



Gowin I3C DDR eXtension **User Guide**

IPUG524-1.0E,2018-12-17

Copyright©2018 Guangdong Gowin Semiconductor Corporation. All Rights Reserved.

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[®], LittleBee[®], Arora[™], and the GOWINSEMI logos are trademarks of GOWINSEMI 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, as described at www.gowinsemi.com.cn. 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

Date	Revision	Description
2018/12/17	1.0E	Initial version released.

Contents

Contents.....	i
List of Figures.....	iv
List of Tables.....	vi
1 About This Manual	1
1.1 Manual Content	1
1.2 Applicable Product.....	1
1.3 Related Documents	1
1.4 Terms and Abbreviations	2
1.5 Technical Support and Feedback	3
2 Overview	4
2.1 I3C DDR EXtension IP Introduction.....	4
3 Features and Performance	5
3.1 Main Features.....	5
3.2 Resource Utilization.....	6
3.3 Clock Frequency.....	6
4 Functional Description.....	7
4.1 Operating Mode	7
4.2 Timing	7
4.2.1 Write Data.....	7
4.2.2 Read Timing.....	8
4.3 Master Communication Programming	8
4.3.1 SDR Write Operation	8
4.3.2 SDR Read Operation.....	10
4.3.3 Dynamic Address Allocation	10
4.3.4 DDR Write Operation	11
4.3.5 DDR Read Operation	12
4.4 Slave Communication Programming	13
4.4.1 SDR Write Operation	13
4.4.2 SDR Read Operation.....	14

4.4.3 Dynamic Address Allocation	16
4.4.4 DDR Write Operation.....	17
4.4.5 DDR Read Operation	17
4.5 Interrupt Flag	18
4.5.1 DONE Flag	18
4.5.2 NACK Flag.....	18
4.5.3 STA Flag	18
4.5.4 SR Flag.....	18
4.5.5 STO Flag	18
4.5.6 DA Flag	18
4.5.7 LOST Flag	18
4.5.8 DDR_DONE Flag.....	19
4.5.9 DDR_SR Flag.....	19
4.5.10 DDR_EXIT Flag	19
4.6 Register Map	19
4.6.1 Basic Control Register 1	20
4.6.2 Basic Control Register 2	22
4.6.3 Call Control Register	23
4.6.4 Hang Up Control Register	24
4.6.5 Pre-scale Register	24
4.6.6 SDR Message Length Register	25
4.6.7 DDR Message Length Register.....	25
4.6.8 Slave Static Address.....	25
4.6.9 Byte Transferred	26
4.6.10 Word Transferred.....	26
4.6.11 SDR Call Address	27
4.6.12 DDR Call Address.....	27
4.6.13 Cache Write Pointer Low	27
4.6.14 Cache Write Pointer High	28
4.6.15 Cache Read Pointer Low.....	28
4.6.16 Cache Read Pointer High.....	28
4.6.17 Cache Pointer Reset Register	29
4.6.18 Status Register	29
4.6.19 Interrupt Flag Register 1	29
4.6.20 Interrupt Flag Register 2.....	30
5 Structure, Interface, and Configuration.....	32
5.1 Microstructure	32
5.2 Interface.....	34
5.2.1 Block Diagram	34
5.2.2 Interface Description.....	34
5.2.3 Configurable Parameters.....	35

6 Compile, Configuration, and Files	36
6.1 Compile I3C DDR EXtension.....	36
6.2 Generated Files	38
6.2.1 File Structure	38

List of Figures

Figure 4-1 Write Timing Diagram	8
Figure 4-2 Reading Sequential Indication.....	8
Figure 4-3 SDR Write Master Process.....	9
Figure 4-4 Timing of SDR Write Master	9
Figure 4-5 SDR read Master Process.....	10
Figure 4-6 Timing of SDR Read Master.....	10
Figure 4-7 Dynamic Address Assignment Master Process.....	11
Figure 4-8 Master Timing Diagram of Dynamic Address Assignment	11
Figure 4-9 Master Timing Diagram a of DDR Write Operation	12
Figure 4-10 Master Timing Diagram b of DDR Write Operation	12
Figure 4-11 Timing Diagram of DDR Read Master	12
Figure 4-12 Flow Chart of SDR Write Slave	13
Figure 4-13 Timing Diagram of SDR Write Operation Slave	14
Figure 4-14 Slave Flow Chart of SDR Read Operation	15
Figure 4-15 Slave timing diagram of SDR read operation	15
Figure 4-16 Slave Flowchart of Dynamic Address Assignment	16
Figure 4-17 Slave Timing Diagram a of Dynamic Address Allocation.....	17
Figure 4-18 Dynamic Address Allocation Slave Timing Diagram b.....	17
Figure 4-19 Slave Timing Diagram a of DDR Write Operation	17
Figure 4-20 Slave Timing Diagram b of DDR Write Operation	17
Figure 4-21 Slave Timing Diagram of DDR Read Operation	18
Figure 4-22 Linear Address Division Diagram	19
Figure 4-23 IP Interface Interaction Mode Diagram.....	19
Figure 5-1 Micro-structure Block Diagram	33

Figure 5-2 I3C DDR EXtension Block Diagram	34
Figure 6-1 FPGA RTL Project Initial Interface Diagram	36
Figure 6-2 Call I3C DDR EXtension	37
Figure 6-3 I3C DDR EXtension Configuration Interface	37
Figure 6-4 Generate I3C DDR EXtension Diagram	38
Figure 6-5 IP Core Generator Generated File	38

List of Tables

Table 1-1 Terms and Abbreviations	2
Table 2-1 I3C DDR EXtension IP	4
Table 3-1 Resource Utilization	6
Table 4-1 Control Register Name and Address	20
Table 5-1 I3C DDR EXtension Interface Description	34
Table 5-2 Configurable Parameters	35

1 About This Manual

1.1 Manual Content

The Gowin I3C DDR eXtension User Guide mainly includes function introduction, signal definition, working principle, GUI call, etc., which are designed to help users quickly understand the features and usage of Gowin I3C DDR eXtension.

1.2 Applicable Product

The information described in this manual applies to the following products:

1. GW1N series of FPGA products
2. GW1NR series of FPGA products
3. GW2A series of FPGA products
4. GW2AR series of FPGA products

Note!

Since the IP requires two BSRAM, some of the devices listed above may not be supported due to too small resource.

1.3 Related Documents

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

1. GW1N Series of FPGA Product Data Sheet
2. GW1NR Series of FPGA Product Data Sheet
3. GW2A Series of FPGA Product Data Sheet
4. GW2AR Series of FPGA Product Data Sheet
5. Gowin YunYuan Software User Guide

1.4 Terms and Abbreviations

The abbreviations and terminology used in this manual are as shown in 错误!未找到引用源。 below.

Table 1-1 Terms and Abbreviations

Terms and abbreviations	Full name
ACK	ACKnowledge
NACK	Not ACKnowledge
SDR	Single Data Rate
HDR	High Data Rate
DDR	Double Data Rate
SRAM	Static Radom Access Memory
SCL	Serial CLock
SDA	Serial DAta
CCC	Common Command Code
CRC	Cyclic Redundancy Check
DAR	Dynamic Address Request
DCR	Device Characteristics Register
BCR	Bus Characteristics Register
BER	Bit Error Rate
IBI	In-Band Interrupt
LCR	Legacy Characteristics Register
Mbps	Megabits per second
MHz	Mega Hertz
MID	MIPI Manufacturer Flag
OD	Open Drain
sysCLK	SYStem CLock
CE	Clock Enable
WRE	Write Enable
DI	Data Input
DO	Data Output
AD	ADdress

1.5 Technical 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 using the information provided below.

URL: www.gowinsemi.com

E-mail: support@gowinsemi.com

Tel: +86 755 8262 0391

2Overview

2.1 I3C DDR EXtension IP Introduction

The Gowin I3C DDR eXtension IP is used in serial communication interfaces and serial sensor interfaces to receive or transmit data. I3C DDR eXtension provides with synchronized SRAM interface I3C bus communication controller. The I3C DDR eXtension IP is shown in Table 2-1.

Table 2-1 I3C DDR EXtension IP

I3C DDR eXtension IP	
IP core application	
Chip supported	GW1N, GW1NR series GW2A, GW2AR series
Logical resource	See Table 3-1
Delivery document	
Designing documents	Verilog (encrypted)
Reference design	Verilog
TestBench	Verilog
Test design flow	
Integrated software	Synplify_Pro
Application	GoWinYunYuan

3Features and Performance

3.1 Main Features

- Only two lines are required to implement the communication system of single-master and multi-slave
 - Bus clock 12.5MHz;
 - Up to 11 devices are online at the same time;
 - Use a normal IO PAD with a 4mA driver (I2C typically requires a specially designed IO);
 - Slave uses the clock on the bus as the clock for the peripheral (so the device can design an internal low or inaccurate clock as needed);
 - Built-in CDC module, the communication controller logic works in the SCL clock domain, and the user interface works in the system clock domain;
 - Slave and Master can be actively terminated when reading data from the slave, so there is no need to know the length of the read in advance (I2C and SPI require the master to know the length of the read before reading).
- Support multiple transmission modes: SDR and HDR
 - The maximum data rate of SDR mode is 12.5Mbps
 - The maximum data rate of DDR mode is 25Mbps.
- Support the latest I3C standards
- Support I3C CCC commands
 - Support dynamic address configuration
 - Other optional CCC commands.
- Support IBI, hot-Join
 - Allow Slave to proactively notify the Master;
 - Allow multiple slaves to notify the Master at the same time, resolve conflicts through their respective dynamic addresses and arbitration mechanisms;
 - No additional interrupt lines are needed, built on top of the two bus mechanisms.

- Support command queuing
- Support for up to 4KB cache (SDR 2KB, DDR 4KB)
- Support synchronous SRAM user interface

3.2 Resource Utilization

Implement RTL code in Verilog language. Performance and resource utilization may vary depending on the density, speed, and grade of the device being used.

Taking Gowin GW1N-9 series FPGA as an example, the resource utilization situation is shown in Table 3-1. For application verification on other Gowin FPGA, please pay attention to the post-release information.

Table 3-1 Resource Utilization

Device series	Speed grade	Device name	Resource utilization	Remarks
GW1N-9	-5	LUT	1059	-
		ALU	93	
		REG	401	
		BSRAM	2	

3.3 Clock Frequency

This IP is designed by using a fixed clock frequency with a frequency of 25MHz. At this frequency, IP can reach the standard I3C bus speed. At this frequency, IP can reach the standard I3C bus speed. Users can increase or decrease the system clock frequency according to actual needs, resulting in higher bus speeds or lower power consumption. The frequency can be increased up to 150%, and choosing a chip with higher performance can help to further increase the frequency.

