# Gowin Programmer

# User Guide

**SUG502-1.4E, 06/01/2022**

## 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. 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 |
|---|---|---|
| 04/06/2017 | 1.0E | Initial version published. |
| 08/06/2017 | 1.1E | Device programming operation modified. |
| 10/28/2019 | 1.2E | ● Slave SPI Mode added.<br>● SVF File Creation added.<br>● User Flash Initialization added. |
| 02/17/2020 | 1.3E | The description of installing and starting Programmer added. |
| 06/01/2022 | 1.4E | ● Section 2.1 Introduction to Programmer Tool Chain added.<br>● Chapter 4 Programmer_cli Programming Download Flow added. |

# Contents

# List of Figures

# List of Tables

# 1 About This Guide

## 1.1 Purpose

This guide describes how to use Gowin Programmer.

The software screenshots and the supported products listed in this guide are based on 1.9.3.01Beta. As the software is subject to change without notice, some information may not remain relevant and may need to be adjusted according to the software that is in use.

## 1.2 Related Documents

The latest user guides are available on GOWINSEMI Website. You can find the related documents at www.gowinsemi.com:

- SUG100, Gowin Software User Guide
- TN653, Gowin FPGA Products JTAG Programming and Configuration Manual
- UG290, Gowin FPGA Products Programming and Configuration Manual

## 1.3 Terminology and Abbreviations

Table 1-1 shows the abbreviations and terminology used in this guide.

**Table 1-1 Terminology and Abbreviations**

| Terminology and Abbreviations | Meaning |
|---|---|
| FPGA | Field Programmable Gate Array |
| SRAM | Static Random Access Memory |
| I/O | Input/Output |
| BSDL | Boundary Scan Description Language |
| GAO | Gowin Analyzer Oscilloscope |

# 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 Introduction

## 2.1 Introduction to Programmer Tool Chain

### 2.1.1 Programmer.exe

The graphical tool Programmer.exe is Gowin FPGA downloader, which provides a graphical operation interface and provides bitstream configuration or download functions conveniently and intuitively.

### 2.1.2 Programmer_cli.exe

Programmer_cli is the command line version of Programmer.

### 2.1.3 JTAGLoading.exe

Gowin SVF command line software tool, currently only Windows version is available; and the current version only supports Gowin USB Cable Version 3.0 and 4.0.

### 2.1.4 jtagserver.exe

jtagserver.exe and jtagserver_lpt.exe\ jtagserver_u2x.exe belong to GAO tool chain.

### 2.1.5 Cable5.uid.up.exe

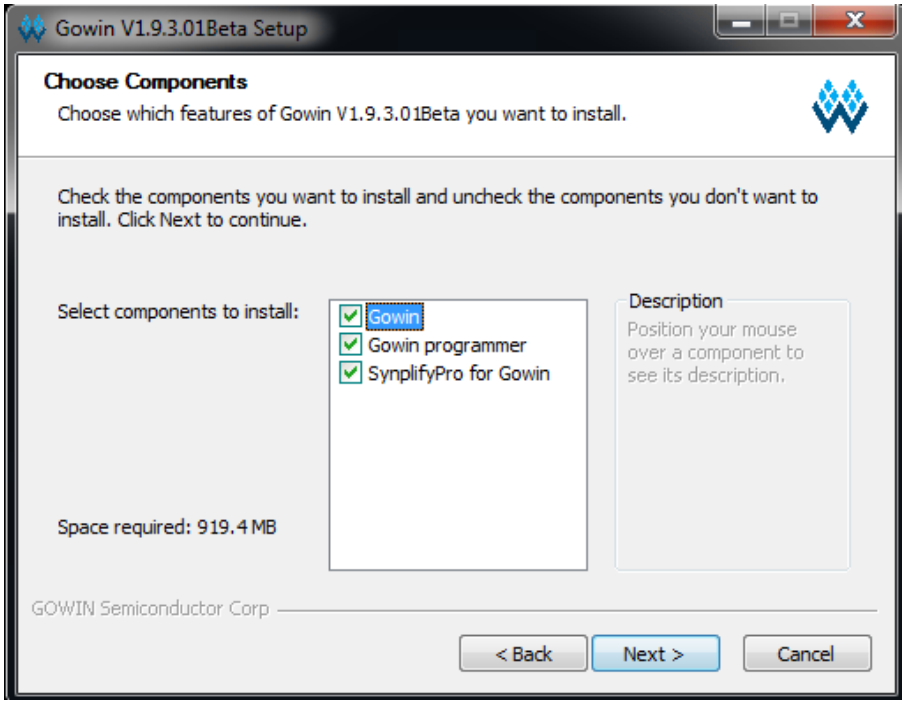
Gowin USB Cable Version 5.0 UID configuration tool

## 2.2 Install and Start Programmer Tool Chain

### 2.2.1 The First Method to Install

When Gowin Software is installed by default, Gowin Programmer is installed, as shown in Figure 2-1. For details on the software installation and application, please refer to *SUG100, Gowin Software User Guide*.

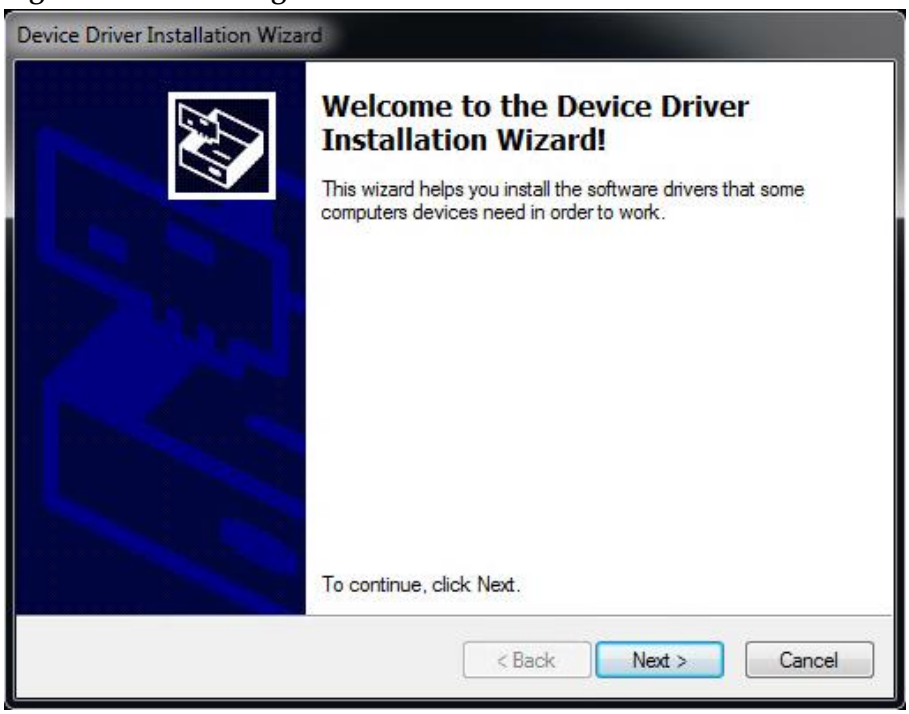
**Figure 2-1 Install Programmer**




## 2.2.2 The Second Method to Install

Download Gowin Programmer installation package at official website: https://www.gowinsemi.com/en/support/download_eda/. After unzipping the installation package, open directory of programmer2\driver. Choose corresponding driver according to your own computer system, as shown in Figure 2-2.

