



Gowin I3C Single Clock IP

User Guide

IPUG786-1.0E,10/28/2021

Copyright © 2021 Guangdong Gowin Semiconductor Corporation. All Rights Reserved.

GOWIN, , Gowin, and GOWINSEMI are trademarks of Guangdong Gowin Semiconductor Corporation and are registered in China, the U.S. Patent and Trademark Office, and other countries. All other words and logos identified as trademarks or service marks are the property of their respective holders. No part of this document may be reproduced or transmitted in any form or by any denotes, electronic, mechanical, photocopying, recording or otherwise, without the prior written consent of GOWINSEMI.

Disclaimer

GOWINSEMI assumes no liability and provides no warranty (either expressed or implied) and is not responsible for any damage incurred to your hardware, software, data, or property resulting from usage of the materials or intellectual property except as outlined in the GOWINSEMI Terms and Conditions of Sale. All information in this document should be treated as preliminary. GOWINSEMI may make changes to this document at any time without prior notice. Anyone relying on this documentation should contact GOWINSEMI for the current documentation and errata.

Revision History

Date	Version	Description
10/28/2021	1.0E	Initial version published.

Contents

Contents	i
List of Figures.....	iv
List of Tables.....	v
1 About This Guide	1
1.1 Purpose	1
1.2 Related Documents	1
1.3 Terminology and Abbreviations.....	2
1.4 Support and Feedback	2
2 Overview.....	3
3 Features and Performance	4
3.1 Features.....	4
3.2 Max. Frequency	4
3.3 Resource Utilization.....	4
4 Functional Description.....	6
4.1 Operating Mode	6
4.2 User Interface Timing.....	6
5 Port Description.....	8
6 Register	10
6.1 Register Address Mapping and Description	10
6.2 Register Description	12
6.2.1 MCFG (0x00)	12
6.2.2 SCFG (0x04).....	14
6.2.3 SSTS (0x08)	16
6.2.4 SCONTROL (0x0C)	18
6.2.5 SIS (0x10)	19

6.2.6 SIC (0x14).....	20
6.2.7 SIM (0x18)	21
6.2.8 SERR (0x1C)	22
6.2.9 SDATACONTROL (0x2C)	22
6.2.10 STXB (0x30)	23
6.2.11 SRXB (0x40)	23
6.2.12 SDA (0x64)	24
6.2.13 SVFVORRV (0x6C)	24
6.2.14 SBCRANDDCR (0x70)	24
6.2.15 SMMID (0x74).....	24
6.2.16 MCONTROL (0x84)	25
6.2.17 MSTs (0x88).....	27
6.2.18 MIBIFORMCFG (0x8C)	29
6.2.19 MIS (0x90)	30
6.2.20 MIC (0x94)	31
6.2.21 MIM (0x98).....	31
6.2.22 MERR (0x9C)	32
6.2.23 MDATACONTROL (0xAC)	33
6.2.24 MTXB (0xB0)	34
6.2.25 MTXBE (0xB4).....	34
6.2.26 MRXB (0xC0).....	34
6.2.27 DID (0xC4).....	35
7 Operating Flow.....	36
7.1 Master Operating Flow	36
7.1.1 SDR Read/Write	36
7.1.2 I ² C Read/Write	37
7.1.3 ENTDAAs Operation.....	37
7.1.4 Other CCC Operation	38
7.1.5 IBI/HJ Operation	38
7.1.6 DDR Read/Write	39
7.2 Slave Operating Flow	41
7.2.1 SDR Read/Write	41

7.2.2 ENTDAAs Operation.....	41
7.2.3 Other CCC Operation	42
7.2.4 IBI / HJ / MR Operation	42
8 Interrupt.....	44
9 GUI	45
9.1 IP Generation.....	45
9.2 Configuration Interface	46
10 Reference Design	47

List of Figures

Figure 4-1 REG Write Operation Timing	7
Figure 4-2 REG Read Operation Timing.....	7
Figure 5-1 IO Ports of I3C Single Clock.....	8
Figure 9-1 Call Gowin I3C Single IP Via Tool Bar	45
Figure 9-2 Options	46

List of Tables

Table 1-1 Terminology and Abbreviations	2
Table 2-1 I3C Single Clock IP	3
Table 3-1 I3C Single Clock Resource Utilization	5
Table 5-1 IO List of I3C Single Clock	8
Table 6-1 Master Register Address Mapping and Description.....	10
Table 6-2 Slave Register Address Mapping and Description.....	11
Table 6-3 MCFG	12
Table 6-4 SCFG	14
Table 6-5 SSTS.....	16
Table 6-6 SCONTROL	18
Table 6-7 SIS.....	19
Table 6-8 SIC	20
Table 6-9 SIM	21
Table 6-10 SERR	22
Table 6-11 SDATACONTROL	22
Table 6-12 STXB.....	23
Table 6-13 SRXB.....	23
Table 6-14 SDA.....	24
Table 6-15 SVFVORRV.....	24
Table 6-16 SBCRANDDCR	24
Table 6-17 SMMID	24
Table 6-18 MCONTROL.....	25
Table 6-19 MSTs	27
Table 6-20 MIBIFORMCFG	29
Table 6-21 MIS	30

Table 6-22 MIC	31
Table 6-23 MIM	31
Table 6-24 MERR.....	32
Table 6-25 MDATACONTROL.....	33
Table 6-26 MTXB	34
Table 6-27 MTXBE	34
Table 6-28 MRXB	34
Table 6-29 DID	35
Table 9-1 Configuration Options	46

1 About This Guide

1.1 Purpose

Gowin I3C Single Clock IP User Guide mainly includes the descriptions of the functions, features, ports, timing, call and configuration, and reference design, etc. It helps you learn the features and usage of Gowin I3C Single Clock IP.

1.2 Related Documents

The latest user guides are available on our Website. Refer to the related documents at www.gowinsemi.com:

- [DS100](#), GW1N series of FPGA Products Data Sheet
- [DS117](#), GW1NR series of FPGA Products Data Sheet
- [DS821](#), GW1NS series of FPGA Products Data Sheet
- [DS861](#), GW1NSR series of FPGA Products Data Sheet
- [DS891](#), GW1NRF series of Bluetooth FPGA Products Data Sheet
- [DS881](#), GW1NSER series of FPGA Products Data Sheet
- [DS102](#), GW2A series of FPGA Products Data Sheet
- [DS226](#), GW2AR series of FPGA Products Data Sheet
- [DS961](#), GW2ANR series of FPGA Products Data Sheet
- [DS976](#), GW2A-55 Data Sheet
- [DS971](#), GW2AN-18X & 9X Data Sheet
- [SUG100](#), Gowin Software User Guide

1.3 Terminology and Abbreviations

The abbreviations and terminology used in this manual are as shown in Table 1-1.

Table 1-1 Terminology and Abbreviations

Terminology and Abbreviations	Meaning
IP	Intellectual Property
LUT	Look-up Table
IBI	In-Band Interrupt
HJ	Hot-join

1.4 Support and Feedback

Gowin Semiconductor provides customers with comprehensive technical support. If you have any questions, comments, or suggestions, please feel free to contact us directly by the following ways.

Website: www.gowinsemi.com

E-mail: support@gowinsemi.com

2 Overview

Introduction

Gowin I3C Single Clock IP is used for serial communication interface and serial sensor interface to receive or transmit data. Gowin I3C Single Clock IP provides an I3C bus communication controller with REG interface and APB interface.

Table 2-1 I3C Single Clock IP

Gowin I3C Single Clock IP	
Logic Resource	See Table 3- 1.
Delivered Doc.	
Design Files	Verilog (encrypted)
Reference Design	Verilog
TestBench	Verilog
Test and Design Flow	
Synthesis Software	GowinSynthesis [®]
Application Software	Gowin Software (V1.9.8.01 and above)

3 Features and Performance

3.1 Features

- Compatible with MIPI I3C V1.1 specification
- Supports SDR and DDR modes
 - SDR mode supports both master and slave, up to 12.5Mbps.
 - DDR mode supports master only, up to 25Mbps.
- Supports address arbitration
- Supports I3C common command code
 - dynamic address assignment
 - other optional CCC
- Supports IBI and Hot-join
- Can emit slave reset pattern
- Backwards compatible with I²C Slave
- Supports two user interfaces
 - APB Interface
 - REG Interface

3.2 Max. Frequency

The maximum frequency of I3C Single Clock is determined by the speed grade of the devices, when the system clock is 100MHz, and I3C SCL speeds up to 12.5 MHz.

3.3 Resource Utilization

I3C Single Clock can be implemented by Verilog. Its performance and resource utilization may vary when the design is employed in different

devices, or at different densities, speeds, or grades.

