	1	0	HID			W=0	A[1,2,3]	
	Address [7:0]								T	
	CMD			R=1	0	0	0	0	T	
	PEC								T	
S or Sr	0	X	1	0	HID			R=1	A/N[4,5]	
	Data								T=1	
	…								T=1	
	Data								T=1	
	PEC								T=0[6]	Sr[7] or P

1. See Figure 11 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Address, bit[7]).
2. The GD30TP139A NACKs if there is a parity error in a previous transaction when Host performs consecutive transactions with Repeat Start.
3. The GD30TP139A device does not check for parity or PEC error in subsequent bytes when it determines 7-bit device

select code issued by the Host does not match with its own device code. The GD30TP139A device ignores the entire packet until STOP or next Repeat Start operation.

4. If Target device NACKs during Repeat Start for any reason, the Host may re-try Repeat Start again. The Host can do the Repeat Start as many times it may desire. If Target device NACKs due to PEC error parity error in previous bytes, it will always NACK regardless of how many times the Host tries Repeat Start. If there were no parity or PEC errors, the GD30TP139A may eventually ACK. The PEC calculation by the Target device only includes device select code of the ACK response of the Repeat start operation. In other words, if there are more than one Repeat Start operation, the Target device includes device select code of only the last Repeat Start from the Host when it ACKs in PEC calculation and all other NACK responses of the device select code of the Repeat Start are not included in PEC calculation.
5. See Figure 13 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK).
6. See Figure 15 to see how Target device ends the operation followed by Host STOP operation.
7. Repeat Start or Repeat Start with 7'h7E.

Table 14 Read Command Data Packet w/ IBI Header; PEC Disabled

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S	1	1	1	1	1	1	0	W=0	A[1,3]	
Sr	0	X	1	0	HID			W=0	A[2,3,4]	
	Address [7:0]								T	
Sr	0	X	1	0	HID			R=1	A/N[5,6]	
	Data								T=1	
	…								T=1	
	Data								T=1[7,8]	Sr[9] or P

1. See Figure 11 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (Repeat Start).
2. See Figure 13 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK) and Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Address, bit[7]).
3. The GD30TP139A NACKs if there is a parity error in a previous transaction when Host performs consecutive transactions with Repeat Start.
4. The GD30TP139A does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the Host does not match with its own device code. The GD30TP139A ignores the entire packet until STOP or next Repeat Start operation.
5. See Figure 13 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK).
6. If Target device NACKs during Repeat Start for any reason, the Host may re-try Repeat Start again. The Host can do the Repeat Start as many times it may desire. If Target device NACKs due to parity error in previous bytes, it will always NACK regardless of how many times the Host tries Repeat Start. If there were no parity errors, the GD30TP139A may eventually ACK.
7. See Figure 14 to see how Host ends Target device operation.
8. When last byte (i.e. MR255) is reached (extreme rare case), the Target device sends T = '0'. See Figure 15 to see how Target device ends the operation followed by Host STOP operation.
9. Repeat Start or Repeat Start with 7'h7E.

Table 15 Read Command Data Packet w/ IBI Header; PEC Enabled

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S	1	1	1	1	1	1	0	W=0	A[1,3]	
Sr	0	X	1	0	HID			W=0	A[2,3,4]	
	Address [7:0]								T	
	CMD			R=1	0000				T	
	PEC								T	
Sr	0	X	1	0	HID			R=1	A/N[5,6]	
	Data								T=1	
	…								T=1	
	Data								T=1	
	PEC								T=0[7]	Sr[8] or P

1. See Figure 11 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (Repeat Start).
2. See Figure 13 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK) and See Figure 7 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (1st bit of Address, bit[7]).
3. The GD30TP139A NACKs if there is a parity or PEC error in a previous transaction when Host performs consecutive transactions with Repeat Start.
4. The GD30TP139A does not check for parity or PEC error in subsequent bytes when it determines 7-bit device select code issued by the Host does not match its own device code. The GD30TP139A ignores the entire packet until STOP or next Repeat Start operation.
5. See Figure 13 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK).
6. If Target device NACKs during Repeat Start for any reason, the Host may re-try Repeat Start again. The Host can do Repeat Start as many times it may desire. If Target device NACKs due to PEC error or parity error in previous bytes, it will always NACK regardless of how many times the Host tries Repeat Start. If there was no parity or PEC error, the GD30TP139A may eventually ACK. The PEC calculation by the Target device only includes device select code of the ACK response of the Repeat Start operation. If there are more than one Repeat Start operation, the Target device includes device select of only the last Repeat Start from the Host when it ACKs in PEC calculation and all other NACK responses of the device select code of the Repeat Start are not included in PEC calculation.
7. See Figure 15 to see how Target device ends the operation followed by Host STOP operation.
8. Repeat Start or Repeat Start with 7'h7E.

6.6.3 Default Read Address Pointer Mode

This mode works the same exact way as explained in Section 6.5.3. Table 16 and Table 17 show the read command data packet for PEC function disabled and enabled respectively. When PEC function is enabled, *MR18*[1] sets the number of bytes that GD30TP139A device sends out followed by the PEC calculation. If PEC is enabled, the Host must complete the burst length as indicated in *MR18*[1] register. In other words, the Host must not interrupt the burst length pre-maturely for default address pointer read operation.

Table 16 Read Command Data Packet w/ Read Address Pointer Mode; PEC Disabled

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	0	X	1	0	HID			R=1	A[1]	
	Data								T=1	
	…								T=1	
	Data								T=1[2,3]	Sr[4] or P

1. The GD30TP139A NACKs if there is a parity error in a previous transaction when Host performs consecutive transactions with Repeat Start.
2. See Figure 14 to see how Host ends Target device operation.
3. When last byte (i.e. MR255) is reached (extreme rare case), the Target device sends T = '0'. See Figure 15 to see how Target device ends the operation followed by Host STOP operation.
4. Repeat Start or Repeat Start with 7'h7E.

Table 17 Read Command Data Packet w/ Read Address Pointer Mode; PEC Enabled

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	0	X	1	0	HID			R=1	A[1]	
	Data								T=1	
	…								T=1	
	Data								T=1	
	PEC								T=0[2]	Sr[3] or P

1. The GD30TP139A NACKs if there is a parity or PEC error in a previous transaction when Host performs consecutive transactions with Repeat Start.
2. See Figure 15 to see how Target device ends the operation followed by STOP operation.
3. Repeat Start or Repeat Start with 7'h7E.

Table 18 Read CMD Data Packet w/ Read Address Pointer Mode and IBI Header; PEC Disabled

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S	1	1	1	1	1	1	0	W=0	A[1,2]	
Sr	0	X	1	0	HID			R=1	A/N[2,3]	
	Data								T=1	
	…								T=1	
	Data								T=1[4,5]	Sr[6] or P

1. See Figure 11 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (Repeat Start).
2. The GD30TP139A NACKs if there is a parity error in a previous transaction when Host performs consecutive transactions with Repeat Start.
3. See Figure 13 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK).
4. See Figure 14 to see how Host ends Target device operation.
5. When last byte (i.e. MR255) is reached (extreme rare case), the Target device sends T = '0'. See Figure 15 to see how Target device ends the operation followed by Host STOP operation.
6. Repeat Start or Repeat Start with 7'h7E.

Table 19 Read CMD Data Packet w/ Read Address Pointer Mode and IBI Header; PEC Enabled

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S	1	1	1	1	1	1	0	W=0	A[1,2]	
Sr	0	X	1	0	HID			R=1	A/N[2,3]	
	Data								T=1	
	…								T=1	
	Data								T=1	
	PEC								T=0[4]	Sr[5] or P

1. See Figure 11 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation (Repeat Start).
2. The GD30TP139A NACKs if there is a parity or PEC error in a previous transaction when Host performs consecutive transactions with Repeat Start.
3. See Figure 13 to see how the transition occurs from Host Push Pull Operation to Target Open Drain (ACK).
4. See Figure 15 to see how Target device ends the operation followed by STOP operation.
5. Repeat Start or Repeat Start with 7’h7E.