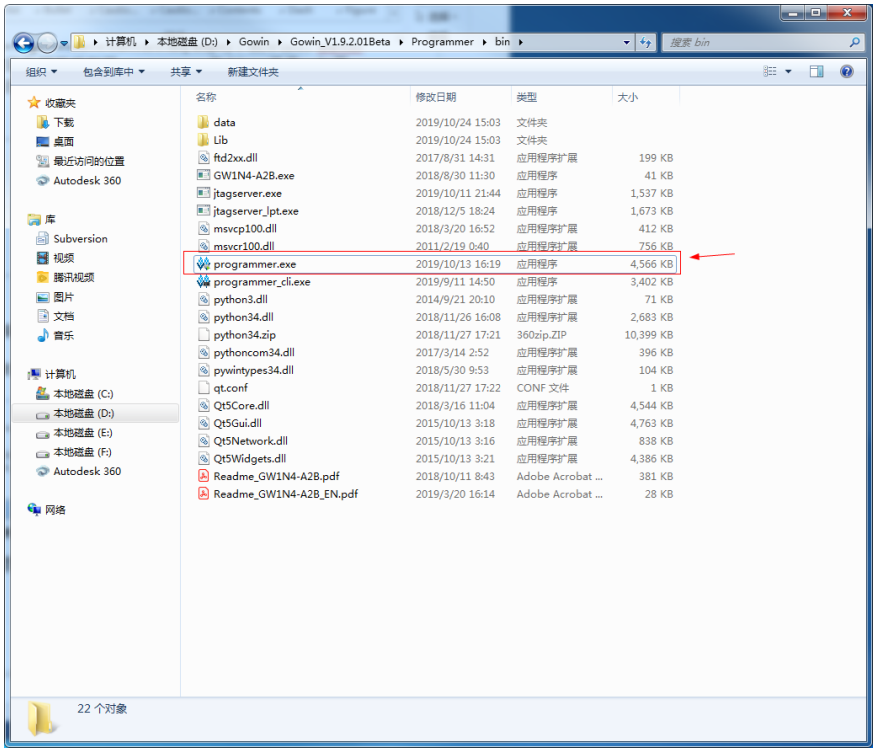
**Figure 2-2 Install Programmer Driver**



### 2.2.3 Start Programmer Tool Chain

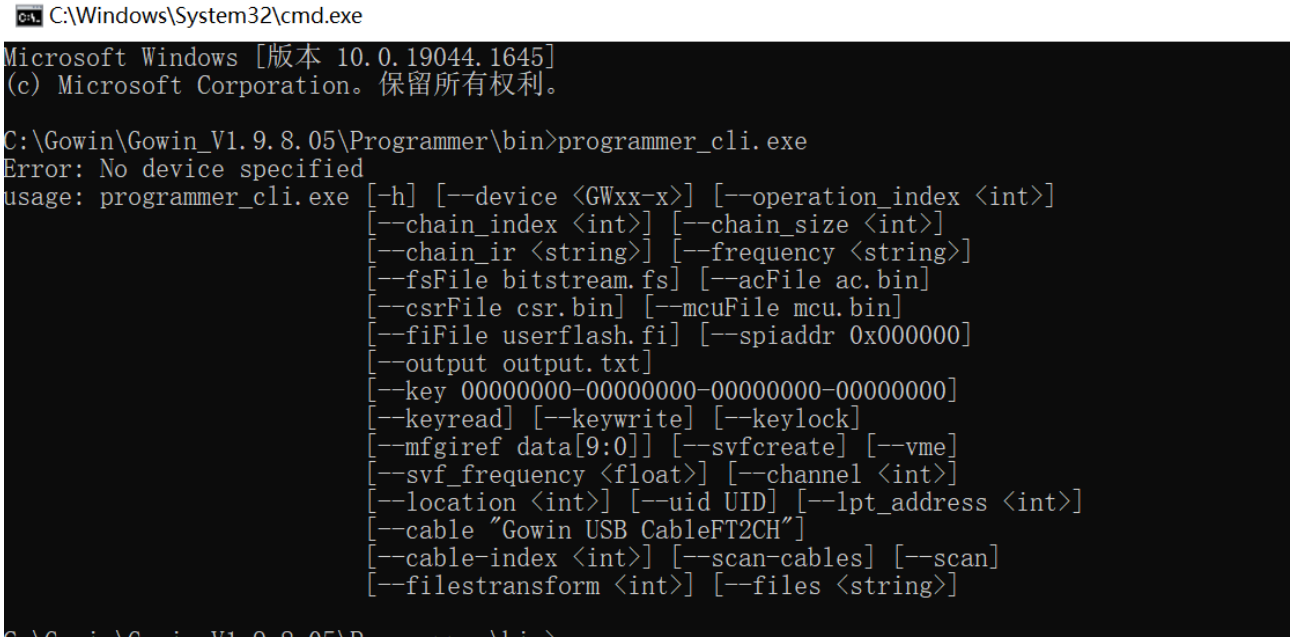
- After installation, the .exe file is under \x.x\Programmer\bin\; Double-click on programmer.exe to start the software, as shown in Figure 2-3.

**Figure 2-3 Start programmer.exe**



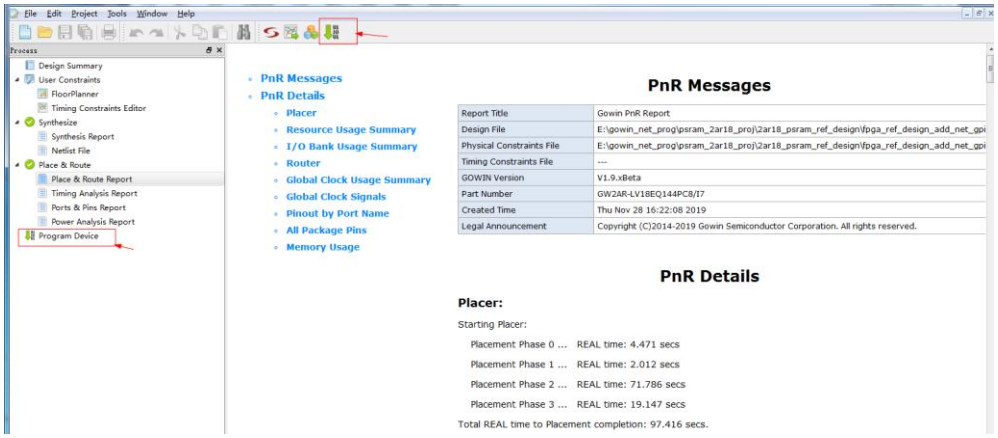

- For command line software, please open it in CMD; for example, open programmer_cli.exe.

**Figure 2-4 Start programmer_cli.exe**



- This software can be started by the shortcut key of Gowin Software, as shown in Figure 2-5.

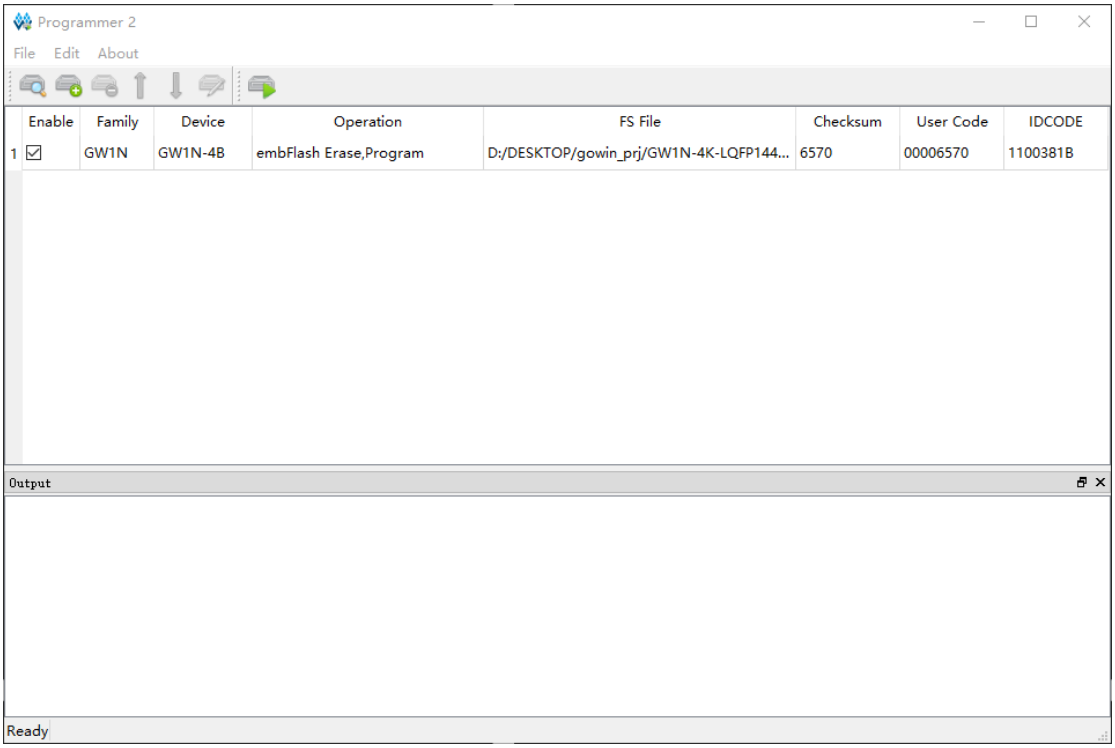
**Figure 2-5 Shortcut Key to Start Programmer**




# 2.3 Software Interface

Gowin Programmer interface includes menu bar, tool bar, device table, output area, as shown in Figure 2-6 .