Taking Gowin GW2A-18 series of FPGA as an example, the resource utilization is shown in Table 3-1. For application verification on other Gowin FPGA, please pay attention to the post-release information.

Table 3-1 I3C Single Clock Resource Utilization

Device Series	Speed Grade	Resource Utilization		Remarks
GW2A-18	C8/I7	LUT	2189	-
		REG	1274	

4 Functional Description

4.1 Operating Mode

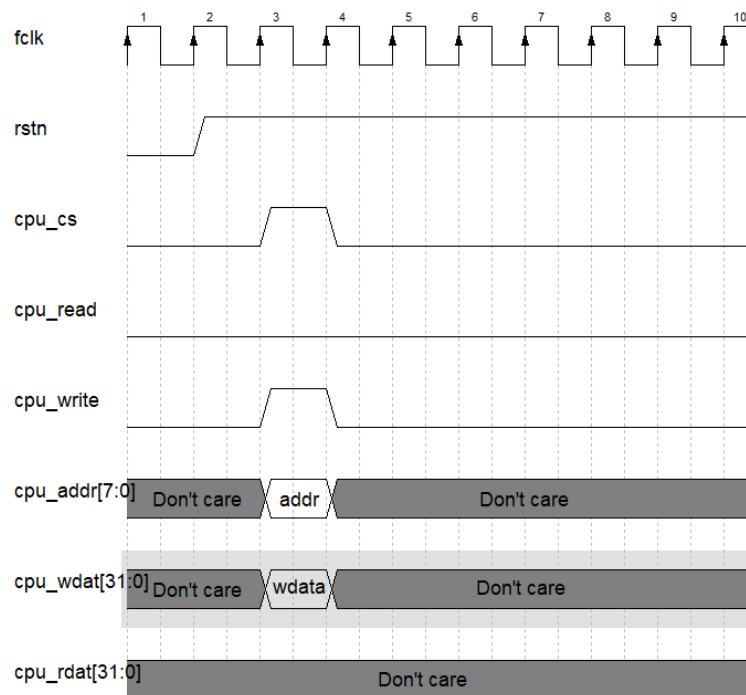
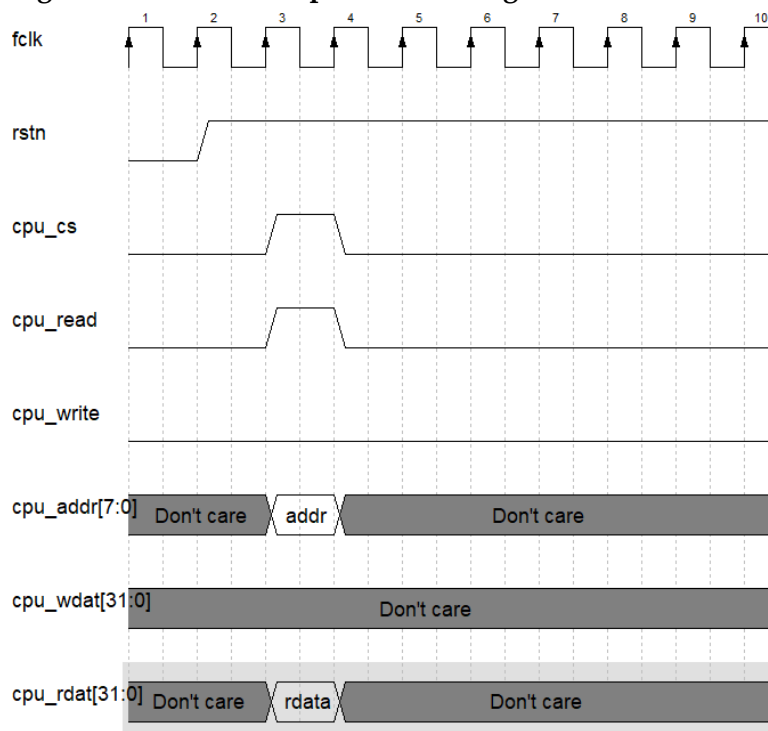
This soft core provides REG interface and APB interface. You can configure IP parameters, control the IP communication and send data through writing control register; you can read IP parameters and the data received through reading control register; IP notifies you of the events on the bus through interrupts; you can handle the interrupts through the control register; after the interrupts are processed, that is, the bus communication is completed.

In order to realize the fast I3C data read and write, this soft core is equipped with two 8-bit read/write FIFOs with a depth of 16. This allows you to write data continuously during sending data, and to read data continuously after data received. Note that for Master, read means that the master reads the data sent by the slave, while for Slave read means that the slave sends the data read by the master to the master; for Master, write means that the master writes data to the slave, and for Slave write means that the slave receives the data sent by the master.

4.2 User Interface Timing

You can configure the user interface to REG or APB, and APB interface is compatible

The timing diagram of REG interface is as shown below.

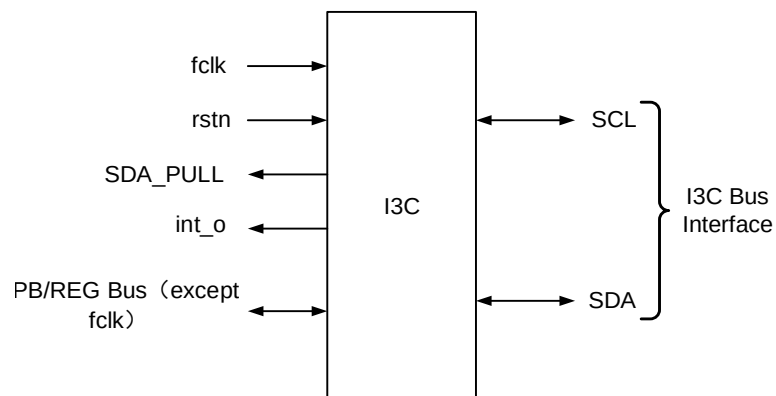
Figure 4-1 REG Write Operation Timing**Figure 4-2 REG Read Operation Timing**

5 Port Description

I3C Single Clock Port

For I3C Single Clock ports description, you can see Figure 5-1.

Figure 5-1 IO Ports of I3C Single Clock



For the details, please refer to Table 5-1.

Table 5-1 IO List of I3C Single Clock

Signal	I/O	Description
fclk	input	Clock
rstn	input	Reset, active low.
SDA_PULL	output	Controllable pull-up of the I3C serial data
int_o	output	Interrupt output, active low.
I3C Bus Interface		
SCL	inout	I3C Serial clock line
SDA	inout	I3C Serial data line
User interface-APB interface (except fclk)		
PADDR[7:0]	input	Byte address
PWDATA[31:0]	input	Write data

Signal	I/O	Description
PRDATA[31:0]	output	Read data
PSEL	input	Slave selection
PWRITE	input	Read / write IO
PENABLE	input	Enable Signal
User interface-REG interface (except fclk)		
cpu_cs	input	Chip selected signal
cpu_read	input	Read enable
cpu_write	input	Write enable
cpu_addr[7:0]	input	Address signal
cpu_wdat[31:0]	input	Write data
cpu_rdat[31:0]	output	Read data

6 Register

The I3C Single Clock IP uses a 256 space and therefore decodes the lower 8 bits of the address as an offset value for the register group. It maps all control registers, status registers, and registers used to access RX and TX FIFO to this address space.

6.1 Register Address Mapping and Description

Table 6-1 Master Register Address Mapping and Description

Offset Address	Name	Access	Description
0x00	MCFG	Read/Write	Master configuration: initializes IP.
0x84	MCONTROL	Read/Write	Master control: controls IP operations on I3C and I ² C.
0x88	MSTS	Read/Write	Master status: IP and I3C status (write 1 to set register to 0).
0x8C	MIBIFORMCFG	Read/Write	Rules for master In-Band Interrupt use
0x90	MIS	Read/Write	Master Interrupt Enable: enables IP interrupt.
0x94	MIC	Write-only	Master Interrupt clear: writes 1 to clear interrupt enable.
0x98	MIM	Read-only	Master masked interrupt: MSTs and MIS.
0x9C	MERR	Read/Write	Master error: writes 1 to set register to 0.
0xAC	MDATACONTR OL	Read/Write	Master data control: controls and shows the FIFO status.

Offset Address	Name	Access	Description
0xB0	MTXB	Write-only	Master TX data: master writes data to TX FIFO.
0xB4	MTXBE	Write-only	Master TX data end: master writes the last data to TX FIFO.
0xC0	MRXB	Read-only	Master RX data: master reads data from RX FIFO
0xC4	DID	Read-only	Device ID: you can read ID of this IP.