4Functional Description

4.1 Operating Mode

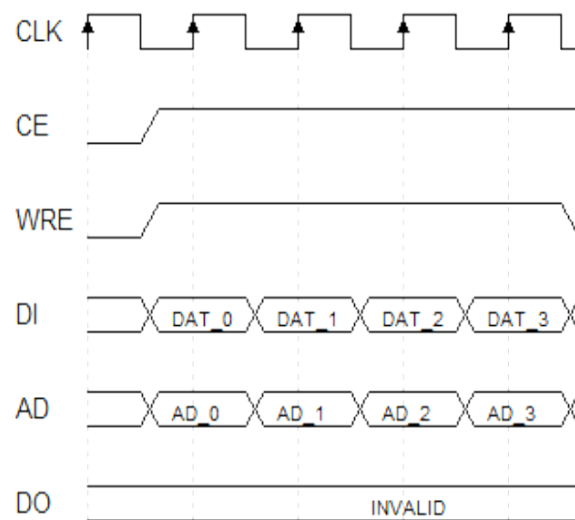
This soft core provides the user with a synchronous SRAM interface. The user configures the IP runtime parameters by writing control registers to control the IP communication behavior; Exchange communication data by reading and writing cache; The IP notifies the user of the event occurring on the bus through the interrupt, and the user processes the interrupt event through the interrupt service function to complete the interaction with the IP, thereby completing the bus communication.

In order to achieve high data throughput, the soft core is equipped with a cache of up to 4KB. In order to prevent mutual coverage of read and write data during large data communication, the cache is divided into two parts: read buffer and write buffer. Note that reading means that the Master reads Slave, and writing means that the Master writes Slave. Therefore, in the data transfer of a read operation, Slave fetches data from the read buffer and sends it to the bus, and Master charges the data on the bus to the write buffer; In a data transfer of a write operation, Master fetches data from the read buffer and sends it to the bus. Slave charges the data on the bus to the write buffer.

4.2 Timing

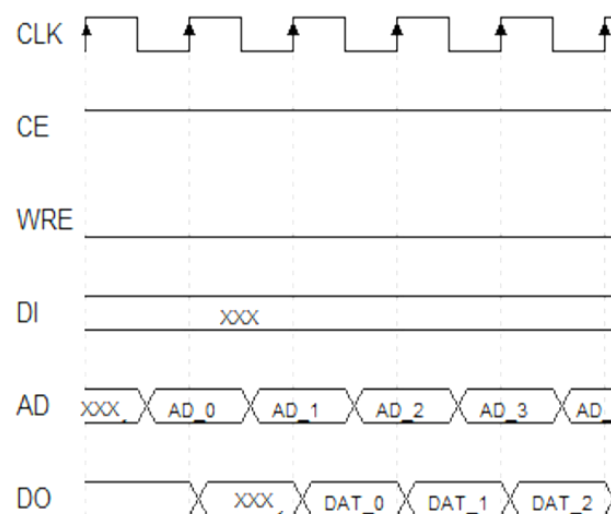
4.2.1 Write Data

The timing for writing data to IP on the user interface is shown in Figure 4-1.

Figure 4-1 Write Timing Diagram

4.2.2 Read Timing

The timing for reading data on the user interface is shown in Figure 4-2.

Figure 4-2 Reading Sequential Indication

4.3 Master Communication Programming

4.3.1 SDR Write Operation

When the write operation is performed in the SDR mode, the master communication flow is completed as shown in Figure 4-3 and Figure 4-4.

Figure 4-3 SDR Write Master Process

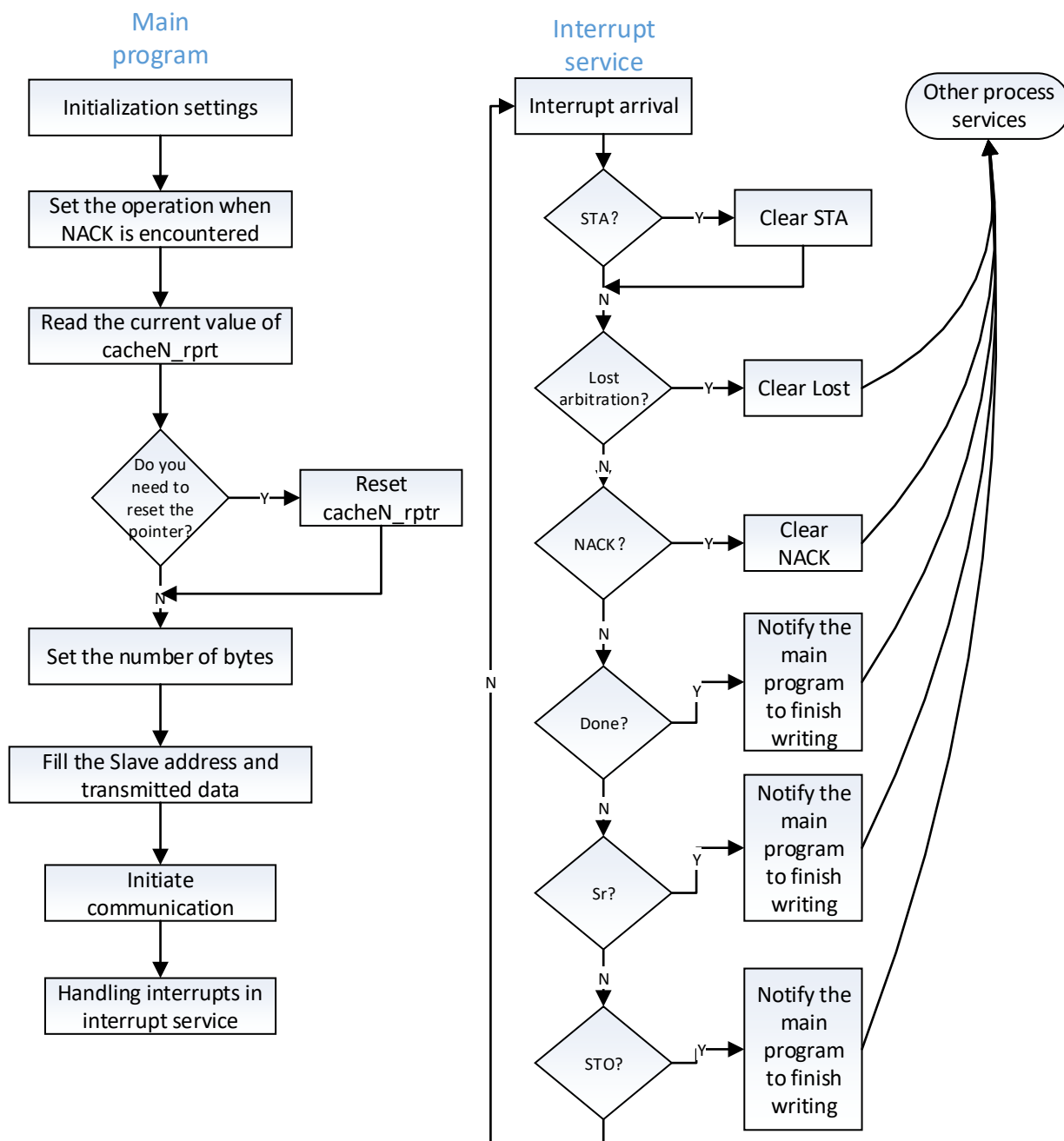
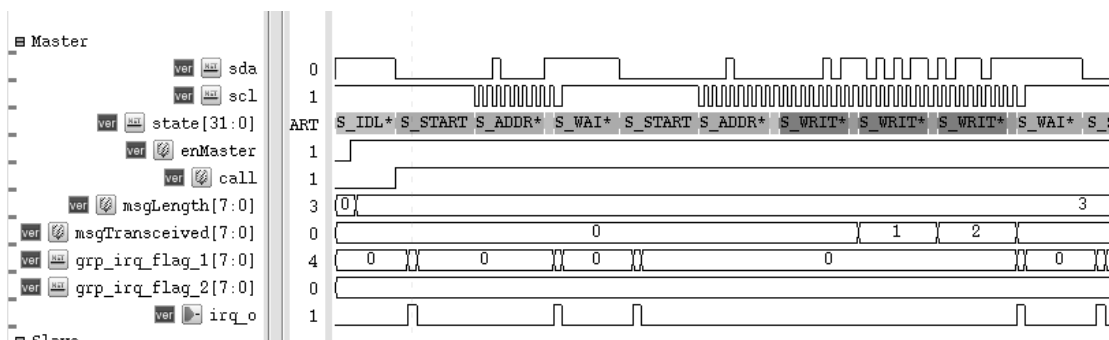


Figure 4-4 Timing of SDR Write Master



4.3.2 SDR Read Operation

When the read operation is performed in the SDR mode, the master process is as shown in Figure 4-5 and Figure 4-6