**Figure 2-6 Programmer Main Window**




In the device table, all the devices that will be programmed in daisy chain are displayed via automatically scanning or manual configuration. Each row of the table represents a device, which can be programmed or not via Enable column.

The device table includes Enable, Family, Device, Operation, FS File, Checksum, User Code, and IDCODE options. Enable, Family, Device, Operation, and FS files are editable and can be edited with clicking.

Double-click to open Device Configuration Dialog to configure the other options. Please refer to 3.4 Device Programming Configuration for the details.

Output area includes Output, Error, Warning, and Info, which respectively displays all information, error information, warning information, and instructions information.

**Note!**

If "Enable" is not checked, the programmer treats the device as if it is not in the chain. The device row is not editable.

# 2.4 Software Version

Gowin Programmer and Gowin IDE have separate software version number, which can be viewed by "About" of software interface, as shown in Figure 2-7.

**Figure 2-7 Version Number**

# 3 Programming Download

Programming download is the process of transmitting the data stream files to SRAM, embedded Flash or external Flash of FPGA via download cable, and programming download process is as follows: Start > Setting download cable > Scanning device > Device programming configuration > Downloading.

1. Download Cable Setting (selectable): Select the download cable type, port, and frequency for the programming download.

**Note!**

The first available port will be selected by default. And the default frequency is 2MHz.

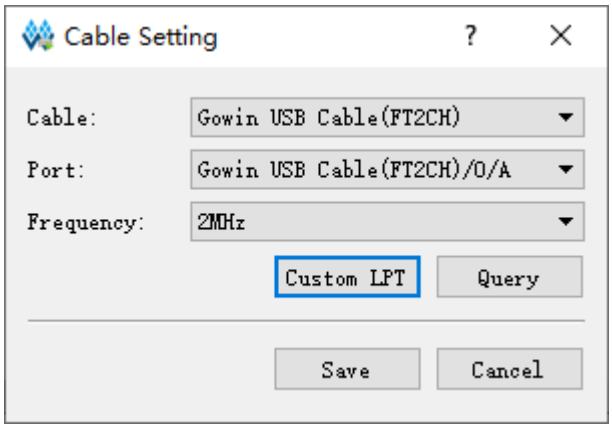
2. Daisy chain configuration and programming: Configure daisy chain in the device table to match the actual physical connection of daisy chain, and select the programming operation and required data file for each device with the uppermost layer being near the Programmer side.
3. Programming download: Download the daisy chain that has been configured, and the final result will be displayed in the output area.

## 3.1 Cable Setting

The Cable Setting allows users to select the available download cable type, port, and frequency for the programming download. Select "Edit > Setting > Cable Setting" in the menu bar to open "Cable Setting". Two types of cables are supported currently: Gowin USB Cable and LPT.

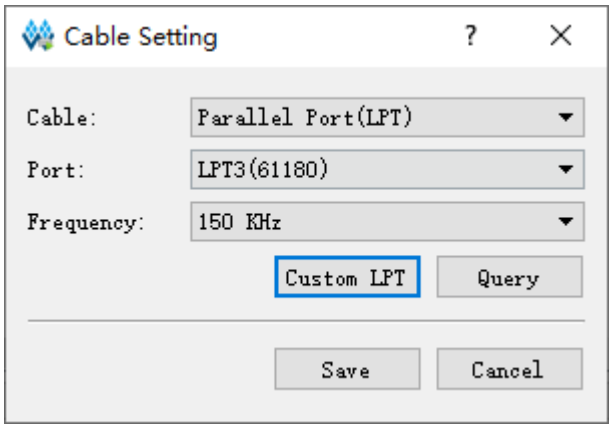
1. Gowin USB Cable (FT2CH) is as shown in Figure 3-1.
   - Cable: Gowin USB Cable.
   - Port: The first available port will be selected by default. The last character A represents the channel number of programmer. There are three channels: S, A and B
   - Frequency: JTAG, 30 MHz, 15 MHz, 2.5 MHz, 2 MHz, 1.5 MHz, 0.75 MHz, 0.1 MHz. The default is 2 MHz.

**Figure 3-1 Gowin USB Cable**




2. LPT is as shown in Figure 3-2.
   - Cable: Parallel Port (LPT)
   - Port: The port available for the download cable is selected according to the PCI property of the computer.
   - Frequency: JTAG frequency simulation, 150KHz.

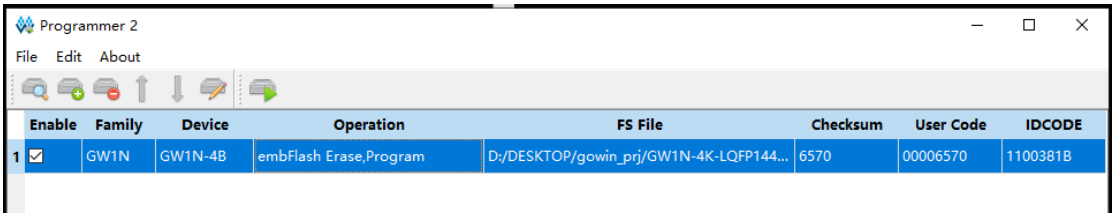
**Figure 3-2 LPT**




# 3.2 Scan Daisy Chain

Programmer automatically scans the daisy chain connected to the computer. Click "  " to scan daisy chain connected to the computer. After scanning, all devices are shown in device table of Gowin Programmer in the order of chain, as shown in Figure 3-3.

**Figure 3-3 Device Table**




**Note!**

Some devices have the same ID (such as GW2A-18/GW2AR-18), which requires users to manually specify the corresponding equipment after scanning.

Programmer supports the manual configuration of daisy chain. It includes the operations of adding device, removing device, and modifying the position of the device in the chain.

# 3.3 Daisy Chain Configuration

## 3.3.1 Add Device

1. Select "Edit > Add Device" or click "  " in menu to add a new device;
2. Click "Family" to select the device family from the drop-down menu;
3. Click "Device" to select a specific device type from the drop-down menu.

**Note!**

When selected, the new device is added to the selected location or the end of the daisy chain.

## 3.3.2 Remove Device

Select the device row and remove the device by clicking "Edit > Remove Device" or the "  ".

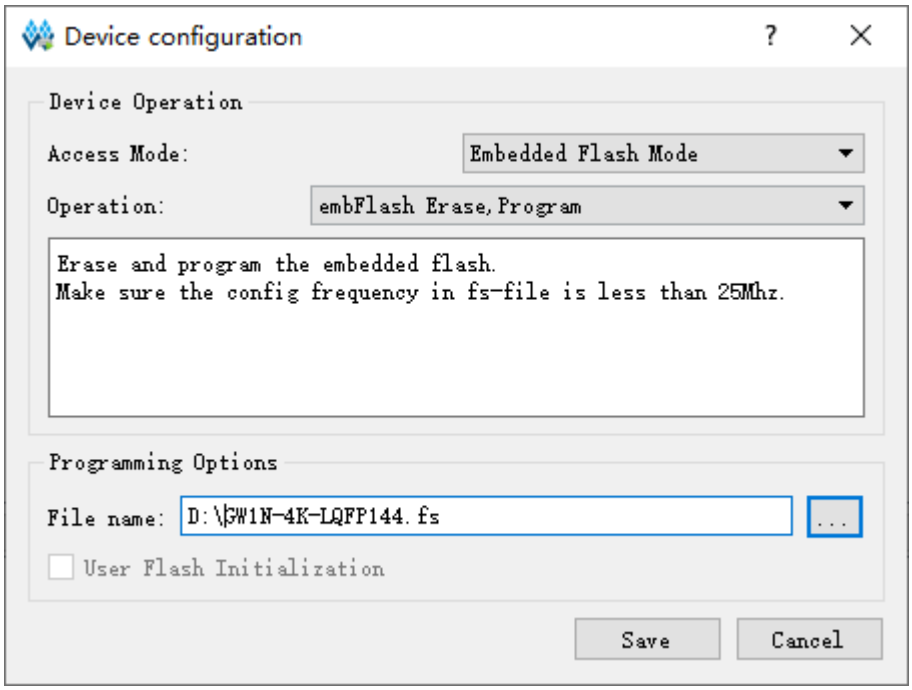
## 3.3.3 Modify Device Position in Chain

Select the device row and modify the device position in the chain by clicking "Edit > Up (or Down) " or the "  " or "  ".