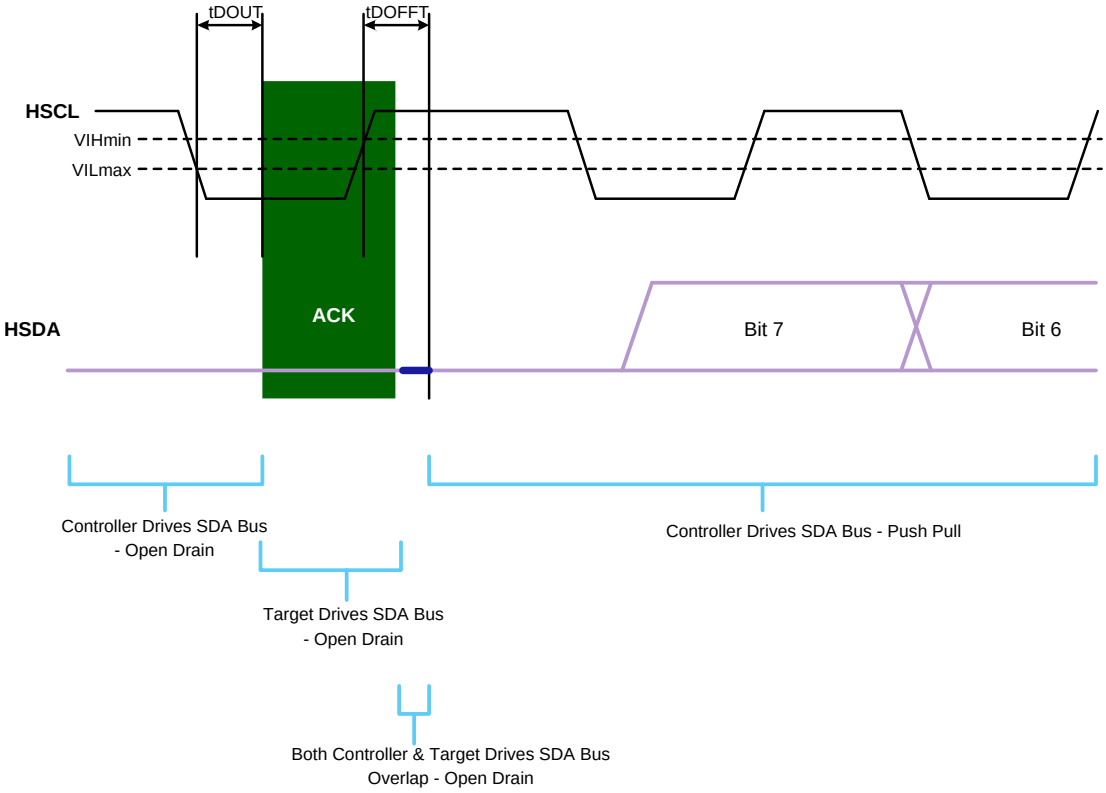


Figure 11 Target Open Drain to Host Push Pull Hand Off Operation

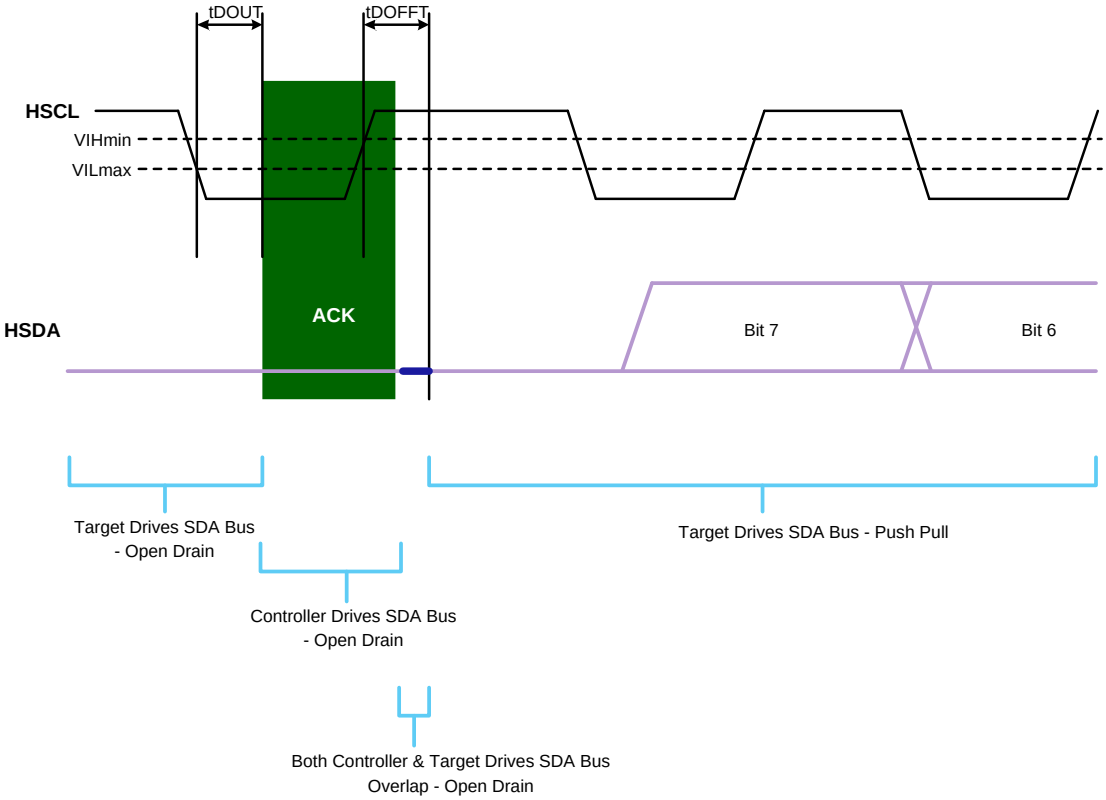


Figure 12 Controller Open Drain (ACK) to Target Push Pull Hand Off Operation

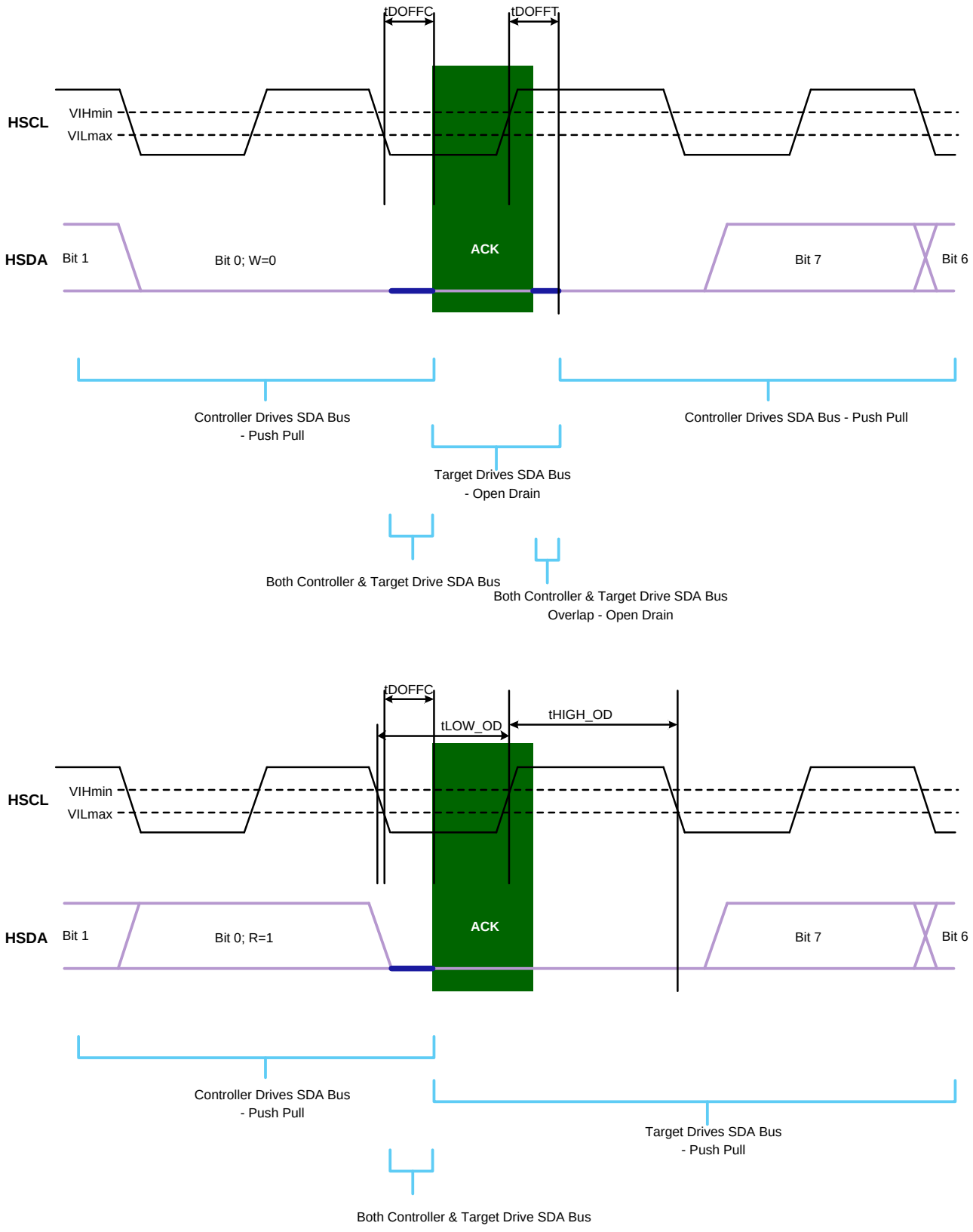


Figure 13 Controller Push Pull to Target Open Drain Hand Off Operation

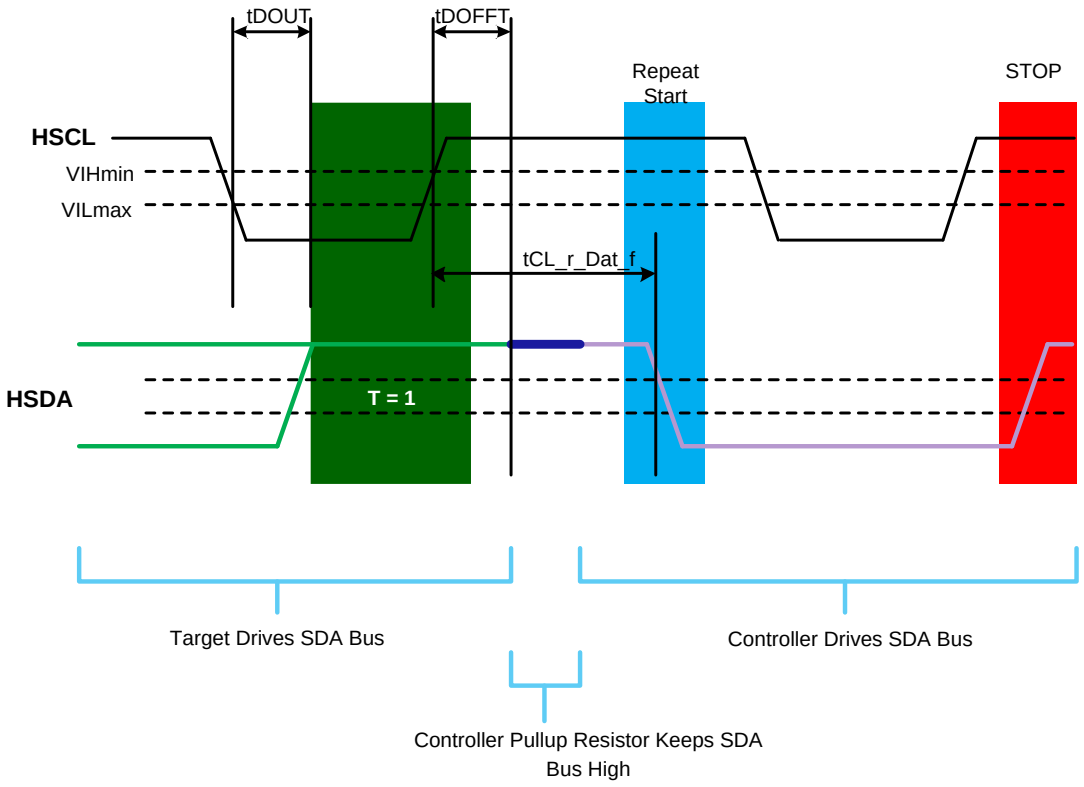


Figure 14 T = 1; Host Ends Read with Repeated START and STOP Waveform

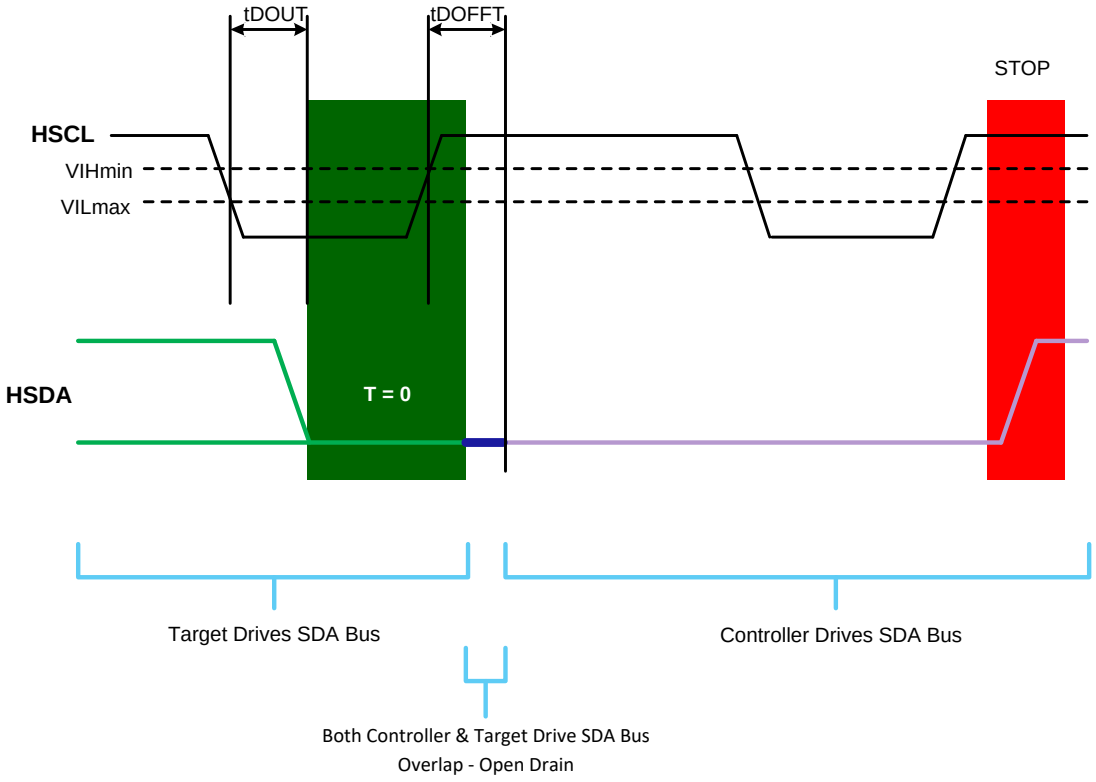


Figure 15 T = 0; Target Ends Read; Host Generates STOP

6.7 I3C Basic Common Command Codes (CCC)

The I3C Basic specification lists large number of Common Command Codes (CCC). Not all CCC are required to be supported. The GD30TP139A device NACKs for all unsupported CCC. The GD30TP139A supports CCC as listed in Table 20 below.

The GD30TP139A device requires STOP operation in between when switching from CCC operation to private device specific Write or Read or Default Read Address Pointer mode operation and vice versa. In other words, any CCC operation must be followed by STOP operation before continuing to any device specific Write or Read or Default Read Address Pointer mode operation. Similarly, any device specific Write or Read or Default Read Address Pointer mode operation must be followed by STOP operation before continuing to any CCC operation. The GD30TP139A device also requires STOP operation between any direct CCC to broadcast CCC.

The GD30TP139A device does allow Repeat Start operation between any direct CCC to any other direct CCC or between any broadcast CCC to any other broadcast CCC or between any private Write or Read or Default Read Address Pointer mode operation to any other private Write or Read or Default Read Address Pointer mode operation.
CCC is an I3C concept by definition, and shall always conform to I3C SDR timings, irrespective of whether the device has switched from I^2C mode or not.

Prior to dynamic address assignment (SETAASA/P), the Target device(s) may drive the ACK/NACK past the Open Drain SCL rising but before the next SCL falling transition, as a longer overlap in Open Drain is harmless. Immediately after the Open Drain ACK (upon the next SCL falling edge), the bus should transition to Push Pull mode (though still at 1 Mhz) as described in the MIPI *I3C Basic Specification V1.0, Section 5.1.2.3 "Handoff from Address ACK to SDR Controller Write Data" and Figure 32 "I3C Data Transfer - ACK by Target"*.

For additional details on how to handle the ACK transition, please refer to the *MIPI I3C Basic Specification V1.0, Section 5.1.2.3.1 "Transition from Address ACK to SDR Controller Write Data"*.

Table 20 GD30TP139A CCC Support Requirement

CCC	Mode	Code	Description
ENEC	Broadcast	0x00	Enable Event Interrupts
	Direct	0x80	
DISEC	Broadcast	0x01	Disable Event Interrupts
	Direct	0x81	
RSTDAA	Broadcast	0x06	Put the device in I2C Mode (aka: Reset Dynamic Address Assignment)
SETAASA	Broadcast	0x29	Put the device in I3C Basic Mode (aka: Set All Addresses to Static Address)
GETSTATUS	Direct	0x90	Get Device Status
DEVCAP[1]	Direct	0xE0	Get Device Capability
SETHID[1]	Broadcast	0x61	GD30TP139A updates 3-bit HID field, updates 'T' bit with updated parity calculation for all devices behind Hub and stops 3-bit HID translation.
DEVCTRL[1]	Broadcast	0x62	Configure GD30TP139A and all devices behind Hub

1 JEDEC specific CCC.

6.7.1 ENEC CCC

The ENEC CCC is only supported after device is put in I3C Basic mode. In I²C mode, it is illegal for Host to issue this CCC. When ENEC CCC is registered by the GD30TP139A, it updates *MR27*[4] = '1' and it takes in effect at the next Start operation (i.e. after STOP condition). Table 21 to Table 24 shows an example of a single ENEC CCC. Table 25 shows the encoding definition for ENEC CCC.

If PEC function is enabled, the PEC calculation starts with Start or Repeat Start operation but does not include 7'h7E with W=0 byte in PEC calculation.

Table 21 ENEC CCC - Broadcast

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	1	1	1	1	1	1	0	W=0	A[1]	
	0x00 (Broadcast)								T	
	7'h00							ENINT	T	Sr[2] or P

1. The GD30TP139A NACKs if there is a parity error in a previous transaction when Host performs consecutive transactions with Repeat Start.
2. Repeat Start or Repeat Start with 7'h7E.

Table 22 ENEC CCC - Broadcast w/ PEC

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	1	1	1	1	1	1	0	W=0	A[1]	
	0x00 (Broadcast)								T	
	7'h00							ENINT	T	
	PEC								T	Sr[2] or P

1. The GD30TP139A NACKs if there is a parity or PEC error in a previous transaction when Host performs consecutive transactions with Repeat Start.
2. Repeat Start or Repeat Start with 7'h7E.