Figure 4-5 SDR read Master Process

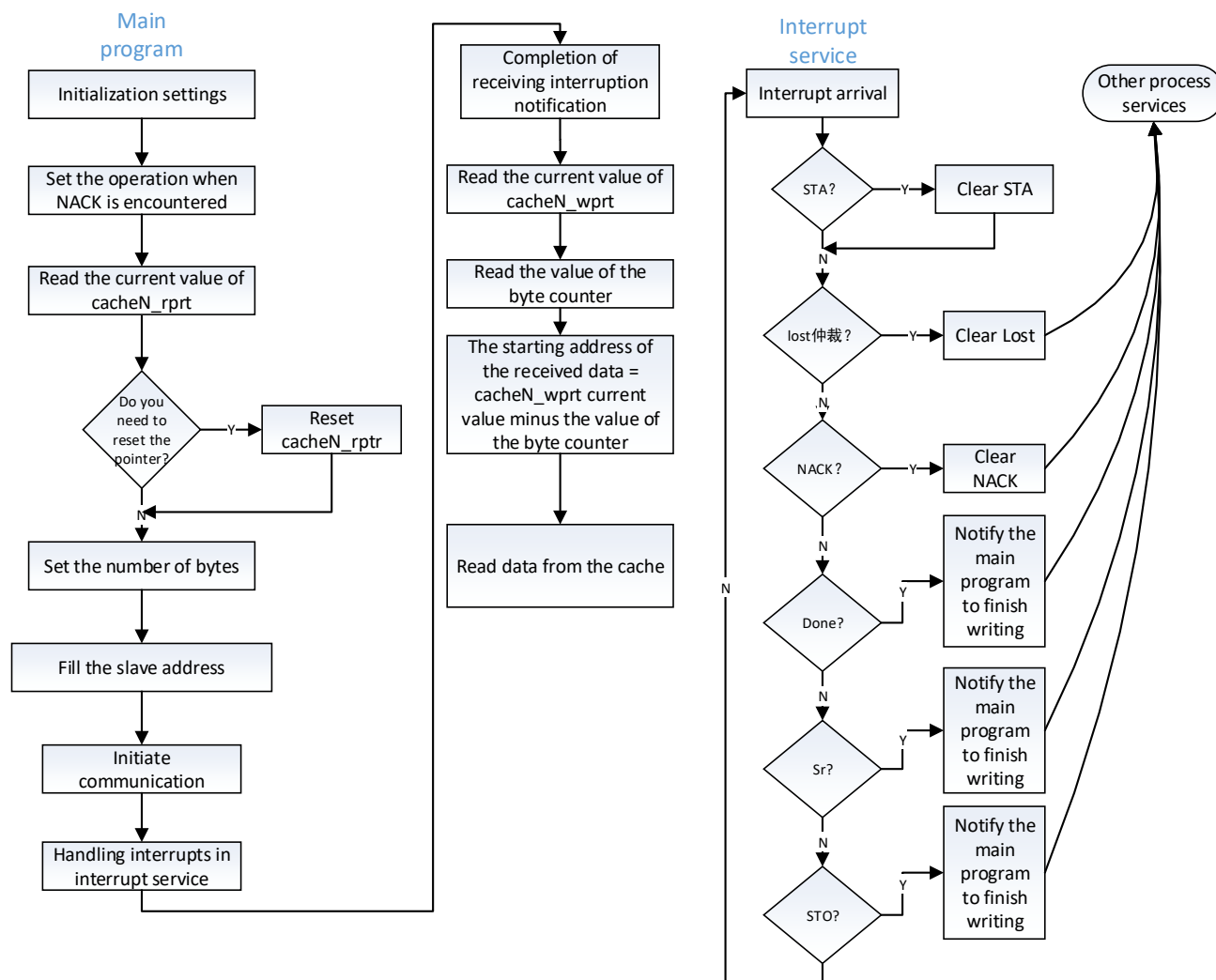
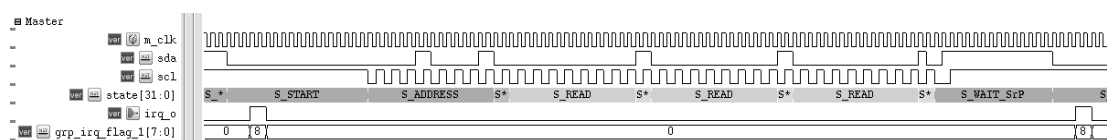


Figure 4-6 Timing of SDR Read Master



4.3.3 Dynamic Address Allocation

When the system needs to perform the allocation of dynamic address, the master communication flow is shown in Figure 4-7 and Figure 4-8.

Figure 4-7 Dynamic Address Assignment Master Process

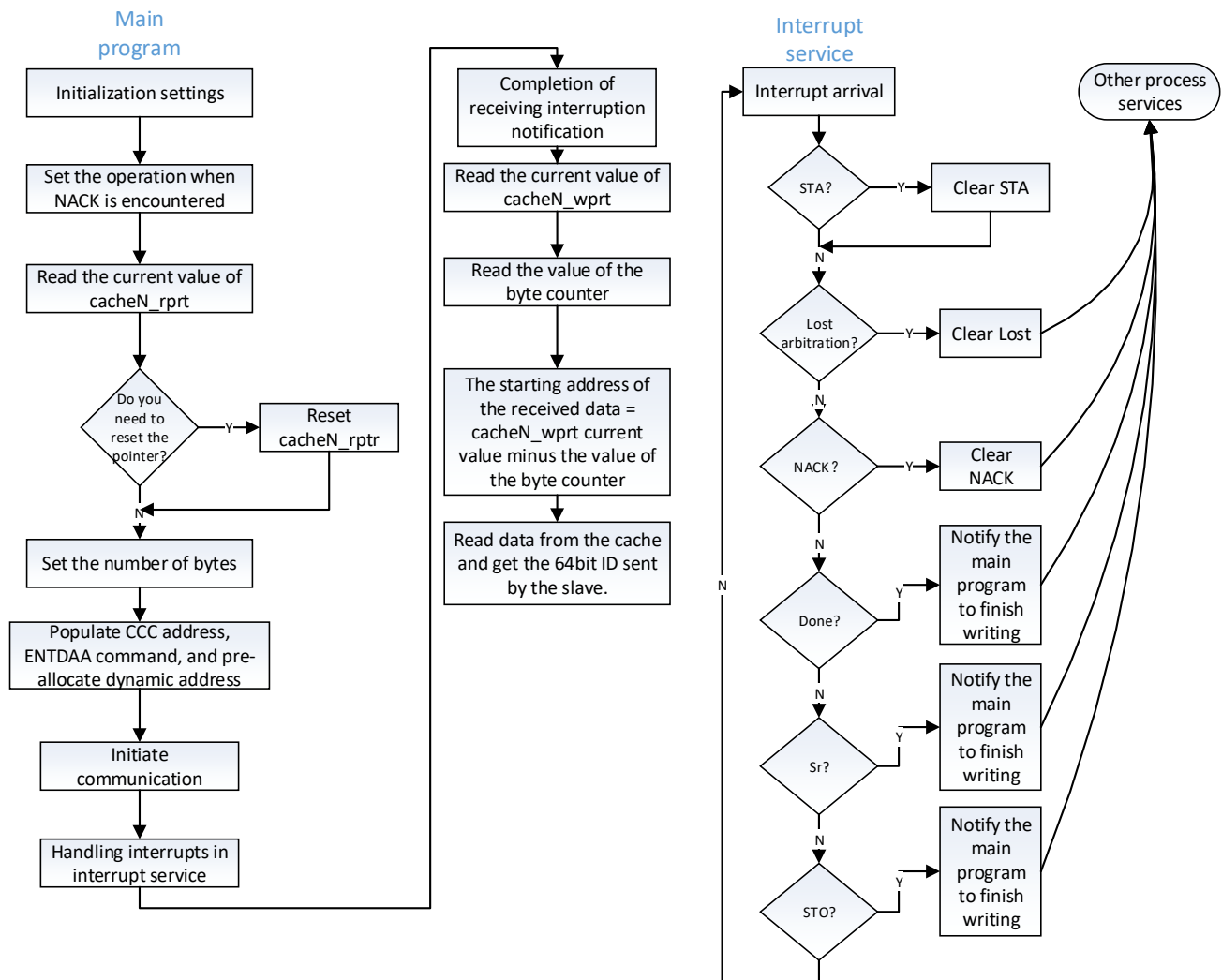
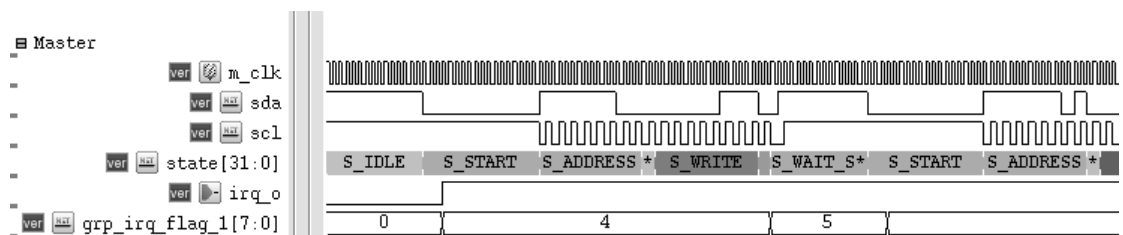


Figure 4-8 Master Timing Diagram of Dynamic Address Assignment



4.3.4 DDR Write Operation

The DDR mode write operation needs to first send the CCC command to the DDR mode by using the SDR mode, and the communication programming is the same as the SDR write operation, as shown in Figure 4-9 and Figure 4-10. Only the communication waveform diagram is drawn here.

4.4 Slave Communication Programming

4.4.1 SDR Write Operation

When writing in SDR mode, slave communication flow is shown in Figures 4-12 and 4-13.