# 3.4 Device Programming Configuration

Select the device row and open the Device Configuration dialog by clicking "Edit > Configure Device" or "  " or double-clicking "Operation", as shown in Figure 3-4.

**Figure 3-4 Device Configuration Interface**




- Access Mode: Select programming mode.
- Operation: Select programming operation, please see Table 3-1 for details.
- Instruction Register Length: When the device is selected as JTAG-NOP, select instruction register length.
- Programming File: Select programming data file.
- Device: When the programming mode is selected as External Flash Mode, select External Flash.
- Start Address: When the programming mode is selected as External Flash Mode, select initial address of External Flash.

**Table 3-1 Device Operations Description**

| Access Mode | Operation | Description |
|---|---|---|
| SRAM Mode | Bypass | Bypass |
| | Read Device Code | Read Device ID, User Code, Status Code. |
| | Read User Code | Read Device User Code |
| | Read Status Register | Read Device Status |
| | Reprogram | – |
| | SRAM Program JTAG 1149 | JTAG writes pure data and does not support CRC validation<br>Encrypted or compressed datastream file is not supported currently. |
| | SRAM Erase | Erase SRAM |
| | SRAM Program | Configure the data stream file to FPGA SRAM |

| Access Mode | Operation | Description |
|---|---|---|
| | SRAM program and Verify | Write data to SRAM and verify |
| Embedded Flash Mode | embFlash Erase, Program | Erase embFlash, then write data. |
| | embFlash Erase, Program, Verify | Erase embFlash, then write data and verify. |
| | embFlash Erase Only | Erase embFlash only |
| External Flash Mode | exFlash Erase, Program | Erase external Flash, then write data to external flash. |
| | exFlash Erase, Program, Verify | Erase external Flash, then write data and verify. |
| | exFlash Program Without Erasure | Write data to external Flash without erasure |
| | exFlash Bulk Erase | Erase external Flash |
| | exFlash Verify | Verify external Flash |
| | exFlash Erase, Program in bscan | Erase external Flash and write data to external Flash in bscan |
| | exFlash Erase, Program, Verify in bscan | Erase external Flash, write data to external flash and verify in bscan. |
| | exFlash Verify in bscan | – |
| | exFlash Program in bscan without erasure. | Write data to external Flash without erasure in bscan. |
| | exFlash Bulk Erase in bscan | Verify external Flash in bscan |
| | exFlash C Bin Erase, Program | Erase external Flash, then write RISC-V bin files to external Flash. |
| | exFlash C Bin Erase, Program, Verify | Erase external Flash, then write RISC-V bin files to external Flash and verify. |
| | exFlash C Bin Program | Write RISC-V bin files to external Flash |
| Slave SPI Mode | Slave SPI Read ID Code | Read ID Code in SSPI |
| | Slave SPI Scan exFlash | Scan exFlash in SSPI |
| | Slave SPI Program SRAM | Write data to SRAM in SSPI |

**Note!**

GW2A/GW2AR series of chips do not have embedded Flash and do not support this mode.

### 3.4.1 SRAM Configuration

1. Select the device row and open the Device Configuration dialog by clicking "Edit > Configure Device" or "  " or double-clicking "Operation".
2. Select SRAM Mode in "Access Mode" drop-down list;
3. Select operation in “Operation” drop-down list as required;
4. For non-Gowin devices, you need to manually specify the length of the instruction register or instruct the programmer to read the length of the instruction register of the BSDL file.
5. Click “Save” to complete the configuration.

**Note!**

Non-Gowin device (JTAG-NOP) only supports Bypass.

### 3.4.2 Embedded Flash Configuration of LittleBee® Series of FPGA Products

The GW1N/GW1NZ series of FPGA products includes EmbFlash; The corresponding programming mode can be Embedded Flash Mode.

1. Select the device row and open the Device Configuration dialog by clicking "Edit> Configure Device" or "  " or double-clicking "Operation".
2. Select Embedded Flash Mode in "Access Mode" drop-down list;
3. Select operation in "Operation" drop-down list as required;
4. Programming File: Select programming data stream file.
5. Click “Save” to complete the configuration.

### 3.4.3 ExFlash Configuration

Gowin programmer supports ExFlash programming. The external flash configuration process is as follows:

1. Select the device row and open the Device Configuration dialog by clicking "Edit> Configure Device" or "  " or double-clicking "Operation".
2. Select "External Flash Mode" in "Access Mode" drop-down list;
3. Select operation in "Operation" as required;
4. If "exFlash Program" selected in "Operation", the corresponding programming data stream file needs to be selected in "Programming File";
5. For external Flash models, the LittleBee® and Arora family support SPI Flash with read commands of 0x03 or 0x0B;
6. If there is no flash selected in the menu, please select Generic Flash to try to automatically program;
7. Select the initial address of the external Flash. Currently, the default is 0x000000;
8. Click “Save” to complete the configuration.

### 3.4.4 Slave SPI Mode

In Slave SPI Mode, the download cable shall be connected to the pin used by SSPI, please see *UG290, Gowin FPGA Products Programming and Configuration Guide*.

1. Select the device row and open the Device Configuration dialog by clicking "Edit> Configure Device" or "  " or double-clicking "Operation".

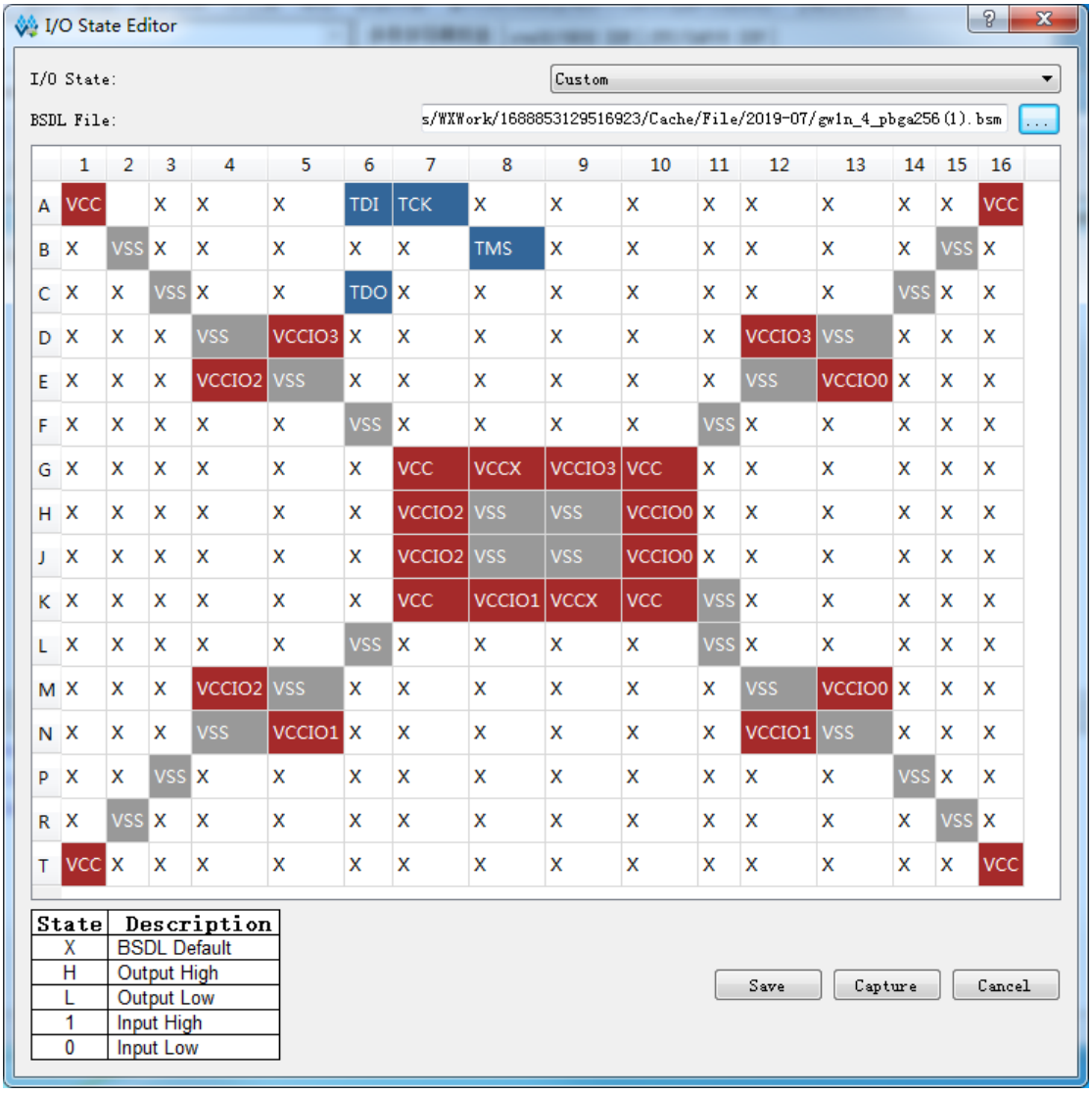
2. Select "Slave SPI Mode" in "Access Mode" drop-down list;
3. Select operation in "Operation" drop-down list as required;
4. If selected "Slave SPI Program SRAM" in "Operation", the corresponding programming data stream file needs to be selected in "Programming File";
5. Click “Save” to complete the configuration.

