1.Programmer Overview

1.1 Programmer hardware parameters

It is a professional programmer with high cost-effective, made by high-density SMD process, unified operation interface, easy to use, complete function, reliable operation of application software program, super small code, faster execution speed, support multi-language menu. Supported operating systems: WINXP/WIN7/WIN8/WIN10 operating system.

1.1.1 **XGecu T56 Parameters**

Note: Be sure to use the original USB cable, if you use HUB, it must be a high quality HUB

- PIC32MZ 32-bit MCU 200MHZ + Xilinx SP6 FPGA structure, USB2.0 HS 480MHZ transmission
- Volume: 12.9cm x 8.5cm x 2.9cm
- 56-pins powerful pins drivers IO design. Maximum operating frequency of 50MHZ for all pins, extremely high reliability.
- Stable and reliable 8P in-circuit programmable ISP interface with wire length of more than 40CM, reliable communication at 25-45MHZ.
- The highest clock frequency can reach up to 40MHZ-50MHZ for 25 series/EMMC in-circuit programming.
- VCC voltage 1.2-6.5V, 64 levels adjustable, VPP voltage 6.5-25V 64 levels adjustable.IO voltage 1.2V-3.6V continuously adjustable. VCC 120MA-320MA multilevel over current protection continuously adjustable. VPP 120MA overcurrent protection.
- Power consumption: 5V <500MA. Static power consumption: 180mA , All chips do not need an external power supply.
- Allow to use an external power supply 5V-5.5V 500MA.
- The programmer has multiple protection for external power supply: over-voltage protection/reverse connection protection/over-current protection/under-voltage prompt.
- Maximum read-write speed can reach up to 25-45MB/s
- Open self-test function, self-test on each pin of VPP/VCC/GND/IO
- Suitable for developers and maintenance personnel, suitable for small batch production, support simultaneously working with 4 Programmers on each computer

1.1.2 **T866II Plus Parameters**

- PIC24 16-bit MCU 32MHZ, USB1.1 FS 12MHZ transmission
- Volume: 10cm x 6cm x 2.5cm
- 40-pin drive IO design. Non-full drive, IO voltage is not adjustable.
- VCC voltage 1.8-6.5V 32 levels adjustable, VPP voltage 9V-18V 32 levels adjustable. VCC 120MA over current protection, VPP 100MA over current protection.
- Power consumption: 5V <500MA. Static power consumption: 60mA. Without external power supply interface.
- Self-test function, self-test each pin of VPP/VCC/GND/IO
- Suitable for developers and maintenance personnel, can do small batch production for small capacity chips, support simultaneously working with 4 Programmers on each computer.
-

1.2 Programmer support chips and functions

1.2.1 XGecu T56

- 26 27 28 29 37 39 49 50 series parallel ROM, EPROM, EEPROM, Paged EPROM, Flash Maximum capacity 2GBits.
- 24 25 35 45 85 93 95 Serial Series EEPROM, Support 25 FLASH chips: Maximum capacity 2GBits
- Support NAND FLASH: Maximum capacity 256Gbits, NAND can be customized parameters to add new model chips.
- Support SPI NAND: Maximum capacity 8Gbits
- Support EMMC/EMCP: Maximum capacity 256GBits, EMMC/EMCP support one-button cloning, support ISP programming
- Microcontroller Series
- GAL/CPLD programmable logic device programming
- Support VGA tool and VGA ISP programming, Can output VGA/HDMI video test signal, read TV or monitor parameters, and ISP in-circuit programming
- Super performance 54/74F/LS/HC CMOS4000 series integrated circuit logic test, Can test common logic integrated circuit, result can be located to the logic gate circuit, customizable logic vector table, can add new chips to support list.
- Support more packages of DIP42/PLCC44/SOP44/QFP44/TSOP48 on base of TL866
- Support TSOP56/BGA48/BGA63/BGA64/BGA100/BGA153/BGA162/BGA169/BGA221 etc.