Table 6-2 Slave Register Address Mapping and Description

Offset Address	Name	Access	Description
0x04	SCFG	Read/Write	Slave configuration: initializes IP.
0x08	SSTS	Read/Write	Slave status: IP and I3C status: writes 1 to set register to 0.
0x0C	SCONTROL	Read/Write	Slave control: controls IP operations on I3C and I ² C.
0x10	SIS	Read/Write	Slave Interrupt Enable: enables IP interrupt.
0x14	SIC	Write-only	Slave Interrupt clear: writes 1 to clear interrupt enable.
0x18	SIM	Read-only	Slave masked interrupt: SSTS and SIS
0x1C	SERR	Read and Write	Slave error: writes 1 to set register to 0.
0x2C	SDATACONTROL	Read and Write	Slave data control: controls and shows the FIFO status.
0x30	STXB	Write-only	Slave TX Data: slave writes data to TX FIFO.
0x40	SRXB	Read-only	Slave RX data: slave reads data from RX FIFO.
0x64	SDA	Read and Write	Slave dynamic address: you can read/write slave dynamic address.

Offset Address	Name	Access	Description
0x6C	SVFVORRV	Read and Write	Master vendor fixed value or random value: you can read/write the vendor fixed value or random value of the slave.
0x70	SBCRAND DCR	Read and Write	Slave BCR or DCR: you can read/write the value of BCR and DCR.
0x74	SMMID	Read and Write	Slave MIPI manufacturer ID: you can read/write MIPI manufacturer ID of the slave.

6.2 Register Description

6.2.1 MCFG (0x00)

Initialize the IP.

Table 6-3 MCFG

Bit	Name	Description	Reset Value
0	MENABLE: Enable or disable master.	1: IP is configured to master. 0: IP is configured to slave.	0x0
2: 1	Reserved	–	–
3	MDISTIMEOUT: Enable or disable timeout.	1: Disable timeout error detection. 0: Enable timeout error detection. After MDISTIMEOUT enabled, If the master has been left in a state other than idle for more than 100 us, then the timeout error will be generated, the IP will send an MERR interrupt.	0x0
7: 4	Reserved	–	–

Bit	Name	Description	Reset Value
11: 8	PPHIGH: The number of fclk occupied by SCL high in push-pull mode	PPHIGH = the number of fclk occupied by SCL high - 1, and PPHIGH is configured to a minimum of 3.	0x0
15: 12	PPLOWEXTRA: In the push-pull mode, the number of fclk occupied by SCL low over SCL high.	Number of fclk occupied by SCL low = Number of fclk occupied by SCL high + PLOWEXTRA	0x0
23: 16	ODSCL: When ODHIGHEQUALPP is 0, the number of fclk occupied by SCL high and low in open-drain mode is ODSCL+1 times the number of fclk occupied by SCL high in push-pull mode.	Number of fclk occupied by SCL high and low in open-drain mode = Number of fclk occupied by SCL high in push-pull mode * (ODSCL+1)	0x00
24	ODHIGHEQUALPP: Whether the number of fclk occupied by SCL high in open-drain mode is the same as the number of fclk occupied by SCL high in push-pull mode.	1: The number of fclk occupied by SCL highs in open-drain mode is the same as the number of fclk occupied by SCL highs in push-pull mode 0: the number of fclk occupied by SCL high in open-drain mode = the number of fclk occupied by SCL high in push-pull mode * (ODSCL+1)	0x0
27: 25	Reserved	-	-
31: 28	I2CSCL ^[1] : The number of fclk occupied by SCL high and low in I ² C mode	When I2CSCL is even: Number of fclk occupied by SCL high and low in I ² C mode = Number of fclk occupied by SCL low in open-drain mode * I2CSCL. When I2CSCL is odd: Number of fclk occupied by SCL high in I ² C mode	0x0

Bit	Name	Description	Reset Value
		= the number of fclk occupied by SCL low in open-drain mode * (I2CSCL-1). Number of fclk occupied by SCL low in I ² C mode = Number of fclk occupied by SCL low in open-drain mode * I2CSCL.	

Note!

[1] The minimum value of I2CSCL is 2.

6.2.2 SCFG (0x04)

After IP is configured to slave, initialize the IP.

Table 6-4 SCFG

Bit	Name	Description	Reset Value
0	MENABLE: Enable or disable slave.	1: Enable slave 0: Disable slave	0x0
1	SNACK: Whether the slave will NACK all requests to it or not.	1: The slave will NACK all requests to it except a Common Command Code (CCC) broadcast. 0: The slave will ACK all requests to it.	0x0
2	MATCHSAORDASS: Match static or dynamic address START or STOP.	1: Only SSTS MATCHSAORDA =1, then SSTS START and STOP will be set to 1, and START and STOP can be used to detect the end of a message to/from this slave. 0: START and STOP will be set to 1 whenever START and STOP are detected on the bus.	0x0
3	ERRIGNORE: Errors ignore.	1: IP will not detect S0	0x0

Bit	Name	Description	Reset Value
		and S1 errors on I3C bus. 0: IP will detect S0 and S1 errors on I3C bus.	
7: 4	Reserved	–	–
8	PIDTYPESELECT: Pre-assigned ID type selection	1: SVFVORRV. VFVORRV is a random value. 0: SVFVORRV. VFVORRV is a fixed value.	0x0
9	HJWAIT: Hot- join waiting	1: If this value is 1, the slave will wait for 200us before detecting START or pulling down the SDA bus. 0: If this value is 0, the slave will not wait for 200us before detecting START or pulling down the SDA bus.	0x0
15: 10	Reserved	–	–
23: 16	PULLDOWNSDACNT: Counting for slave pull-down bus.	The slave will wait for PULLDOWNSDACNT fclk cycles to pull down the SDA bus when the bus is idle.	0x00
24	Reserved	–	–
31: 25	SA: Static address	You can configure slave static address.	0x00

6.2.3 SSTS (0x08)

Show the status of the slave, and you can use this register to set the status register to 0 (write 1 to its field to set the register to 0).

Table 6-5 SSTS

Bit	Name	Description	Reset Value
0	STSBUSY: bus busy	1: bus busy 0: bus idle	0x0
1	STSMMSG: Status communication	1: The bus is communicating with this slave. 0: The bus is idle or is communicating with other slave.	0x0
2	STSCCAH: Status Common Command Code handler	1: Common Command Code (CCC) message is being handled automatically. 0: Common Command Code (CCC) message is not being handled automatically.	0x0
3	STSREAD: Status read communication	1: The device is performing read operation or IBI operation. 0: The device is not performing a read operation or IBI operation.	0x0
4	STSWRITE: Status write communication	1: The device is performing write operation (static address, dynamic address and broadcast address). 0: The device is not performing write operation (static address, dynamic address and broadcast address).	0x0
5	STSDAA: Status dynamic address assignment	1: Enter dynamic address assignment mode. 0: Not enter dynamic address assignment mode.	0x0
6	Reserved	–	–

Bit	Name	Description	Reset Value
7	START: START pattern detected	1: START pattern detected 0: START pattern not detected	0x0
8	MATCHEDBA: Status broadcast address match	1: Broadcast address write detected. 0: Broadcast address write not detected.	0x0
9	MATCHEDSAORDA: Status static address or dynamic address match	1: Static address or dynamic address detected. 0: Static address or dynamic address not detected.	0x0
10	STOP: STOP pattern detected	1: STOP pattern detected 0: STOP pattern not detected	0x0
11	RFIFONOTEMPTY: RX FIFO non-empty	1: RX FIFO non-empty 0: RX FIFO empty	0x0
12	SFIFONOTFULL: TX FIFO non-full	1: TX FIFO non-full 0: TX FIFO full	0x1
13	DAVALID: Dynamic address has been assigned in Dynamic Address Assignment Mode.	1: The dynamic address has been assigned in dynamic address assignment mode. 0: The dynamic address has not been assigned in dynamic address assignment mode.	0x0
14	CCCRCV: CCC commands other than DAA are received.	1: CCC commands other than DAA are received. 0: CCC commands other than DAA are not received.	0x0
15	ERR: Status error	1: The device in error. 0: The device not in error.	0x0
16	Reserved	–	–
17	CCCAH: Status Command Code handler	1: Common Command Code is being handled automatically. 0: Common Command Code is not being handled	0x0

Bit	Name	Description	Reset Value
		automatically.	
18	DATANEED: Status data	1: The device being read, and you need to write data. 0: You do not need to write data.	0x0
19	SLVRST: Status slave reset	1: The device has received slave reset pattern. 0: The device has not received slave reset pattern.	0x0
20	REQUEST: Status slave request	1: IBI, HJ and MR requests have been sent. 0: IBI, HJ and MR requests have not been sent.	0x0
21	REQUEST: Status slave ACK request	1: Master requests ACK from IBI, HJ, MR. 0: Master requests NACK from IBI, HJ, MR.	0x0
31: 22	Reserved	–	–

6.2.4 SCONTROL (0x0C)

Control slave.