# 3.5 Edit Pin State

Programmer edits the I/O pin value via the I/O State Editor, which can be set before programming.

1. Select the device row and open I/O State Editor by clicking "Edit > I/O State" or right clicking "I/O State";
2. Select the BSM file that conforms with the device model and package;
3. Change the pin state by clicking on the cell location or set the same state for all pins by right-clicking on the menu.

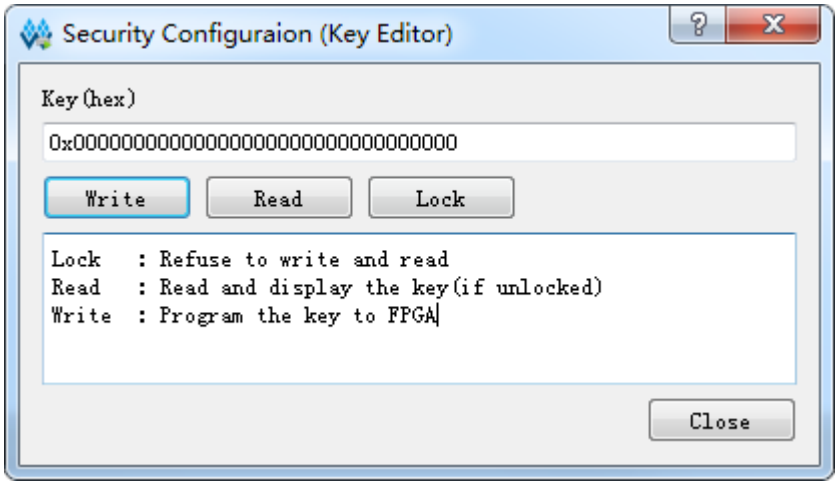
**Figure 3-5 I/O State Editor**



# 3.6 Device Security

When programming with encrypted bitstream files, you need to write the key of the bitstream file to FPGA. Select the device row and open Security Configuration dialog by clicking "Edit > Configure Security" from the menu or right-clicking "Security Key Setting", as shown in Figure 3-6.

**Figure 3-6 Security Configuration**




- Write: Write the specified key value to the FPGA.
- Read: Read and display the key value in the FPGA in the case of unlocking.
- Lock: Lock the key in the FPGA, and the key is unable to be read or written after being locked.

After the decryption key is written successfully, readback the written value via the "Read" button on the interface to verify.

After the key is written successfully, lock it in the FPGA via the Lock command. Once you have performed this operation, any read and write key operations will be invalid, the key value cannot be modified, and the read bits are all 1.

After the decryption key is set, the encrypted bitstream data will only work when the data matches the decryption key. The key does not affect the non-encrypted bitstream data.

**Note!**

All bits of the initial value of the Gowin FPGA keys are 0. If a certain bit of the key value is changed to 1, it cannot be changed back to 0. For example, the key value written during an operation is 00000000-00000000-00000000-00000001, and the last bit of the modified key must be 1. For more detailed information, please refer to *TN654, GW2A(R) series of FPGA Products AES Programming Guide*.

# 3.7 Programming Download

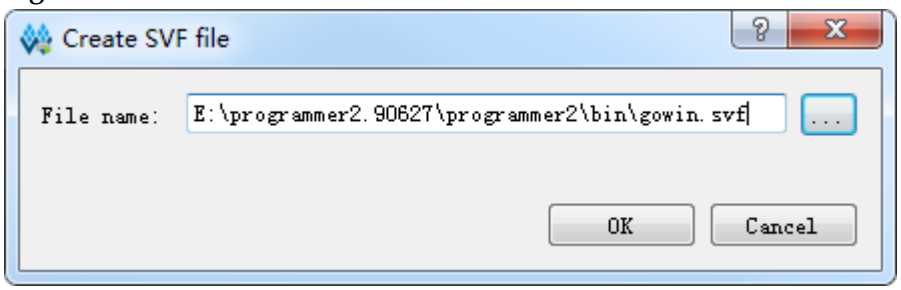
After configuring the download cable and daisy chain, select "Design > Run" from the menu or click "▶" to program and download the device. The final result will be displayed in the output panel.

# 3.8 SVF File Creation

SVF file creation is supported by fs file. Currently, only GW1N-4 SVF file creation is supported.

1. Configure embedded Flash mode according to the GW1N/GW1NZ and select GW1N4 device.
2. Select the device chain and open the Create SVF File dialog by clicking, "Edit> SVF File Create" or right-clicking "SVF File Create".
3. The generated SVF file can be named in File Name and it can choose SVF save path, as shown in Figure 3-7.
4. Click "OK" to complete the creation of the SVF file.

**Figure 3-7 Create SVF File**




# 3.9 User Flash Initialization

LittleBee® series provides users with User Flash space. User Flash data can be used to program embedded Flash and User Flash space at the same time. For the security of the design, this operation only supports user flash programming on the Programmer side and readback is not supported. You can choose user flash initialization file with .fi suffix when programming, as shown in Figure 3-8 .

