Gowin FPGA Products

JTAG Programming and Configuration Guide

TN653-1.07E, 11/18/2019

Disclaimer

Revision History

Date	Version	Description
4/13/2017	1.0E	Initial version published.
4/24/2017	1.01E	The process of programming the internal Flash of the JTAG improved.
6/5/2017	1.02E	The process of erasing internal Flash of the GW1N-1 added.
7/28/2017	1.03E	The process of erasing internal Flash of the GW1N-2/4/6/9, GW1NZ-1 added.
10/22/2017	1.04E	The description of each configuration/programming/reading process improved.
8/2/2018	1.05E	GW1N4 Background Programming flow chart and Transfer JTAG Instrction Sample & Extest flow chart updated.
11/15/2018	1.06E	The status code of GW1NZ and GW1NS added.
11/18/2019	1.07E	● GW1NZ-1 related content added. ● The content of Table 2-2 Gowin FPGA IDCODE updated.

Contents

List of Figures

List of Tables

1 About This Guide

1.1 Purpose

This manual mainly introduces the related information on the JTAG configuration and programming of the Gowin FPGA products, including the JTAG configuration mode, the configuration process, and the related routine files.

1.2 Supported Products

The information in the guide applies to all Gowin FPGA products.

1.3 Related Documents

Before reading, please visit www.gowinsemi.com for JTAG 1149.1 configuration.

Gowin FPGA Products Programming and Configuration Manual

1.4 Terminology and Abbreviation

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

Table 1-1 Abbreviations and Terminology

Terminology and Abbreviation	Full Name	Meaning
FS file	Fuses file	ASCII file including the configuration data
Configuration	Configuration	The process of configuring the FPGA SRAM area
Configuration Data	Configuration Data	The data of configuring the FPGA SRAM
Bitstream	Bitstream Data	The data of configuring the FPGA SRAM
Configuration Mode	Configuration Mode	Configuration mode, determine the Configuration Data source
EFlash/EmbFlash	Embedded Flash Memory	Internal Flash Memory
Internal Flash	Internal Flash Memory	Same with Embedded Flash
Programming	Programming	The process of programming the Configuration Data to the Embedded Flash or the External Flash memory
Edit Mode	Edit Mode	The FPGA in Configuration mode or Programming mode
User Mode	User Mode	The mode after the FPGA finishes the Configuration or the Programming and the logic function is executed.
LSB	Least Significant Bit	Least Significant Bit (priority)
MSB	Most Significant Bit	Most Significant Bit (priority)
TAP	Test Access Port	Test Access Port
Security Bit	Security Bit	Security Bit (readback the SRAM to keep high level)
Bscan	Boundary Scan	Boundary Scan

1.5 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.

Website: www.gowinsemi.com

E-mail:support@gowinsemi.com

+Tel: +86 755 8262 0391

2 Configuration&Programming

2.1 JTAG Configuration Mode

The JTAG mode of Gowin FPGA products complies with IEEE1149.1 boundary scan standard.

The bitstream data is loaded to SRAM in JTAG mode, and the data is lost if power off. The relevant pins for the JTAG configuration mode are shown in Table 2-1.

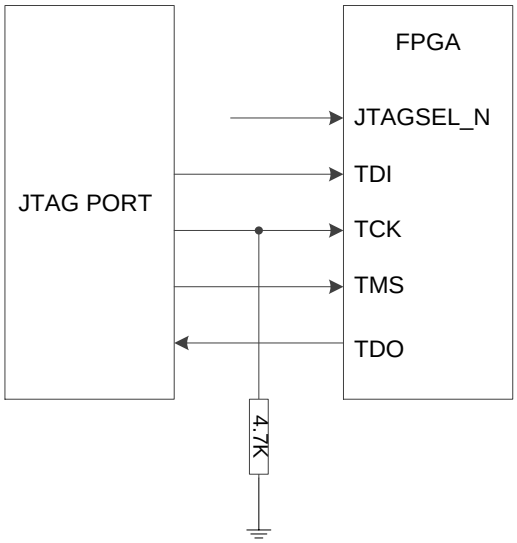
Table 2-1 Pin Description in JTAG Configuration Mode

Pin Name	I/O	Description
JTAGSEL_N[1]	I, internal weak pull-up	Revert JTAG pin from GPIO to configuration pin. Low active
TCK[2]	I	Serial clock input in JTAG mode
TMS	I, internal weak pull-up	Serial mode input in JTAG mode
TDI	I, internal weak pull-up	Serial data input in JTAG mode
TDO	O	Serial data output in JTAG mode

Note!

- [1] Only when JTAG pin is set as RECOVERY and the device starts sucessfully, the JTAGSEL_N signal is valid;
- [2] TCK needs to connect 4.7 K pull down resistance on PCB.

Figure 2-1 Connection Diagram for the JTAG Configuration Mode



Note!

- For devices unbonded by JTAGSEL_N, when debugging the JTAG pin reuse, users are advised to set the MODE value to non-auto configuration mode (AUTOBOOT, DUALBOOT or MSPI) before powering up the device to avoid the other bitstream data affecting configuration. The device entries User MODE, and JTAG pin will be a GPIO pin after the device is powered up and the JTAG is configured manually.
- The clock frequency of JTAG mode shall not be higher than 25MHz.

2.2 Configuration Process

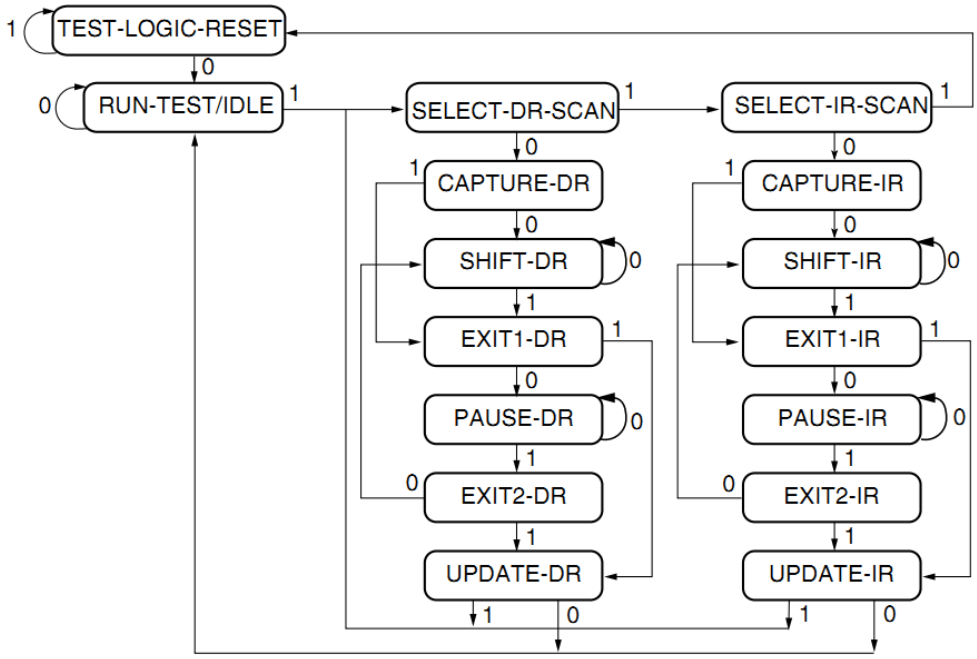
2.2.1 JTAG Pins Definition

- TCK: Test clock input. Sample the data of TMS and TDI at the rising edge of TCK, and output the data to TDO at the falling edge of clock;
- TMS: test mode selection. TMS is used to set a specific test mode of JTAG;
- TDI: Test data input. Input data to JTAG via TDI;
- TDO: Test data output. Output data from JTAG via TDO.

2.2.2 TAP State Machine

The state machine at the test access port is designed to select an instruction register or a data register to connect it between TDI and TDO. In general, the instruction register is used to select the data register to be scanned. In the state machine diagram, the number on the side of the arrow indicates the logic state of the TMS when the TCK goes high, as shown in Figure 2-2.

Figure 2-2 TAP State Machine



2.2.3 TAP Reset

After the TMS is keep high (logic "1") and at least 5 strobes are input (higher and then low) at the TCK terminal, the TAP logic is reset, and the TAP state machine in other states is converted into the state of test logic reset, the JTAG port and the test logic are reset.

Note!

The CPU and periphrals are not reset in this state.

Note!

- The data on the TDO is valid from the falling edge of TCK in the Shift_DR or Shift_IR state;

- The data is not shifted in the Shift_DR or Shift_IR state;
- The data is shifted when leaving the Shift_DR or Shift_IR;
- The first to be shifted is the least significant bit (LSB) of the data;
- Once reset, all instructions will be reset or disabled.

2.2.4 Instruction Register and Data register

In addition to the test logic reset, the state machine can also control two basic operations:

- Instruction register (IR) scan;
- Data Register (DR) scan.