Table 6-6 SCONTROL

Bit	Name	Description	Reset Value
1: 0	REQUEST: Slave request	2'H00: No request 2'H01: IBI is being requested. 2'H02: MR is being requested. 2'H03: HJ is being requested.	0x0
7: 2	Reserved	–	–
15: 8	IBIMDATA	IBI mandatory data byte (if data is 0 then no mandatory byte is sent; if data is not 0, then it is sent)	0x00
31: 16	Reserved	–	–

6.2.5 SIS (0x10)

User Interrupt enable set. To activate an interrupt enable, write 1 to its field in this register.

Table 6-7 SIS

Bit	Name	Description	Reset Value
6: 0	Reserved	–	–
7	START	START interrupt enable. Detect START pattern on the bus.	0x0
8	MATCHEDBA	Broadcast write address match interrupt enable, and the bus address is broadcast write address.	0x0
9	MATCHEDSAORDA	Static address or dynamic address match interrupt enable, the address on the bus is the static address or dynamic address of the slave.	0x0
10	STOP	STOP interrupt enable. Detect STOP pattern on the bus.	0x0
11	RFIFONOTEMPTY	RX FIFO non-empty enable. Detect RX FIFO non-empty.	0x0
12	SFIFONOTFULL	TX FIFO non-empty enable. Detect TX FIFO non-empty.	0x0
13	DAVALIDID	Dynamic address assignment valid enable. Detect valid dynamic address assignment.	0x0
14	CCCRCV	CCC other than DAA reception interrupt enable. Detect the device has received CCC other than DAA.	0x0
15	ERR	Error interrupt enable. Detect error.	0x0
16	Reserved	–	–
17	CCCAH	CCC automatic handler interrupt enable. The device receives CCC and handles it automatically.	0x0

Bit	Name	Description	Reset Value
18	DATANEED	Data demand interrupt enable. You need to write data for communication.	0x0
19	SLVRST	Slave reset interrupt enable. Detect slave reset pattern.	0x0
20	REQUEST	Request interrupt enable. The device has sent IBI, HJ, MR requests.	0x0
31: 21	Reserved	–	–

6.2.6 SIC (0x14)

User Interrupt enable clear. To clear an interrupt enable, write 1 to its field in this register.

Table 6-8 SIC

Bit	Name	Description	Reset Value
6: 0	Reserved	–	–
7	START	START interrupt enable clear	–
8	MATCHEDBA	Broadcast address write match interrupt enable clear	–
9	MATCHEDSAORDA	Status static address or dynamic address match interrupt enable clear	–
10	STOP	STOP interrupt enable clear	–
11	RFIFONOTEMPTY	RX FIFO non-empty interrupt enable clear	–
12	SFIFONOTFULL	TX FIFO non-full interrupt enable clear	–
13	DAVALID	Dynamic address valid interrupt enable clear	–
14	CCCRCV	CCC other than DAA RX Interrupt enable clear	–
15	ERR	Error interrupt enable clear	–
16	Reserved	–	–
17	CCCAH	Common Command Code handler Interrupt enable clear	–

Bit	Name	Description	Reset Value
18	DATANEED	Data demand interrupt enable clear	–
19	SLVRST	Slave reset interrupt enable clear	–
20	REQUEST	Request interrupt enable clear	–
31: 21	Reserved	–	–

6.2.7 SIM (0x18)

Interrupt mask. Return the status of enabled interrupts with the value of SSTS & SIS.

Table 6-9 SIM

Bit	Name	Description	Reset Value
6: 0	Reserved	–	–
7	START	START Interrupt Mask	0x0
8	MATCHEDBA	Broadcast address write match interrupt mask	0x0
9	MATCHEDSAORDA	Status static address or dynamic address match interrupt mask	0x0
10	STOP	STOP Interrupt mask	0x0
11	RFIFONOTEMPTY	RX FIFO non-empty Interrupt mask	0x0
12	SFIFONOTFULL	TX FIFO non-full Interrupt mask	0x0
13	DAVALID	Dynamic address valid interrupt mask	0x0
14	CCRCRV	CCC other than DAA RX Interrupt mask	0x0
15	ERR	Error Interrupt mask	0x0
16	Reserved	–	–
17	CCCAH	CCC handler Interrupt mask	0x0
18	DATANEED	Data demand Interrupt mask	0x0
19	SLVRST	Slave reset Interrupt mask	0x0
20	REQUEST	Request Interrupt mask	0x0
31: 21	Reserved	–	–

6.2.8 SERR (0x1C)

Slave error. When there is an error, you can read the error type through the error register; when SERR register is set to 1, SSTS.ERR will also be set to 1. You can set the register to 0 by writing 1 to its field in this register.

Table 6-10 SERR

Bit	Name	Description	Reset Value
0	OVERRCV: Overrun	RX FIFO is full and data is still coming in.	0x0
1	Reserved	-	-
2	NACKWITHOUTDATA: NACK master without data	IP NACK master because there is no data provided to the IP.	0x0
7: 3	Reserved	-	-
8	SDRPARERR: Parity error in SDR mode.	Parity error in SDR mode.	0x0
109	Reserved	-	-
11	S0ORS1ERR: S0 or S1 error	S0 or S1 error	0x0
1512	Reserved	-	-
16	READEMPTY: Read empty error	When RX FIFO is empty, you still read the RX FIFO.	0x0
17	WRITEFULL: Write full error	When TX FIFO is full, you still write data to TX FIFO.	0x0
31: 18	Reserved	-	-

6.2.9 SDATACONTROL (0x2C)

Slave data control; control and show the status of FIFO with depth of 16.

Table 6-11 SDATACONTROL

Bit	Name	Description	Reset Value
0	SFIFOCLR: Clear TX FIFO	Write 1 to clear TX FIFO	0x0
1	RFIFOCLR: Clear RX FIFO	Write 1 to clear RX FIFO	0x0
15: 2	Reserved	-	-

Bit	Name	Description	Reset Value
20: 16	SFIFOCNT: Number of data in TX FIFO	Number of data in TX FIFO	0x00
23: 21	Reserved	–	–
28: 24	RFIFOCNT: Number of data in RX FIFO	Number of data in RX FIFO	0x00
29	Reserved	–	–
30	SFIFOFULL: TX FIFO full	1: TX FIFO full 0: TX FIFO non-full	0x0
31	RFIFOEMPTY: RX FIFO empty	1: RX FIFO empty 0: RX FIFO non-empty	0x1

6.2.10 STXB (0x30)

Slave TX data. You can write data to TX FIFO via this register.

Table 6-12 STXB

Bit	Name	Description	Reset Value
7: 0	WDATA: Write data	Data to be written in the TX FIFO (IP will automatically end the transmission if there is no data in the TX FIFO).	–
31: 8	Reserved	–	–

6.2.11 SRXB (0x40)

You can read data from RX FIFO via this register.

Table 6-13 SRXB

Bit	Name	Description	Reset Value
7: 0	RDATA: Read data	Read data from RX FIFO	0x00
31: 8	Reserved	–	–

6.2.12 SDA (0x64)

Slave dynamic address. You can read/write slave dynamic address via this register.

Table 6-14 SDA

Bit	Name	Description	Reset Value
0	DAVALID	1: Dynamic address valid (write 1 to set 1) 0: Dynamic address invalid (write 0 to set 0)	0x0
7: 1	DA	Dynamic address value	0x00
31: 8	Reserved	–	–

6.2.13 SVFVORRV (0x6C)

Slave vendor fixed value or random value. You can read/write the vendor fixed value or random value of the slave.

Table 6-15 SVFVORRV

Bit	Name	Description	Reset Value
31: 0	VFVORRV	Slave vendor fixed value or random value	0x0000

6.2.14 SBCRANDDCR (0x70)

Slave BCR or DCR. You can read/write the value of BCR and DCR.

Table 6-16 SBCRANDDCR

Bit	Name	Description	Reset Value
7: 0	Reserved	–	–
15: 8	DCR	Device characteristics register value	0x00
23: 16	BCR	Bus characteristics register value	0x00
31: 24	Reserved	–	–

6.2.15 SMMID (0x74)

Slave MIPI manufacturer ID. You can read/write MIPI manufacturer ID of the slave.