**Figure 3-8 User Flash Initialization**

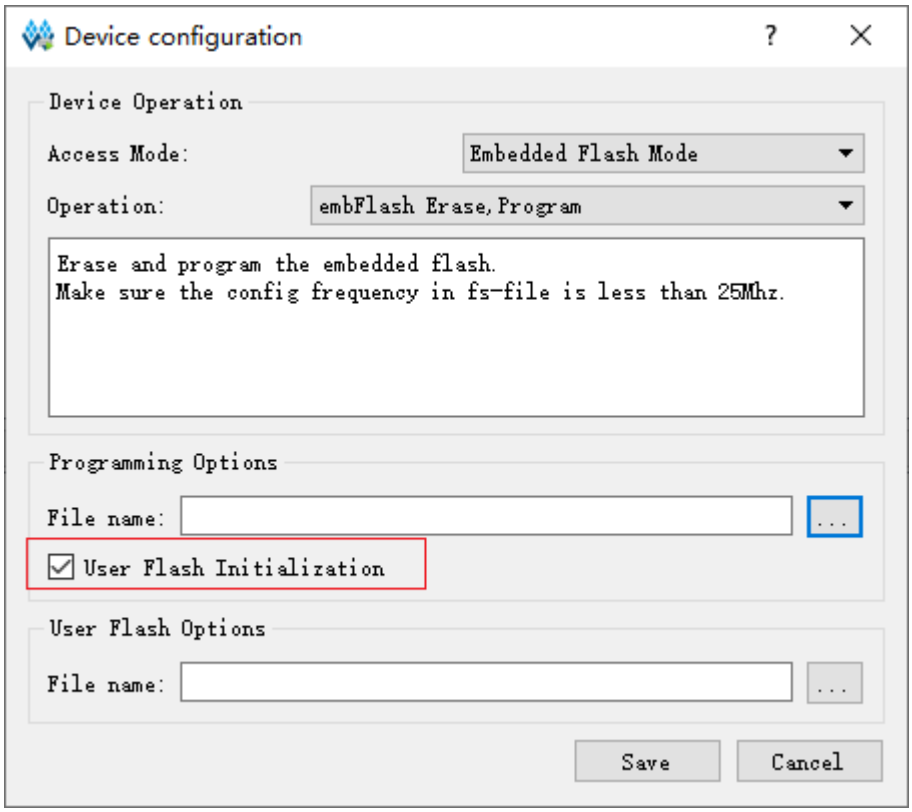



**Table 3-2 User Flash Reference**

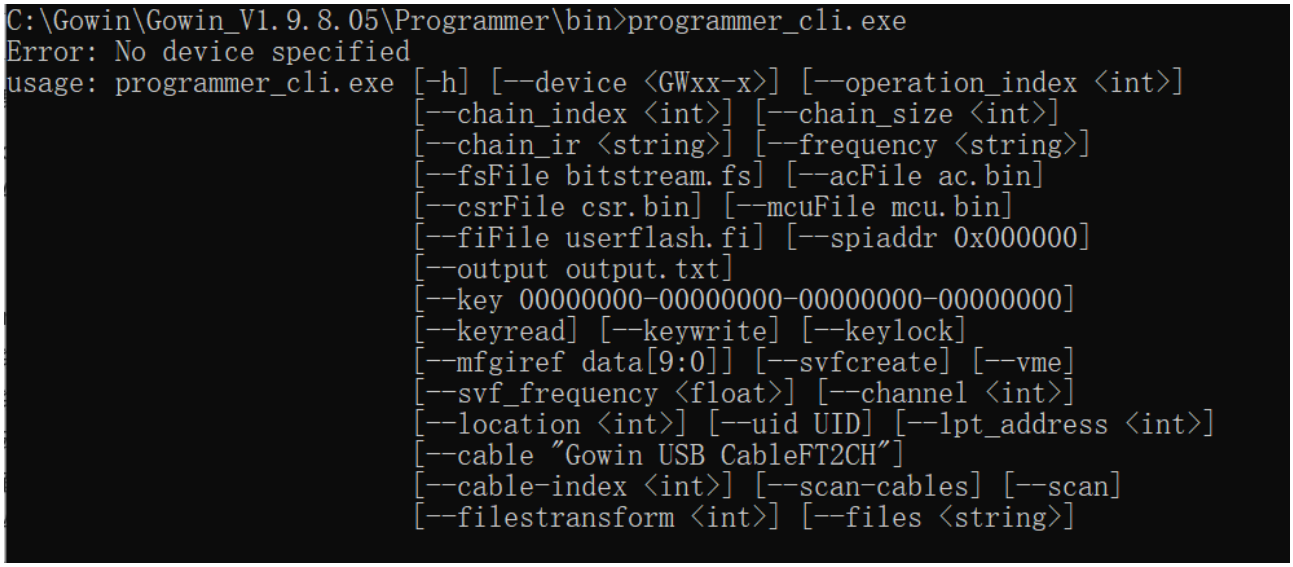
| Series | Device | Flash Type | Address | Data Width |
|---|---|---|---|---|
| GW1N | GW1N-1 | FLASH96K | 48* 64 | 32Bits |
| | GW1N-1S | | | |
| | GW1N-2 | FLASH256K | 128* 64 | |
| | GW1N-2B | | | |
| | GW1N-4 | | | |
| | GW1N-4B | | | |
| | GW1N-6 | FLASH608K | 304* 64 | |
| | GW1N-9 | | | |
| GW1NR | GW1NR-4 | FLASH256K | 128* 64 | |
| | GW1NR-4B | | | |
| | GW1NR-9 | FLASH608K | 304* 64 | |
| GW1NS | GW1NS-2 | FLASH128K | 32786 | |
| | GW1NS-2C | | | |
| GW1NSR | GW1NSR-2 | FLASH128K | 32786 | |
| | GW1NSR-2C | | | |
| GW1NZ | GW1NZ-1 | FLASH64KZ | 32* 64 | |

# 4 Programmer_cli Programming Download Flow

## 4.1 Preview and Help

Open the programmer_cli tool in CMD. When no parameters are used, a brief help description will be prompted, and no device specified will also be prompted..

**Figure 4-1 Open programmer_cli in CMD**




Use the parameter –help to get more help information.