Table 23 ENEC CCC - Direct

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	1	1	1	1	1	1	0	W=0	A[1]	
	0x80 (Direct)								T	
Sr	DevID[6:0]							W=0	A[1,2]	
	7'h00							ENINT	T	Sr[3] or P

1. The GD30TP139A NACKs if there is a parity error in a previous transaction when Host performs consecutive transactions with Repeat Start.
2. The GD30TP139A device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the Host does not match with its own device code. The GD30TP139A device ignores the entire packet until STOP or next Repeat Start operation.
3. Repeat Start or Repeat Start with 7'h7E.

Table 24 ENEC CCC - Direct w/ PEC

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	1	1	1	1	1	1	0	W=0	A[1]	
	0x80 (Direct)								T	
	PEC								T	
Sr	DevID[6:0]							W=0	A[1,2]	
	7'h00							ENINT	T	
	PEC								T	Sr[3] or P

1. The GD30TP139A NACKs if there is a parity or PEC error in a previous transaction when Host performs consecutive transactions with Repeat Start.
2. The GD30TP139A device does not check for parity or PEC error in subsequent bytes when it determines 7-bit device select code issued by the Host does not match with its own device code. The GD30TP139A device ignores the entire packet until STOP or next Repeat Start operation.
3. Repeat Start or Repeat Start with 7'h7E.

Table 25 ENEC CCC Byte Encoding

Bit	Encoding	Notes
ENINT	0 = No Action 1 = Enable IBI Interrupt	It is illegal for Host to issue ENEC CCC with ENINT bit = '0'

6.7.2 DISEC CCC

The DISEC CCC is only supported after device is put in I3C Basic mode. In I²C mode, it is illegal for Host to issue this CCC. When DISEC CCC is registered by the GD30TP139A, it updates *MR27*[4] = '0'and it takes in effect at the next Start operation (i.e. after STOP condition). Table 26 to Table 29 shows an example of a single DISEC CCC. Table 30 shows the encoding definition for DISEC CCC.

If PEC function is enabled, the PEC calculation starts with Start or Repeat Start operation but does not include 7'h7E with W=0 byte in PEC calculation.

Table 26 DISEC CCC - Broadcast

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	1	1	1	1	1	1	0	W=0	A[1]	
	0x01 (Broadcast)								T	
	7'h00							DISINT	T	Sr[2] or P

1. The GD30TP139A NACKs if there is a parity error in a previous transaction when Host performs consecutive transactions with Repeat Start.
2. Repeat Start or Repeat Start with 7'h7E.

Table 27 DISEC CCC - Broadcast w/ PEC

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	1	1	1	1	1	1	0	W=0	A[1]	
	0x01 (Broadcast)								T	
	7'h00							DISINT	T	
	PEC								T	Sr[2] or P

1. The GD30TP139A NACKs if there is a parity or PEC error in a previous transaction when Host performs consecutive transactions with Repeat Start.
2. Repeat Start or Repeat Start with 7'h7E.

Table 28 DISEC CCC - Direct

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	1	1	1	1	1	1	0	W=0	A[1]	
	0x81 (Direct)								T	
Sr	DevID[6:0]							W=0	A[1,2]	
	7'h00							ENINT	T	Sr[3] or P

1. The GD30TP139A NACKs if there is a parity error in a previous transaction when Host performs consecutive transactions with Repeat Start.
2. The GD30TP139A device does not check for parity error in subsequent bytes when it determines 7-bit device select code issued by the Host does not match with its own device code. The GD30TP139A device ignores the entire packet until STOP or next Repeat Start operation.
3. Repeat Start or Repeat Start with 7'h7E.

Table 29 DISEC CCC - Direct w/ PEC

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	1	1	1	1	1	1	0	W=0	A[1]	
	0x81 (Direct)								T	
	PEC								T	
Sr	DevID[6:0]							W=0	A[1,2]	
	7'h00							DISINT	T	
	PEC								T	Sr[3] or P

1. The GD30TP139A NACKs if there is a parity or PEC error in a previous transaction when Host performs consecutive transactions with Repeat Start.
2. The GD30TP139A device does not check for parity or PEC error in subsequent bytes when it determines 7-bit device select code issued by the Host does not match with its own device code. The GD30TP139A device ignores the entire packet until STOP or next Repeat Start operation.
3. Repeat Start or Repeat Start with 7'h7E.

Table 30 DISEC CCC Byte Encoding

Bit	Encoding	Notes
DISINT	0 = No Action 1 = Disable IBI Interrupt	It is illegal for Host to issue DISEC CCC with DISINT bit = '0'

6.7.3 RSTDAA CCC

The RSTDAA CCC is only supported after device is put in I3C Basic mode. In I²C mode, this CCC is ignored. When RSTDAA CCC is registered by the GD30TP139A, it updates MR18[5] = '0'and it takes in effect at the next Start operation (i.e. after STOP condition). Further it disables IBI and PEC function (*MR27*[4] = '0', *MR18*[7] = '0' respectively) and clears parity function (MR18[6] = '0'). Table 31 to Table 32 shows an example of a single RSTDAA CCC.

If PEC function is enabled, the PEC calculation starts with Start or Repeat Start operation but does not include 7'h7E with W=0 byte in PEC calculation.

Table 31 RSTDAA CCC - Broadcast

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	1	1	1	1	1	1	0	W=0	A[1]	
	0x06 (Broadcast)								T	Sr[2] or P

1. The GD30TP139A NACKs if there is a parity error in a previous transaction when Host performs consecutive transactions with Repeat Start.
2. Repeat Start or Repeat Start with 7'h7E.

Table 32 RSTDAA CCC - Broadcast w/ PEC

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	1	1	1	1	1	1	0	W=0	A[1]	
	0x06 (Broadcast)								T	
	PEC								T	Sr[2] or P

1. The GD30TP139A NACKs if there is a parity or PEC error in a previous transaction when Host performs consecutive transactions with Repeat Start.
2. Repeat Start or Repeat Start with 7'h7E.

6.7.4 SETAASA CCC

The SETAASA CCC is only supported when device is in I²C mode; however, it still follows I3C SDR timings compliant to CCC definitions. In I²C mode, when the Host issues this CCC, to guarantee that this CCC is registered by the device without any error, the Host shall limit the maximum speed operation for this CCC to 1 MHz. In I3C Basic mode, this CCC is ignored. When SETAASA CCC is registered by the GD30TP139A, it updates MR18[5] = '1'and it takes in effect at the next Start operation (i.e. after STOP condition). Table 33 shows an example of a single SETAASA CCC.

SETAASA CCC does not support PEC function as device is in I²C mode and there is no PEC function in I²C mode.

Table 33 SETAASA CCC - Broadcast

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	1	1	1	1	1	1	0	W=0	A	
	0x29 (Broadcast)								T	P

6.7.5 GETSTATUS CCC

The GETSTATUS CCC is supported in I3C Basic mode. In I²C mode, this CCC is ignored (i.e. it is not executed internally and GETSTATUS CCC code is not acknowledged and Host must do STOP operation). Table 34 to Table 35 shows an example of a single GETSTATUS CCC. Table 36 shows the encoding definition for GETSTATUS CCC.

If PEC function is enabled, the PEC calculation starts with Start or Repeat Start operation but does not include 7'h7E with W=0 byte in PEC calculation.

When the GD30TP139A device responds to GETSTATUS CCC, after it completes the response, the PEC_Err, P_Err and Pending Interrupt Bits [3:0] do not automatically get cleared. The Host must explicitly clear the appropriate status register through Clear command by writing '1' to corresponding register or by issuing Global Clear command. Once the GD30TP139A device clears the appropriate status register, the PEC_Err, P_err and Pending Interrupt Bits [3:0] get cleared.

After Host issues Clear command, if the condition is still present, the device will again set the appropriate status register, sets the IBI status register to '1' and Pending Interrupt Bits [3:0] to '0001'.

Table 34 GETSTATUS CCC - Direct

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	1	1	1	1	1	1	0	W=0	A[1]	
	0x90 (Direct)								T	
Sr	DevID[6:0]							R=1	A[1]	
	PEC_Err	0	0	0	0	0	0	0	T=1	
	0	0	P_Err	0	Pending Interrupt				T=0	Sr[2] or P

1. The GD30TP139A NACKs if there is a parity error in a previous transaction when Host performs consecutive transactions with Repeat Start.
2. Repeat Start or Repeat Start with 7'h7E.

Table 35 GETSTATUS CCC - Direct w/ PEC[1]

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	1	1	1	1	1	1	0	W=0	A[2]	
	0x90 (Direct)								T	
	PEC								T	
Sr	DevID[6:0]							R=1	A[1]	
	PEC_Err	0	0	0	0	0	0	0	T=1	
	0	0	P_Err	0	Pending Interrupt				T=1	
	PEC								T=0	Sr[3] or P

1. GETSTATUS CCC with PEC check is only supported in I3C Basic mode.
2. The GD30TP139A NACKs if there is a parity or PEC error in a previous transaction when Host performs consecutive transactions with Repeat Start.
3. Repeat Start or Repeat Start with 7'h7E.

Table 36 GETSTATUS CCC Byte Encoding

Bit	Encoding	Notes
PEC_Err	0 = No Error 1 = PEC Error occurred	This register is cleared when Host issues clear command to MR20[1] for PEC error.
P_Err	0 = No Error 1 = Protocol Error; Parity Error occurred	This register is cleared when Host issues clear command to MR20[0] for Parity error.
Pending Interrupt	0000 = No Pending Interrupt 0001 = Pending Interrupt All other encodings are reserved	This register is cleared when Host issues clear command to any appropriate device status register that causes IBI status register to get cleared.

6.7.6 DEVCAP CCC

The DEVCAP CCC is only supported after device is put in I3C Basic mode. In I^2C mode, it is illegal for Host to issue this CCC. Table 37 to Table 38 shows an example of a single DEVCAP CCC. Table 39 shows the encoding definition for DEVCAP CCC.

If PEC function is enabled, the PEC calculation starts with Start or Repeat Start operation but does not include 7'h7E with W=0 byte in PEC calculation.

Table 37 DEVCAP CCC - Direct

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	1	1	1	1	1	1	0	W=0	A[1]	
	0xE0 (Direct)								T	
Sr	DevID[6:0]							R=1	A[1]	
	MSB (Each bit defines capability)								T=1	
	LSB (Each bit defines capability)								T=0	Sr[2] or P

1. The GD30TP139A NACKs if there is a parity error in a previous transaction when Host performs consecutive transactions with Repeat Start.
2. Repeat Start or Repeat Start with 7'h7E.

Table 38 DEVCAP CCC - Direct w/ PEC

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	1	1	1	1	1	1	0	W=0	A[1]	
	0xE0 (Direct)								T	
	PEC								T	
Sr	DevID[6:0]							R=1	A[1]	
	MSB (Each bit defines capability)								T=1	
	LSB (Each bit defines capability)								T=1	
	PEC								T=0	Sr[3] or P

1. The GD30TP139A NACKs if there is a parity or PEC error in a previous transaction when Host performs consecutive transactions with Repeat Start.
2. Repeat Start or Repeat Start with 7'h7E.

Table 39 DEVCAP CCC Byte Encoding

Bit	Encoding	Notes
MSB[7]	RFU	Coded as '0'
MSB[6]	RFU	Coded as '0'
MSB[5]	RFU	Coded as '0'
MSB[4]	RFU	Coded as '0'
MSB[3]	RFU	Coded as '0'
MSB[2]	0 = No Support for Timer based Reset 1 = Supports Timer based Reset	Coded as '1'
MSB[1:0]	RFU	Coded as '0'
LSB[7:0]	RFU	Coded as '0'

6.7.7 SETHID CCC

The SETHID CCC is supported only when device is in I²C mode. In I²C mode, when the Host issues this CCC, to guarantee that this CCC is registered by the device without any error, the Host shall limit the maximum speed operation for this CCC to 1 MHz. In I3C Basic mode, it is illegal for Host to issue this CCC. When SETHID CCC is registered by the GD30TP139A, it updates *MR7*[3:1] with the HID code received by the GD30TP139A and it takes in effect at the next Start operation (i.e. after STOP condition). Table 40 shows an example of a single SETHID CCC. As the device is in I²C mode when SETHID CCC is issued, the PEC function is not supported.

Once GD30TP139A receives SETHID CCC and updates its 3-bit HID code, after the Stop operation, GD30TP139A device only responds to updated 7-bit address. The 4-bit LID code of the GD30TP139A device remains as is.

The Host may issue SETHID CCC more than one time.

Table 40 SETHID CCC - Broadcast

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	1	1	1	1	1	1	0	W=0	A	
	0x61 (Broadcast)								T	
	0	0	0	0	0	0	0	0	T	P

6.7.8 DEVCTRL CCC

On a typical I3C Basic bus there can be up to 120 devices. For DDR5 DIMM application environment, there are up to 8 GD30TP139A devices and behind each GD30TP139A device, there are 4 local Target devices totaling up to 40 or more devices on I3C Basic bus. For certain operation such as enable or disable functions that are common to all devices (i.e. Packet Error Check), the Host must go through one device at a time which takes significant amount of time at initial power up. Further, it requires additional complexity on the Host because it must speak different protocol depending on how it may access the device until all devices are configured identically.

To help expedite this configuration operation and to simplify the Host complexity, the device supports the DEVCTRL CCC. The DEVCTRL CCC is supported either in I^2C mode or I3C Basic mode of operation. In I^2C mode, when the Host issues this CCC, to guarantee that this CCC is registered by the device without any error, the Host shall limit the maximum speed operation for this CCC to 1 MHz. Table 41 to Table 42 shows an example of a single DEVCTRL CCC.