Figure 4-12 Flow Chart of SDR Write Slave

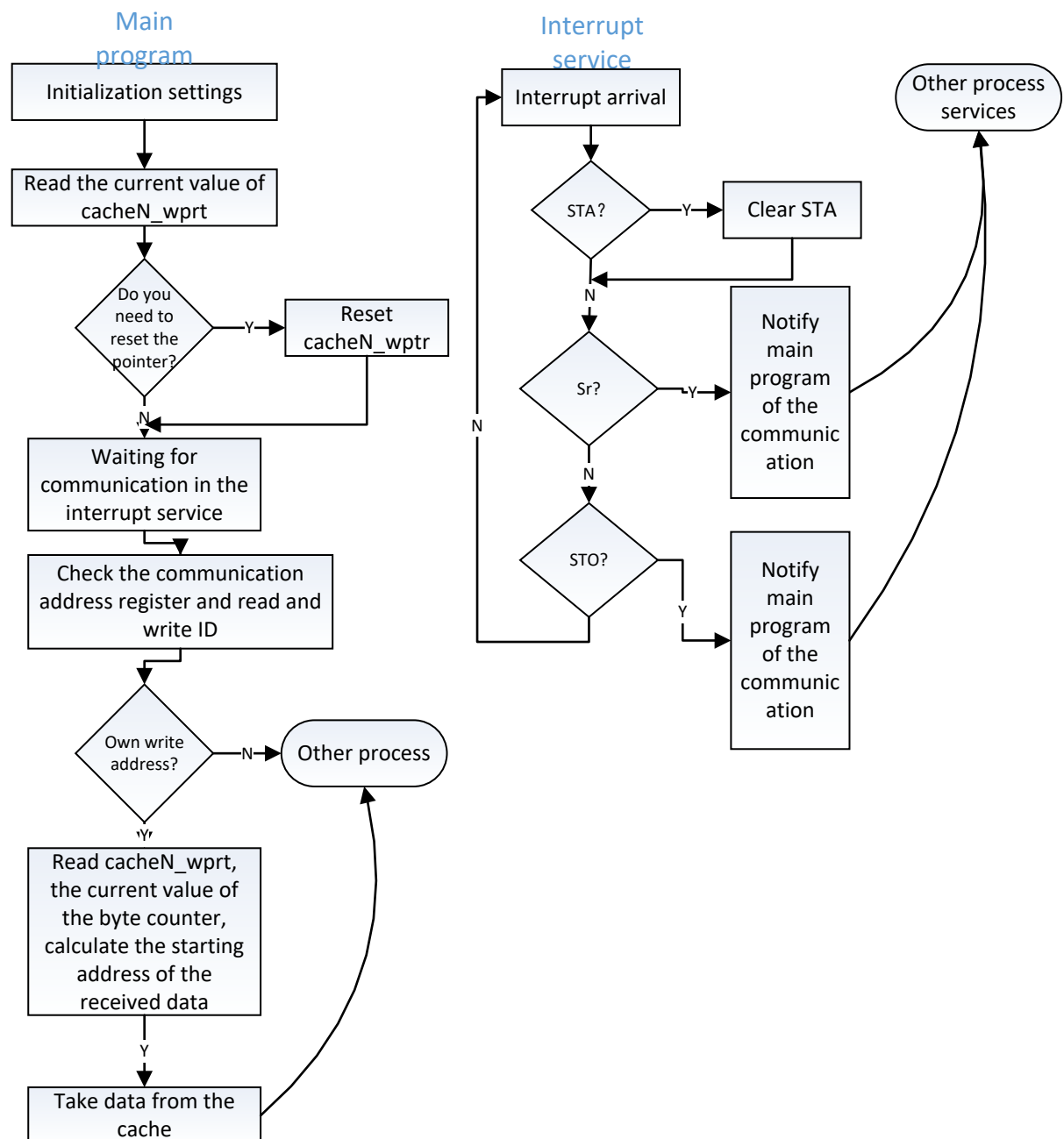
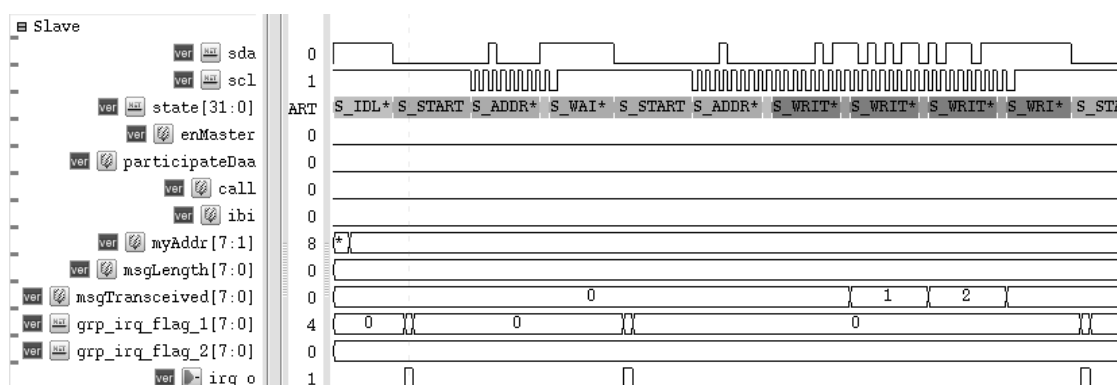


Figure 4-13 Timing Diagram of SDR Write Operation Slave



4.4.2 SDR Read Operation

When the read operation is performed in the SDR mode, the slave communication flow is as shown in Figure 4-14 and Figure 4-15.

Figure 4-14 Slave Flow Chart of SDR Read Operation

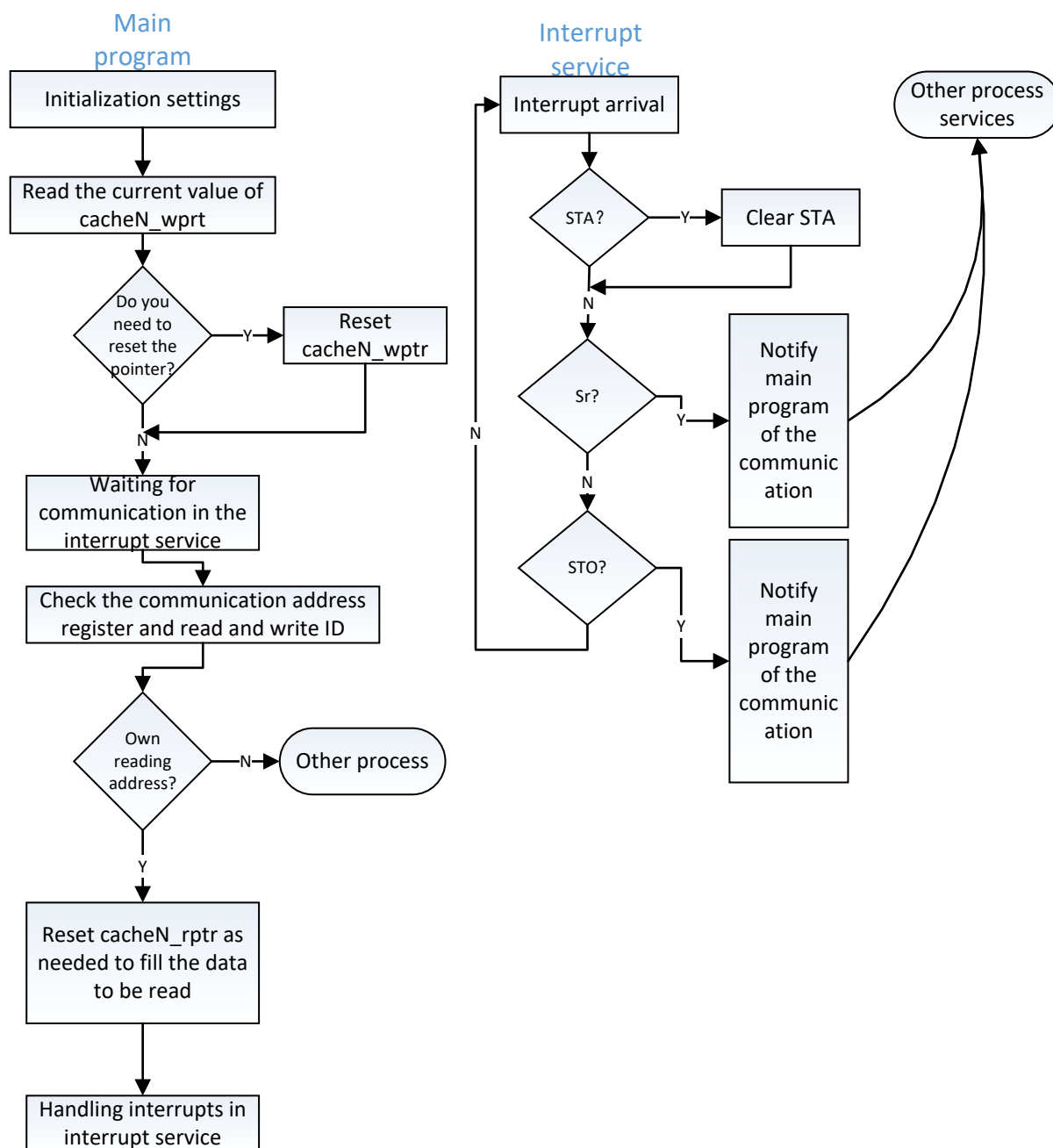
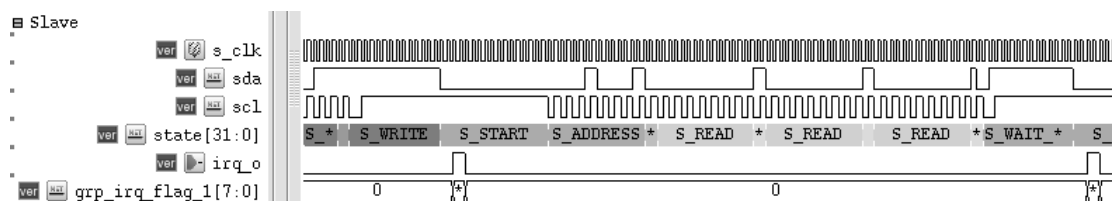


Figure 4-15 Slave timing diagram of SDR read operation



4.4.3 Dynamic Address Allocation

When the read operation is performed in the mode, the master communication flow is as shown in Figure 4-16, Figure 4-17, and Figure 4-18.