**Figure 4-2 Help Information**

```
C:\Gowin\Gowin_V1.9.8.06\Programmer\bin>programmer_cli -h
usage: programmer_cli [-h] [--device <GWxx-x>] [--operation_index <int>]
                      [--chain_index <int>] [--chain_size <int>]
                      [--chain_ir <string>] [--frequency <string>]
                      [--fsFile bitstream.fs] [--acFile ac.bin]
                      [--csrFile csr.bin] [--mcuFile mcu.bin]
                      [--fiFile userflash.fi] [--spiaddr 0x000000]
                      [--output output.txt]
                      [--key 00000000-00000000-00000000-00000000] [--keyread]
                      [--keywrite] [--keylock] [--mfgiref data[9:0]]
                      [--svfcreate] [--vme] [--svf_frequency <float>]
                      [--channel <int>] [--location <int>] [--uid UID]
                      [--lpt_address <int>] [--cable "Gowin USB CableFT2CH"]
                      [--cable-index <int>] [--scan-cables] [--scan]
                      [--filestransform <int>] [--files <string>]

Gowin FPGA Programmer command-line interface. Version 1.9.8.06 build(20526);
Copyright (C) 2014-2022 Gowin Semiconductor Corporation

optional arguments:
  -h, --help            show this help message and exit
  --device <GWxx-x>, -d <GWxx-x>
                        Define a GOWIN FPGA device from:
                        GW1N-1 GW1N-1P5 GW1N-1P5B GW1N-1P5C
                        GW1N-1S GW1N-2 GW1N-2B GW1N-2C
                        GW1N-4 GW1N-4B GW1N-4D GW1N-9
                        GW1N-9C GW1NR-1 GW1NR-2 GW1NR-2B
                        GW1NR-2C GW1NR-4 GW1NR-4B GW1NR-4D
                        GW1NR-9 GW1NR-9C GW1NRF-4B GW1NS-2
                        GW1NS-2C GW1NS-4 GW1NS-4C GW1NSE-2C
                        GW1NSER-4C GW1NSR-2 GW1NSR-2C GW1NSR-4
                        GW1NSR-4C GW1NZ-1 GW1NZ-1C GW2A-18
                        GW2A-18C GW2A-55 GW2A-55C GW2AN-18X
                        GW2AN-55C GW2AN-9X GW2ANR-18C GW2AR-18
                        GW2AR-18C
  --operation_index <int>, --run <int>, -r <int>
                        0: Read Device Codes;
                        1: Reprogram;
                        2: SRAM Program;
                        3: SRAM Read;
                        4: SRAM Program and Verify;
                        5: embFlash Erase,Program;
                        6: embFlash Erase,Program,Verify;
                        7: embFlash Erase Only;
                        8: exFlash Erase,Program;
                        9: exFlash Erase,Program,Verify;
                        10: exFlash Bulk Erase;
                        11: exFlash Verify;
                        12: exFlash Erase,Program in bscan;
                        13: exFlash Erase,Program,Verify in bscan;
                        14: exFlash Bulk Erase in bscan;
                        15: exFlash Verify in bscan;
                        16: SRAM Program JTAG 1149;
                        17: SRAM Program,Verify JTAG 1149;
                        18: bsdl read;
                        19: embFlash 2nd Erase,Program;
                        20: embFlash 2nd Erase,Program,Verify;
                        21: embFlash 2nd Erase Only;
                        22: -R-;
                        23: Connect to JTAG of MCU;
                        24: SRAM Erase;
                        25: Authentication Code Erase,Program,Verify;
```

```
                      26: Authentication Code Read;
                      27: Firmware Erase,Program Securely;
                      28: Firmware Erase Only;
                      29: Firmware Erase,Program;
                      30: Firmware Erase,Program,Verify;
                      31: exFlash C Bin Erase,Program;
                      32: exFlash C Bin Erase,Program,Verify;
                      33: -R-;
                      34: MFG Write iRef;
                      35: CSR File Erase,Program,Verify;
                      36: exFlash Erase,Program thru GAO-Bridge;
                      37: exFlash Erase,Program,Verify thru GAO-Bridge;
                      38: exFlash C Bin Erase,Program thru GAO-Bridge;
                      39: exFlash C Bin Erase,Program,Verify thru GAO-Bridge;
                      40: DK-GoAI-GW1NSR4C_QN48 v1.1;
                      41: DK-GoAI-GW1NSR4C_QN48 v2.2;
                      42: DK-GoAI-GW2AR18_QN88P v1.1;
                      43: I2C Program SRAM;
                      44: sFlash Erase,Program;
                      45: sFlash Erase,Program,Verify;
                      46: sFlash Verify Only;
                      47: sFlash Bulk Erase;
                      48: sFlash Background Erase,Program;
                      49: sFlash Background Erase,Program,Verify;
                      50: -R-;
                      51: -R-;
  --chain_index <int>, -i <int>
                      Define the device index on the chain. The default is 0.
                      It must be used in combination with option: "--chain_size".
  --chain_size <int>, -l <int>
                      Define the device index on the chain. The minimum  length is 1.
                      It must be used in combination with option: "--chain_index".
  --chain_ir <string>   Define the IR_LENGTH of every device. example: 8,8,8,8
  --frequency <string>, --freq <string>
                      default is 2.5MHz. More options:
                      0.75MHz; 1.5MHz; 0.02MHz; 0.3MHz; 10MHz; 2.5MHz; 0.4MHz; 15MHz; 0.5MHz; 0.9MHz; 2MHz; 1.1MHz; 0.
1MHz;
  --fsFile bitstream.fs, --fs bitstream.fs, -f bitstream.fs
                      Define the .fs file path.
  --acFile ac.bin, --ac ac.bin, -a ac.bin
                      Define the Authentication-Code file path.
  --csrFile csr.bin     Define the CSR file path.
  --mcuFile mcu.bin, --fw mcu.bin, --mcu mcu.bin, -m mcu.bin
                      Define firmware file path of MCU.
  --fiFile userflash.fi
                      Define Userflash initialization file path.
  --spiaddr 0x000000    Define starts address of spi-flash.
  --output output.txt, -o output.txt
                      Define output file path. Default is output.txt
  --key 00000000-00000000-00000000-00000000
                      16 byte key string in HEX format.
  --keyread             Read key from FPGA
  --keywrite            Write key to FPGA
  --keylock             Lock key setting
  --mfgiref data[9:0]   Write data[9:5] to tune iref;data[9:0]=itrim[9:5]+freq[4:0]
  --svfcreate           Create SVF file only.
  --vme                 Create VME file after SVF file created.
  --svf_frequency <float>
                      Define a frequency for SVF, default is 2.5 (MHz).
  --channel <int>       Define download cable channel. Default is 0. Only works for Gowin USB Cable(FT2CH)
  --location <int>      Define location number of USB Cable.
                             when use location option, programmer will open the corresponding cable.
                             Default works for Gowin USB Cable(FT2CH) .
```

```
                          Will ignore --channel option
 --uid UID, --unique-id UID
                    Define Unique-ID of USB Cable.
                           when use this option, programmer will open the corresponding cable.
                           Default works for Gowin USB Cable(FT2CH) .
                           Will ignore --location and --channel option
 --lpt_address <int>   Define GOWIN LPT cable address. Default is 0x0378.
 --cable "Gowin USB Cable(FT2CH)"
                    Select a type of USB cable(including quotation marks):
                    "Gowin USB Cable(GWU2X)"
                    "Gowin USB Cable(FT2CH)"
                    "Parallel Port(LPT)"
                    "Digilent USB Device"
                    "Unverified FT Device"
                    Default cable is "Gowin USB Cable(FT2CH)"
 --cable-index <int>   Select a number for USB cable:
                    0: Gowin USB Cable(GWU2X);
                    1: Gowin USB Cable(FT2CH);
                    2: Parallel Port(LPT);
                    3: Digilent USB Device;
                    4: Unverified FT Device;
                    Higher priority than --cable, default cable-index is 0
 --scan-cables, --show-channel
                    List GOWIN USB download cables
 --scan             Scan and list GOWIN FPGA devices
 --filestransform <int>
                    1: Convert/Merge .fs to .bin;
                    2: Convert/Merge .bin(binary) to .hex(HEX);
                    3: Convert/Merge .bin(binary) to .h(hpp);
                    4: Convert .bin(binary) to .intelhex(Intel HEX);
                    5: Merge multiple ".fs" files to one ".fs"  ;
                    6: Append User Flash Init File(.fi) to a BitStreamFile(.bin);
                    7: Append a MCU FW File(.bin) to a BitStreamFile(.bin);
                    8: Append GW1NS4C M3 Core File(.bin) to a BitStreamFile(.fs);
 --files <string>   Used with the parameter --filestransform together, multiple files are separated by ",",
                    such as: file1.fs,file2.fs
```

# 4.2 Scan USB Cable Device

Scan and display the connected USB Cable information.

Use programmer_cli.exe --scan-cables

**Figure 4-3 Scan USB Cable Device**

```
C:\Gowin\Gowin_V1.9.8.06\Programmer\bin>programmer_cli --scan-cables
 Cable found:  Gowin USB Cable(FT2CH)/0/5922/GW-eHwVp (USB location:5922) (SN: GW-eHwVp)
 Cost 0.05 second(s)
```

- Gowin USB Cable(FT2CH) is the Cable type.
- /0/ is the channel number.
- 5922 is the USB location information.
- /GW-eHwVp is the USB Cable UID.
- (USB location:5922) (SN: GW-eHwVp) is the description information.

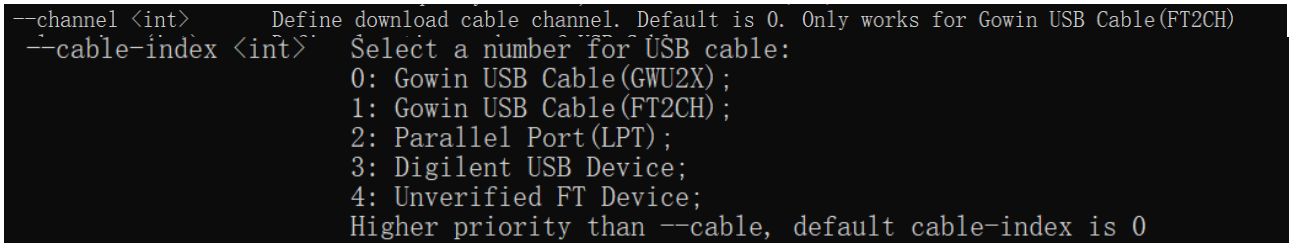
# 4.3 Specify USB Cable Type and Port

When scanning or configuring FPGA, you can specify the USB Cable type using the parameter "--cable-index". When --cable-index is 0, the cable type is Gowin USB Cable (GWU2X); when --cable-index is 1, the cable type is Gowin USB Cable (FT2CH), and so on; and --cable-index is 0 by default.