If PEC function is enabled, the PEC calculation starts with Start or Repeat Start operation but does not include 7'h7E with W=0 byte in PEC calculation.

The Host shall pay attention to DEVCTRL CCC. If DEVCTRL CCC is used to access device specific registers (e.g. RegMod = '1'), the Host shall still follow any device specific register restriction. For example, if device specific register requires STOP operation for device to take in the effect of the setting, the Host must also use STOP operation when using DEVCTRL CCC to access device specific register.

Table 41 DEVCTRL CCC - Broadcast

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	1	1	1	1	1	1	0	W=0	A[1]	
	0x62 (Broadcast)								T	
	AddrMask[2:0]			StartOffset[1:0]		PEC BL[1:0]		RegMod	T	
	DevID[6:0]							0	T[2]	
	Byte 0 Data Payload								T	
	Byte 1 Data Payload								T	
	Byte 2 Data Payload								T	
	Byte 3 Data Payload								T	Sr[3] or P

1. The GD30TP139A NACKs if there is a parity error in a previous transaction when Host performs consecutive transactions with Repeat Start.
2. An exception is made for DEVCTRL CCC where device does report a parity error when it determines 7-bit device select code issued by the Host does not match with its own device code. If 7-bit device select code does not match but if parity is still valid, the device does not check for parity error in subsequent bytes; ignores the entire packet and waits until

STOP or next Repeat Start operation.

3. Repeat Start or Repeat Start with 7'h7E.

Table 42 DEVCTRL CCC - Broadcast w/ PEC[1]

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S or Sr	1	1	1	1	1	1	0	W=0	A[2]	
	0x62 (Broadcast)								T	
	AddrMask[2:0]			StartOffset[1:0]		PEC BL[1:0]		RegMod	T	
	DevID[6:0]							0	T[3]	
	Byte 0 Data Payload								T	
	Byte 1 Data Payload								T	
	Byte 2 Data Payload								T	
	Byte 3 Data Payload								T	
	PEC								T	Sr[4] or P

1. DEVCTRL CCC with PEC check is only supported in I3C Basic mode.
2. The GD30TP139A NACKs if there is a parity or PEC error in a previous transaction when Host performs consecutive transactions with Repeat Start.
3. An exception is made for DEVCTRL CCC where device does report a parity error when it determines 7-bit device select code issued by the Host does not match with its own device code. The device does not check for PEC as all subsequent bytes are discarded due to parity error. If 7-bit device select code does not match but if parity is still valid, the device does not check for parity error in subsequent bytes; ignores the entire packet and waits until STOP or next Repeat Start operation.
4. Repeat Start or Repeat Start with 7'h7E.

Table 43 DEVCTRL CCC Command Definition

Parameter	Definition
AddrMask[2:0]	Broadcast, Unicast or Multicast Command Selection 000 = Unicast Command; GD30TP139A device responds if DevID[6:0] field matches with GD30TP139A device's own 7-bit address (4-bit LID + 3-bit HID) 011 = Multicast Command; GD30TP139A device and possible other device respond if DevID[6:3] field matches with GD30TP139A device's own 4-bit LID address 111 = Broadcast Command; All devices respond to this command All other encodings are reserved.
StartOffset[1:0]	Only applicable if RegMod = '0' Identifies the starting Byte (Byte 0 or Byte 1 or Byte 2 or Byte 3) for DEVCTRL CCC. Host can start at any Byte (from Byte 0 to Byte 3) and has continuous access to next byte until STOP operation. If Byte 3 is reached, the Host is responsible for applying STOP operation. 00 = Byte 0 01 = Byte 1 10 = Byte 2 11 = Byte 3
PEC BL[1:0]	Only applicable if RegMod = '0' and PEC function is enabled. Identifies the burst length just for this DEVCTRL CCC. The device uses the setting in this field to know when the PEC byte is expected after the data bytes. 00 = 1 Byte 01 = 2 Byte 10 = 3 Byte 11 = 4 Byte
RegMod	Identifies if DEVCTRL is going to be used for General Registers as identified in Byte 0 to Byte 3 or device specific address offset register. 0 = Access to General Registers in Byte 0 to Byte 3 (i.e. StartOffset[1:0] = Valid) 1 = Device Specific Offset Address (i.e. StartOffset[1:0] and PECBL[1:0] is a don't care and does not apply). The Host shall NOT use RegMod = '1' with Broadcast Command if there are different types of devices on the I3C Basic bus.
DevID[6:0]	Identifies 7-bit device address. Device responds to DEVCTRL CCC data packet depending on AddrMask[2:0]. If AddrMask[2:0] = '111', DevID[6:0] is a don't care and device always responds. If AddrMask[2:0] = '000', DevID[6:0] must match for device to respond. If AddrMask[2:0] = '011', DevID[6:3] must match for device to respond. DevID[2:0] is don't care. For any other codes for AddrMask[2:0], the device always NACKs.

Table 44 DEVCTRL CCC Data Payload Definition

Byte #	Bit #	Function	Definition	Comment
Byte 0	[7]	PEC Enable	0 = Disable 1 = Enable	MR18[7] is updated
	[6]	Parity Disable	0 = Disable 1 = Enable	MR18[6] is updated
	[5:2]	RFU	RFU	
	[1]	RSVD	RSVD	GD30TP139A device always ignores this bit.
	[0]	RFU	RFU	
Byte 1	[7:4]	RFU	RFU	
	[3]	Global and IBI Clear	0 = No Action 1 = Clear All Event and pending IBI[1]	MR27[7] is updated.
	[2:0]	RFU	RFU	
Byte 2	[7:0]	RFU	RFU	
Byte 3	[7:0]	RFU	RFU	

1. After Target device clears the event, the device can still have certain registers set to '1' if the event is still present in which case, the device will generate an IBI again at the next opportunity.

DEVCTRL CCC Examples - RegMod = '0'

Table 45 shows an example of DEVCTRL CCC data packet. It assumes that all devices on the bus are already in I3C Basic mode with PEC function disabled and parity function enabled. In this example, the Host uses DEVCTRL CCC as Multicast command. Host sends Multicast command to all devices with 4-bit LID code of '1001' on I3C Basic bus to do VR Enable followed by all devices with 4-bit LID code of '0110' to disable parity function. The Host sends AddrMask = '011' to indicate Multicast command with DevID[6:3] match; StartOffset = '00' to indicate starting Byte 0 and RegMod = '0' to indicate general register. Upon receiving this command, all devices with DevID[6:3] that matches to '1001' will do the VR Enable command and DevID[6:3] that matches to '0110' with disable the parity function.

Table 45 DEVCTRL CCC Example - Multicast Command to '1001' and '0110' Devices

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S	1	1	1	1	1	1	0	W=0	A[1]	
	0x62 (Broadcast)								T	
	011			00		00		0	T	
	1001 000							0	T	
	0000 0010								T	
Sr	1	1	1	1	1	1	0	W=0	A[1]	
	0x62 (Broadcast)								T	
	011			00		00		0	T	
	0110 000							0	T	
	0100 0000								T	P

1. See Figure 11 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation.

Table 46 shows an example of DEVCTRL CCC data packet. It assumes that all devices on the bus are already in I3C Basic mode with PEC function disabled and parity function enabled. In this example, the Host uses DEVCTRL CCC as Broadcast command to enable PEC function. The Host sends AddrMask = '111' to indicate Broadcast command; StartOffset = '00' to indicate starting Byte 0 and RegMod = '0' to indicate general register. Upon receiving this command, all devices will enable PEC function.

Table 46 DEVCTRL CCC Example - Broadcast Command to all Devices

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S	1	1	1	1	1	1	0	W=0	A[1]	
	0x62 (Broadcast)								T	
	111			00		00		0	T	
	0000 000							0	T	
	1000 0000								T	P

1. See Figure 11 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation.

Table 47 shows an example of DEVCTRL CCC data packet. It assumes that all devices on the bus are already in I3C Basic mode with PEC function disabled and parity function enabled. In this example, the Host uses DEVCTRL CCC as Unitcast command to enable VR on DIMM5. The Host sends AddrMask = '000' to indicate Unicast command; StartOffset = '00' to indicate starting Byte 0 and RegMod = '0' to indicate general register. Upon receiving this command, PMIC on DIMM5 will enable its regulator.

Table 47 DEVCTRL CCC Example - Unicast Command to PMIC on DIMM5

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S	1	1	1	1	1	1	0	W=0	A1	
	0x62 (Broadcast)								T	
	000			00		00		0	T	
	1001 101							0	T	
	0000 0010								T	P

1. See Figure 11 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation.

DEVCTRL CCC Examples - RegMod = '1'

Table 48 shows an example of DEVCTRL CCC data packet for the purpose of configuring device specific address offset register. It assumes that all devices on the bus are already in I3C Basic mode with PEC function enabled and parity function enabled. In this example, the Host sends Multicast command to all devices with 4-bit LID code of '0010' on the I3C Basic bus to write to address offset of 0x1C and 0x1D with data 0xFF and 0x55 respectively followed by all devices with 4-bit LID of '1001' on the I3C Basic bus to write to address offset of 0x15 with data 0x78.

The PEC calculation starts with Start or Repeat Start operation but does not include 7'h7E with W=0 byte in PEC calculation.

Table 48 DEVCTRL CCC Example - Multicast Command to '0010' and '1001' Devices

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S	1	1	1	1	1	1	0	W=0	A[1]	
	0x62 (Broadcast)								T	
	011			00		00		1	T	
	0010 000							0	T	
	0001 1100 (address offset 0x1C)								T	
	0010 0000 (CMD field = 2 bytes of data)								T	
	1111 1111 (data)								T	
	0101 0101 (data)								T	
	PEC								T	
Sr	1	1	1	1	1	1	0	W=0	A[1]	
	0x62 (Broadcast)								T	
	011			00		00		1	T	
	1001 000							0	T	
	0001 0101 (address offset 0x15)								T	
	0000 0000 (CMD field = 1 byte of data)								T	
	0111 1000 (data)								T	
	PEC								T	P

Table 49 shows an example of DEVCTRL CCC data packet for the purpose of configuring device specific address offset register. It assumes that all devices on the bus are already in I3C Basic mode with PEC function disabled and parity function enabled. In this example, the Host sends Multicast command to all devices with 4-bit LID code of '1001' on the I3C Basic bus to write to address offset of 0x13 with data 0xFF and it continues to write data 0x01 to the next address.

Table 49 DEVCTRL CCC Example - Multicast Command to '1001' Devices

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S	1	1	1	1	1	1	0	W=0	A[1]	
	0x62 (Broadcast)								T	
	011			00		00		1	T	
	1001 000							0	T	
	0001 0011 (address offset 0x13)								T	
	1111 1111 (data)								T	
	0000 0001 (data)								T	P

1. See Figure 11 to see how the transition occurs from Target Open Drain (ACK) to Host Push Pull Operation.

6.8 In Band Interrupt (IBI)

In I²C mode, in band interrupt function is not supported. Only I3C Basic mode supports in band interrupt function.

6.8.1 Enabling and Disabling In Band Event Interrupt Function

By default, all interrupt sources are disabled (i.e. set to '0'). The Host may enable following interrupts in the GD30TP139A device. Once enabled, the GD30TP139A device sends an IBI when that event occurs.

1. Error Interrupt Enable in *MR27*[4]:

- When *MR27*[4] = '1', the device sends the IBI at next available opportunity when any of the register bit in *MR52*[1:0] is set to '1' and sets *MR48*[7] = '1' and updates Pending Interrupt Bits [3:0] = '0001' for GETSTATUS CCC.
- When *MR27*[4] = '0', the device does not send the IBI regardless of the register bit status in *MR52*[1:0]. However, the device does set *MR48*[7] = '1' and updates Pending Interrupt Bits [3:0] = '0001' for GETSTATUS CCC.

2. Temperature Sensor Interrupt Enable in *MR27*[3:0]: The Host can set any combination of register bits to '1':

- When any of the register bits in *MR27*[3:0] = '1' and if *MR27*[4] = '1', the device sends the IBI at next available opportunity when the corresponding register bit in *MR51*[3:0] is set to '1' and sets *MR48*[7] = '1' and updates Pending Interrupt Bits [3:0] = '0001' for GETSTATUS CCC.
- When any of the register bits in *MR27*[3:0] = '0' or *MR27*[4] = '0', the device does not send the interrupt regardless of the corresponding register bit status in *MR51*[3:0]. However, the device does set *MR48*[7] = '1' and updates Pending Interrupt Bits [3:0] = '0001' for GETSTATUS CCC if any of the bits in *MR27*[3:0] = '1' and *MR27*[4] = '0'.

6.8.2 Mechanics of Interrupt Generation

Event interrupts may be generated by the local device if IBI is enabled. When there is a pending interrupt (i.e. *MR48*[7] = '1) and if IBI is enabled (i.e. *MR27*[4] = '1'), the GD30TP139A device requests an interrupt after detecting START condition by transmitting its 7-bit binary address (LID bits followed by HID bits) followed by R/W = '1' on the SDA bus serially (synchronized by SCL falling transitions).

If GD30TP139A device detects no START condition but if the I3C Basic bus (SDA and SCL) has been inactive (no edges seen) for t_{AVAL} period, then the GD30TP139A device may assert SDA low by t_{IBI_ISSUE} time to request an interrupt. When the GD30TP139A device requests an interrupt, the Host toggles the SCL. The GD30TP139A device transmits its 7-bit binary address (LID bits followed by HID bits) followed by R/W bit = '1' to the Host.