Figure 4-16 Slave Flowchart of Dynamic Address Assignment

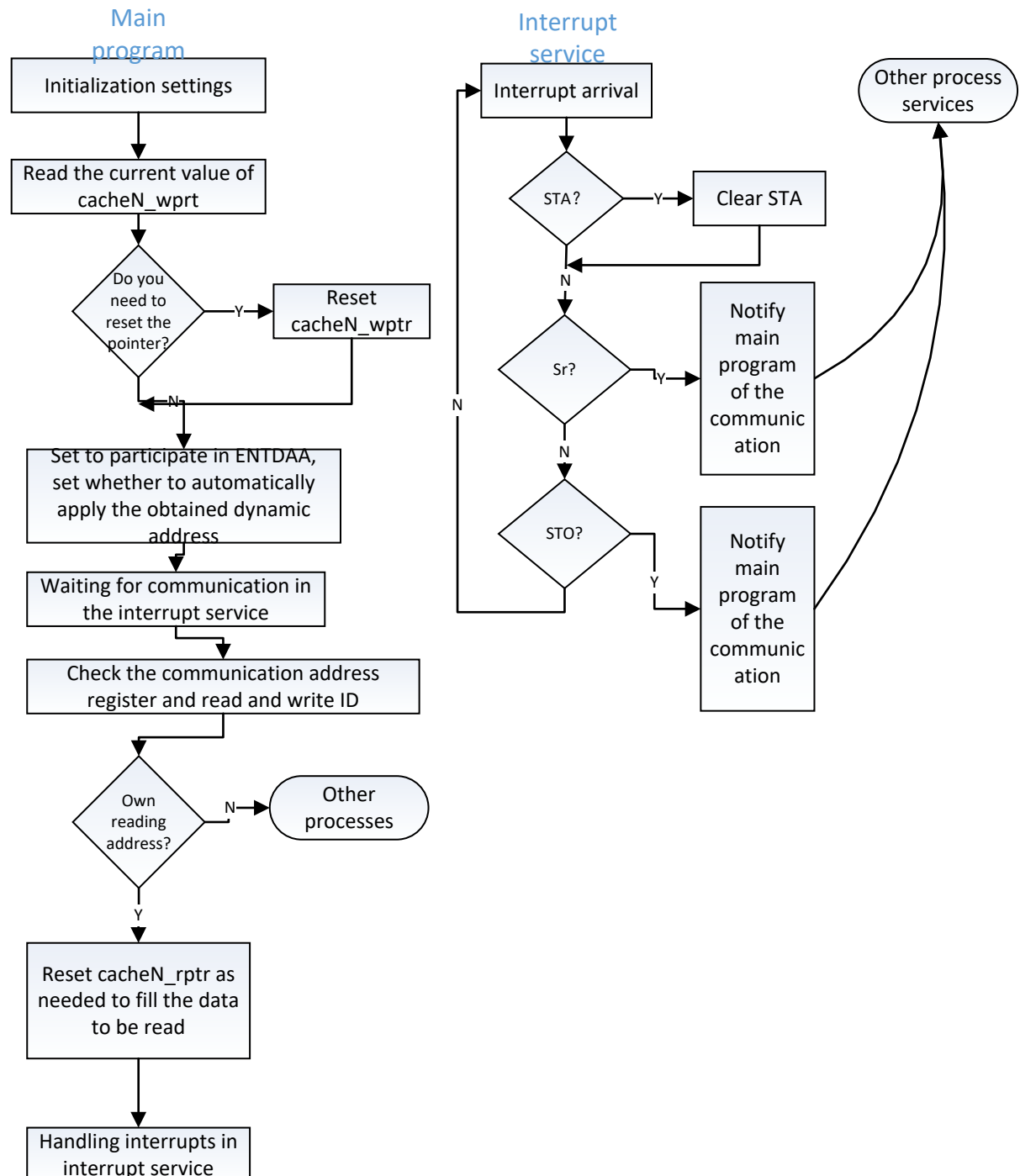


Figure 4-17 Slave Timing Diagram a of Dynamic Address Allocation

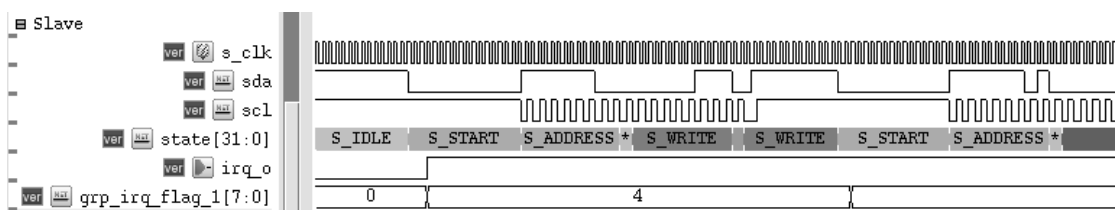
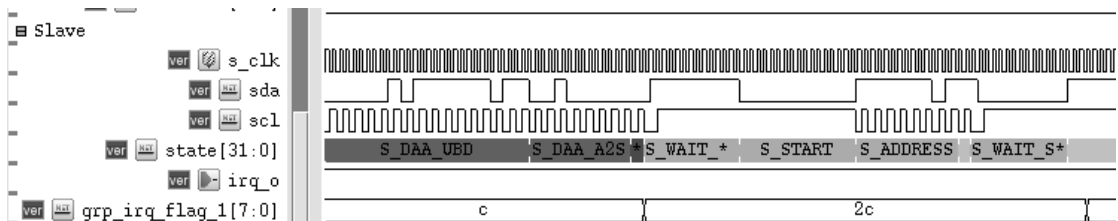


Figure 4-18 Dynamic Address Allocation Slave Timing Diagram b



4.4.4 DDR Write Operation

For DDR mode write, you need to first send the CCC command to the DDR mode by using SDR mode, and the communication programming is the same as the SDR write operation, as shown in Figure 4-19 and Figure 4-20. Only the communication waveform diagram is drawn here.

Figure 4-19 Slave Timing Diagram a of DDR Write Operation

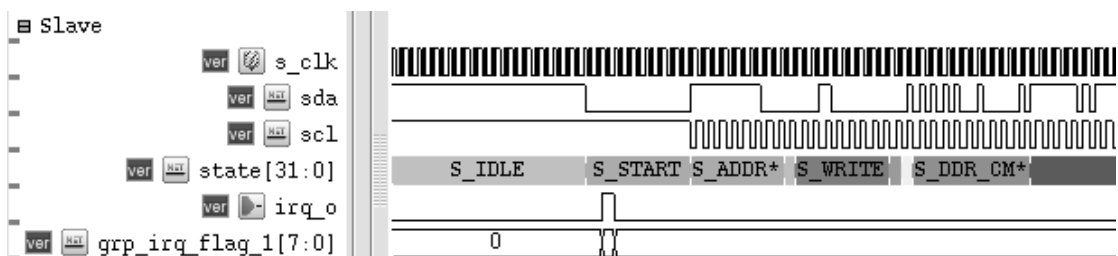
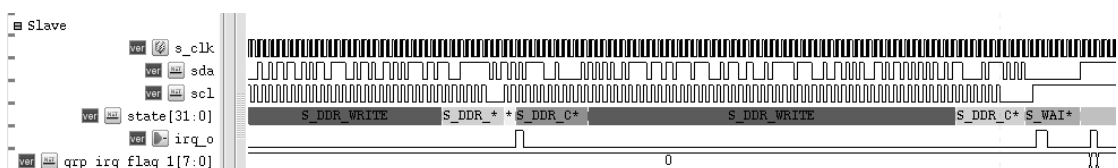


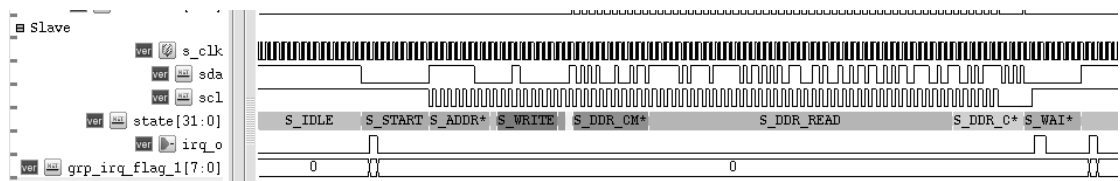
Figure 4-20 Slave Timing Diagram b of DDR Write Operation



4.4.5 DDR Read Operation

For DDR mode read, you need to use SDR mode to send CCC command to import bus into DDR mode. Communication programming is the same as SDR write operation, as shown in Figure 4-21. Only the communication waveform diagram is drawn here.

Figure 4-21 Slave Timing Diagram of DDR Read Operation



4.5 Interrupt Flag

When receiving an event on the bus, the IP will send an IRQ request to the user side, and at the same time set the corresponding flag to inform the user which events have occurred.

For the definition of events on the bus, please refer to "mipi_I3C_specification".

4.5.1 DONE Flag

The data transmission completion interrupt flag in SDR mode. Only the Master will receive this event.

4.5.2 NACK Flag

Bus NACK event interrupt flag in SDR mode. Only the Master will receive this event.

4.5.3 STA Flag

Start condition event interrupt flag in SDR mode. Both Master and Slave will receive this event.

4.5.4 SR Flag

Re-start condition event interrupt flag in SDR mode. Both Master and Slave will receive this event.

4.5.5 STO Flag

Stop condition event interrupt flag in SDR mode. Both Master and Slave will receive this event.

4.5.6 DA Flag

During the dynamic address allocation process, the dynamic address event interrupt flag is received. This event will only happen to Slave.

4.5.7 LOST Flag

Arbitration loss event interrupt flag when arbitration occurs on the bus. Both Master and Slave will receive this event. The device participating in the bus arbitration sets the interrupt when it loses control of the bus.

4.5.8 DDR_DONE Flag

In DDR mode, the data transfer completes the event interrupt flag. This event will only happen to the Master.

4.5.9 DDR_SR Flag

Restart condition event interrupt flag in DDR mode. Both Master and Slave will receive this event.

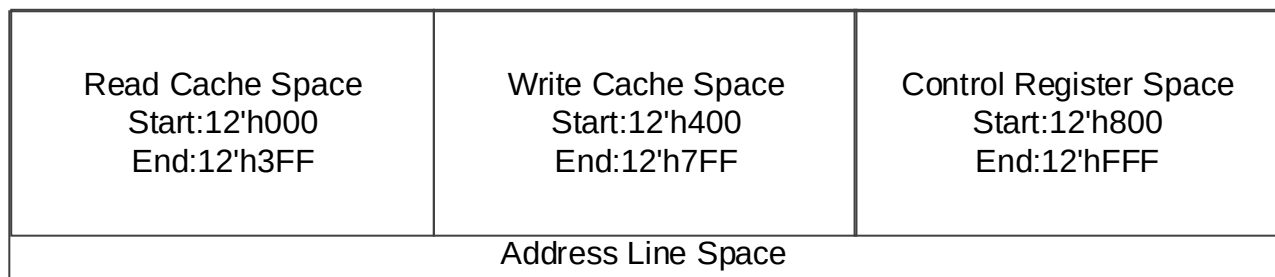
4.5.10 DDR_EXIT Flag

The exit condition event interrupt flag in DDR mode. Both Master and Slave will receive this event.

4.6 Register Map

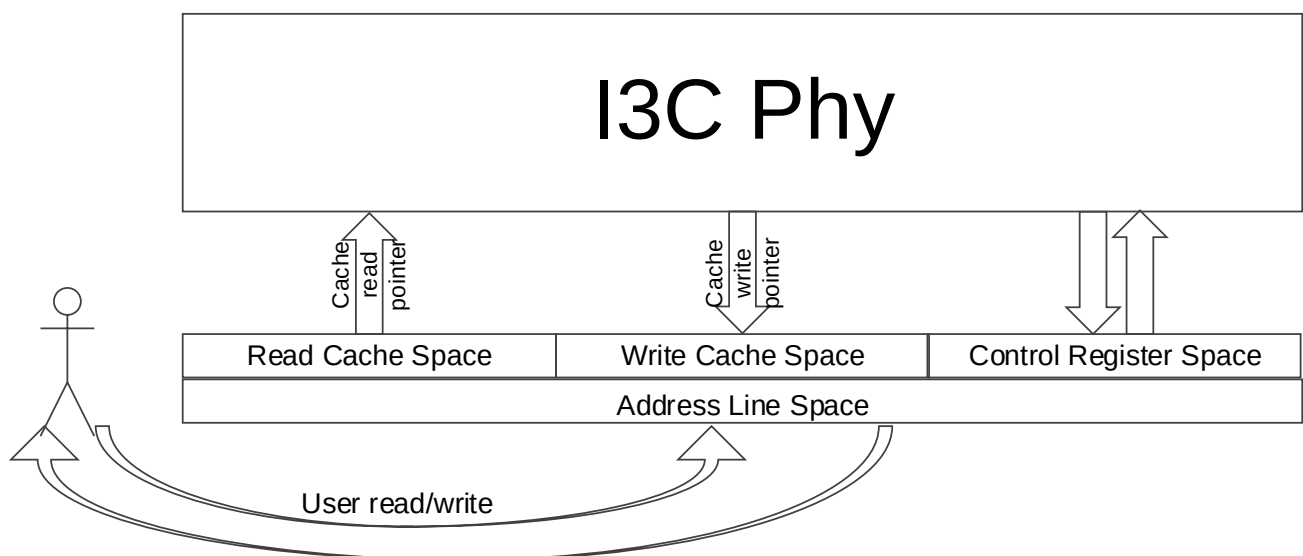
This soft core simplifies the user interface by designing the cache and control registers in the same address space. The control register and the read/write buffer are allocated to different address segments by dividing the address control, as shown in Figure 4-2.