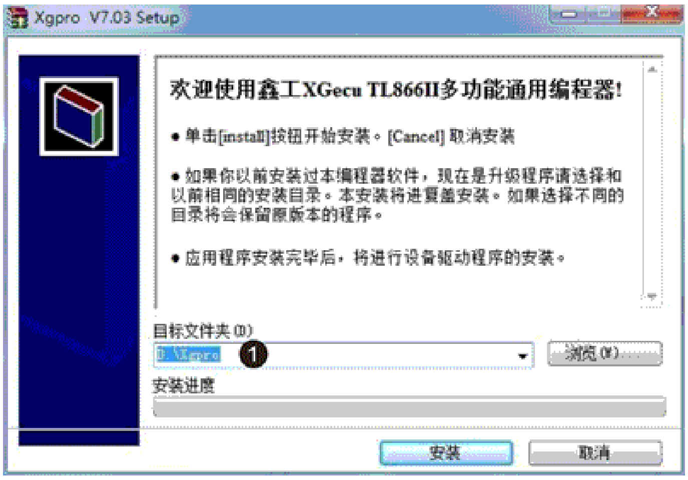
1.2.2 TL866II Plus

- 26 27 28 29 37 39 49 50 Series Parallel ROM, EPROM, EEPROM, Paged EPROM, Flash Maximum capacity 64MBits.
- 24 25 35 45 85 93 95 Serial Series EEPROM. Support 25FLASH chips maximum capacity 512MBits
- NAND FLASH TSOP48 chips support up to 8Gbits, NAND can be customized parameters to add new model chips.
- Microcontroller Series
- GAL programmable logic device programming
- SRAM test, support common 24 61 62 DS12 series SRAM test, support data line test, address line test, memory cell incremental test and cell test.
- Super performance 54/74F/LS/HC CMOS4000 series integrated circuits logic test.
- Can test common logic integrated circuits, test can be located to the logic gate circuit, can customize the logic vector table.

2. Software Installation

2.1 Download and install the application software

Download the XgproVxxxx_setup.exe file from HTTP://www.xgecu.com, and double-click the file to enter the installation interface. As follows: (The installation file contains the contents of this manual.) Click [Install] to start installing the application, and the following screen will pop up: Enter the folder where you want to install. Click [Install] to start the installation of the application



Xgpro V7.03 Setup

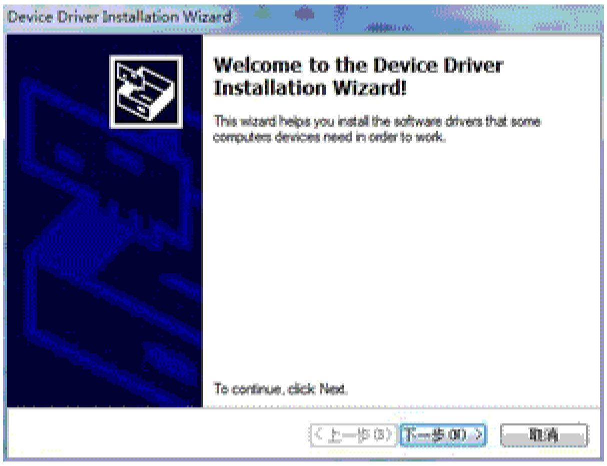
欢迎使用鑫工XGecu TL866II多功能通用编程器!

- 单击[install]按钮开始安装。[Cancel] 取消安装
- 如果你以前安装过本编程器软件，现在是升级程序请选择和以前相同的安装目录。本安装将进覆盖安装。如果选择不同的目录将会保留原版本的程序。
- 应用程序安装完毕后，将进行设备驱动程序的安装。

目标文件夹 (D)

D:\Xgpro

Click [Install], after the file is installed to the hard disk. The USB driver installation dialog box is automatically performed. As below:

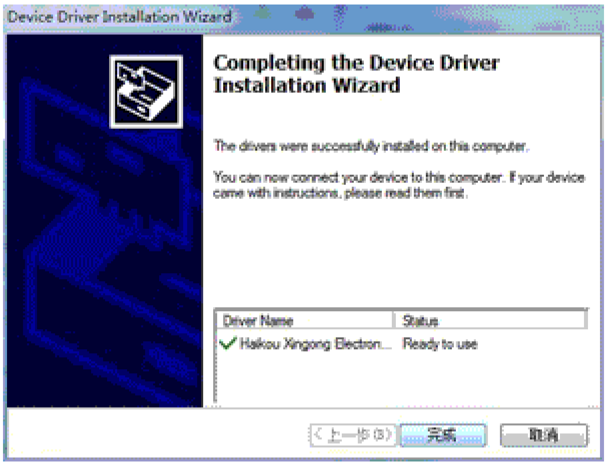


Device Driver Installation Wizard

Welcome to the Device Driver Installation Wizard!

This wizard helps you install the software drivers that some computers devices need in order to work.

To continue, click Next.



Device Driver Installation Wizard

Completing the Device Driver Installation Wizard

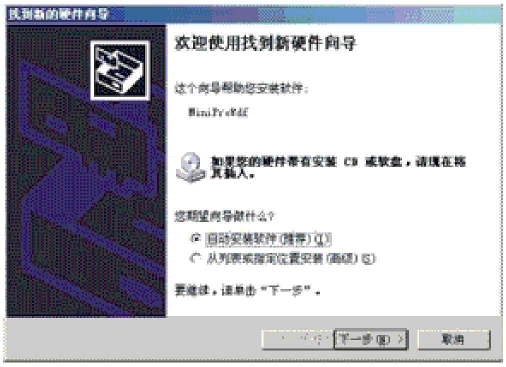
The drivers were successfully installed on this computer.