When the GD30TP139A device requests an interrupt, the Host may take one of the two actions below.

- The Host sends ACK on 9th bit to accept the interrupt request. At this point, if the GD30TP139A device confirms that it has won the arbitration, the GD30TP139A device transmits the IBI payload as shown in Table 50 and Table 51 for PEC disabled and PEC enabled configuration, respectively. See Figure 16. It just shows only first two data bits of the MDB byte to illustrate the timing. The interrupt payload contains MDB followed by 8-bit register contents of MR51 and *MR52* in order. The Host then issues the STOP command. Note the timing waveform in Figure 16. The Host then accepts the IBI payload if it sends an ACK on 9th bit to accept the interrupt request. The Host can interrupt the IBI payload at 'T'. If Host stops the IBI payload at 'T' bit in the middle of payload, the GD30TP139A device retains the IBI status flag *MR48*[7] = '1' and Pending Interrupt Bits [3:0] internally and waits for the next opportunity to request an interrupt. If the GD30TP139A device

successfully transmits the entire IBI payload, it then clears IBI status flag *MR48*[7] = '0' and Pending Interrupt Bits [3:0] = '0000' on its own and does not request for an IBI again unless there is another different event occurs; for another same event, the device does not request for an IBI.

- The Host sends NACK on the 9th bit as shown in Figure 17 followed by a STOP command. In this case, the GD30TP139A device does not transmit the IBI payload and waits for the next opportunity to request an interrupt. At this point, though Host sent a NACK, it does have a knowledge of which GD30TP139A device sent the IBI request. The GD30TP139A retains the IBI status flag *MR48*[7] = '1' and Pending Interrupt Bits [3:0] = '0001'.

Table 50 GD30TP139A IBI Payload Packet; PEC is Disabled

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S	0	X	1	0	HID			R=1	A[1]	
	MDB = 0x00								T=1	
	MR51 [7:0]								T=1	
	MR52 [7:0]								T=0[2]	P

1. See Figure 12 to see how the transition occurs from Host Open Drain (ACK) to Target Push Pull Operation (1st bit of MDB, bit [7]).
2. See Figure 15 to see how Target device ends the operation followed by Host STOP operation.

Table 51 GD30TP139A IBI Payload Packet; PEC is Enabled

Start	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	A/N/T	Stop
S	0	X	1	0	HID			R=1	A[1]	
	MDB = 0x00								T=1	
	MR51 [7:0]								T=1	
	MR52 [7:0]								T=1	
	PEC								T=0[2]	P

1. See Figure 12 to see how the transition occurs from Host Open Drain (ACK) to Target Push Pull Operation (1st bit of MDB, bit [7]).
2. See Figure 15 to see how Target device ends the operation followed by Host STOP operation.

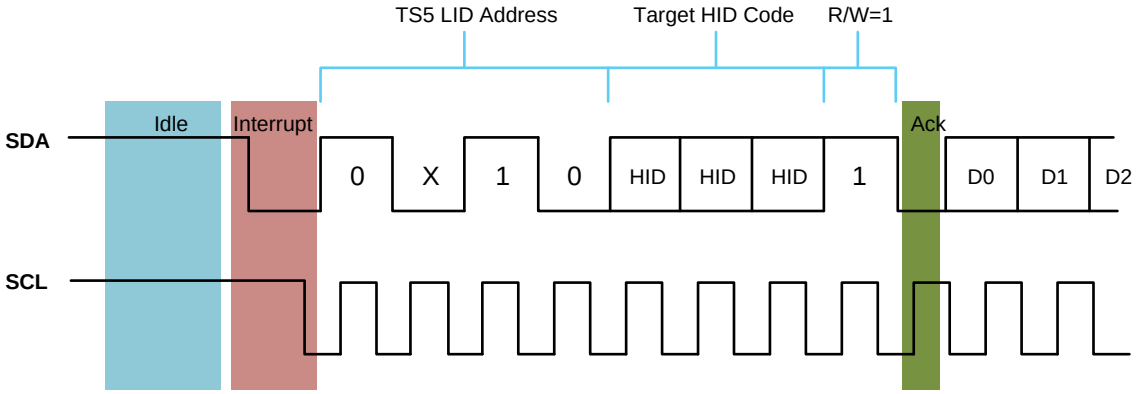


Figure 16 GD30TP139A Interrupt; Host Ack Followed by GD30TP139A Device IBI Payload

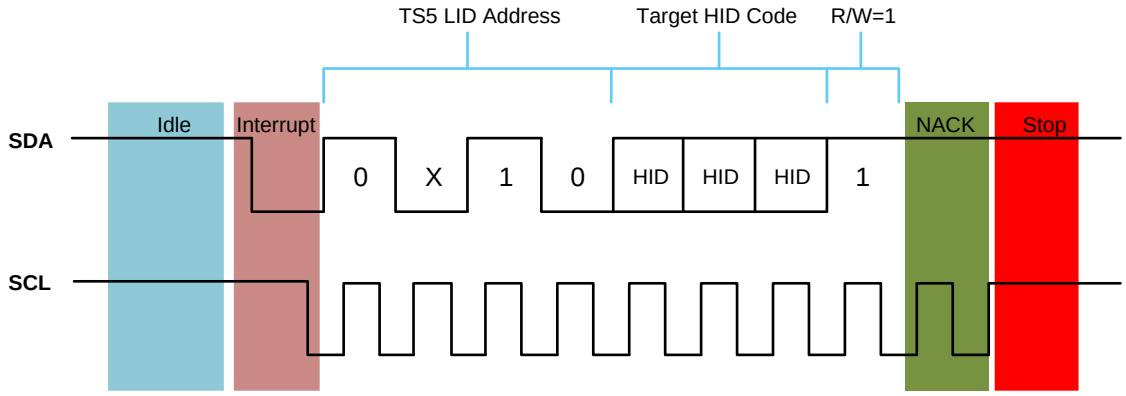


Figure 17 GD30TP139A Interrupt; Host NACK followed by STOP

6.8.3 Interrupt Arbitration

As there are multiple devices I3C Basic bus, multiple devices may request an interrupt when the Host I3C Basic bus is inactive for t_{AVAL} period. Arbitration process is required.

For DDR5 DIMM application environment, there could be up to total of 13 difference devices including the GD30TP139A on I3C Basic bus.

On a typical DDR5 DIMM application environment, all devices have the same 3-bit HID code. Hence the arbitration is always won by the lowest 4-bit LID code. For example, if one GD30TP139A Target device has LID code of '0010' and other device (PMIC) has a LID code of '1001', through the arbitration process, the LID code of GD30TP139A '0010' wins. The other device (PMIC) with a LID code of '1001' must release the bus and wait for next opportunity to request an interrupt. Table 52 shows the arbitration priority based on the LID code for the local devices. The Green color cells in Table 52 are the likely devices that will be on a standard DDR5 RDIMM or DDR5 LRDIMM.

Table 52 Interrupt Arbitration - Among Local Target Devices

Device	Target Device LID Code	Target Device HID Code	Arbitration Priority
N/A	0000	N/A	N/A
RFU	0001	111	1
TS0	0010	111	2
RFU	0011	111	3
RFU	0100	111	4
RFU	0101	111	5
TS1	0110	111	6
RFU	0111	111	7
PMIC1	1000	111	8
PMIC0	1001	111	9
SPD Hub	1010	HID	N/A
RCD	1011	111	10
PMIC2	1100	111	11
RFU	1101	111	12
RFU	1110	111	13
N/A	1111	N/A	N/A

In an uncommon but possible scenario would be that at the exact same time as when the Hub or local Target devices (i.e. GD30TP139A) are requesting an interrupt, the Host is starting an operation to the Hub or local Target devices (i.e. GD30TP139A). When this happens, Host also gets involved in the arbitration process along with the Hub or the local Target devices (i.e. GD30TP139A). During the arbitration phase, there will be always only one winning device and it could be either the Hub or the local Target device (i.e. GD30TP139A) or the Host.

If the Host wins during the arbitration phase, it continues with normal operation. The losing Hub or local Target device waits for next opportunity to send an interrupt.

If the Host loses during the arbitration phase, it must let go of the bus. When the Host loses during the arbitration, the Host must let the Hub or local Target device (i.e. GD30TP139A) finish sending its 4-bit LID code followed by 3-bit HID code followed by R/W = '1'. At this point, during the 9th bit, the Host has two options to take the action as noted below:

- Host sends an ACK to accept the interrupt and hence accepts the IBI payload from the winning SDP5 Hub or local Target device (i.e. GD30TP139A). After the IBI payload, the Host issues STOP operation.
- Host sends an NACK followed by STOP operation.

In a rare but still possible scenario would be that at the exact same time as when the GD30TP139A device is requesting an interrupt, the Host is starting an operation to the same GD30TP139A. When this happens, neither Host nor the GD30TP139A knows it is a winner until the 8th bit and Host always wins. This is because, the GD30TP139A sends R=1 (8th bit) during the interrupt. The Host sets W=0 (8th bit) during the operation. As a result, the Host wins and the GD30TP139A must let go of the bus and wait for the next opportunity to send an interrupt.
In an extreme rare but still possible scenario would be that at the exact same time as when the GD30TP139A device is requesting an interrupt, the Host is requesting a read operation with default read address pointer mode

to the same GD30TP139A device. When this happens, there is no winning device. This is the only time there is no winning device. This is because, the GD30TP139A device sends R=1 (8th bit) during the interrupt and Host also sends R=1 for read request with default read address pointer mode. As a result, there is no winner because Host is waiting for GD30TP139A to ACK and GD30TP139A is waiting for Host to ACK. In this case, neither Host nor GD30TP139A will ACK. Since there is no ACK (i.e. NACK) by either device, the Host must time out and repeat the read request with Repeat Start. When Host repeats the read request with Repeat Start, the GD30TP139A does not send an interrupt because of Repeat Start.

6.8.4 Clearing Device Status and IBI Status Registers

The GD30TP139A device provides the IBI status in *MR48*[7] by setting it to '1'. The GD30TP139A device clears the IBI status register *MR48*[7] to '0' automatically when it sends a complete IBI (including payload and without interruption) and it also clears Pending Interrupt Bits [3:0] to '0000'. Once IBI status register is cleared, the GD30TP139A does not request for an IBI again unless another event occurs.

The GD30TP139A device provides the device status in MR51 and *MR52* registers. The status information in MR51 and *MR52* are latched and remains set even after the GD30TP139A device sends IBI payload and clears the IBI status register *MR48*[7] to '0'. The Host must explicitly clear the status register through Clear command by writing '1' for appropriate status or by issuing a Global Clear command.

After Host issues Clear command, if the condition is no longer present, the GD30TP139A device clears the appropriate status register, clears the IBI status register to '0' and Pending Interrupt Bits [3:0] to '0000' even if the GD30TP139A device has not sent the IBI. After Host issues Clear command, if the condition is still present, the device will again set the appropriate status register, sets the IBI status register to '1' and Pending Interrupt Bits [3:0] to '0001' even if the device has already sent the IBI and entire IBI payload.

6.9 Error Check Function

6.9.1 Packet Error Check (PEC) Function

In I²C mode, packet error checking is not supported. Only I3C Basic mode supports packet error checking.

The GD30TP139A device implements an 8-bit Packet Error Code (PEC) which is appended at the end of all transactions if PECs is enabled through DEVCTRL CCC. The PEC is a CRC-8 value calculated on all the message bytes except for START, REPEATED START, STOP conditions or 'T' bits, ACK and NACK and IBI header (7'h7E followed by W=0) bits.

The polynomial for CRC-8 calculations is:

$$C(X) = X^8 + X^2 + X^1 + 1$$

The seed value for PEC function is all zero.

When Host calculates PEC for GD30TP139A device, it includes LID and HID bits followed by R/W bit.

6.9.2 Parity Error Check Function

In I²C mode, parity error checking is not supported except for supported CCCs. Only I3C Basic mode supports parity error checking.

By default, when GD30TP139A device is put in I3C Basic mode, parity function is automatically enabled. The Host can disable the function after it is enabled. Host can only disable the parity function with DEVCTRL CCC (RegMod = '0'). When parity function is disabled, the GD30TP139A device simply ignores the 'T' bit information from the Host. The Host may actually choose to compute the parity and send that information during 'T' bit or simply drive static low or high in 'T' bit.

The GD30TP139A device implements ODD parity. If an odd number of bits in the byte are '1', the parity bit value is '0'.

If even number of bits in the byte are '1', the parity bit value is '1'. The Host computes the parity and sends it during 'T' bit.

6.9.3 Packet Error Check and Parity Error Handling

There are two types of error checking done by the GD30TP139A device: parity error checking and packet error checking. By default, the parity error checking is always enabled and packet error checking is disabled when the GD30TP139A device is put in I3C Basic mode. The Host may enable the packet error checking at any time. The parity error is calculated for each byte. The Host sends parity error information in 'T' bit.

I3C Basic defines TE0, TE1, TE2, TE3, TE4, TE5, TE6 error detection for Target devices. Only TE1 and TE2 error detection is supported by the GD30TP139A for parity checking. All other errors are not supported and not applicable.

Write Command Data Packet Error Handling - PEC Disabled

The GD30TP139A device checks for the parity error for each byte in a packet except for the device select code byte that it receives from the Host as shown in Table 8.

Write command - if no parity error:

- The GD30TP139A device executes the command.

Write command - if parity error:

- The GD30TP139A device discards the byte in the packet that had a parity error.
- The GD30TP139A device discards all subsequent bytes in that packet until the STOP operation. The GD30TP139A device may or may not check parity for all sub-sequent bytes in that packet.
- Note that as the packet contains more than one byte, if first byte had no parity error but the second byte had a parity error, the GD30TP139A device may or may not execute the first byte operation but second byte and all subsequent bytes operations are discarded.
- The GD30TP139A device sets the *MR52*[0], *MR48*[7] and P_Err in GETSTATUS CCC to '1'; updates Pending Interrupt Bits [3:0] to '0001' and waits for the next opportunity to send an in band interrupt if IBI is enabled.