When using a multi-port cable, such as an FTDI cable, there will be multiple ports to choose. You can use the parameter "--channel" to specify

the port, and the value of -channel is 0 by default; the help information is as follows:

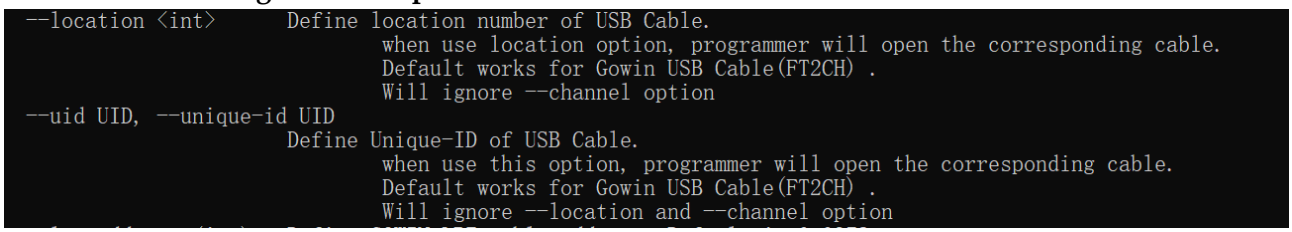
**Figure 4-4 Help Information**




# 4.4 Specify USB Cable Location or UID

When using multiple USB Cables, you can specify the USB port location or Cable UID to confirm a USB Cable device, and the help information is as follows:

**Figure 4-5 Help Information**

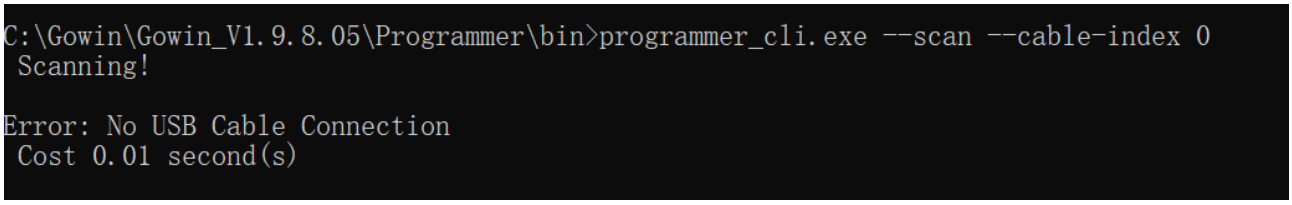



# 4.5 Scan FPGA Device

Scan the device with the following instruction.

programmer_cli.exe –scan

**Figure 4-6 Scan FPGA Device**




# 4.6 Specify Programmer Execution Mode

Use the parameter "--operation_index" or "--run" or "-r" to specify the execution mode, such as SRAM, Flash configuration, etc. The help information is as follows:

**Figure 4-7 Help Information**

```
--operation_index <int>, --run <int>, -r <int>
                      0: Read Device Codes;
                      1: Reprogram;
                      2: SRAM Program;
                      3: SRAM Read;
                      4: SRAM Program and Verify;
                      5: embFlash Erase,Program;
                      6: embFlash Erase,Program,Verify;
                      7: embFlash Erase Only;
                      8: exFlash Erase,Program;
                      9: exFlash Erase,Program,Verify;
                      10: exFlash Bulk Erase;
                      11: exFlash Verify;
                      12: exFlash Erase,Program in bscan;
                      13: exFlash Erase,Program,Verify in bscan;
                      14: exFlash Bulk Erase in bscan;
                      15: exFlash Verify in bscan;
                      16: SRAM Program JTAG 1149;
                      17: SRAM Program,Verify JTAG 1149;
                      18: bsdl read;
                      19: embFlash 2nd Erase,Program;
                      20: embFlash 2nd Erase,Program,Verify;
                      21: embFlash 2nd Erase Only;
                      22: -R-;
                      23: Connect to JTAG of MCU;
                      24: SRAM Erase;
                      25: Authentication Code Erase,Program,Verify;
                      26: Authentication Code Read;
                      27: Firmware Erase,Program Securely;
                      28: Firmware Erase Only;
                      29: Firmware Erase,Program;
                      30: Firmware Erase,Program,Verify;
                      31: exFlash C Bin Erase,Program;
                      32: exFlash C Bin Erase,Program,Verify;
                      33: -R-;
                      34: MFG Write iRef;
```

When configuring the FPGA, use the parameter "--device" to specify FPGA Device, and the help information is as follows:

**Figure 4-8 Help Information**

```
--device <GWxx-x>, -d <GWxx-x>
                      Define a GOWIN FPGA device from:
                      GW1N-1 GW1N-1P5 GW1N-1P5B GW1N-1P5C
                      GW1N-1S GW1N-2 GW1N-2B GW1N-2C
                      GW1N-4 GW1N-4B GW1N-4D GW1N-9
                      GW1N-9C GW1NR-1 GW1NR-2 GW1NR-2B
                      GW1NR-2C GW1NR-4 GW1NR-4B GW1NR-4D
                      GW1NR-9 GW1NR-9C GW1NRF-4B GW1NS-2
                      GW1NS-2C GW1NS-4 GW1NS-4C GW1NSE-2C
                      GW1NSER-4C GW1NSR-2 GW1NSR-2C GW1NSR-4
                      GW1NSR-4C GW1NZ-1 GW1NZ-1C GW2A-18
                      GW2A-18C GW2A-55 GW2A-55C GW2AN-18X
                      GW2AN-55C GW2AN-9X GW2ANR-18C GW2AR-18
                      GW2AR-18C
```

You can use the following command formats to configure SRAM or Flash:

programmer_cli.exe --device <GWxx-x> --run <int> --fsFile <bitstream.fs> --cable-index <int> --location <int> --uid <UID> --chain_index <int> --chain_size <int> --chain_ir <string> --frequency <string>

- --frequency specifies JTAG frequency, currently only applicable to FTDI Cable; U2X Cable frequency is fixed at 1.33MHz.
- --chain_index <int> --chain_size <int> --chain_ir <string> can be used in conjunction to specify the location of the target device in the daisy chain.
- --location specifies the USB port where the target device is located, and the priority is higher than the UID.
- --uid specifies the USB Cable used by the target device.
- --cable-index <int> specifies USB Cable type.
- --fsFile <bitstream.fs> specifies bitstream file path.
- --run <int> and --operation_index specify execution mode.
- --device <GWxx-x> specifies the target device name, case-sensitive.

# 4.7 Configure SRAM

Specify a bitstream file, corresponding device and SRAM configuration mode to configure FPGA SRAM, for example:

Configure SRAM Program; the "SRAM Program" value in the --operation_index parameter is 2, so the operation command is as follows:

programmer_cli.exe --device <GWxx-x> --run <int> --fsFile <bitstream.fs> --cable-index <int> --location <int>

Among them, --cable-index and --location have default values when they are specified, and they can be omitted.

An example is as follows:

programmer_cli.exe --device GW1N-4B --run 2 --fsFile d:\bitstream.fs --cable-index 0

The parameter meanings are as follows:

- --device GW1N-4B specifies the target FPGA,
- --fsFile d:\bitstream.fs specifies the bitstream file d:\bitstream.fs.
- --cable-index 0 specifies USB Cable as "Gowin USB Cable(GWU2X)".

# 4.8 Configure LittleBee® Family Embedded Flash

## 4.8.1 Configure Flash

The --operation_index label for programming Flash is as follows:

5: embFlash Erase,Program;

6: embFlash Erase,Program,Verify;

7: embFlash Erase Only;

An example is as follows:

programmer_cli –run 5 –fsFile d:\bitstream.fs –device GW1N-4B –cable-index 1

### 4.8.2 Configure Flash and UserFlash Initialization File

Use the parameter "--fiFile userflash.fi” to specify UserFlash initialization file, and configure UserFlash at the same time when programming Flash, for example:

programmer_cli --run 5 --fsFile d:\bitstream.fs --device GW1N-4B --cable-index 1

## 4.9 Configure External SPI Flash

The --operation_index label for programming external SPI Flash is as follows, and it is recommended to use "thru GAO-Bridge" type.

8 exFlash Erase,Program;

9: exFlash Erase,Program,Verify;

10: exFlash Bulk Erase;

11: exFlash Verify;

12: exFlash Erase,Program in bscan;

13: exFlash Erase,Program,Verify in bscan;

14: exFlash Bulk Erase in bscan;

15: exFlash Verify in bscan;

36: exFlash Erase,Program thru GAO-Bridge;

37: exFlash Erase,Program,Verify thru GAO-Bridge;

38: exFlash C Bin Erase,Program thru GAO-Bridge;

39: exFlash C Bin Erase,Program,Verify thru GAO-Bridge;

An example is as follows:

programmer_cli --run 36 --fsFile d:\bitstream.fs --device GW1N-4B --cable-index 1

**Note!**

For more information, please see programmer_cli –help.


GOWIN
PROGRAMMING FOR THE FUTURE