Table 6-17 SMMID

Bit	Name	Description	Reset Value
14: 0	MMID	MIPI manufacturer ID	0x000
31: 15	Reserved	–	–

6.2.16 MCONTROL (0x84)

Master control. You can control IP operations on I3C and I²C.

Table 6-18 MCONTROL

Bit	Name	Description	Reset Value
2: 0	REQUEST: Master request	3'h0: Set 0 after request completed. 3'h1: MCONTROL. COMTYPE in non-DDR, the master requests to send START + ADDR pattern, after the master ends the read, only ADDR is sent; MCONTROL. COMTYPE in DDR, when I3C does not enter DDR mode, the master sends START+ 7E/W+ACK+ENTHDR0 CCC+DDR COMMAND; and when I3C enters DDR mode, the master sends HDR Restart+ DDR COMMAND. 3'h2: Master requests to send STOP. 3'h3: Master requests manual IBI response. 3'h4: Master requests DAA, if the master not in DAA mode, the master sends START+7E/W+ACK+ENTDAA+RSTART+7E/R, and ends before the master sends the dynamic address to be assigned; if the master in DAA mode, the master sends DA+ACK+ RSTART+7E/R, and ends before the master sends the dynamic address to be assigned; if the slave NACK 7E/R, the master will automatically end the DAA. 3'h5: Master sends START + slave reset pattern. 3'h6: Master sends HDR Exit. 3'h7: Reserved	0x0
3	Reserved	-	-
5: 4	COMTYPE: Communication type	2'H0: I3C SDR 2'H1: I²C	0x0

Bit	Name	Description	Reset Value
		2'h2: DDR 2'h3: Reserved	
7: 6	IBIRSPTYPE: IBI response type	2'h0: Auto ACK. Whether to follow the forced byte after ACK is decided by MIBIFORMCFG register. 2'h1: Auto NACK 2'h2: Auto ACK, and the master can ensure that the slave will send the forced byte (only used in manual ACK). 2'h3: Manual ACK. Communication will be suspended when IBI occurs, waiting for the master to configure MCONTROL.REQUEST to 3'h3 for in-band interrupt response.	0x0
8	DIRECTION: Communication direction	0: Write 1: Read	0x0
15: 9	COMADDR: Communication address	Slave address sent by the master	0x00
23: 16	READTERMCNT:	The master will end this read communication after READTERMCNT bytes, and READTERMCNT must be an even number in DDR.	0x00
24	DDRENDWITHCRC: Communication in DDR ends with/without CRC	0: Master ends read communication in DDR, and slave ends write communication without CRC. 1: Master ends read communication in DDR, and slave ends write communication with CRC.	0x0
31: 25	Reserved	–	–

6.2.17 MSTS (0x88)

Show the status of the master. You can use this register to set the status register to 0 (write 1 to its field to set the status register to 0).

Table 6-19 MSTS

Bit	Name	Description	Reset Value
2: 0	MSTE: Status master	3'H0: Idle 3'H1: SDA pull-down by slave 3'h2: Reserved 3'H0: SDR or I ² C 3'H0: DDR 3: Dynamic address assignment 3'H6: Wait for you to decide whether to response IBI, MR and HJ. 3'h7: Receive IBI, MR, HJ requests from the slave (IBI with forced bytes will exit this status after receiving, the rest is to exit this status after the master ACK).	0X0
3	Reserved	–	–
4	BWN: In communication	1: When the master in dynamic address assignment, this bit is 1, then the master waits for next operation; when the master is in SDR read and write, this bit is 1, then the master waits for read data from the TX FIFO or write data to the RX FIFO. 0: Master in other status.	0x0
5	NACK	1: In SDR mode, slave NACK address sent by master; in DDR, slave NACK command sent by master. 0: In SDR mode, slave ACK address sent by master; in DDR, slave ACK command sent by master.	0x0

Bit	Name	Description	Reset Value
7: 6	SRTYPE: Slave request type	2'H0: No request 2'H1: IBI request 2'H2: Slave requests a master role 2'H3: Slave requests hot-join.	0x0
8	SSTART: Slave pulls down SDA bus to send a START signal.	1: SDA bus is pulled down by the slave to generate START pattern. 0: SDA bus is not pulled down by the slave to generate START pattern.	0x0
9	MCONTROLFINISH: Master control completed	1: After MCONTROL.REQUEST operation is completed, this bit is set to 1; for example, if MCONTROL.REQUEST is 3'h1, the master will set this bit to 1 after the START + ADDR pattern is sent and the slave ACK/NACK the address. 0: Write 1 to this bit to clear	0x0
10	COMCOMPLETE: Communication completed	1: After a read/write communication or DAA communication is completed, this bit is set to 1 (contains slave NACK DDR read/write command). 0: Write 1 to this bit to clear	0x0
11	RFIFONOTEMPTY: RX FIFO non-empty	1: RX FIFO non-empty, this bit is set to 1. 0: RX FIFO empty, this bit is set to 0.	0x0
12	SFIFONOTFULL: TX FIFO non-full	1: TX FIFO non-full 0: TX FIFO full	0x1
13	IBIRCV: Master receives IBI, HJ and MR requests.	1: Master receives IBI, HJ, MR requests. 0: Write 1 to this bit to clear	0x0
14	Reserved	–	–
15	ERR: Status error	1: Master in error (The error type	0x0

Bit	Name	Description	Reset Value
		can be viewed through the MERR register) 0: Master not in error (Write 1 to this bit to clear)	
23: 16	Reserved	–	–
30: 24	IBIADDRESS: Address of IBI, HJ and MR requests received by master	Requests address of IBI, HJ and MR received by master	0x00
31	Reserved	–	–

6.2.18 MIBIFORMCFG (0x8C)

Master IBI format configuration. When MIBIFORMCFG.DAMSB0 is 1 and MCONTROL.IBIRSPTYPE is 0, NOIBIMBYTE=1, then all the addresses in MIBIFORMCFG have IBI forced bytes except for those in MIBIFORMCFG; NOIBIMBYTE = =0, only the address in MIBIFORMCFG has an IBI forced byte.

Table 6-20 MIBIFORMCFG

Bit	Name	Description	Reset Value
5: 0	SADDRESS0: Slave address 0	Whether the slave at this address has a forced byte is determined by DAMSB0 and NOIBIMBYTE.	0x00
11: 6	SADDRESS1: Slave address 1	Whether the slave at this address has a forced byte is determined by DAMSB0 and NOIBIMBYTE.	0x00
17: 12	SADDRESS2: Slave address 2	Whether the slave at this address has a forced byte is determined by DAMSB0 and NOIBIMBYTE.	0x00
23: 18	SADDRESS3: Slave address 3	Whether the slave at this address has a forced byte is determined by DAMSB0 and NOIBIMBYTE.	0x00
29: 24	SADDRESS4: Slave address 4	Whether the slave at this address has a forced byte is determined by DAMSB0 and	0x00

Bit	Name	Description	Reset Value
		NOIBIMBYTE.	
30	DAMSB0: Whether the highest bit of the dynamic address is all 0.	1: The highest bit of the dynamic address is all 0. 0: The highest bit of the dynamic address is not all 0.	0x0
31	NOIBIMBYTE: No forced byte	1: The slave IBI of the corresponding address in this register has no forced bytes. 0: The slave IBI of the corresponding address in this register has forced bytes.	0x0

6.2.19 MIS (0x90)

User Interrupt enable set. To activate an interrupt enable, write 1 to its field in this register.

Table 6-21 MIS

Bit	Name	Description	Reset Value
7: 0	Reserved	-	-
8	SSTART	SSTART Interrupt enable. Slave pull down SDA bus to send START pattern.	0x0
9	MCONTROLFINISH	Master control operation completed interrupt enable	0x0
10	COMCOMPLETE	Communication completed Interrupt enable One communication completed	0x0
11	RFIFONOTEMPTY	RX FIFO non-empty interrupt enable	0x0
12	SFIFONOTFULL	TX FIFO non-full interrupt enable	0x0
13	IBIRCV	Interrupt enable of IBI, HJ, MR requests received by master	0x0
14	Reserved	-	-
15	ERR	Error interrupt enable	0x0
31: 16	Reserved	-	-

6.2.20 MIC (0x94)

User Interrupt enable clear. To clear an interrupt enable, write 1 to its field in this register..

Table 6-22 MIC

Bit	Name	Description	Reset Value
7: 0	Reserved	–	–
8	SSTART	SSTART interrupt enable clear	–
9	MCONTROLFINISH	Master control operation completed interrupt enable clear	–
10	COMCOMPLETE	Communication completed Interrupt enable clear	–
11	RFIFONOTEMPTY	RX FIFO non-empty interrupt enable clear	–
12	SFIFONOTFULL	TX FIFO non-full interrupt enable clear	–
13	IBIRCV	Clear interrupt enable of IBI, HJ, MR requests received by master	–
14	Reserved	–	–
15	ERR	Error interrupt enable clear	
31: 16	Reserved	–	–

6.2.21 MIM (0x98)

Interrupt mask. Return the status of enabled interrupts with the value of MSTs & MIS.