Read Command Data Packet Error Handling - PEC Disabled

The GD30TP139A device checks for parity error for each byte in a packet except for the device select code byte that it receives from the Host prior to Repeat Start as shown in Table 12.

The GD30TP139A device does not compute the parity when it sends the data to the Host. The Host does not check for parity error for the bytes that GD30TP139A device sends. The GD30TP139A device sends Continuous ('1') or Stop ('0') information during 'T' bit.

Read Command - If no parity error:

- The GD30TP139A sends ACK back to the Host when Host performs Start Repeat operation.
- The GD30TP139A device executes the command and sends the data as shown in Table 12.

Read Command - If parity error:

- The GD30TP139A device discards the byte in the packet that had a parity error.
- The GD30TP139A device discards second byte in that packet if the parity error occurred in first byte. The GD30TP139A device may or may not check the parity for second byte in that packet.
- The GD30TP139A sends NACK back to the Host when Host performs a Start Repeat operation. This is shown in the RED color cell in Table 12. The NACK represents either a parity error in one of the two bytes or that GD30TP139A is not able to start the read operation. The Host may re-try Repeat Start again. The Host may do the Repeat Start as many times as it may desire. If the GD30TP139A device NACKs due to parity error in a previous byte from the Host, it will always NACK regardless of how many times Host tries Repeat Start.
- The GD30TP139A does not send the data shown in Table 12 and instead expects Host to perform STOP operation.
- The GD30TP139A device sets *MR52*[0] and *MR48*[7] and P_Err in GETSTATUS CCC to '1'; updates Pending Interrupt Bits [3:0] in GETSTATUS to '0001' and waits for the next opportunity to send an in band interrupt if IBI is enabled.

Write Command Data Packet Error Handling - PEC Enabled

The GD30TP139A device checks for the parity error for each byte in a packet except for the device select code byte that it receives from the Host as shown in Table 9. Further, the GD30TP139A device checks for the packet error for the entire packet (from Start condition until last byte of Data) that receives from the Host as shown in Table 9.

Write command - if no parity error:

- The GD30TP139A device waits for the entire packet. If no error in packet, the GD30TP139A device executes the command. If there is an error in the packet, the GD30TP139A device discards the entire packet and does not execute that packet and waits for STOP, sets the *MR52*[1] and *MR48*[7] to '1' and PEC_Err in GETSTATUS CCC to '1' and updates Pending Interrupt Bits [3:0] in GETSTATUS CCC to '0001' and waits for the next opportunity to send in band interrupt if IBI is enabled.

Write command - if parity error:

- The GD30TP139A device discards that byte and the entire packet until STOP operation.
- The GD30TP139A device sets *MR52*[0] and *MR48*[7] and P_Err in GETSTATUS CCC to '1'; updates Pending Interrupt Bits [3:0] in GETSTATUS CCC to '0001' and waits for the next opportunity to send in band interrupt if IBI is enabled.
- The GD30TP139A device may or may not check the error for the packet. If the GD30TP139A device checks for the packet error, likely it will detect an error in the packet and the device may also set *MR52*[1] and PEC_Err in GETSTATUS CCC to '1' as well.

Read Command Data Packet Error Handling - PEC Enabled

The GD30TP139A device checks for the parity error for each byte in a packet except for the device select code byte that it receives from the Host prior to Repeat Start as shown in Table 13.

The GD30TP139A device does not compute the parity when it sends the data to the Host. It does not check for parity error for the bytes shown in Table 13. The GD30TP139A device sends Continuous ('1') or Stop ('0') information during 'T' bit when GD30TP139A device is sending the read data.

The GD30TP139A device checks for the PEC error for a packet that it receives from the Host from Start condition to Repeat Start condition (from first device select code followed by the address offset and CMD byte).

The GD30TP139A device computes the packet error code for the entire packet starting with Repeat Start (device select code and the data GD30TP139A device transmits back to Host)

Read command - If no parity error and no PEC error:

- The GD30TP139A sends ACK back to the Host when Host performs a Start Repeat operation.
- The GD30TP139A device executes the command and sends the data as shown in **Table 13**.
- The GD30TP139A computes PEC for the bytes (from Start condition to PEC byte prior to Repeat Start) shown in Table 13.

Read command - if parity error or PEC error:

- The GD30TP139A device discards the byte in the packet that had a parity error.
- The GD30TP139A device discards second byte in that packet if a parity error occurred in first byte. The SPD5
- Hub device may or may not check parity for the second byte in that packet.
- The GD30TP139A device discards the packet if there is a PEC error.

- The GD30TP139A sends NACK back to the Host when Host performs Start Repeat operation. This is shown in the RED color cell in Table 13. The NACK represents either PEC error or a parity error in one of the three bytes or that GD30TP139A is not able to start the read operation. The Host may re-try Repeat Start again. The Host may do the Repeat Start as many times it may desire. The PEC calculation by GD30TP139A device only includes device select code of the ACK responses of the Repeat Start operation. In other words, if there are more than one Repeat Start operation, the GD30TP139A device includes the device select of only the last Repeat Start from the Host when it ACKs in PEC calculation and other NACK responses of the device select codes of the Repeat Start are not included in PEC calculation. If the GD30TP139A device NACKs due to PEC error or a parity error in previous bytes from Host, it will always NACK regardless of how many times Host tries Repeat Start.
- The GD30TP139A does not send any data shown in Table 13 and instead expects Host to perform STOP operation.
- The GD30TP139A device sets *MR52*[0] and *MR48*[7] and P_Err in GETSTATUS CCC to '1' for parity error and *MR52*[1] and *MR48*[7] and PEC_Err in GETSTATUS CCC to '1' for PEC error. Further, the GD30TP139A updates Pending Interrupt Bits [3:0] in GETSTATUS CCC to '0001' and waits for the next opportunity to send an in band interrupt if IBI is enabled.

CCC Packet Error Handling

Parity error and PEC error detected in a CCC packet are handled in the same way as described for normal Read/Write operations.

Error Reporting

All error conditions including PEC error check and parity error check detected by the GD30TP139A devices are captured in *MR51* and *MR52* registers.

There are three different possible ways error information can be communicated to the Host.

1. The Host makes the read request to *MR51* and *MR52* registers.
2. The Host starts any transaction with 7'h7E IBI header (only applicable in I3C mode).
3. The GD30TP139A device sends in band interrupt if enabled, when its SCL and SDA input has been idle for t_{AVAL} time (only applicable in I3C Basic mode).

6.10 Command Truth Table

The command truth table as shown in Table 53 only applies in I3C Basic mode with PEC enabled. In I²C mode and I3C Basic mode with PEC disabled, the command truth table does not apply.

Table 53 For I3C Basic Mode Only w/ PEC Enabled - Command Truth Table

Command	Command Name	Command Code	RW	Address
		2nd Byte Bits [7:5]	2nd Byte Bit [4]	1st Byte Bits [5:0]
Write 1 Byte to Register	W1R	000	0	V
Read 1 Byte from Register	R1R		1	V
Write 2 Byte to Register	W2R	001	0	V
Read 2 Byte from Register	R2R		1	V
Reserved	RSVD	010 to 111	RSVD	RSVD

7 Registers

7.1 Register Attribute Definition

All volatile registers have Base Attributes as defined in Table 54. Some register attributes are further modified with Attribute Modifiers, as defined in Table 55.

The volatile register space has a continuous address.

Table 54 Register Base Attributes

Attribute	Abbreviation	Description
Read Only	R	This bit can be read by software. Writes have no effect.
Read/Write	RW	This bit can be read or written by software.
Write Only	W	This bit can only be written by software.
Reserved	RV	This bit is reserved for future expansion and its value must not be modified by software. The bit will return ‘0’ when read. When writing this bit, software must preserve the value read unless otherwise indicated.

Table 55 Register Attribute Modifier

Attribute	Abbreviation	Description
Write 1 Only	1O	This bit can only be set (i.e. write ‘1’) but not reset (i.e. write ‘0’)
Protected	P	This bit is protected by the password registers TBD. This bit cannot be written to unless the password code has been written into the password registers
Persistent	E	Persistent.

7.2 Register Map

Table 56 Register Map

Register Name	Register Address (Hex)	Attribute	Description
MR0	0x00	ROE	Device Type; Most Significant Byte
MR1	0x01	ROE	Device Type; Least Significant Byte
MR2	0x02	ROE	Device Revision
MR3	0x03	ROE	Vendor ID Byte 0
MR4	0x04	ROE	Vendor ID Byte 1
MR5 to MR6	0x05 to 0x06	RV	Reserved
MR7	0x07	RO	Device Configuration; HID
MR8 to MR17	0x08 to 0x11	RV	Reserved
MR18	0x12	RO, RW	Device Configuration
MR19	0x13	1O	Clear Register MR51 Temperature Status Command
MR20	0x14	1O	Clear Register MR52 Error Status Command
MR21 to MR25	0x15 to 0x19	RV	Reserved
MR26	0x1A	RW	TS Configuration
MR27	0x1B	1O, RO, RW	Interrupt Configurations
MR28	0x1C	RW	TS Temperature High Limit Configuration - Low Byte
MR29	0x1D	RW	TS Temperature High Limit Configuration - High Byte
MR30	0x1E	RW	TS Temperature Low Limit Configuration - Low Byte
MR31	0x1F	RW	TS Temperature Low Limit Configuration - High Byte
MR32	0x20	RW	TS Critical Temperature High Limit Configuration – Low Byte
MR33	0x21	RW	TS Critical Temperature High Limit Configuration – High Byte
MR34	0x22	RW	TS Critical Temperature Low Limit Configuration – Low Byte
MR35	0x23	RW	TS Critical Temperature Low Limit Configuration – High Byte
MR36 to MR47	0x24 to 0x2F	RV	Reserved for Device Configuration Type of Registers
MR48	0x30	RO	Device Status
MR49	0x31	RO	TS Current Sensed Temperature - Low Byte
MR50	0x32	RO	TS Current Sensed Temperature - High Byte
MR51	0x33	RO	TS Temperature Status
MR52	0x34	RO	Misc. Error Status

7.3 Thermal Sensor Registers Read Out Mechanism

All thermal registers are sixteen bit quantities stored in two consecutive registers; low byte first and then high byte. Five bits are reserved for future use. Reserved bits are Read only bits and must be set to '0' when Host writes to low and high byte. The device returns '0' in reserved bits when Host reads from the low and high byte. Remaining eleven bits in these paired registers form a signed value of multiples of 0.25 ranging from -256.00 to + 255.75. Units for all thermal registers are °C.

The format of each pair of thermal registers is shown in Table 57 below.

Table 57 Thermal Register - Low Byte and High Byte

Register		bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
MR49	Low Byte	8	4	2	1	0.5	0.25	RSVD	RSVD
MR50	High Byte	RSVD	RSVD	RSVD	Sign	128	64	32	16

Bit 'Sign' decides whether the readout temperature is positive or negative. The examples (reserved bits in grey, sign bit highlighted in blue) is shown in Table 58 below.

Table 58 Thermal Register Examples

High Byte	Low Byte	Value	Unit
0000 0101	1111 0000	+ 95.00	°C
0000 0101	0101 0000	+ 85.00	°C
0000 0100	1011 0000	+ 75.00	°C
0000 0000	0001 0000	+ 1.00	°C
0000 0000	0000 1100	+ 0.75	°C
0000 0000	0000 1000	+ 0.50	°C
0000 0000	0000 0100	+ 0.25	°C
0000 0000	0000 0000	0.00	°C
0001 1111	1111 1100	- 0.25	°C
0001 1111	1111 1000	- 0.50	°C
0001 1111	1111 0100	- 0.75	°C
0001 1111	1111 0000	- 1.00	°C
0001 1101	1000 0000	- 40.00	°C

7.4 Register Description

Table 59 MR0

Addr	MR0		Device Type; Most Significant Byte[1]
Bits	Attr	Default	Description
7:0	ROE	-	MR0[7:0]: MSB_DEV_TYPE Device Type - These are hard coded. 0x51: Temperature Sensor without Serial Number 0x52: Temperature Sensor with Serial Number

1. The code in this register is used in conjunction with any device type in *MR1* register.

Table 60 MR1

Addr	MR1		Device Type; Least Significant Byte[1]
Bits	Attr	Default	Description
7:0	ROE	0x11	MR1[7:0]: LSB_DEV_TYPE Device Type – Temperature Sensor - These are hard coded. 0x11: Grade A Temperature Sensor

1. The code in this register is used in conjunction with any device type in *MR0* register.

Table 61 MR2

Addr	MR2		Device Revision
Bits	Attr	Default	Description
7:6	RV	0	MR2[7:6]: Reserved
5:4	ROE	00	MR2[5:4]: DEV_REV_MAJOR Major Revision 00 = Revision 1 01 = Revision 2 10 = Revision 3 11 = Revision 4
3:1	ROE	000	MR2[3:1]: DEV_REV_MINOR Minor Revision 000 = Revision 0 001 = Revision 1 010 = Revision 2 … 111 = Revision 8
0	RV	0	MR2[0]: Reserved

Table 62 MR3

Addr	MR3		Vendor ID Byte 0
Bits	Attr	Default	Description
7:0	ROE	0x0E	MR3[7:0]: VENDOR_ID_BYTE0 Vendor ID Byte 0

Table 63 MR4

Addr	MR4		Vendor ID Byte 1
Bits	Attr	Default	Description
7:0	ROE	0x8F	MR4[7:0]: VENDOR_ID_BYTE1 Vendor ID Byte 1

Table 64 MR7

Addr	MR7		Device Configuration; HID[1]
Bits	Attr	Default	Description
7:4	RV	0	MR7[7:4]: Reserved
3:1	RO	111	MR7[3:1]: DEV_HID_CODE Device HID Code. The GD30TP139A device responds to unique 7-bit address as formed by 4 bit LID code as in Table 4 and 3-bit HID code as configured in this register. This register is updated when SETHID CCC is registered by the GD30TP139A device or when GD30TP139A device goes through bus reset as described in Section 6.2.3.
0	RV	0	MR7[0]: Reserved

1. The write (or update) transaction to this register must be followed by STOP operation to allow the GD30TP139A device to update the setting.