During the IR scanning operation, in Shift_IR state, the data or instructions are sent to the IR in the LSB way. The lower data bits are sent first. The instructions will be all sent when the sate machine returns to Run-Test-Idle, as shown in Figure 2-3

During the data register scanning operation, the data or instructions are sent to the DR in the Shift_DR state, as shown in Figure 2-4. The data is sent in LSB way or MSB way depending on specific operations.

Figure 2-3 Instruction Register Access Timing

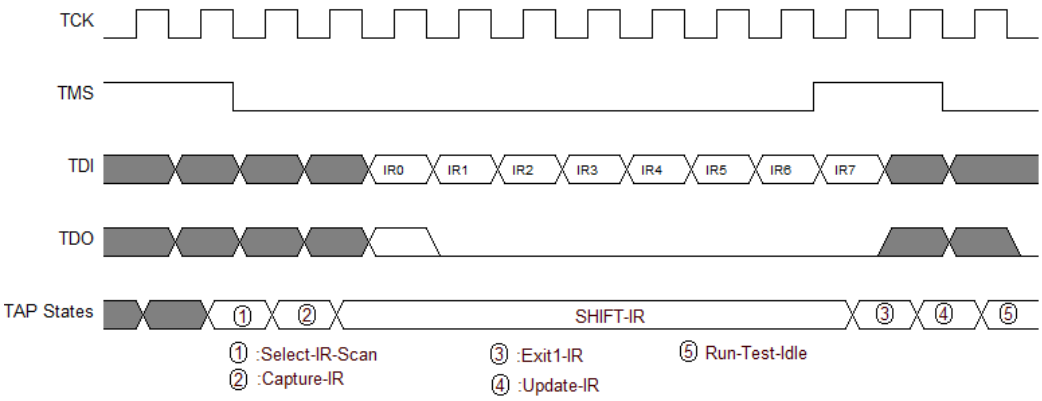
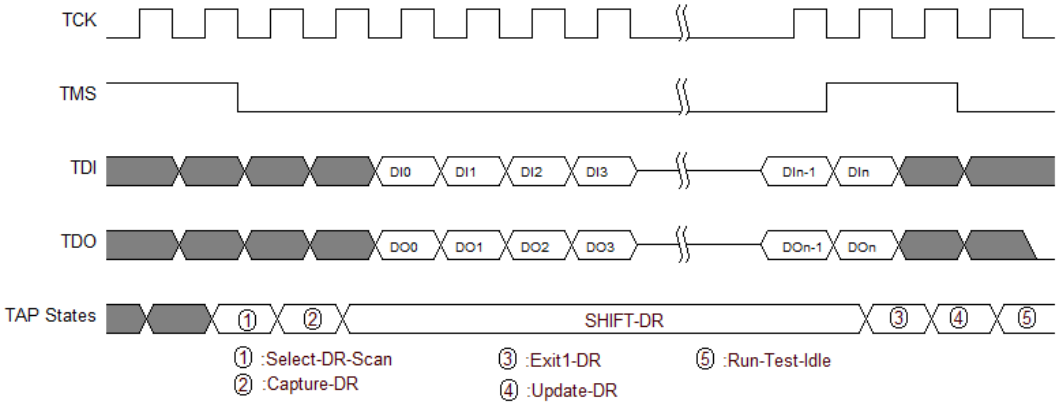


Figure 2-4 Data Register Access Timing



Note!

- The total length of the instruction register is 8 bits in the GW1N(R) and GW2A(R) series of the FPGA;
- The length of the data register can vary depending on the selected register.

2.2.5 Read ID CODE Instance

ID Code, i.e. JEDEC ID Code, is the basic logo of FPGA products.

The length of the Gowin FPGA ID Code is 32 bits. The ID Codes of the FPGA are listed in the following table.

Table 2-2 Gowin FPGA IDCODE

Gowin FPGA Device Family IDCODE			
Device Family	Device Part	Manufacturer ID	IDCODE
	Bits 31-12	Bits 11-0	
		h81B	
GW1N-1	h09002	h81B	h0900281B
GW1N-1S	h09003	h81B	h0900381B
GW1NZ-1	h01006	h81B	h0100681B
GW1N(R)-2	h01001	h81B	h0100181B
GW1N(R)-2B	h11001	h81B	h1100181B
GW1NS-2	H03000	h81B	h0300081B
GW1NS(R)-2C	H03001	h81B	h0300181B
GW1NSE-2C	H03001	h81B	h0300181B
GW1N(R)-4B	h11003	h81B	h1100381B
GW1N(R)-6	h01004	h81B	h0100481B
GW1N(R)-9	h11005	h81B	h1100581B
GW2A(R)-18	h00000	h81B	h0000081B
GW2A-55	h00002	h81B	h0000281B

The instruction that FPGA reads is 0x11. Take the GW1N-4B ID Code as an example to illustrate the following steps of the working mode of the JTAG.

1. TAP reset: TMS is set to high level and at least 5 clock cycles are continuously transmitted;
2. Move the state machine from Test-Logic-Reset to Run-Test-Idle;
3. Move the state machine to Shift-IR. Send Read ID 0x11 beginning with LSB. When MSB (the last bit) is being sent, move state machine to Exit1-IR at the same time, i.e., TMS should be high level before sending MSB. Table 2-3 lists the change of TDI and TMS value during sending 0x11 in 8-clock cycle, as shown in Figure 2-6.

Table 2-3 Change of TDI and TMS Value in The Process of Sending Instructions

	TCK 1	TCK 2	TCK 3	TCK 4	TCK 5	TCK 6	TCK 7	TCK 8
TDI value (0x11)	1	0	0	0	1	0	0	0
TMS value	0	0	0	0	0	0	0	1

4. Move the state machine, back to Run-Test-Idle after going from Exit1-IR to Update-IR, and then run the state machine at least 3 clock cycles in Run-Test-Idle.
5. Move the state machine to Shift-DR, send 32 clock cycles, and set TMS to high level before the 32nd clock is sent. When the 32 clock

cycles are completed, jump from Shift-DR to Exit1-DR. During this period, sending 32 clocks can read 32 bits data, that is, 0x0100381B, as shown in Figure 2-7;

6. Move the state machine back to Run-Test-Idle;

Figure 2-5 Read Machine Flow Chart in ID Code State

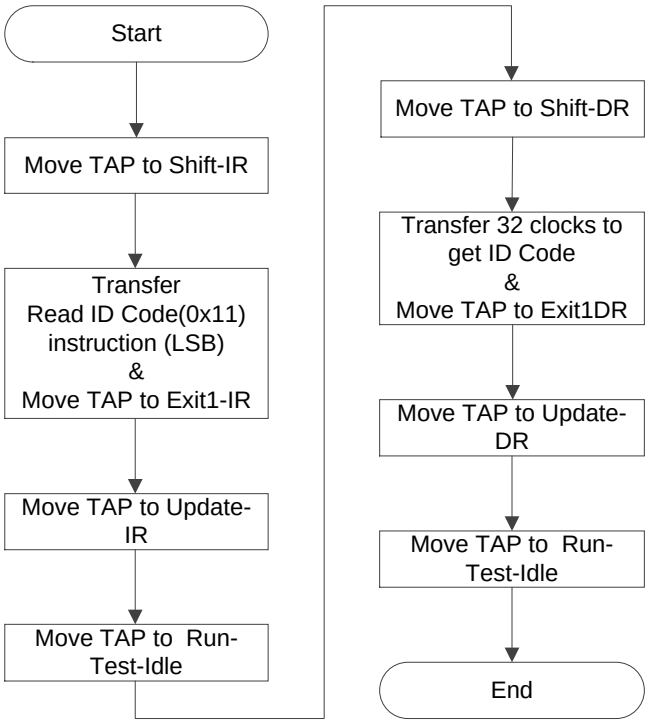


Figure 2-6 Read - 0x11 Access Timing of ID Code Instruction

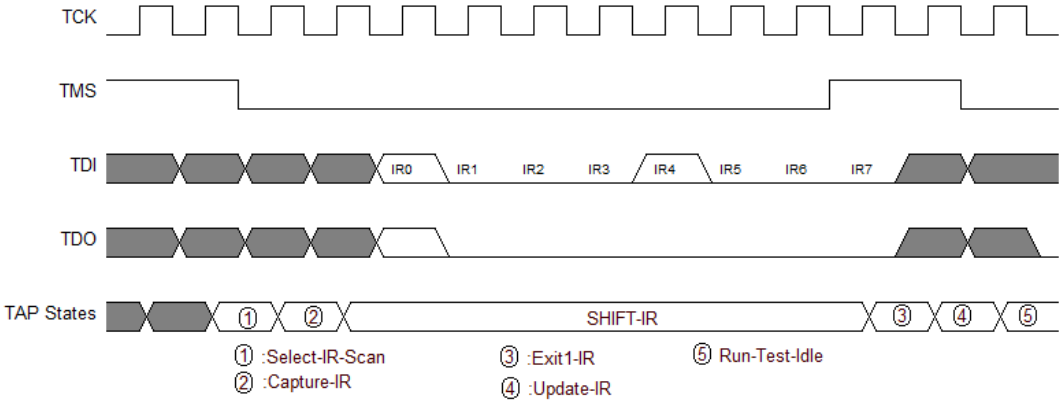
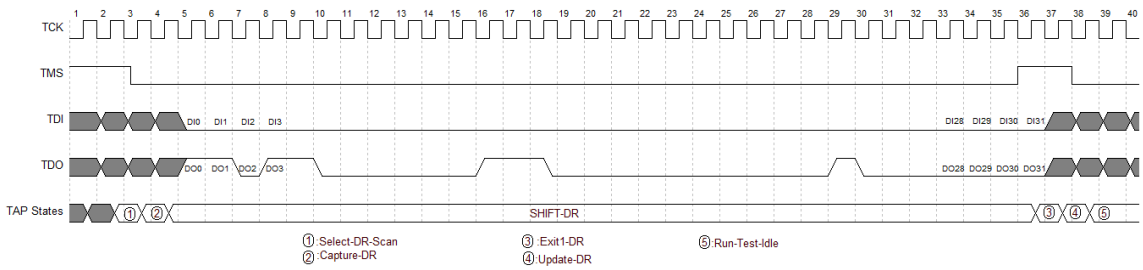