Figure 4-22 Linear Address Division Diagram



Defining the control register base address control_base_prt = 0x800.
Read and write data of users is performed as shown in Figure 4-23.

Figure 4-23 IP Interface Interaction Mode Diagram



The list of control register part address assignments is shown in Table 4-1.

Table 4-1 Control Register Name and Address

Name	Offset	Features
Basic control register 1	0x00	Basic control register 1
Basic control register 2	0x01	Basic control register 2
Call control register	0x02	Call control register
Hang up control register	0x03	Hang up control register
Pre-scale register	0x04	Pre-scale register
SDR Message Length register	0x05	SDR Message Length register
DDR Message Length register	0x06	DDR Message Length register
Slave Static Address	0x07	Slave Static Address
Byte transferred	0x08	Byte transferred
Word transferred	0x09	Word transferred
SDR Call Address	0x0a	SDR Call Address
DDR Call Address	0x0b	DDR Call Address
cache_wptrL	0x10	cache_wptrL
cache_wptrH	0x11	cache_wptrH
cache_rptrL	0x12	cache_rptrL
cache_rptrH	0x13	cache_rptrH
Cache pointer reset register	0x18	Cache pointer reset register
State register	0x20	State register
Interrupt flag register 1	0x30	Interrupt flag register 1
Interrupt flag register 2	0x31	Interrupt flag register 2

4.6.1 Basic Control Register 1

Offset address: 0x00

	7	6	5	4	3	2	1	0
Name	Res.	Res.	Res.	Res.	Res.	enSEvent	lgcyMode	enMaster
Read and write	Res.	Res.	Res.	Res.	Res.	rw	rw	rw

Bit 0: enable master

Write 1 to enable master

Write 0 to enable slave

Can only be written at IDLE state

Read return current setting

Bit 1: legacy mode

Write 1 to enter legacy mode (only when enMaster = 1)

Write 0 to enable I3C mode

Can only be written at IDLE state
Read return current setting

Bit 2: enable slave event

Write 1 to enable acknowledge slave IBI(only when enMaster = 1)

Write 0 to disable acknowledge slave IBI(only when enMaster = 1)

Can only be written at IDLE state

Read return current setting

4.6.2 Basic Control Register 2

Offset address: 0x01

	7	6	5	4	3	2	1	0
Name	Res.	Res.	Res.	Res.	Res.	Res.	autoApplyDaa	participateDaa
Read and write	Res.	Res.	Res.	Res.	Res.	Res.	rw	rw

Bit 0: participate Dynamic Address Assignment

Write 1 to enable Slave participate Dynamic Address Assignment

Write 0 to disable Slave participate Dynamic Address Assignment

Can only be written at IDLE state

Read return current setting

Will be cleared by the hardware after ENTDAAC

Bit 1: automatically apply dynamic address

Write 1 to automatically apply dynamic address

Write 0 to manually apply dynamic address

Can only be written at IDLE state

Read return current setting

4.6.3 Call Control Register

Offset address: 0x02

	7	6	5	4	3	2	1	0
Name	Res.	Res.	Res.	Res.	Res.	abortRead	ibi	call
Read and write	Res.	Res.	Res.	Res.	Res.	wr	wr	wr

Bit 0: master call

Write 1 to start a transfer

Write 0 ignore

Can only be written at IDLE state

Read return current setting

Can only operate when enMaster =1

Will be cleared by the hardware after call

Bit 1: slave start in-band-Interrupt

Write 1 to start a transfer

Write 0 ignore

Can only be written at IDLE state

Read return current setting

Can only operate when enMaster =0

Will be cleared by the hardware after ibi

Bit 3: abort a read operation

Write 1 to abort a read operation

Write 0 ignore

Can be written at any state, but only take effect in read operation

Read return current setting

Will be cleared by the hardware after abort

4.6.4 Hang Up Control Register

Offset address: 0x03

	7	6	5	4	3	2	1	0
Name	Res.	Res.	Res.	Res.	Res.	srWhenAbort	srWhenDone	srWhenNack
Read and write	Res.	Res.	Res.	Res.	Res.	rw	rw	rw

Bit 0: srWhenNack

Write 1 to force master re-start when encounter a NACK

Write 0 to force master stop when encounter a NACK

Can be written at any state, only work with master

Read return current setting

Bit 1: srWhenDone

In SDR mode

Write 1 to force master re-start when data transfer over

Write 0 to force master stop when data transfer over

In DDR mode

Write 1 to force master re-start when data transfer over

Write 0 to force master exit DDR when data transfer over

Can be written at any state, only work with master

Read return current setting

Bit 2: srWhenAbort

In SDR MODE

Write 1 to force master re-start when abort a read

Write 0 to force master stop when abort a read

In DDR MODE

Write 1 to force master re-start when abort a read

Write 0 to force master exit DDR when abort a read

Can be written at any state, only work with master

Read return current setting

4.6.5 Pre-scale Register

Offset address: 0x04

	7	6	5	4	3	2	1	0
Name	Res.	Res.	Res.	prescale				
Read and write	Res.	Res.	Res.	rw				

Bit 4-0: prescale

Writing the prescale number to this register will take effect immediately when enMaster =1.
 $SCL_clock_cycle/2 = prescale + 1$
 So the actual SCL frequency depends on your system clock and prescale.
 Can only be written at IDLE state
 Read return current setting

4.6.6 SDR Message Length Register

Offset address: 0x05

	7	6	5	4	3	2	1	0
Name	Msg length							
Read and write	rw							

Bit 7-0: SDR mode byte try to transfer
 Write the number before call
 Read return current setting
 Can be written at any state
 Do not write in the middle of a transfer

4.6.7 DDR Message Length Register

Offset address: 0x06

	7	6	5	4	3	2	1	0
Name	DDR length							
Read and write	rw							

Bit 7-0: DDR mode word try to transfer
 Write the number before DDR transfer
 Read return current setting
 Can be written at any state
 Do not write in the middle of a DDR transfer
 Must be greater than 0 when perform a DDR read

4.6.8 Slave Static Address

Offset address: 0x07

	7	6	5	4	3	2	1	0
Name	Slave address							Res.
Read and write								Res.

Bit 7:1: Slave Static Address

Write new address will take effect immediately

If automatically apply Dynamic Address, this register will change by

hardware. If manually apply Dynamic Address, you should write new address to this register.

Can only be written at IDLE state

Read return current setting

4.6.9 Byte Transferred

Offset address: 0x08

	7	6	5	4	3	2	1	0
Name	Byte transferred							
Read and write	r							

Bit 7-0: Byte transferred

Write ignore

Read return current value of transferred bytes after a SDR write or read

Can be read at any state

4.6.10 Word Transferred

Offset address: 0x09

	7	6	5	4	3	2	1	0
Name	Word transferred							
Read and write	r							

Bit 7-0: Word transferred

Write ignore

Read return current value of transferred words after a DDR write or read.

Can be read at any state.

4.6.11 SDR Call Address

Offset address: 0x0a

	7	6	5	4	3	2	1	0
Name	SDR Call Address							RnW
Read and write	r							r

Bit 7-1: SDR Call Address

Write ignore

Read return current value of master calling address or slave ibi address in SDR mode.

Bit 0: SDR read/write bit

Write ignore

Read return current value of master calling or slave ibi operation in SDR mode.

4.6.12 DDR Call Address

Offset address: 0x0b

	7	6	5	4	3	2	1	0
Name	DDR Call Address							Res.
Read and write	r							Res.

Bit 7-1: DDR Call Address

Write ignore

Read return current value of master calling address in DDR mode.

4.6.13 Cache Write Pointer Low

Offset address: 0x10

	7	6	5	4	3	2	1	0
Name	cache_wptrL							
Read and write	r							

Bit 7-0: cache write pointer low

Write ignore

Read return current value of the low 8 bits of the cache write pointer. Can be read at any state.

4.6.14 Cache Write Pointer High

Offset address: 0x11

	7	6	5	4	3	2	1	0
Name	Res.	Res.	Res.	Res.	cache_wptrH			
Read and write	Res.	Res.	Res.	Res.	r			

Bit 3-0: cache write pointer high

Write ignore

Read return current value of the high 4 bits of the cache write pointer

Can be read at any state

4.6.15 Cache Read Pointer Low

Offset address: 0x12

	7	6	5	4	3	2	1	0
Name	cache_rptrL							
Read and write	r							

Bit 7-0: cache read pointer low

Write ignore

Read return current value of the low 8 bits of the cache read pointer

Can be read at any state

4.6.16 Cache Read Pointer High

Offset address: 0x13

	7	6	5	4	3	2	1	0
Name	Res.	Res.	Res.	Res.	cache_rptrH			
Read and write	Res.	Res.	Res.	Res.	r			

Bit 3-0: cache read pointer high

Write ignore

Read return current value of the high 4 bits of the cache read pointer

Can be read at any state

4.6.17 Cache Pointer Reset Register

Offset address: 0x18

	7	6	5	4	3	2	1	0
Name	Res.	Res.	Res.	Res.	Res.	Res.	cacheN_rprt_rst	cacheN_wprr_rst
Read and write	Res.	Res.	Res.	Res.	Res.	Res.	w	w

Bit 0: cache N write pointer reset

Write 1 ignore

Write 0 to reset the pointer of the cache N writer

Can only be written at IDLE state

Read return 0

Bit 1: cache N read pointer reset

Write 1 ignore

Write 0 to reset the pointer of the cache N reader

Can only be written at IDLE state

Read return 0

4.6.18 Status Register

Offset address: 0x20

	7	6	5	4	3	2	1	0
Name	Res.	Res.	Res.	state				
Read and write	Res.	Res.	Res.	r				

Bit 4-0: current state

Write ignore

Read return current state

4.6.19 Interrupt Flag Register 1

Offset address: 0x30

	7	6	5	4	3	2	1	0
Name	Res.	lost	da	sto	sr	sta	nack	done
Read and write	Res.	wr	wr	wr	wr	wr	wr	wr

Bit 0: DONE flag

Write 1 ignore

Write 0 to clear the flag

Can be written at any state

Read return current flag

Bit 1: NACK flag

Write 1 ignore
 Write 0 to clear the flag
 Can be written at any state
 Read return current flag