Table 65 MR18

Addr	MR18		Device Configuration[1]
Bits	Attr	Default	Description
7	RO	0	MR18[7]: PEC_EN PEC Enable[2,3] 0 = Disable 1 = Enable
6	RO	0	MR18[6]: PAR_DIS Parity (T bit) Disable[3,4] 0 = Enable 1 = Disable
5	RO	0	MR18[5]: INF_SEL Interface Selection 0 = I^2C Protocol (Max speed of 1 MHz) 1 = I3C Basic Protocol[5]
4	RW	0	MR18[4]: DEF_RD_ADDR_POINT_EN Default Read Address Pointer Enable 0 = Disable Default Read Address Pointer (Address pointer is set by the Host)[6] 1 = Enable Default Read Address Pointer; Address selected by register bits [3:2]
3:2	RW	00	MR18[3:2]: DEF_RD_ADDR_POINT_START Default Read Pointer Starting Address[7] 00 = *MR49* 01 = Reserved 10 = Reserved 11 = Reserved
1	RW	0	MR18[1]: DEF_RD_ADDR_POINT_BL Burst Length for Read Pointer Address for PEC Calculation[7] 0 = 2 Bytes 1 = 4 Bytes
0	RV	0	MR18[0]: Reserved

1. The write (or update) transaction to this register must be followed by STOP operation to allow the GD30TP139A device to update the setting.
2. This register is only applicable if *MR18*[5] = '1'.
3. This register is updated when RSTDAA CCC is registered by GD30TP139A device or when GD30TP139A device goes through bus reset as described in Section 6.2.3.
4. This register is only applicable if *MR18*[5] = '1'. When Parity function is disabled, the GD30TP139A device simply ignores the 'T' bit information from the Host. The Host may actually choose to compute the parity and send that information in 'T' bit or simply drive static low or high in 'T' bit.
5. This register is automatically updated when SETAASA CCC or RSTDAA CCC is registered by the GD30TP139A device or when GD30TP139A device goes through bus reset as described in Section 6.2.3. This register can be read by the Host through normal Read operation but it cannot be written with normal write operation either in I^2C mode or I3C Basic

mode of operation. When this register is updated, it takes in effect when there is a next START operation (i.e. after STOP operation).

6. The setting in register *MR18*[3:1] is don't care.
7. This register is only applicable if *MR18*[4] = '1'. This register is only applicable if *MR18*[7, 4] = '11'.

Table 66 MR19

Addr	MR19		Clear Register Command[1]
Bits	Attr	Default	Description
7:4	RV	0	MR19[7:4]: Reserved
3	1O	0	MR19[3]: CLR_TS_CRIT_LOW Clear Temperature Sensor Critical Low Status 1 = Clear *MR51*[3] Register
2	1O	0	MR19[2]: CLR_TS_CRIT_HIGH Clear Temperature Sensor Critical High Status 1 = Clear *MR51*[2] Register
1	1O	0	MR19[1]: CLR_TS_LOW Clear Temperature Sensor Low Status 1 = Clear *MR51*[1] Register
0	1O	0	MR19[0]: CLR_TS_HIGH Clear Temperature Sensor High Status 1 = Clear *MR51*[0] Register

1. This entire register is self-clearing register after corresponding register is cleared.

Table 67 MR20

Addr	MR20		Clear Register Command[1]
Bits	Attr	Default	Description
7	1O	0	MR20[7]: CLR_SPD_BUSY_ERROR Clear Write or Read Attempt while SPD Device Busy Error Status 1 = Clear *MR52*[7] Register
6	1O	0	MR20[6]: CLR_WR_NVM_BLK_ERROR Clear Write Attempt to Protected NVM Block Error Status 1 = Clear *MR52*[6] Register
5	1O	0	MR20[5]: CLR_WR_NVM_PRO_REG_ERROR Clear Write Attempt to NVM Protection Register Error Status 1 = Clear *MR52*[5] Register
4:3	RV	0	MR20[4:2]: Reserved
1	1O	0	MR20[1]: CLR_PEC_ERROR Clear Packet Error Status 1 = Clear *MR52*[1] Register
0	1O	0	MR20[0]: CLR_PAR_ERROR Clear Parity Error Status 1 = Clear *MR52*[0] Register

1 This entire register is self-clearing register after corresponding register is cleared.

Table 68 MR26

Addr	MR26		Thermal Sensor Configuration
Bits	Attr	Default	Description
7:1	RV	0	MR26[7:1] Reserved
0	RW	0	MR26[0]: DIS_TS Disable Temperature Sensor[1] 0 = Enable thermal sensor 1 = Disable thermal sensor

1. If this bit is set to '1' and then reset to '0', the Host must wait minimum of t_{INIT} before accessing samples on the thermal sensor.

Table 69 MR27

Addr	MR27		Interrupt Configuration
Bits	Attr	Default	Description
7	1O	0	MR27[7]: CLR_GLOBAL Global Clear Event Status and In Band Interrupt Status[1,2] 1 = Clear *MR48*[7], *MR51*[3:0] and MR52[7:5, 1:0] Register
6:5	RV	0	MR27[6:5]: Reserved
4	RO	0	MR27[4]: IBI_ERROR_EN In Band Error Interrupt Enable for *MR52* Error Log[3] 0 = Disable; Errors logged in *MR52*[7:5, 1:0] registers do not generate an IBI to Host 1 = Enable; Errors logged in *MR52*[7:5, 1:0] registers generates an IBI to Host
3	RW	0	MR27[3]: IBI_TS_CRIT_LOW_EN In Band Error Interrupt Enable for Temperature Sensor Critical Low 0 = Disable; *MR51*[3] = '1' does not generate an IBI to Host 1 = Enable; *MR51*[3] = '1' and *MR27*[4] = '1' generates an IBI to Host
2	RW	0	MR27[2]: IBI_TS_CRIT_HIGH_EN In Band Error Interrupt Enable for Temperature Sensor Critical High 0 = Disable; *MR51*[2] = '1' does not generate an IBI to Host 1 = Enable; *MR51*[2] = '1' and *MR27*[4] = '1' generates an IBI to Host
1	RW	0	MR27[1]: IBI_TS_LOW_EN In Band Error Interrupt Enable for Temperature Sensor Low 0 = Disable; *MR51*[1] = '1' does not generate an IBI to Host 1 = Enable; *MR51*[1] = '1' and *MR27*[4] = '1' generates an IBI to Host
0	RV	0	MR27[0]: IBI_TS_HIGH_EN In Band Error Interrupt Enable for Temperature Sensor High 0 = Disable; *MR51*[0] = '1' does not generate an IBI to Host 1 = Enable; *MR51*[0] = '1' and *MR27*[4] = '1' generates an IBI to Host

1. This register is a self-clearing register after corresponding registers are cleared. Writing '0' in this register has no effect.
2. After this command is issued, the device does not generate an IBI for any pending event. But if new event occurs, the device does generate an IBI.
3. This register is automatically updated when ENEC CCC or DISEC CCC or RSTDAA CCC is registered by the GD30TP139A device or when GD30TP139A device goes through bus reset as described in Section 6.2.3. This register

can be read by the Host through normal read operation but cannot be written with normal write operation either in I²C mode or I3C Basic mode. When this register is updated, it takes effect when there is a next START operation (i.e. after STOP operation).

Table 70 MR28

Addr	MR28		Thermal Sensor High Limit Configuration - Low Byte[1,2]
Bits	Attr	Default	Description
7:0	RW	0x70	MR28[7:0]: TS_HIGH_LIMIT_LOW *MR28* and *MR29* - 16 bit thermal registers define the high limit for thermal sensor. See Table 57 *Thermal Register - Low Byte and High Byte*.

1. Critical temperature High Limit value must have a higher value than temperature High Limit (*MR28*[7:0] and *MR29*[7:0]).
2. The Reserved bits are Read Only bits. The Host must write '0' in reserved bits when writing and device always returns '0' from reserved bits when reads this byte.

Table 71 MR29

Addr	MR29		Thermal Sensor High Limit Configuration - High Byte[1,2]
Bits	Attr	Default	Description
7:0	RW	0x03	MR29[7:0]: TS_HIGH_LIMIT_HIGH *MR28* and *MR29* - 16 bit thermal registers define the high limit for thermal sensor. See Table 57 *Thermal Register - Low Byte and High Byte*.

1. Critical temperature High Limit value must have a higher value than temperature High Limit (*MR28*[7:0] and *MR29*[7:0]).
2. The Reserved bits are Read Only bits. The Host must write '0' in reserved bits when writing and device always returns '0' from reserved bits when reads this byte.

Table 72 MR30

Addr	MR30		Thermal Sensor Low Limit Configuration - Low Byte[1,2]
Bits	Attr	Default	Description
7:0	RW	0	MR30[7:0]: TS_LOW_LIMIT_LOW *MR30* and *MR31* - 16 bit thermal registers define the low limit for thermal sensor. See Table 57 *Thermal Register - Low Byte and High Byte*.

1 Critical temperature Low Limit value must have a lower value than temperature Low Limit (*MR30*[7:0] and *MR31*[7:0]).

2 The Reserved bits are Read Only bits. The Host must write '0' in reserved bits when writing and device always returns '0' from reserved bits when reads this byte.

Table 73 MR31

Addr	MR31		Thermal Sensor Low Limit Configuration - High Byte[1,2]
Bits	Attr	Default	Description
7:0	RW	0	MR31[7:0]: TS_LOW_LIMIT_HIGH *MR30* and *MR31* - 16 bit thermal registers define the low limit for thermal sensor. See Table 57 *Thermal Register - Low Byte and High Byte*.

1 Critical temperature Low Limit value must have a lower value than temperature Low Limit (*MR30*[7:0] and *MR31*[7:0]).

2 The Reserved bits are Read Only bits. The Host must write '0' in reserved bits when writing and device always returns '0' from reserved bits when reads this byte.

Table 74 MR32

Addr	MR32		Thermal Sensor Critical Temperature High Limit Configuration - Low Byte[1,2]
Bits	Attr	Default	Description
7:0	RW	0x50	MR32[7:0]: TS_CRIT_HIGH_LIMIT_LOW *MR32* and *MR33* - 16 bit thermal registers define the critical temperature high limit for thermal sensor. See Table 57 *Thermal Register - Low Byte and High Byte*.

1 Critical temperature High Limit value must have a higher value than temperature High Limit (*MR28*[7:0] and *MR29*[7:0]).

2 The Reserved bits are Read Only bits. The Host must write '0' in reserved bits when writing and device always returns '0' from reserved bits when reads this byte.

Table 75 MR33

Addr	MR33		Thermal Sensor Critical Temperature High Limit Configuration - High Byte[1,2]
Bits	Attr	Default	Description
7:0	RW	0x05	MR33[7:0]: TS_CRIT_HIGH_LIMIT_HIGH *MR32* and *MR33* - 16 bit thermal registers define the critical temperature high limit for thermal sensor. See Table 57 *Thermal Register - Low Byte and High Byte.*

1 Critical temperature High Limit value must have a higher value than temperature High Limit (*MR28*[7:0] and *MR29*[7:0]).

2 The Reserved bits are Read Only bits. The Host must write '0' in reserved bits when writing and device always returns '0' from reserved bits when reads this byte.

Table 76 MR34

Addr	MR34		Thermal Sensor Critical Temperature Low Limit Configuration - Low Byte[1,2]
Bits	Attr	Default	Description
7:0	RW	0	MR34[7:0]: TS_CRIT_LOW_LIMIT_LOW *MR34* and *MR35* - 16 bit thermal registers define the critical temperature low limit for thermal sensor. See Table 57 *Thermal Register - Low Byte and High Byte.*

1 Critical temperature Low Limit value must have a lower value than temperature Low Limit (*MR30*[7:0] and *MR31*[7:0]).

2 The Reserved bits are Read Only bits. The Host must write '0' in reserved bits when writing and device always returns '0' from reserved bits when reads this byte.

Table 77 MR35

Addr	MR35		Thermal Sensor Critical Temperature Low Limit Configuration - High Byte[1,2]
Bits	Attr	Default	Description
7:0	RW	0	MR35[7:0]: TS_CRIT_LOW_LIMIT_HIGH *MR34* and *MR35* - 16 bit thermal registers define the critical temperature low limit for thermal sensor. See Table 57 *Thermal Register - Low Byte and High Byte.*

1 Critical temperature High Limit value must have a higher value than temperature High Limit (*MR30*[7:0] and *MR31*[7:0]).

2 The Reserved bits are Read Only bits. The Host must write '0' in reserved bits when writing and device always returns '0' from reserved bits when reads this byte.