Figure 2-7 Read ID Code Data Register Access Timing



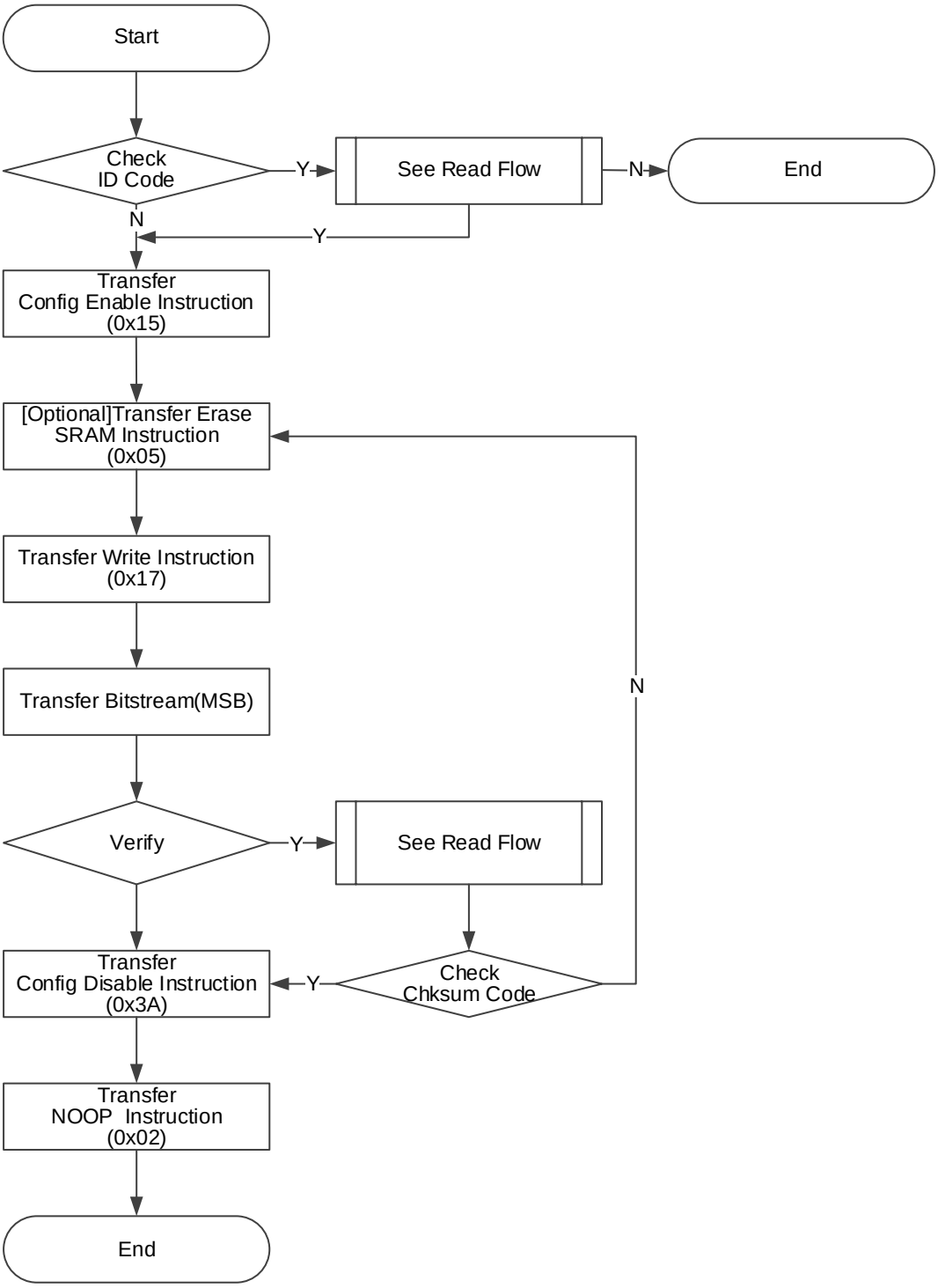
2.2.6 SRAM Configuration

The FPGA SRAM is configured using an external Host to enable the FPGA functions. Configuration Mode Pins have no impact on SRAM configuration.

Generate the FS file using Gowin Designer. Configure the SRAM according to JTAG 1149.1. The process of SRAM configuration using the external Host is as follows, as shown in Figure 2-8.

1. Establish a JTAG link and reset the TAP;
2. Read the device IDCODE and check if it matches. This step is optional.
3. Move the state machine to Shift-IR (instruction register), send the " 0x15" instruction of ConfigEnable, and return to Run-Test-Idle. Please refer to 2.2.4 and 2.2.5 for the timing.
4. Send the "0x17" instruction of Transfer Configuration Data. The Timing is same with the step 3 above.
5. Move the state machine to Shift-DR (Data Register). Send Configuration Data from the MSB bit by bit till all the bitstream file content is sent.
6. Move the state machine to back to Run-Test-Idle;
7. Please refer to 2.2.7Process of Reading SRAM (The process of reading SRAM) if read back Configuration Data is needed to verify.
8. Send the "0x3A" instruction of ConfigDisabled;
9. Send the "0x02 " instruction of Noop to end the configuration process.

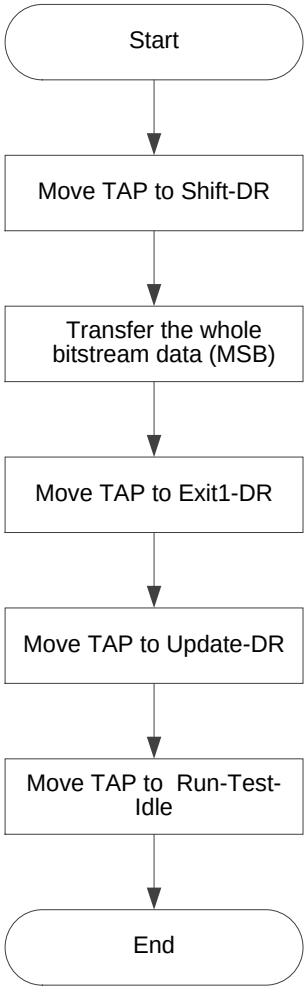
Figure 2-8 SRAM Configuration Flow



Tansfer Configuration DataProcess (Non-JTAG Mode)

Tansfer the configuration data in the SHIFT-DR.

Figure 2-9 Tansfer Configuration Data Process View



2.2.7 Process of Reading SRAM

Warning: SRAM data is not allowed to be read back by default.

Read the SRAM data from the SRAM area of the FPGA. First ensure that the security bit is not configured when the data are written to the SRAM. The security bit is used to protect the runtime data and ensure the data security. After the safety bit is set, the data received from the SRAM are 1 (high level).

The data verification adopts the Checksum-16 calculation method, i.e., accumulate per 16 bits and the lower 16 bits is taken as the final result.