You can now connect your device to this computer. If your device came with instructions, please read them first.

Driver Name	Status
✓ Haikou Xingong Electron...	Ready to use

2.2 Device Installation

After the above installation is completed, connect programmer into the USB port and the operating system will discover the hardware as below:



找到新的硬件向导

欢迎使用找到新硬件向导

这个向导帮助您安装软件：

MiniPro[illegible]

如果您的硬件带有安装 CD 或软盘，请现在将其插入。

您期望向导做什么？

● 自动安装软件（推荐）(I)
○ 从列表或指定位置安装（高级）(S)

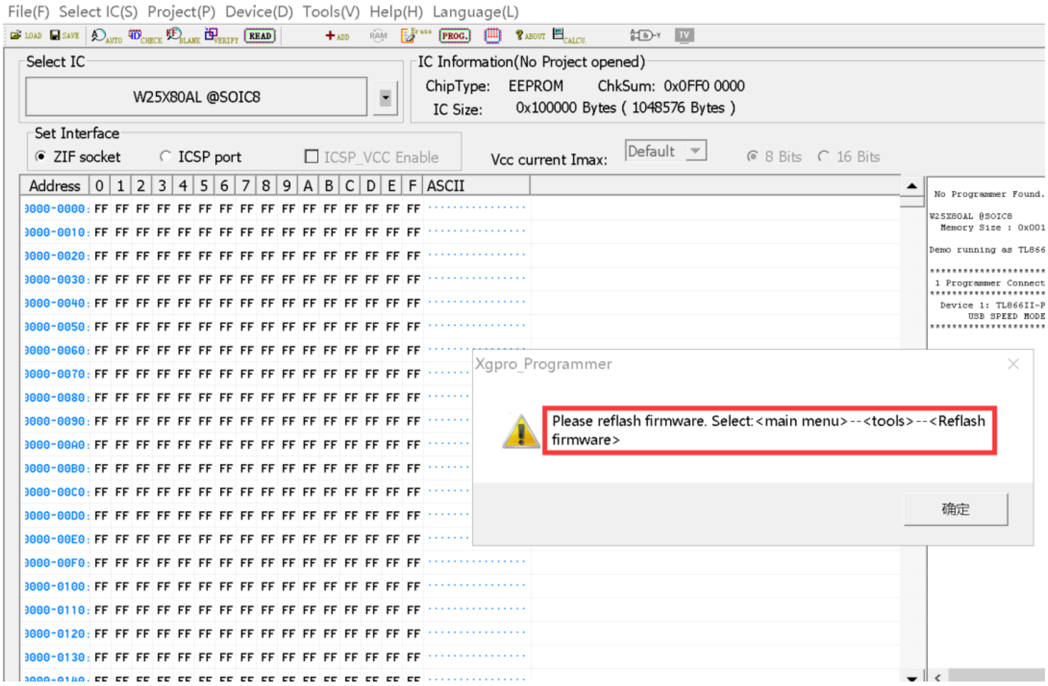
要继续，请单击“下一步”。

Select [Auto Install Software], click [**next**] button, it will automatically install until the installation is successfully completed. (Note: In this process, the driver will change the registry. If the firewall software is installed, please release this software. Windows will also prompt the driver have no digital visa, confirm and continue the installation).

If the automatic installation is not successful, you can also choose to install in <from list or specified location>, USB driver files are in the installation directory (D:\xgpro\drv). After installation, double-click the desktop icon Xgpro Program... and run the programmer software. Ready to use.

2.3 Firmware FLASH Refresh

When using at the first time or upgrading, you may need to refresh the firmware. In the main menu of the software, do a refresh operation as shown below (**Tools**-**Reflash firmware**):



File(F) Select IC(S) Project(P) Device(D) Tools(V) Help(H) Language(L)
LOAD SAVE AUTO CHECK BLANK VERIFY READ ADD RAM Erase PROG. ABOUT CALCU. TV

Select IC: W25X80AL @SOIC8

IC Information(No Project opened)
ChipType: EEPROM ChkSum: 0x0FF0 0000
IC Size: 0x100000 Bytes (1048576 Bytes)