Table 78 MR48

Addr	MR48		Device Status
Bits	Attr	Default	Description
7	RO	0	MR48[7]: IBI_STATUS Device Event In Band Interrupt Status 0 = No pending IBI 1 = Pending IBI
6:0	RV	0	MR48[6:0]: Reserved

Table 79 MR49

Addr	MR49		Current Sensed Temperature - Low Byte[1]
Bits	Attr	Default	Description
7:0	RO	0	MR49[7:0]: TS_SENSE_LOW *MR49* and *MR50* - 16 bit thermal registers return the most recent conversion of the thermal sensor. See Table 57 *Thermal Register - Low Byte and High Byte.*

1 The device always returns '0' from reserved bits.

Table 80 MR50

Addr	MR50		Current Sensed Temperature - High Byte[1]
Bits	Attr	Default	Description
7:0	RO	0	MR50[7:0]: TS_SENSE_HIGH *MR49* and *MR50* - 16 bit thermal registers return the most recent conversion of the thermal sensor. See Table 57 *Thermal Register - Low Byte and High Byte.*

1. The device always returns '0' from reserved bits.

Table 81 MR51

Addr	MR51		Thermal Sensor Temperature Status
Bits	Attr	Default	Description
7:4	RV	0	MR51[7:4]: Reserved
3	RO	0	MR51[3]: TS_CRIT_LOW_STATUS Temperature Sensor Critical Low 0 = Temperature is above the limit set in *MR34* and *MR35* 1 = Temperature is below the limit set in *MR34* and *MR35*
2	RO	0	MR51[2]: TS_CRIT_HIGH_STATUS Temperature Sensor Critical High 0 = Temperature is below the limit set in *MR32* and *MR33* 1 = Temperature is above the limit set in *MR32* and *MR33*
1	RO	0	MR51[1]: TS_LOW_STATUS Temperature Sensor Low 0 = Temperature above limit set in registers *MR30* and *MR31* 1 = Temperature below limit set in registers *MR30* and *MR31*
0	RO	0	MR51[0]: TS_HIGH_STATUS Temperature Sensor High 0 = Temperature is below the limit set in registers *MR28* and *MR29* 1 = Temperature is above the limit set in registers *MR28* and *MR29*

Table 82 MR52

Addr	MR52		Hub and Thermal Sensor Error Status
Bits	Attr	Default	Description
7:2	RO	0	MR52[7:2]: Reserved
1	RO	0	MR52[1]: PEC_ERROR_STATUS Packet Error[1,2] 0 = No PEC Error 1 = PEC Error in one or more packets
0	RO	0	MR52[0]: PAR_ERROR_STATUS Parity Check Error[2,3] 0 = No Parity Error 1 = Parity Error in one or more bytes

1 Only applicable *MR18*[5] = '1' and if PEC function is enabled.

2 This register is updated when GD30TP139A device goes through bus reset as described in Section 6.2.3.

3 Only applicable in *MR18*[5] = '1' and if Parity function is not disabled or for supported CCC in I^2C mode.

8 Application Information

The GD30TP139A is a precision temperature sensor targeted at DDR5 DIMMs. The device is intended to operate up to 12.5 MHz on a 1.0V I3C Basic bus (supporting in band interrupts) or up to 1 MHz on a 1.0 V to 3.3 V I²C bus. As the GD30TP139A operates on the I3C bus, the device does not require an external pull up resistor on the SDA or SCL pin.

8.1 Typical Application Circuit

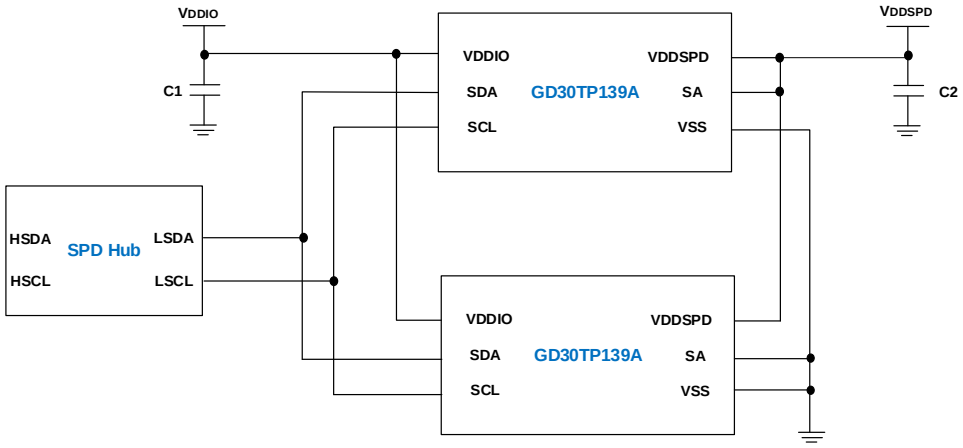


Figure 18. Typical Application

8.2 Design Example

For this design example, use the parameters in *Table 83*.

Table 83. Design Parameters

PARAMETER	EXAMPLE VALUE
V_{DDSPD}	1.8V
V_{DDIO}	1.0V
C1	0.01μF
C2	0.01μF

1. GD's customers need to test and verify whether the selected components meet their intended use to ensure stable system operation.

8.3 Detailed Design Description

The GD30TP139A operates with dual supply pins. The supply V_{DDIO} is used for the bus interface and operates in the range of 0.95 V to 1.05 V. The pin V_{DDSPD} is used as the supply for the core and operates in the range of 1.7 V to 1.98 V. A power-supply bypass capacitor is required for precision and stability. Place these power-supply capacitors as close to the supply and ground pins of the device as possible. A typical value of these supply bypass capacitor is 0.01μF. Applications with noisy or high-impedance power supplies can require a bigger bypass capacitor to reject power-supply noise. It is best to add a 0Ω resistor outside to the SA pin.

9 Layout Guidelines and Example

Efficient PCB layout is critical for stable operation. A poor layout design can result in poor line or load regulation and stability issues.

1) Place the input/output capacitor and inductor should be placed as close to IC.
2) Keep the power traces as short as possible.
3) The low side of the input and output capacitor must be connected properly to the power GND avoid a GND potential shift.

For best results, follow the layout example below.

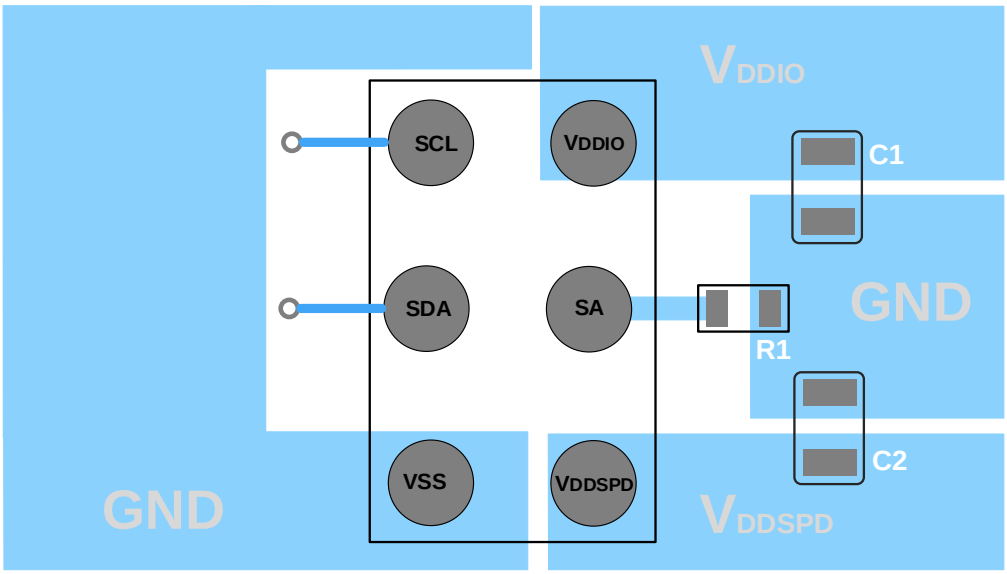
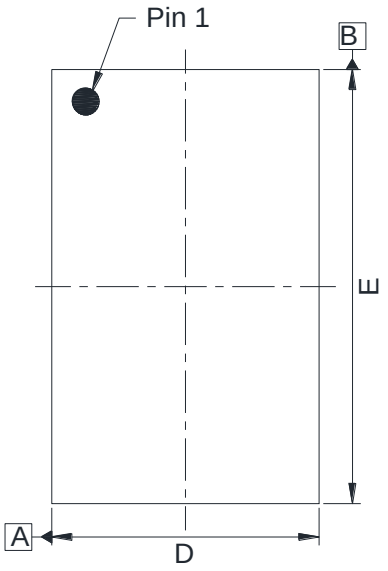


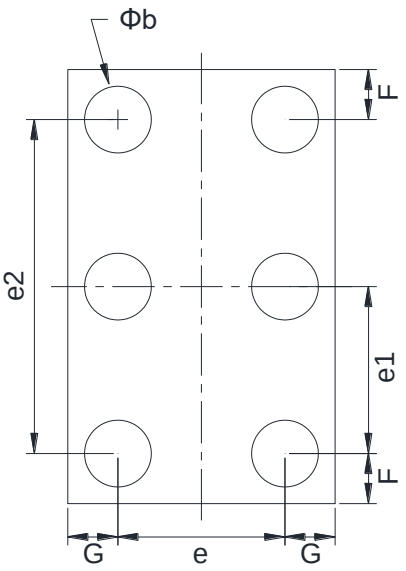
Figure 19. Typical GD30TP139A Example Layout

10 Package Information

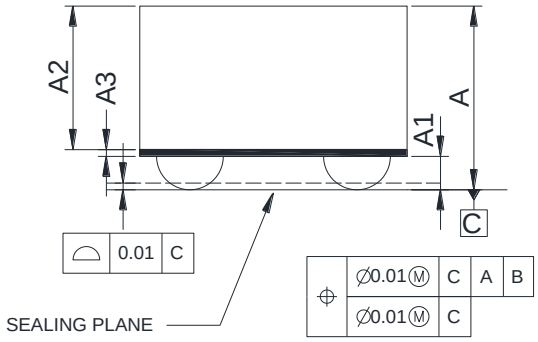
WLCSP Package Outline[1,2]



TOP VIEW



BOTTOM VIEW



SIDE VIEW

SYMBOL	DIMENSIONS IN MILLIMETER		
	MIN	NOM	MAX
A	0.409	0.440	0.471
A1	0.122	0.142	0.162
A2	0.283	0.293	0.303
A3	0.004	0.005	0.006
b	0.176	0.196	0.216
D	0.793	0.808	0.823
E	1.293	1.308	1.323
e		0.500	
e1		0.500	
e2		1.000	
F	0.147	0.154	0.162
G	0.147	0.154	0.162

Note:

1. Controlling dimension is mm.
2. Drawing is not to scale.
3. Seating plane is defined by the spherical crowns of the bump.

NOTES:

1. All dimensions are in millimeters.
2. Package dimensions does not include mold flash, protrusions, or gate burrs.

11 Ordering Information

Ordering Code	Package Type	ECO Plan	Packing Type	MOQ	OP Temp(°C)
GD30TS139NSYTR-I	6-ball WLCSP	Green	Tape & Reel	4000	Industrial -40°C to +105°C

12 Revision History

REVISION NUMBER	DESCRIPTION	DATE
1.0	Initial release and device details	2024

Important Notice

This document is the property of GigaDevice Semiconductor Inc. and its subsidiaries (the "Company"). This document, including any product of the Company described in this document (the “Product”), is owned by the Company according to the laws of the People’s Republic of China and other applicable laws. The Company reserves all rights under such laws and no IntellectμAl Property Rights are transferred (either wholly or partially) or licensed by the Company (either expressly or impliedly) herein. The names and brands of third party referred thereto (if any) are the property of their respective owner and referred to for identification purposes only.

The Company makes no representations or warranties of any kind, express or implied, with regard to the merchantability and the fitness for a particular purpose of the Product, nor does the Company assume any liability arising out of the application or use of any Product described in this document. Any information provided in this document is provided only for reference purposes. It is the sole responsibility of the user of this document to determine whether the Product is suitable and fit for its applications and products planned, and properly design, program, and test the functionality and safety of its applications and products planned using the Product. Unless otherwise expressly specified in the datasheet of the Product , the Product is designed, developed, and/or manufactured for ordinary business, industrial, personal, and/or household applications only, and the Product is not designed or intended for use in (i) safety critical applications such as weapons systems, nuclear facilities, atomic energy controller, combustion controller, aeronautic or aerospace applications, traffic signal instruments, pollution control or hazardous substance management; (ii) life-support systems, other medical equipment or systems (including life support equipment and surgical implants); (iii) automotive applications or environments, including but not limited to applications for active and passive safety of automobiles (regardless of front market or aftermarket), for example, EPS, braking, ADAS (camera/fusion), EMS, TCU, BMS, BSG, TPMS, Airbag, Suspension, DMS, ICMS, Domain, ESC, DCDC, e-clutch, advanced-lighting, etc.. Automobile herein means a vehicle propelled by a self-contained motor, engine or the like, such as, without limitation, cars, trucks, motorcycles, electric cars, and other transportation devices; and/or (iv) other uses where the failure of the device or the Product can reasonably be expected to result in personal injury, death, or severe property or environmental damage (collectively "Unintended Uses"). Customers shall take any and all actions to ensure the Product meets the applicable laws and regulations. The Company is not liable for, in whole or in part, and customers shall hereby release the Company as well as its suppliers and/or distributors from, any claim, damage, or other liability arising from or related to all Unintended Uses of the Product. Customers shall indemnify and hold the Company, and its officers, employees, subsidiaries, affiliates as well as its suppliers and/or distributors harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of the Product.

Information in this document is provided solely in connection with the Product. The Company reserves the right to make changes, corrections, modifications or improvements to this document and the Product described herein at any time without notice. The Company shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. Information in this document supersedes and replaces information previously supplied in any prior versions of this document.