Table 2-4 Count of Address and Length of One Address

Device	Length of One Address (bits/address)	Count of Address
GW1N-1/GW1N-1S/ GW1NZ-1	1216	274
GW1N-2/GW1N(R)-4B/ GW1NS(E/R)-2(C)	2296	494
GW1N(R)-6/GW1N(R)-9	2836	712
GW2A(R)-18	3376	1342
GW2A(R)-55(ES)	5536	2038

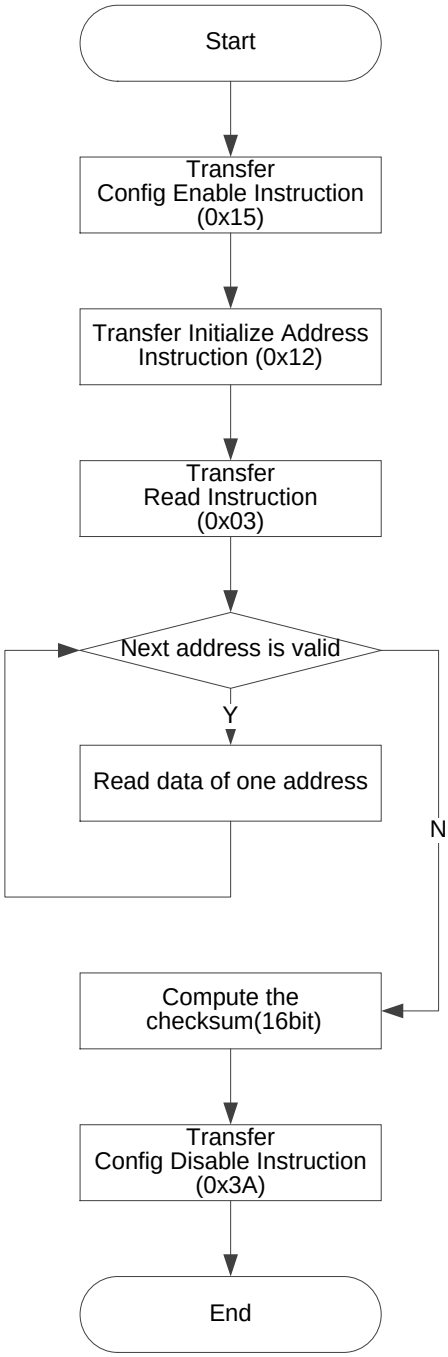
The reading process is described in detail below, as shown in Figure 2-10.

1. If you continue after the process in 2.2.6, please go to Step 3 directly;
2. Send the "0x15" instruction of ConfigDisble;
3. Send the "0x12" instruction of Address Initialize;
4. Send the "0x 03" instruction of SRAM Read;
5. Move the state machine to Shift-DR (data register) and send it to one clock period with one SRAM length, see Table 2-4. When the last clock is sent, pull up TMS at the same time. The state machine jumps to Exit1-DR, and TDO reads data with corresponding length. The state machine will back to Run-Test-Idle state finally.
6. Repeat the step 5, the address will be automatically accumulated when the data of an address are read each time;
7. Send the "0x3A" instruction of ConfigDisabled;
8. Send the "0x02 " instruction of Noop to end the configuration process.
9. Starting from the highest bit, calculate the Checksum of the read data and compare it with the Checksum in the corresponding FS file. Same means same data. (Optional)

Note!

Steps 7, 8 and 9 have no sequence.

Figure 2-10 Process of reading SRAM



2.2.8 Erase Internal Flash

In the GW1N series of internal Flash memory, the built-in Flash needs to be erased before each programming task. For data security, the built-in Flash must be erased entirely.

The requirements for JTAG programming frequency are different according to the different processes of the GW1N series of the internal Flash. Please refer to Table 2-5.

Table 2-5 TCK Frequency Requirements for JTAG

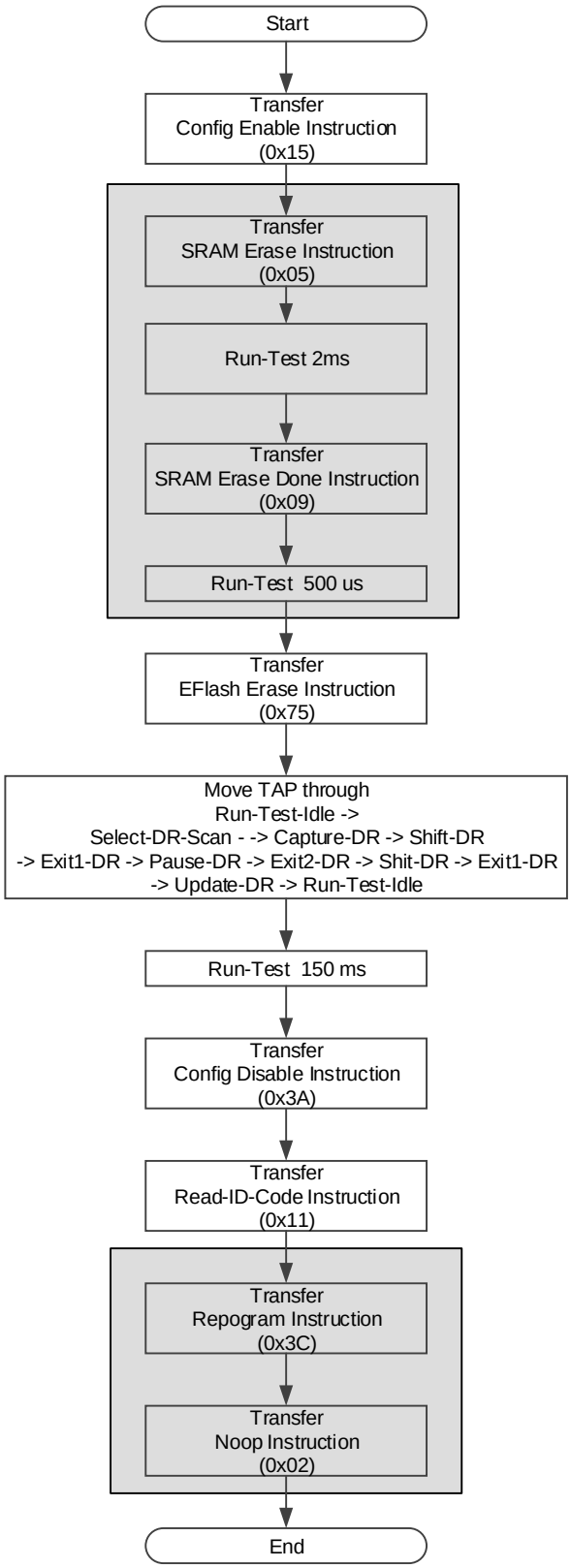
Device	TCK Frequency Range
GW1N-1/GW1N-1S	1.4Mhz ~ 5Mhz
GW1N(R)-2/4	2Mhz ~ 5Mhz
GW1N(R)-2B/4B/9 GW1NZ-1/GW1NS(E)-2(C)	1Mhz ~ 5Mhz

GW1N-2/4/6/9, GW1NZ-1 Erasure Flow

The following describes the erase flow of the GW1N-2/4/6/9 series and GW1NZ-1 chips in detail, as shown in Figure 2-11.

1. Establish a JTAG link and reset the TAP;
2. Read the device IDCODE and check if it matches. This step is optional.
3. Send the "0x15" instruction of ConfigEnable. Please refer to 2.2.5 for the timing.
4. If the SRAM has not been configured (including internal and external Flash configuration) or background programming is adopted, step 5 to step 10 can be skipped;
5. Send the "0x05" instruction of SRAM Erase;
6. Send the "0x02 " instruction of Noop;
7. The clock（Run-Test） is continuously generated in Run-Test-Idle for 6ms;
8. Send the "0x09" instruction of SRAM Erase Done;
9. Send the "0x02 " instruction of Noop;
10. The clock（Run-Test） is continuously generated in Run-Test-Idle for 500μs;
11. Send the "0x75" instruction of EFlash Erase;
12. Move the state machine in turn: Run-Test-Idle -> Select-DR-Scan-> Update-DR -> Capture-DR -> Shift-DR -> Transfer 32 bits-> Exit1-DR -> Update-DR -> Run-Test-Idle;
13. The clock（Run-Test）is continuously generated in Run-Test-Idle for 120ms. Please refer to Table 2-5 for the frequency requirements;
14. Send the "0x3A" instruction of ConfigDisabled;
15. Send the "0x02 " instruction of Noop;
16. Send the "0x3C" instruction of Repogram to load the data in Flash to SRAM;
17. Send the "0x02 " instruction of Noop.

Figure 2-11 The Process of Erasing Internal Flash of GW1N-2(B)/4(B)/6/9 and GW1NZ-1



Note!