Table 6-23 MIM

Bit	Name	Description	Reset Value
7: 0	Reserved	–	–
8	SSTART	SSTART Interrupt mask	0x0
9	MCONTROLFINISH	Master control operation completed interrupt mask	0x0
10	COMCOMPLETE	Communication completed interrupt mask	0x0
11	RFIFONOTEMPTY	RX FIFO non-empty Interrupt mask	0x0

Bit	Name	Description	Reset Value
12	SFIFONOTFULL	TX FIFO non-full Interrupt mask	0x0
13	IBIRCV	Interrupt mask of IBI, HJ, MR requests received by master	0x0
14	Reserved	–	–
15	ERR	Error Interrupt mask	0x0
31: 16	Reserved	–	–

6.2.22 MERR (0x9C)

Master error. When there is an error, you can read the error type through the error register; when MERR register is set to 1, SSTS.ERR will also be set to 1. You can set the register to 0 by writing 1 to its field in this register.

Table 6-24 MERR

Bit	Name	Description	Reset Value
1: 0	Reserved	–	–
2	DAABANACK	Broadcast write address is NACKed during DAA.	0x0
3	I2CWNACK	Slave NACK I2C writes data.	0x0
8: 4	Reserved	–	–
9	DDRFRAME	DDR frame error. The start pattern and the first 4 bits of the CRC are not 4'hC.	0x0
10	DDRCRC5	DDR CRC5 error	0x0
11	DDRCOMMANDDATA	DDR command and data error, inconsistent TX and RX.	0x0
15: 12	Reserved	–	–
16	READEMPTY	When RX FIFO is empty, you still read the RX FIFO.	0x0
17	WRITEFULL	When TX FIFO is full, you still write data to TX FIFO.	0x0
18	Reserved	–	–
19	ERRREQUEST	Request error. You use	0x0

Bit	Name	Description	Reset Value
		the wrong request, e.g. the bus is in DAA and you requests to send START + ADDR (MCONTROL.REQUEST = 3'h1).	
20	COMTIMEOUT	Communication timeout. IP stays in a communication status other than the idle for 100us.	0x0
31: 21	Reserved	–	–

6.2.23 MDATACONTROL (0xAC)

Master data control; control and show the status of FIFO with depth of 16.

Table 6-25 MDATACONTROL

Bit	Name	Description	Reset Value
0	SFIFOCLR: Clear TX FIFO	Write 1 to clear TX FIFO	0x0
1	RFIFOCLR: Clear RX FIFO	Write 1 to clear RX FIFO	0x0
15: 2	Reserved	–	–
20: 16	SFIFOCNT: Number of data in TX FIFO	Number of data in TX FIFO	0x00
23: 21	Reserved	–	–
28: 24	RFIFOCNT: Number of data in RX FIFO	Number of data in RX FIFO	0x00
29	Reserved	–	–
30	SFIFOFULL: TX FIFO full	1: TX FIFO full 0: TX FIFO non-full	0x0
31	RFIFOEMPTY: RX FIFO empty	1: RX FIFO empty 0: RX FIFO non-empty	0x1

6.2.24 MTXB (0xB0)

Master TX data. You can write data to TX FIFO via this register.

Table 6-26 MTXB

Bit	Name	Description	Reset Value
7: 0	WDATA: Write data	Data to be written TX FIFO	–
8	LAST: Indicate whether this data is the last byte of this communication.	1: This byte is the last byte of this communication (the byte of the even of DDR can be the last byte). 0: This byte is not the last byte of this communication.	–
31: 9	Reserved	–	–

6.2.25 MTXBE (0xB4)

The master TX the last byte register. You can write the last byte of this communication to the TX FIFO via this register.

Table 6-27 MTXBE

Bit	Name	Description	Reset Value
7: 0	WDATA: Write data	The last byte to be written to the TX FIFO (the byte of the even in DDR can only be the last byte).	–
31: 8	Reserved	–	–

6.2.26 MRXB (0xC0)

Master read register. You can read data from RX FIFO via this register.

Table 6-28 MRXB

Bit	Name	Description	Reset Value
7: 0	RDATA: Read data	Read data from RX FIFO	0x00
31: 8	Reserved	–	–

6.2.27 DID (0xC4)

Device ID. You can read ID of this IP via this register.

Table 6-29 DID

Bit	Name	Description	Reset Value
1: 0	CLOCKNUMBER	0: Single clock domain device 1: Dual clock domain device 2: reserved 3: reserved	0x0
3: 2	ROLE	0: Support master 1: Support slave 2: Support master and slave 3: reserved	0x2
5: 4	FUNCTION	0: Support SDR 1: Support DDR 2: reserved 3: reserved	0x1
11: 6	VERSIONNUMBER	Version	0x00
31: 12	Reserved	–	–

7 Operating Flow

7.1 Master Operating Flow

7.1.1 SDR Read/Write

1. IP initialization configuration: Write MCFG register for IP initialization configuration.
 - MENABLE: Master enable
 - PPHIGH, PPLOWEXTRA, ODSCL, ODHIGHEQUALPP, I2CSCL: Configure the data rate of SDR read/write communication.
2. Interrupt enable: Write MIS register to enable the interrupt.
 - MCONTROLFINISH: Master control operation completion interrupt enable.
 - COMCOMPLETE: Communication completion interrupt enable.
 - RFIFONOTEMPTY: RX FIFO non-empty interrupt enable.
 - SFIFONOTFULL: TX FIFO non-full interrupt enable.
3. Read and write operations: Write MCONTROL register to enable IP to complete the read and write operations.
 - REQUEST: Write 1 to send START + ADDR operation.
 - COMTYPE: Write 0 to send I3C SDR communication.
 - DIRECTION: Write 0 for SDR write, write 1 for SDR read.
 - COMADDR: Slave address sent by the master.
 - READTERMCNT: End read communication by counting bytes.
4. Write data: Write the data to be sent through MTXB or MTXBE. The data can be written either before or after the read/write operation, and you can configure which byte is the last byte of this write

communication through MTXB or MTXBE. The master will end this communication after sending the last byte.

5. Interrupt reception: After receiving the interrupt, you can read the MSTS register to know what kind of interrupt is received.
 - MCONTROLFINISH interrupt: Indicate that the master has sent START + ADDR and the slave has ACKed/NACKed.
 - COMCOMPLETE interrupt: Indicate you can continue to send STOP (MCONTROL.REQUEST is 2) or RESTART + ADDR (MCONTROL.REQUEST is 1) after the read/write operation is completed.
 - SFIFONOTFULL interrupt: Indicate that the TX FIFO is full and you can not write data.
 - RFIFONOTEMPTY interrupt: Indicate that data has been received in the RX FIFO, and you can read the data in the RX FIFO through the MRXB register.

7.1.2 I²C Read/Write

The I²C read/write operation is similar to those of SDR by simply writing 1 to MCONTROL.COMTYPE to issue an I²C read/write operation.

7.1.3 ENTDAAs Operation

1. IP initialization configuration: Write MCFG register for IP initialization configuration.
 - MENABLE: Master enable
 - PPHIGH, PPLOWEXTRA, ODSCL, ODHIGHEQUALPP, I2CSCL: Configure the data rate of SDR read/write communication.
2. Interrupt enable: Write MIS register to enable the interrupt.
 - MCONTROLFINISH: Master control operation completion interrupt enable.
 - COMCOMPLETE: Communication completion interrupt enable.
 - RFIFONOTEMPTY: RX FIFO non-empty interrupt enable.
 - SFIFONOTFULL: TX FIFO non-full interrupt enable.
3. Read and write operations: Write MCONTROL register to enable IP to complete ENTDAAs operations.
 - REQUEST: If not in the ENTDAAs mode, write 4, the master will send START+7E/W+ACK+ENTDAAs+RSTART+7E/R+ 64-bit read data received from the slave; if in the ENTDAAs mode, write 4, the

master will send dynamic address+ACK+ RESTART+7E/R+ 64-bit read data received from the slave (7E/R will receive 64-bit read data from the slave if ACKed, and the master will automatically send STOP pattern if NACKed).