Bit 2: START flag

Write 1 ignore
 Write 0 to clear the flag
 Can be written at any state
 Read return current flag

Bit 3: RE-START flag

Write 1 ignore
 Write 0 to clear the flag
 Can be written at any state
 Read return current flag

Bit 4: STOP flag

Write 1 ignore
 Write 0 to clear the flag
 Can be written at any state
 Read return current flag

Bit 5: DYNAMIC_ADDRESS_ASSIGNED flag

Write 1 ignore
 Write 0 to clear the flag
 Can be written at any state
 Read return current flag

Bit 6: LOST flag

Write 1 ignore
 Write 0 to clear the flag
 Can be written at any state
 Read return current flag

4.6.20 Interrupt Flag Register 2

Offset address: 0x31

	7	6	5	4	3	2	1	0
Name	Res.	Res.	Res.	Res.	Res.	DDR_EXIT	DDR_SR	DDR_DONE
Read and write	Res.	Res.	Res.	Res.	Res.	rw	rw	rw

Bit 0: DDR_DONE flag

Write 1 ignore
 Write 0 to clear the flag
 Can be written at any state
 Read return current flag

Bit 1: DDR_SR flag

Write 1 ignore

Write 0 to clear the flag

Can be written at any state

Read return current flag

Bit 2: DDR_EXIT flag

Write 1 ignore

Write 0 to clear the flag

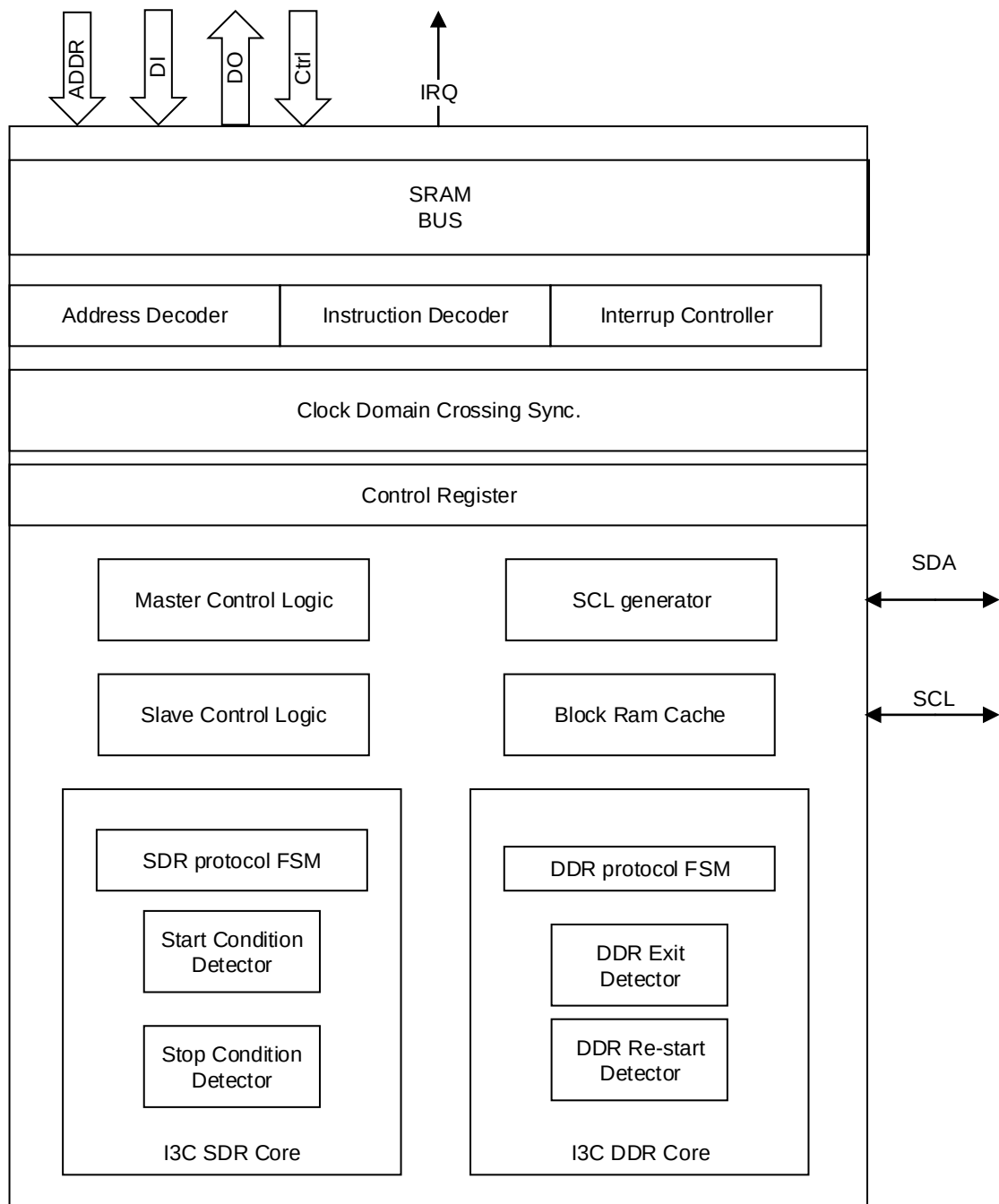
Can be written at any state

Read return current flag

5 Structure, Interface, and Configuration

5.1 Microstructure

The I3C DDR eXtension IP divides the RTL in terms of design basis functions, as shown in Figure 5-1.

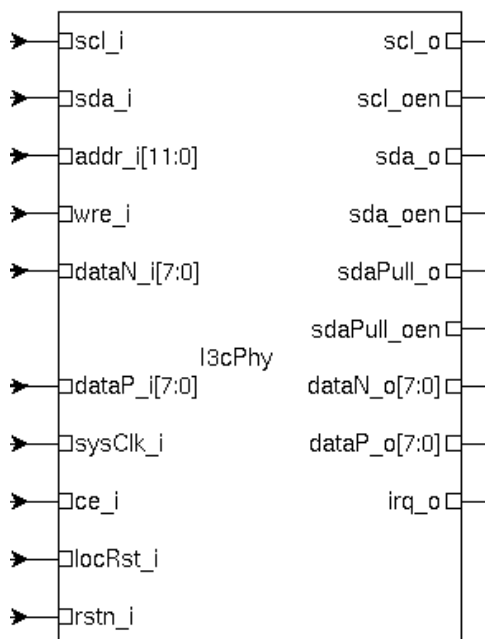
Figure 5-1 Micro-structure Block Diagram

5.2 Interface

5.2.1 Block Diagram

The block diagram of the I3C DDR eXtension is shown in Figure 5-2.

Figure 5-2 I3C DDR EXTension Block Diagram



5.2.2 Interface Description

The description of I3C DDR eXtension interface is shown in Table 5-1.

Table 5-1 I3C DDR EXTension Interface Description

Name	Direction	Bit width	Features
scl_i	Input	1	I3C bus tri-state SDA
scl_o	Output	1	
scl_oen	Output	1	
sda_i	Input	1	I3C bus tri-state SCL
sda_o	Output	1	
sda_oen	Output	1	
sdaPull_o	Output	1	Sda pull-up output
sdaPull_oen	Output	1	Sda pull-up high resistance control
addr_i	Input	[11:0]	Address
wre_i	Input	1	Read and write control
dataN_i	Input	[7:0]	CacheN data input
dataN_o	Output	[7:0]	CacheN data output
dataP_i	Input	[7:0]	CacheP data input
dataP_o	Output	[7:0]	CacheP data output
irq_o	Output	1	Interrupt signal output

Name	Direction	Bit width	Features
sysClk_i	Input	1	System clock
ce_i	Input	1	Clock enable
locRst_i	Input	1	Synchronous reset, active high
rstn_i	Input	1	Asynchronous service, active low

5.2.3 Configurable Parameters

The parameter configuration in this section describes the Power-On parameters. Since these parameters are parameters for RTL synthesis, these parameters are valid only in the FPGA soft core mode. In hard mode, these parameters will all be 0. The user can only change the relevant configuration by writing the corresponding register at runtime.

Note!

When the system is globally reset, the configuration registers are also regressed to the value of the corresponding Power-On parameter configuration.

The configurable parameters are shown in Table 5-2.

Table 5-2 Configurable Parameters

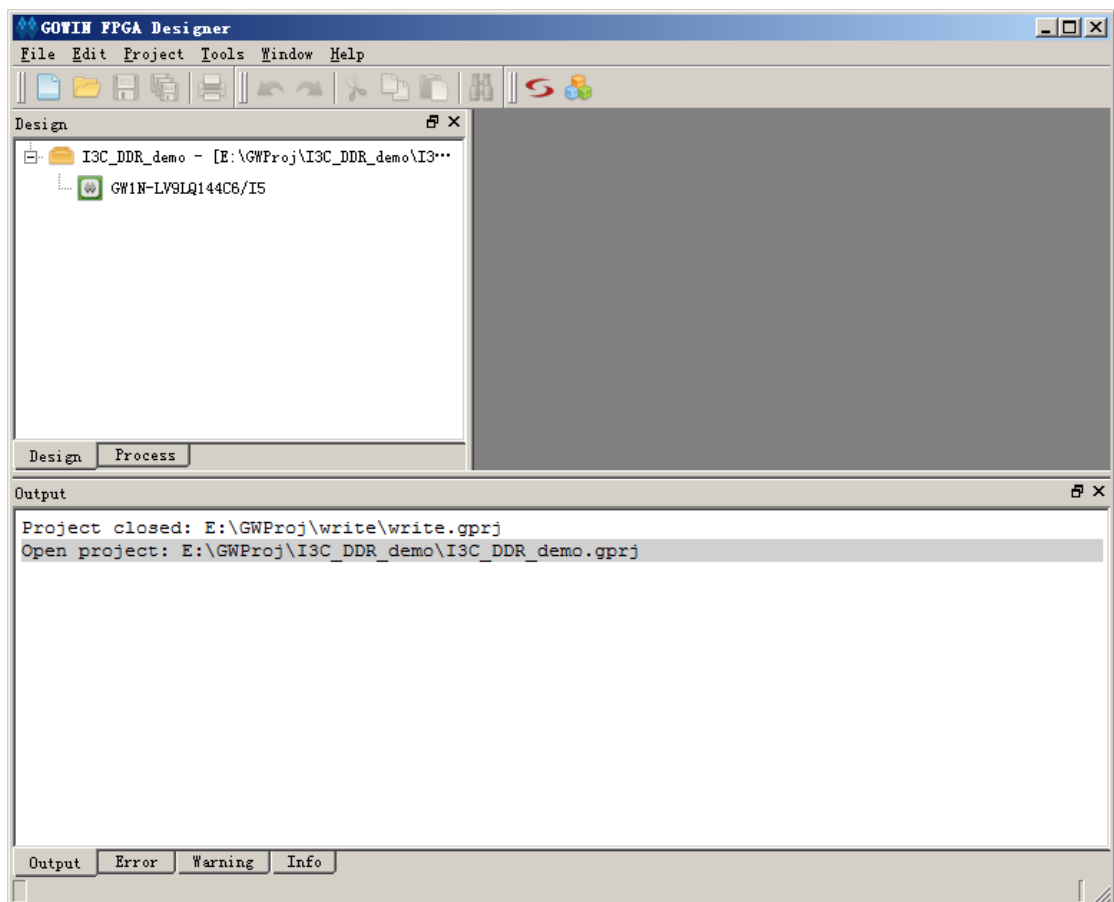
Parameter	Features
INI_enMaster	This parameter determines the role of the device after the system is reset, that is, the initial value of enMaster.
INI_participateDaa	This parameter determines whether the dynamic address allocation is involved after the system is reset, that is, the initial value of the participantDaa.
INI_autoApplyDaa	This parameter determines whether the obtained dynamic address is automatically applied after the system is reset, that is, the initial value of autoApplyDaa.
INI_srWhenNack	This parameter determines the behavior when NACK is encountered after a system reset, that is, the initial value of srWhenNack.
INI_srWhenDone	This parameter determines the behavior when Done is encountered after system reset, that is, the initial value of srWhenDone
INI_srWhenAbort	This parameter determines the behavior when the Abort is encountered after the system reset, that is, the initial value of srWhenAbort
INI_enSEvent	This parameter determines whether the slave IBI is responded after the system reset, that is, the initial value of enSEvent.
INI_plsNumber	This parameter determines the SCL frequency value after system reset, that is, the initial value of plsNumber
INI_myAddress	This parameter determines the static address after the system is reset, that is, the initial value of myAddress.