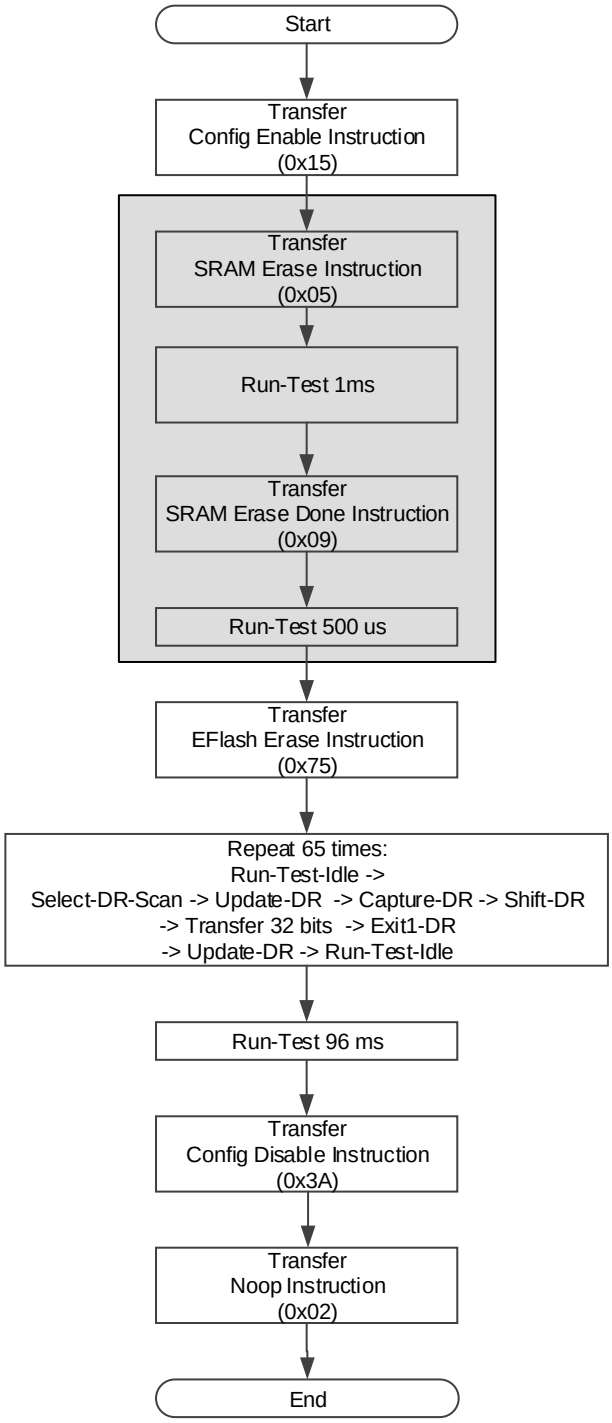
Ignore the shading area operation during Background Programming.

Erasure Process of GW1N-1(S)

The erasure process of GW1N-1(S) chip is different from that of GW1N-2/4/6/9 and GW1NZ-1, as shown in Figure 2-12.

1. Refer to GW1N-2/4/6/9, GW1NZ-1 Erasure Flow, repeat Step 1 to Step 11;
2. Move the state machine from Run-Test-Idle to Shift-DR; 32 clocks are generated (TDI signal keeps low level). Move the state machine to Exit1DR at the 32th clock, and then return to Run-Test-Idle going from Update-DR;
3. Repeat the steps above, 65 times in all;
4. The clock (Run-Test) is continuously generated in Run-Test-Idle for 120ms. Please refer to Table 2-5 for the frequency requirements;
5. Send the "0x3A" instruction of ConfigDisabled;
6. Send the "0x02 " instruction of Noop;
7. Send the "0x3C" instruction of Repogram to load the data in Flash to SRAM;
8. Send the "0x02 " instruction of Noop.

Figure 2-12 The process of Erasing Internal Flash of GW1N-1(S)



Erasure Process of GW1NS(E)-2(C)

GW1NS(E)-2(C) offers two built-in Flash. Note the different Flash when programming. Refer to the process below:

1. Check if the device ID is matched; (Optional)
2. Send the "0x15" instruction of ConfigEnable. Please refer to 2.2.5 for the timing.
3. If the second flash needs to be erased, send the "0x78" instruction of Flash 2nd Enable.

   Note!

   The condition to erase the second Flash is that FPGA should be in the Wakeup state. (Done Final of Status Code should be 1.)
4. Send the "0x09" instruction of SRAM Erase Done;
5. Move the state machine to Shift-DR. Generate a clock of 110ms. Please refer to Table 2-5 for the frequency requirements;
6. Send the "0x3A" instruction of ConfigDisabled;
7. Send the "0x02 " instruction of Noop to end the process.

2.2.9 Process of Programming Internal Flash

The internal Flash uses 256Bytes as an X-page. Each X-page is divided into 64 Y-pages, and each Y-page contains 4Bytes.

The first Y-page of the first X-page is used to identify whether the Flash can configure the Autoboot or the Readback. As shown in Table 2-6. When the first Y-page is written to the Readable-pattern, the Flash data can be read; when the first Y-page is written to the Autoboot-pattern, the device automatically loads the Flash data into the SRAM in the autoboot mode; The flash can be read after the Readable-pattern is wrote, and the flash cannot be read in other cases. The devices with Backgroud programming just need to use Autoboot-pattern.

The requirements for JTAG programming frequency are different according to the different processes of the the internal Flash of GW1N series. Please refer to *2.2.8 Erase* Internal Flash*> Table 2-5* TCK Frequency Requirements for JTAG.

Table 2-6 Readback-pattern / Autoboot-pattern Device

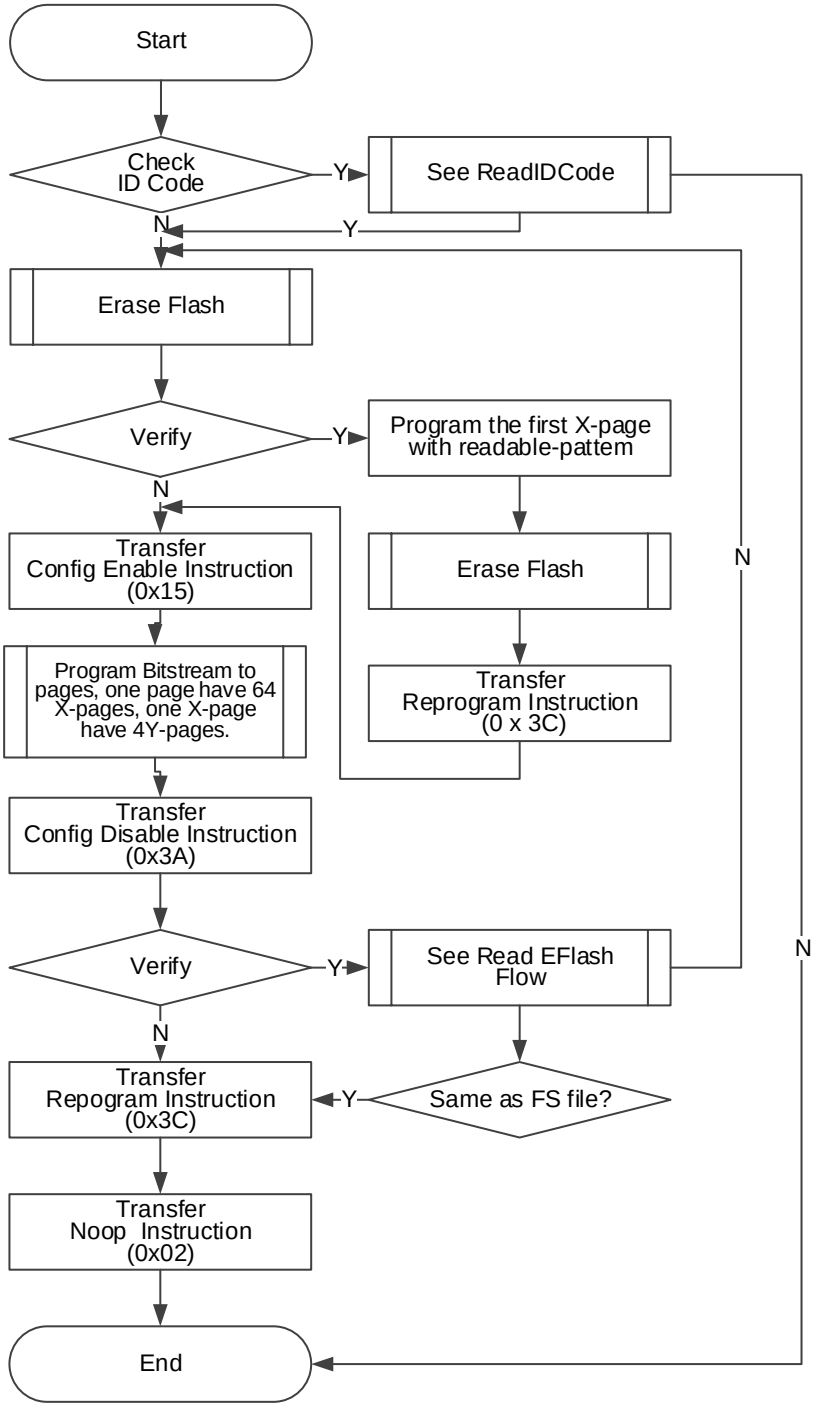
Device	Readable-pattern(4 Bytes)	Autoboot-pattern(4 Bytes)
GW1N-1 / GW1N-1S	0x07,0x07,0x30,0x40	0x47,0x57,0x31,0x4E
GW1N(R)-2/4 GW1N(R)-2B/4B/9 GW1NZ-1 GW1NS(E)-2(C)	0xF7,0xF7,0x3F,0x4F	

The process of programming internal flash is shown in Figure 2-13:

1. Check if the IDcode matches;
2. Erase Flash;
3. Verify if the erasure is successful by reading the Status register to check if the device has been restored to the initial state of the die;the background programming devices and the GW1NS series of devices can not be checked by reading the Status register;
4. Send the "0x15" instruction of ConfigDisble;

5. Write one x-page at a time until the programming is completed;
6. Send the "0x3A" instruction of ConfigDisabled;
7. Send a Reprogram instruction (0x3C) to load the Flash data into the SRAM;
8. Read Status Code/User Code to check if the loading is sucessful.