- COMTYPE: Write 0 to send I3C SDR communication.
- 4. Write dynamic address: Write the dynamic address needed by ENDAA via MTXB (LAST must be 0) (dynamic address is 7:1 bits of data, bit 0 has no role), dynamic address writing needs to be done after ENDAA (7E+R is ACKed).
- 5. Interrupt reception: After receiving the interrupt, you can read the MSTs register to know what kind of interrupt is received.
 - MCONTROLFINISH interrupt: Indicate that the master has issued the operation corresponding to a REQUEST of 4.
 - COMCOMPLETE interrupt: Indicate that the bus has entered the IDLE after this ENDAA operation.
 - RFIFONOTEMPTY interrupt: Indicate that data has been received in the RX FIFO, and you can read the data in the RX FIFO through the MRXB register.

7.1.4 Other CCC Operation

The other CCC operations are implemented through SDR read and write.

7.1.5 IBI/HJ Operation

1. IP initialization configuration: Write MCFG register for IP initialization configuration.
 - MENABLE: Master enable
 - PPHIGH, PPLOWEXTRA, ODSCL, ODHIGHEQUALPP, I2CSCL: Configure the data rate of SDR read/write communication.
2. Interrupt enable: Write MIS register to enable the interrupt.
 - MCONTROLFINISH: Master control operation completion interrupt enable.
 - COMCOMPLETE: Communication completion interrupt enable.
 - RFIFONOTEMPTY: RX FIFO non-empty interrupt enable.
 - SFIFONOTFULL: TX FIFO non-full interrupt enable.
 - SSTART: Interrupt enable of SDA signal pulled down by slave.
 - IBIRCV: Interrupt enable of IBI, HJ, MR requests received by

master.

3. IBI, HJ, MR request ACK configuration.
 - MCONTROL.IBIRSPTYPE: IBI ACK type configuration, which can be configured as Auto ACK, Auto NACK and manual. When configured as auto ACK, you can configure whether to has IBI forced byte after IBI ACK or not with MIBIFORMCFG register; when configured as manual, I3C bus communication will be suspended at IBI address, and you can use MCONTROL.REQUEST (set to 3) and MCONTROL.IBIRSPTYPE to ACK manually IBI (MCONTROL.IBIRSPTYPE 0 is without forced byte ACK; MCONTROL.IBIRSPTYPE 1 is with NACK; MCONTROL.IBIRSPTYPE is 2 with forced byte ACK).
4. IBI, HJ and MR request operations.
 - After the slave pulls down SDA, the master receives the SSTART interrupt.
 - After the slave sends the IBI, HJ, MR request address, the master receives the IBIRCV interrupt.
 - When the IBIRCV interrupt is received, you can read the MSTS register to know if it is a manual ACK. You can initiate a manual ACK using MCONTROL.REQUEST (set to 3), and the IP can initiate an IBI ACK automatically.
 - After IBI, HJ, MR request completed, you can receive COMCOMPLETE interrupt.
 - You can read IBI data from the RX FIFO when the RFIFONOTEMPTY interrupt is received.

7.1.6 DDR Read/Write

1. IP initialization configuration: Write MCFG register for IP initialization configuration.
 - MENABLE: Master enable
 - PPHIGH, PPLOWEXTRA, ODSCL, ODHIGHEQUALPP, I2CSCL: Configure the data rate of DDR read/write communication.
2. Interrupt enable: Write MIS register to enable the interrupt.
 - MCONTROLFINISH: Master control operation completion interrupt enable.
 - COMCOMPLETE: Communication completion interrupt enable.
 - RFIFONOTEMPTY: RX FIFO non-empty interrupt enable.

- SFIFONOTFULL: TX FIFO non-full interrupt enable.
3. Read and write operations: Write MCONTROL register to enable IP to complete the read and write operations.
 - REQUEST: Write 1, when I3C communication is not in the DDR mode, the master will send START+7E/W+ACK+ENTHDR0 CCC+DDR COMMAND; when in the DDR mode, the master will send HDR Restart+ DDR COMMAND.
 - COMTYPE: Write 2 to send I3C DDR communication.
 - COMADDR: The slave address in the DDR COMMAND sent by the master.
 - READTERMCNT: End read communication by counting bytes, must be an even number.
 - DDRENDWITHCRC: Read and write end earlier with/without CRC.
 4. Write data: You must write the command byte of DDR COMMAND through MTXB before or after initiating the read/write operation, then you can write the data to be sent through MTXB or MTXBE, and can configure which byte is the last byte through MTXB or MTXBE (must end at an even number of bytes); the master completes the communication after the last two bytes are sent.
 5. Interrupt reception: After receiving the interrupt, you can read the MSTs register to know what kind of interrupt is received.
 - MCONTROLFINISH interrupt: Indicate that the master has issued START+7E/W+ACK+ENTHDR0 CCC+DDR COMMAND or HDR Restart+ DDR COMMAND or DDR EXIT pattern.
 - COMCOMPLETE interrupt: Indicate you can continue to send HDR EXIT (MCONTROL.REQUEST is 6) or HDR Restart+ DDR COMMAND (MCONTROL.REQUEST is 1) after the read/write operation is completed.
 - SFIFONOTFULL interrupt: Indicate that the TX FIFO is full and you can not write data.
 - RFIFONOTEMPTY interrupt: Indicate that data has been received in the RX FIFO, and you can read the data in the RX FIFO through the MRXB register.

7.2 Slave Operating Flow

7.2.1 SDR Read/Write

1. IP initialization configuration: Write SCFG register for IP initialization configuration.
2. Write slave vendor fixed value or random value: Write vendor fixed value or random value of the slave by writing SVFVORRV register.
3. Write slave BCR and DCR value: Write BCR and DCR values by writing SBCRANDDCR register.
4. Write slave MIPI manufacturer ID: Write MIPI manufacturer ID by writing SMMID register.
5. Interrupt enable: Write SIS register to enable the interrupt.
6. Read and write operations: You can know the bus status by interrupts and perform read and write operations by register.
 - Receiving the START interrupt means that the slave detects the START pattern on the I3C bus.
 - Receiving the MATCHEDSAORDA interrupt means that the address on the I3C bus matches its static address or dynamic address, and at this time you can read the SSTS register to know it is a read or write.
 - Receiving RFIFONOTEMPTY interrupt means that there is data in the RX FIFO and you can read the RX data through the SRXB register.
 - Receiving DATANEED interrupt indicates that the master is performing read operation. You need to write the data to be sent through STXB register, otherwise the IP will automatically NACK the master read or end it at the ninth bit of the data.
 - Receiving the STOP interrupt means that the slave detects the STOP pattern on the I3C bus.

7.2.2 ENTDA A Operation

1. IP initialization configuration: Write SCFG register for IP initialization configuration.
2. Write slave vendor fixed value or random value: Write vendor fixed value or random value of the slave by writing SVFVORRV register.
3. Write slave BCR and DCR value: Write BCR and DCR values by writing SBCRANDDCR register.

4. Write slave MIPI manufacturer ID: Write MIPI manufacturer ID by writing SMMID register.
5. Interrupt enable: Write SIS register to enable the interrupt.
6. ENTDA operation: You can know the bus status by interrupts and perform ENTDA by register.
 - Receiving the START interrupt means that the slave detects the START pattern on the I3C bus.
 - Receiving the MATCHEDBA interrupt indicates that the address on the I3C bus is a broadcast write address.
 - Receiving CCCAH interrupt means that CCC command has been processed automatically, at this time, you can know that the slave has entered to DAA mode by register status.
 - IP automatically sends 64-bit read data and receives dynamic address.
 - Receiving the STOP interrupt means that the slave detects the STOP pattern on the I3C bus, then ENDAA is completed.

7.2.3 Other CCC Operation

The other CCC operations are similar to the SDR read and write operation; the difference is that you can receive CCCRCV interrupt, indicating that CCC commands other than ENDAA are received. You can read the CCC from the RX FIFO.

7.2.4 IBI/HJ/MR Operation

1. IP initialization configuration: Write SCFG register for IP initialization configuration.
2. Write slave vendor fixed value or random value: Write vendor fixed value or random value of the slave by writing SVFVORRV register.
3. Write slave BCR and DCR value: Write BCR and DCR values by writing SBCRANDDCR register.
4. Write slave MIPI manufacturer ID: Write MIPI manufacturer ID by writing SMMID register.
5. Interrupt enable: Write SIS register to enable the interrupt.
6. Perform IBI/HJ/MR operation: Write SCONTROL register to perform /IBI/HJ/MR operation (REQUEST is 1 to issue IBI; REQUEST is 2 to issue MR; REQUEST is 3 to issue HJ; SCFG. HJWAIT should be 1 for HJ).