6Compile, Configuration, and Files

6.1 Compile I3C DDR EXTension

1. Create an FPGA RTL project, as shown in Figure 6-1.

Figure 6-1 FPGA RTL Project Initial Interface Diagram



2. Open the IP Core Generator and open the I3C DDR eXtension configuration interface, as shown in Figure 6-2 and Figure 6-3.

Figure 6-2 Call I3C DDR Extension

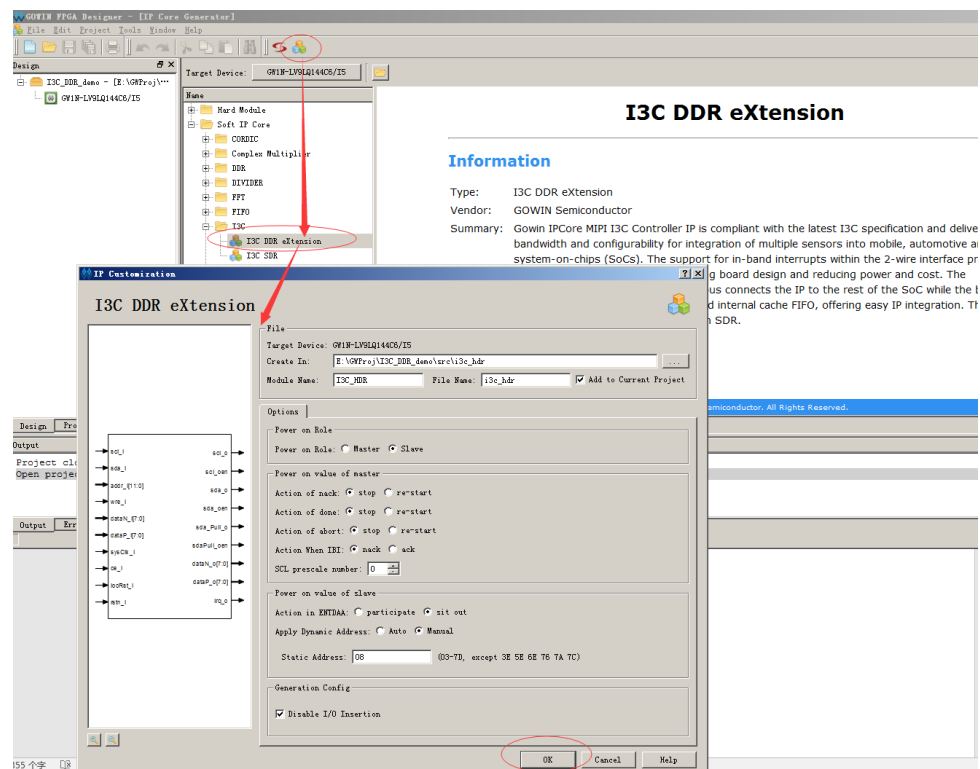
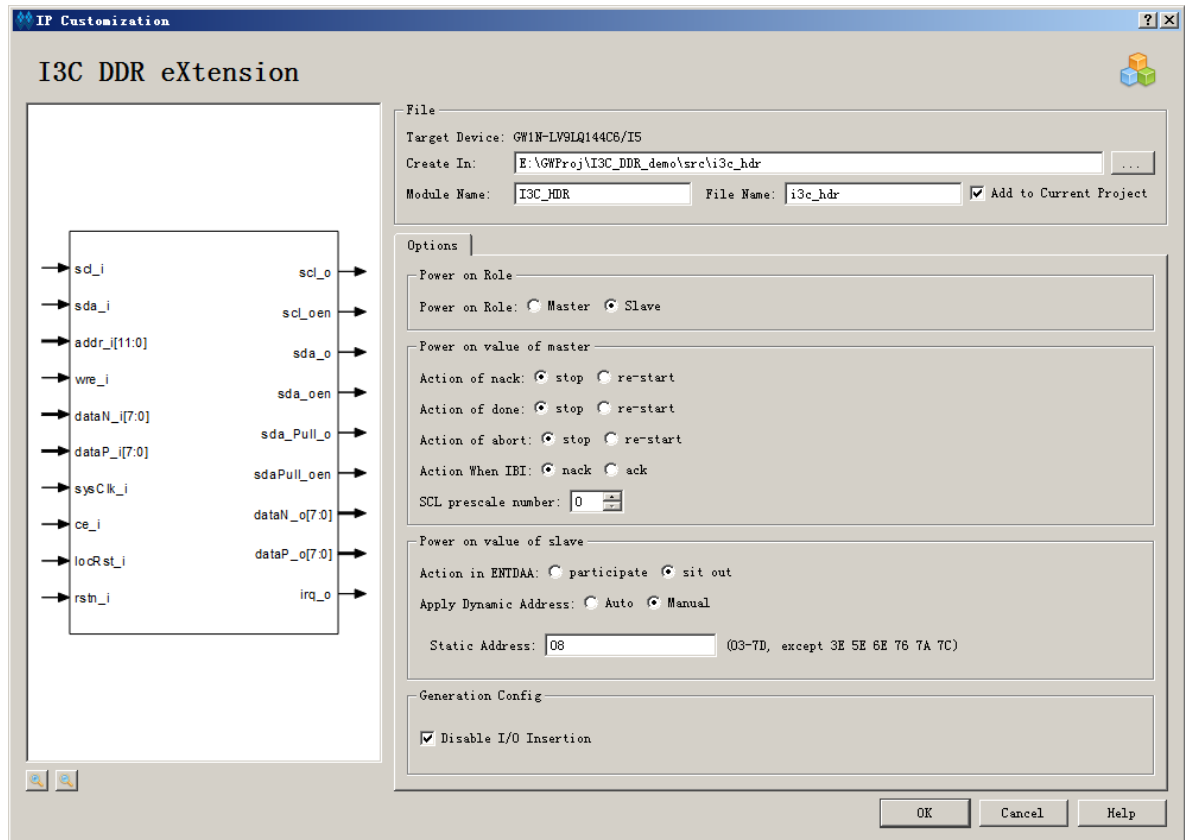
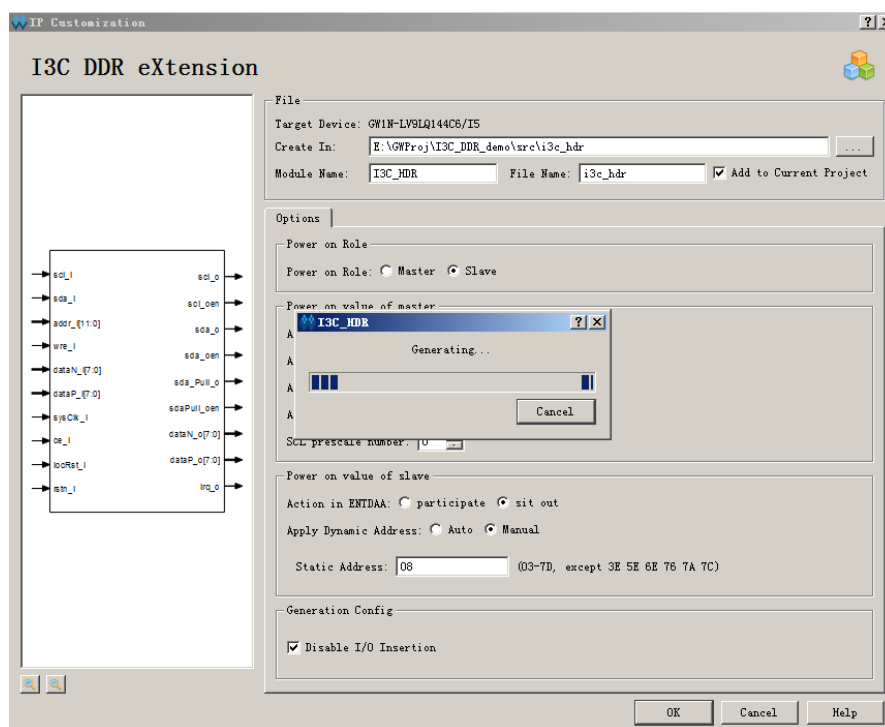


Figure 6-3 I3C DDR EXTENSION Configuration Interface



3. After configuring the parameters according to requirements, click “OK” to generate, as shown in Figure 6-4.

Figure 6-4 Generate I3C DDR EXtension Diagram



4. When asked if you want to add the generated IP to the project, click "OK" to add to the current project.

6.2 Generated Files

6.2.1 File Structure

Taking the src of the RTL project as the root directory, the main files generated by the IP Core Generator and their functions are described below, as shown in Figure 6-5.

Figure 6-5 IP Core Generator Generated File