Figure 2-13 Process of Programming Internal Flash View



Process of Programming A X-page

The process of programming a X-page is as shown

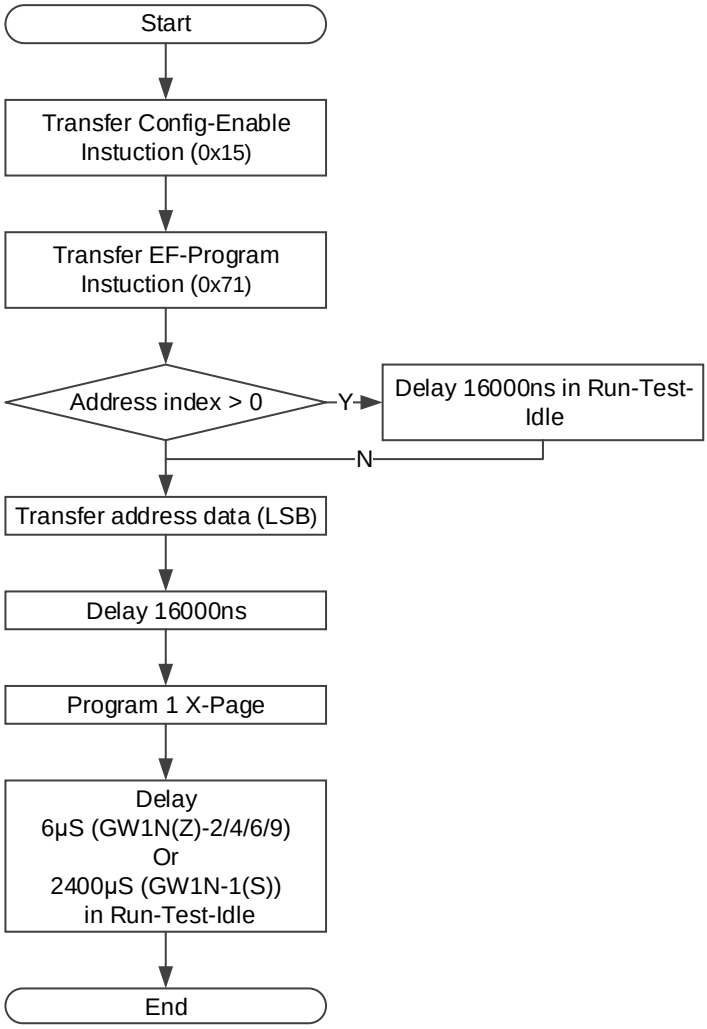
Figure 2-14.

1. Send the "0x15" instruction of ConfigDisble;
2. Send the "0x71" instruction of EF-Program;
3. Enter into Shift-DR and send address data[1];
4. Write an X-page data.
   One Y-page has 256 bytes in all. Program 4 Bytes and program 64 times for one Y-page. The Y-page data is written in LSB way. Refer to Figure 2-15.
5. After one X-page is written in, GW1N-1(S) needs to perform Run-Test for 2400us; GW1N(Z)-2/4/6/9 needs to perform Run-Test for 6us. No extra clock is required for the other series of devices.
6. This X-page programming ends.

Note!

Address data format is 32 bits altogether, and the lower 6 bits are reserved. For example, when the address is b'**00010011**（0x13）, the write-in address is **00000000000000000000010011**000000. The address data is written in LSB way. Jump out of Shift-DR at the last bit.

Figure 2-14 X-page Programming



Programming a Y-page Process

Y-page programming is the smallest unit in programming process. 4 Bytes are written each time in the LSB way, as shown in Figure 2-15.

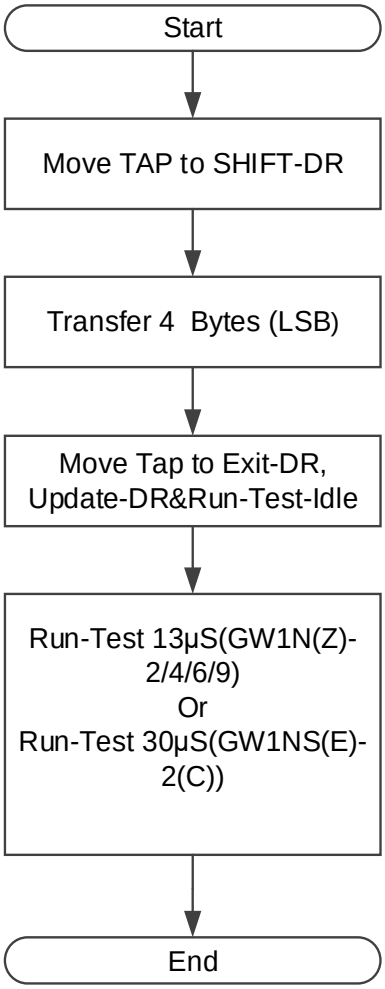
Different series of devices all need to perform Run-Test to wait for writing all Bytes, and the JTAG clock needs to meet minimum frequency requirements. Refer to Table 2-5.

After one Y-page is written in each time, GW1N(Z)-2/4/6/9 needs to perform Run-Test for 13-15μS; GW1N(S)-2(C) needs to perform Run-Test for 30-35 μS. No extra clock is required for the other series of devices.

Note!

If you want to read data from Configuration Data, high 4 Bytes will be taken. If you want to write data during Shift-DR, write from the lowest bit (LSB).

Figure 2-15 Y-page Programming



2.2.10 Process of Reading internal Flash

This chapter introduces the internal Flash process overview with no requirement for TCK of JTAG. As shown in Figure 2-16.

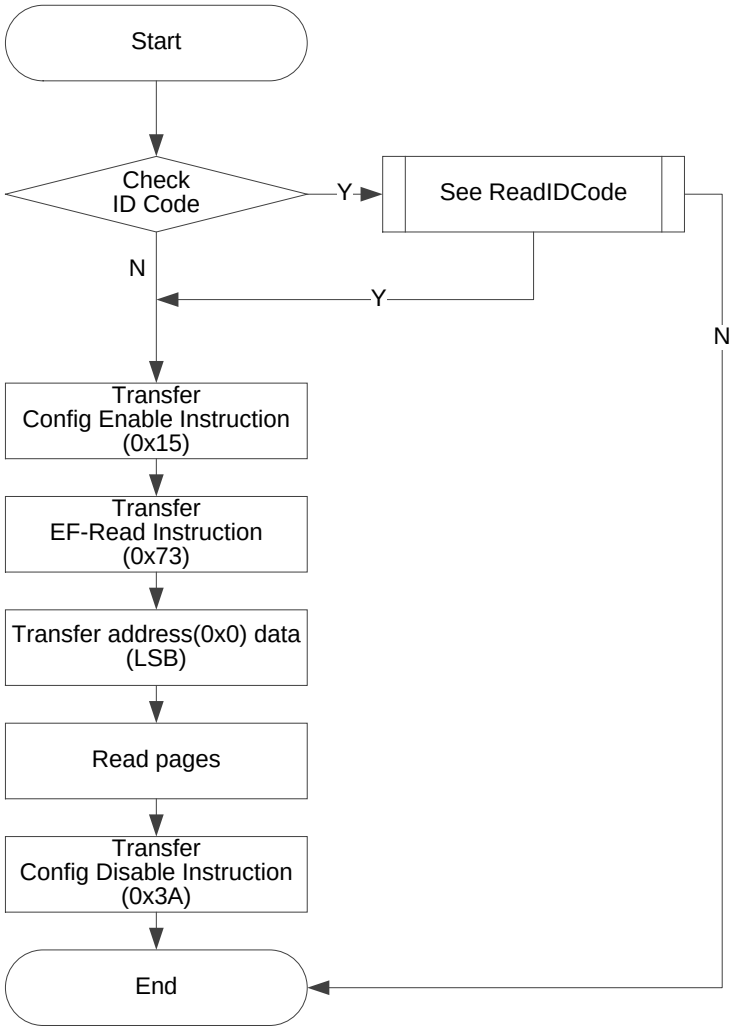
Reading the internal Flash can be understood as the reverse process of programming flash. Firstly, you should make sure that the Readable-pattern that has been written is available. For GW1N, the

Reprogram (0x3C) and Noop (0x02) can be sent to make the internal flash be Readable.

Process Discription:

1. Check IDCode. (optional);
2. Send the "0x15" instruction of ConfigDisble;
3. Send EF-Read instruction 0x73;
4. Send read Flash starting address 0x0. The method is same as write X-address in 0;
5. 64 Y-page is an X-page;
6. After reading one X-page, need not to send address again. The address will recurse automatically;
7. After reading, send the "0x3A" instruction of ConfigDisble to end the process.