7. Perform IBI/HJ/MR operation: You can know the bus status by receiving interrupts and perform IBI/HJ/MR operation via register.
 - Receiving the START interrupt means that the slave detects the START pattern on the I3C bus.
 - Receiving the REQUEST interrupt indicates that the IBI/HJ/MR requests have been sent, and you can know whether the master ACK IBI/HJ/MR or not via the register.
 - Receiving the DATANEED interrupt indicates that the IBI request requires data without forced bytes (write forced bytes through the SCONTROL register), and you can write data to the TX FIFO or not, if not, the slave will end the IBI operation.
 - Receiving the STOP interrupt means that the slave detects the STOP pattern on the I3C bus, then this communication is completed.

8 Interrupt

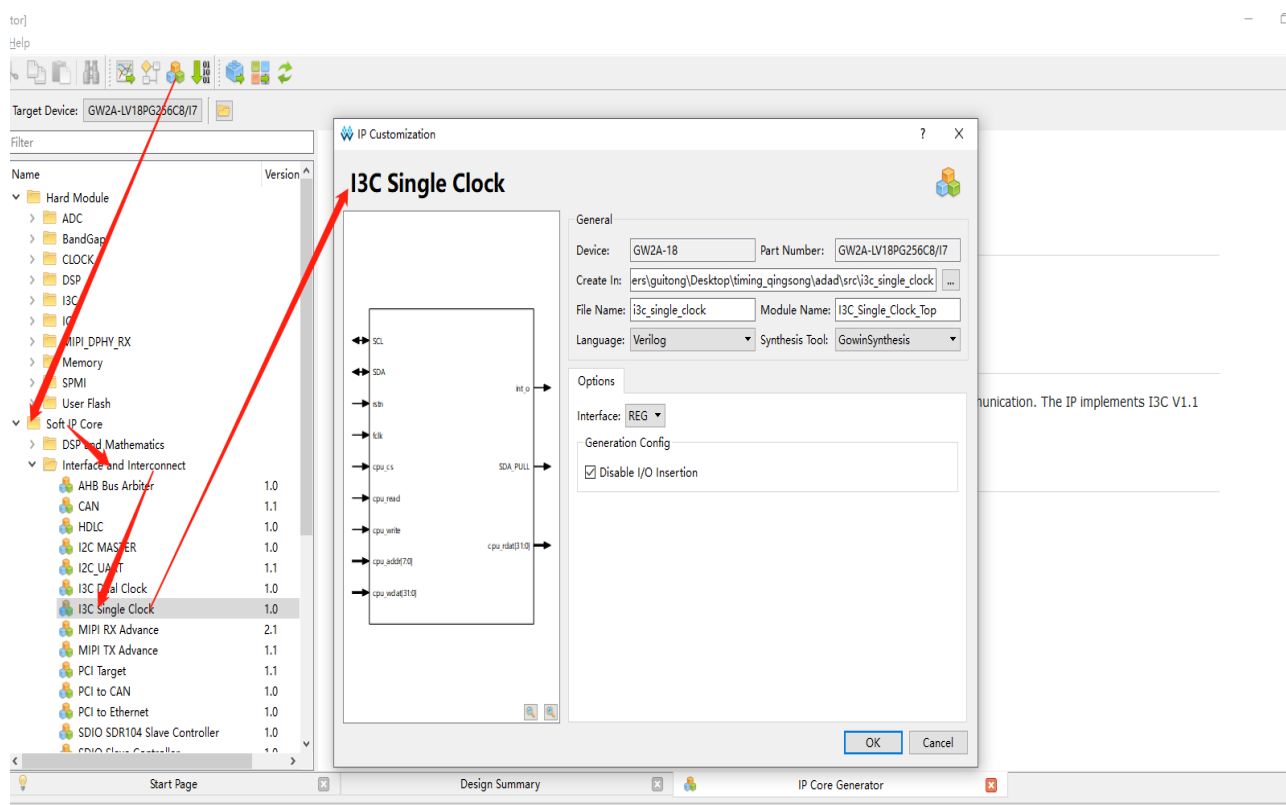
The I3C Single Clock IP implements active-low interrupt. It is a standard mechanism where all enabled single interrupt sources perform an OR operation to produce the interrupt output. The interrupt sources are defined in the MIS and SIS registers, and the interrupt enable has a corresponding mask.

9 GUI

9.1 IP Generation

Click "IP Core Generator" from the "Tools" menu in the Gowin Software and to call and configure Gowin I3C Single Clock IP. Or you can start via the icons as shown in Figure 9-1.

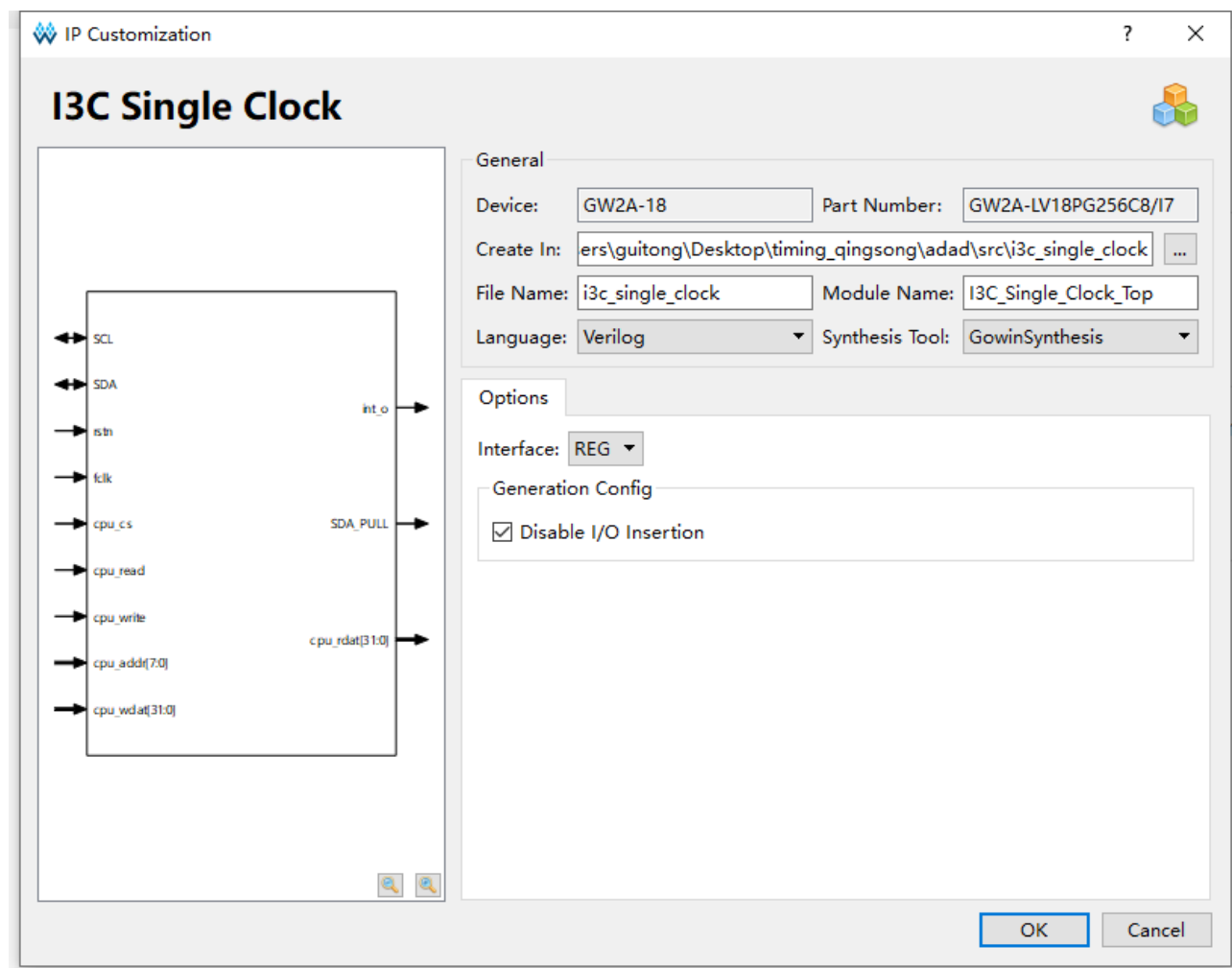
Figure 9-1 Call Gowin I3C Single IP Via Tool Bar



9.2 Configuration Interface

The configuration interface of Gowin I3C Single Clock IP is as shown in Figure 9-2.

Figure 9-2 Options



The option description is shown in Table 9-1.

Table 9-1 Configuration Options

Options		Description
Interface	REG	Configure the IP to REG interface
	APB	Configure the IP to APB interface

10 Reference Design

Please see the I3C Single Clock Reference Design for details at Gowinsemi website.