Figure 2-16 Process of Reading Internal Flash

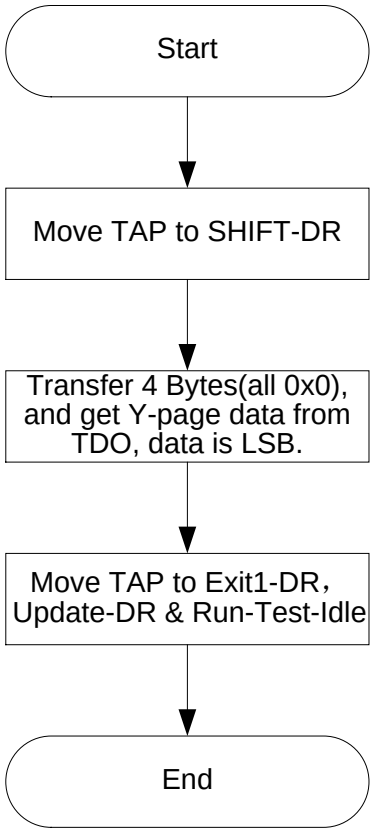


Process of Reading a Page (Y-page) Flash

Reading a Y-page is similar to writing a Y-page, but there is no waiting time for writing in Flash. As shown in Figure 2-17.

the lowest bit in the data is outputted first.

Figure 2-17 Process of Reading a Y-page



2.2.11 Background Programming

The device sometimes needs to upgrade the data file and program the Flash without affecting current functions. It can maintain the I/O state when adding a new data stream file. The following is the flow of GW1N4 that upgrades the internal Flash data using the Background Programming.

Figure 2-18 GW1N-4 Background Programming Flow

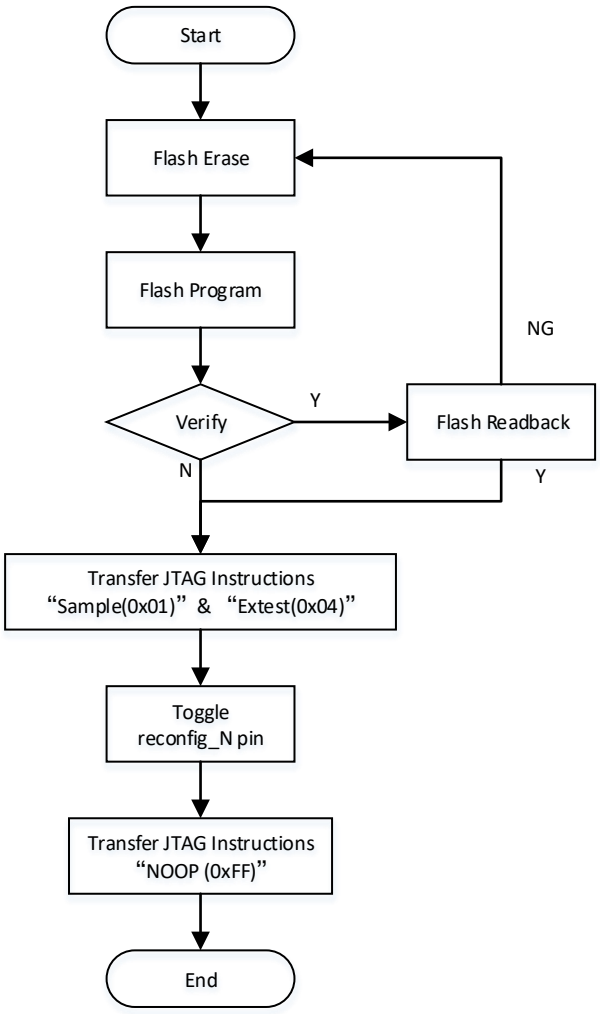
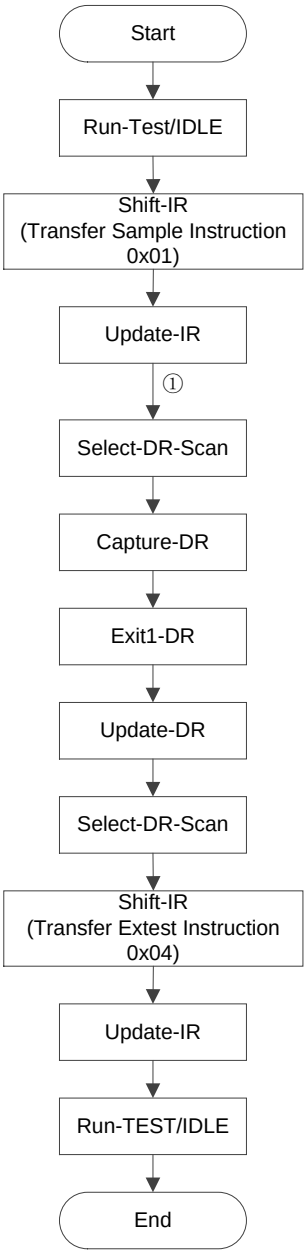


Figure 2-19 Transfer JTAG Instrction Sample & Extest Flow Chart



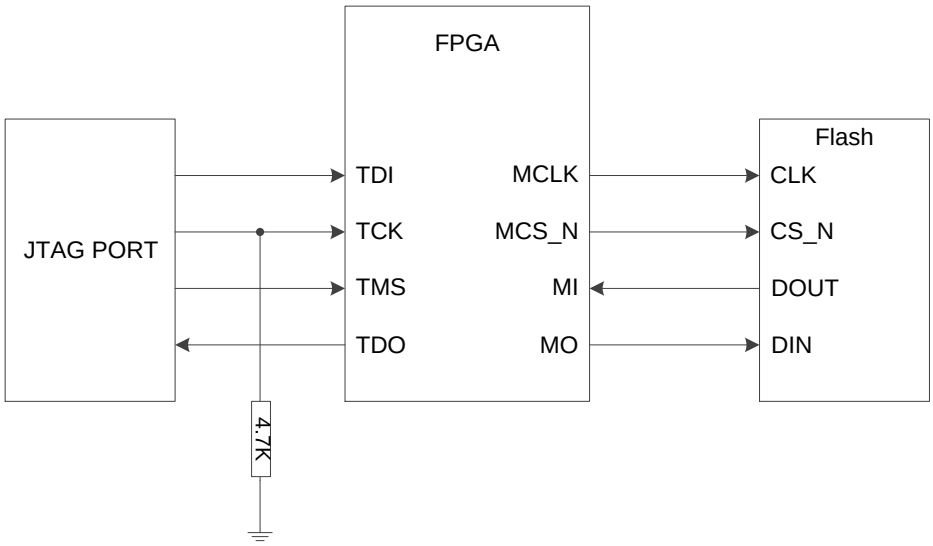
Note!

① Jump directly from Update-IR to Select-DR-Scan.

2.2.12 ExFlash Programming

Gowin FPGA can load data flow files from external Flash and program external Flash through JTAG directly.

Figure 2-20 Connection Diagram of JTAG Programming External Flash



Note!

The figure above shows the minimum system diagram of external Flash of JTAG interface programming. The MODE value is set to ″011″ (programming external Flash via boundary scan is independent of MODE value).

Program SPI Flash Using config-mode[2:0]=011 Mode

In this mode, the external Flash can be programed through JTAG port and the specific value in MODE [2:0] is 011.

The principle of this mode is to connect the JTAG interface to the flash interface in the form of transmission. User operates the FSM at JTAG port to analog SSPI timing to program SPI Flash.

Figure 2-21 Program SPI Flash Using config-mode [2:0]=011 Mode

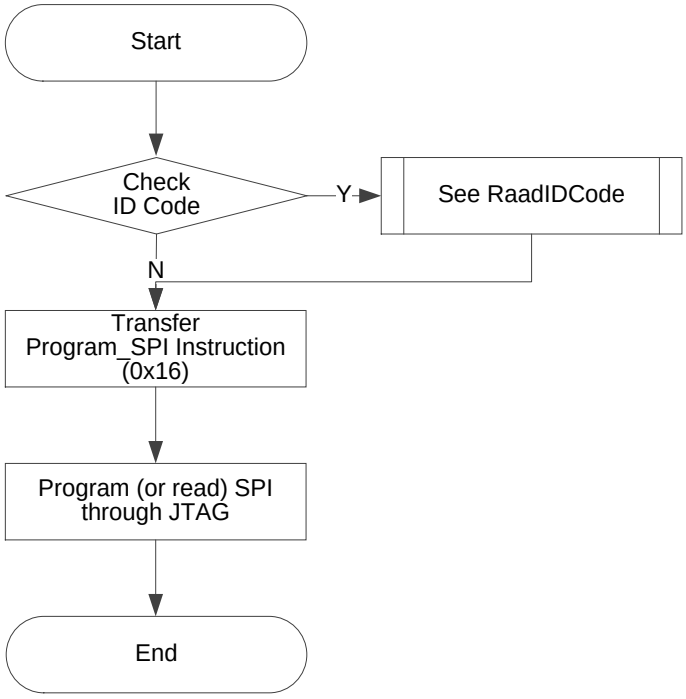


Figure 2-22 Timing diagram of GW2A series JTAG Send 0x06 via SPI

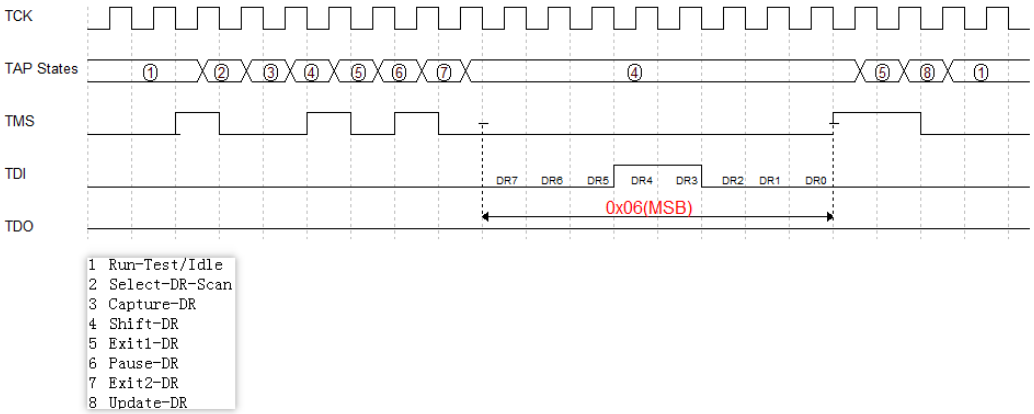
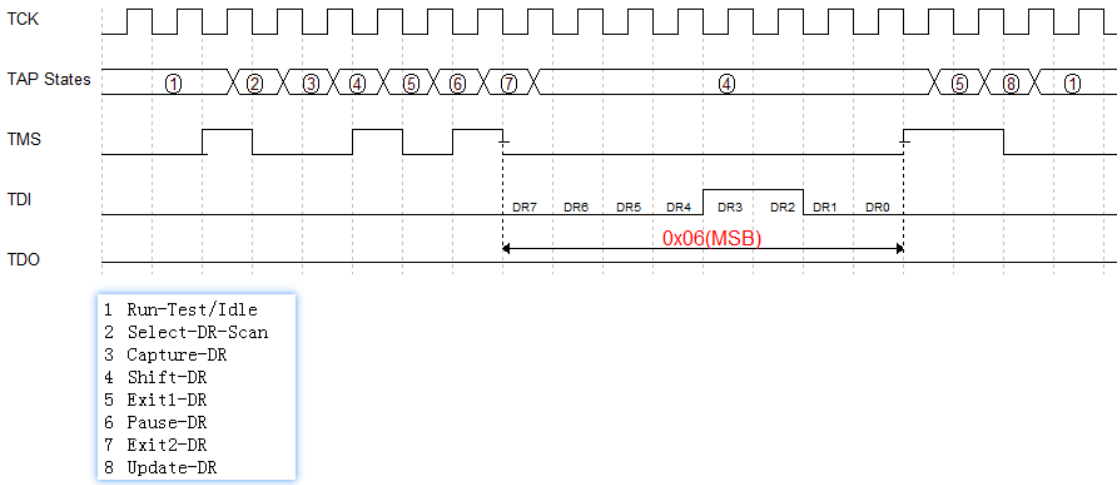


Figure 2-23 Timing diagram of GW1N series JTAG Send 0x06 via SPI



Program SPI Flash Using Boundart Scan Mode

The principle of this mode is changing the state of the pins connected to SPI by using Boundary Scan method to achieve SSPI timing, and then to program the internal Flash.

The length of the Boundary Scan Chain used in this mode is 8-bit. Every 2-bit combination corresponds to the pin state. as shown in Table 2-7. One SCLK drive completes every two times sending of Boundary Scan Chain.

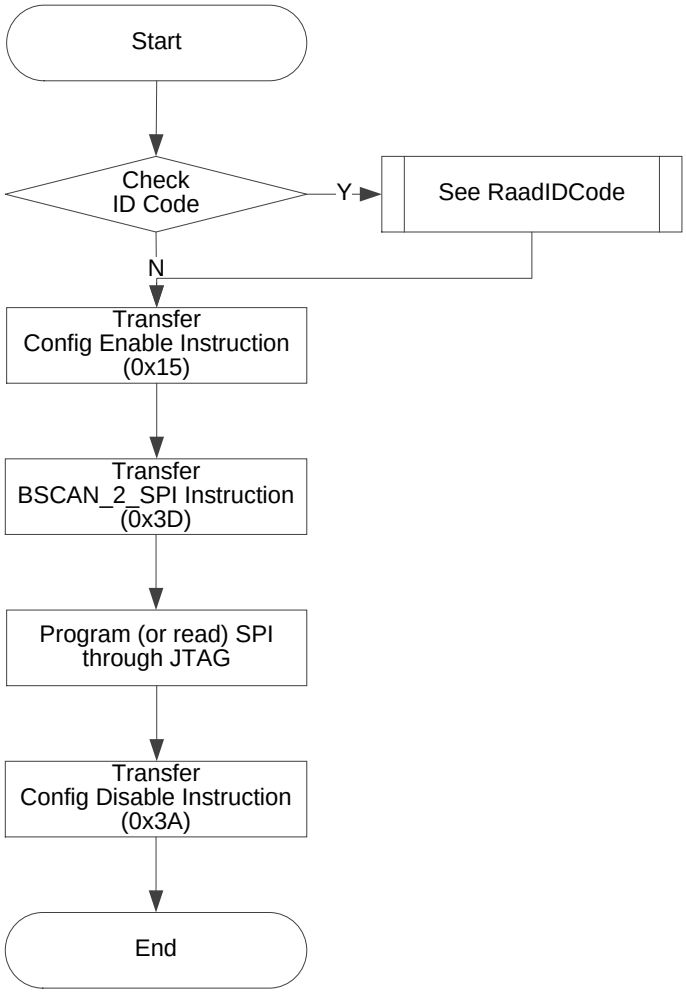
Table 2-7 Pin State

Pins Name of SPI Flash	SCLK		CS		DI		DO	
Bscan Chain[7:0]	7	6	5	4	3	2	1	0
(ctrl & data)	0		0		0		1	

Note!

- ctrl:0 means output, 1 means input;
- data:0 means low, 1 means high.

Figure 2-24 Process of Use Boundary Scan Mode To Program SPI Flash



2.2.13 Read Status Register 0x41

The status of the device can be preliminarily determined by reading the Status Register, such as whether it is successful wakeup, whether there is a timeout error, id check error, crc check error, etc.

The Status Register is 32 bits, the read instruction is 0x41 and the timing is the same as that of Read ID Code.

The meaning of the Status Register is as shown in Table 2-8.

Table 2-8 Status Register Meaning

Device / Status Register[31:0]	GW1N(R)-1/2/4	GW1NS-2 GW1NS(R)-2C	GW1N(R)-6/9 GW1NZ-1	GW2A-18/55
0	CRC Error			
1	Bad Command			
2	ID Verify Failed			
3	Timeout			
4	0			
5	Memory Erase			
6	Preamble			
7	System Edit Mode			
8	Program spi flash derectly			
9	0			
10	Non-jtag configuration is active			
11	Bypass			
12	Gowin VLD(1)			0
13	Done Final			
14	Security Final			
15	Ready(1)	Ready(0)	Ready(1)	Encrypted format
16	POR(1)			Encrypted key is right
17	0	Flash1 Lock	Flash Lock	0
18	0	Flash2 Lock	0	0
19-31	0			

2.2.14 Read Code 0x13

The user code is 32 bits, the read instruction is 0x13 and the timing is the same as that of Read ID Code.

The user code adopts the checksum value in the FS file by default. It can be redefined using Gowin Designer.

2.2.15 Reload 0x3C

This instruction is used to read the bitstream files from Flash and write to SRAM.

Send the instructions of Reprogram (0x3C) and Noop (0x02) to reload the device via JTAG. You can also reload the device by triggering the Reconfig_N pin.

2.2.16 Erase SRAM 0x15

This instruction is used to erase SRAM.

Send the instructions of ConfigEnable(0x15), EraseSram(0x05), Noop(0x02), ConfigDisable(0x3A), Noop(0x02) TO erase SRAM via JTAG.

Note!

You need to wait enough time for the device to finish erasing after the instructions of EraseSram(0x05) and Noop(0x02) are sent.

- The reference time for GW1N-1 is 1ms;
- The reference time for GW1N-2/4 is 2ms;
- The reference time for GW1N-6/9 is 4ms.

3 Example File

For the example file, please contact GOWINSEMI technical support or the local office.


GOWIN
PROGRAMMING FOR THE FUTURE