Set Interface: ● ZIF socket ○ ICSP port □ ICSP_VCC Enable

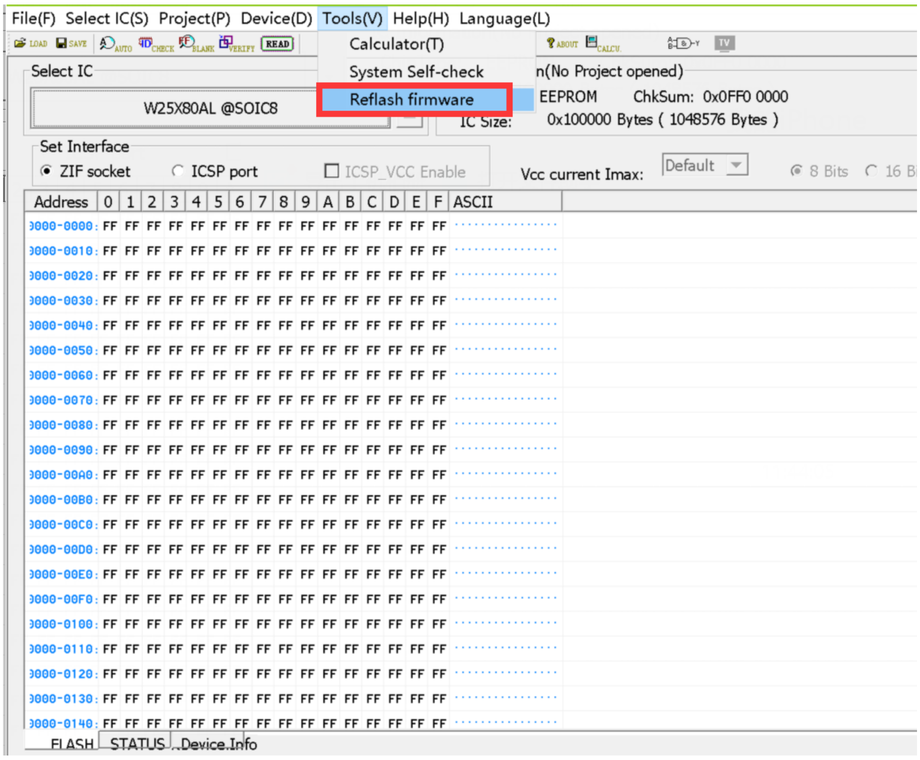
Vcc current Imax: Default ● 8 Bits ○ 16 Bits

```
Address   0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  ASCII
0000-0000: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0010: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0020: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0030: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0040: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0050: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0060: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0070: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF .......
0000-0080: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF .......
0000-0090: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF .......
0000-00A0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF .......
0000-00B0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF .......
0000-00C0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF .......
0000-00D0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF .......
0000-00E0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF .......
0000-00F0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0100: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0110: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0120: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0130: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
```

```
 No Programmer Found.

W25X80AL @SOIC8
  Memory Size : 0x001

Demo running as TL866

*********************
 1 Programmer Connect
*********************
  Device 1: TL866II-P
       USB SPEED MODE
*********************
```

Xgpro_Programmer

Please reflash firmware. Select:<main menu>--<tools>--<Reflash firmware>


File(F) Select IC(S) Project(P) Device(D) Tools(V) Help(H) Language(L)
LOAD SAVE AUTO CHECK BLANK VERIFY READ
ABOUT CALCU.
Calculator(T)
System Self-check
Reflash firmware
FLASH STATUS Device.Info

Select IC
W25X80AL @SOIC8

n(No Project opened)
EEPROM ChkSum: 0x0FF0 0000
IC Size: 0x100000 Bytes (1048576 Bytes)

Set Interface
● ZIF socket ○ ICSP port □ ICSP_VCC Enable
Vcc current Imax: Default
● 8 Bits ○ 16 B

```
Address  0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  ASCII
0000-0000: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0010: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0020: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0030: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0040: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0050: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0060: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0070: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0080: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0090: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-00A0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-00B0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-00C0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-00D0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-00E0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-00F0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0100: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0110: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0120: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0130: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0140: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
```

If there is an error during firmware refresh, it is often because the programmer cannot be reset during the refresh and the programmer cannot do the refresh operation.

The reason of error:

It is usually caused by using an external USB hub. Please do not use an external HUB when refreshing. Please directly connect the programmer to the USB port of your computer to refresh. If it still cannot solve your problem, please try to restart your computer or do the refresh operation on another computer.

3. Quick Start

3.1 Hardware preparation and select the correct chip model

Firstly, please make sure the universal programmer application software and USB driver are correctly installed. Then connect programmer to your PC by using USB cable. At this point, the “POW” power indicator of the programmer keeps on and the “RUN” indicator goes off, that means the power connection is normal and you can go to the next step.

(If the RUN indicator blinks, that means the USB driver is installed incorrectly or don't install the USB device driver)

Note: Please be sure to use the original USB cable. If you use HUB, make sure it is the high quality HUB and the USB voltage should not be lower than 4.90V. Low voltage is usually caused by using poor quality USB cable or HUB, that will make the programmer work unstably.

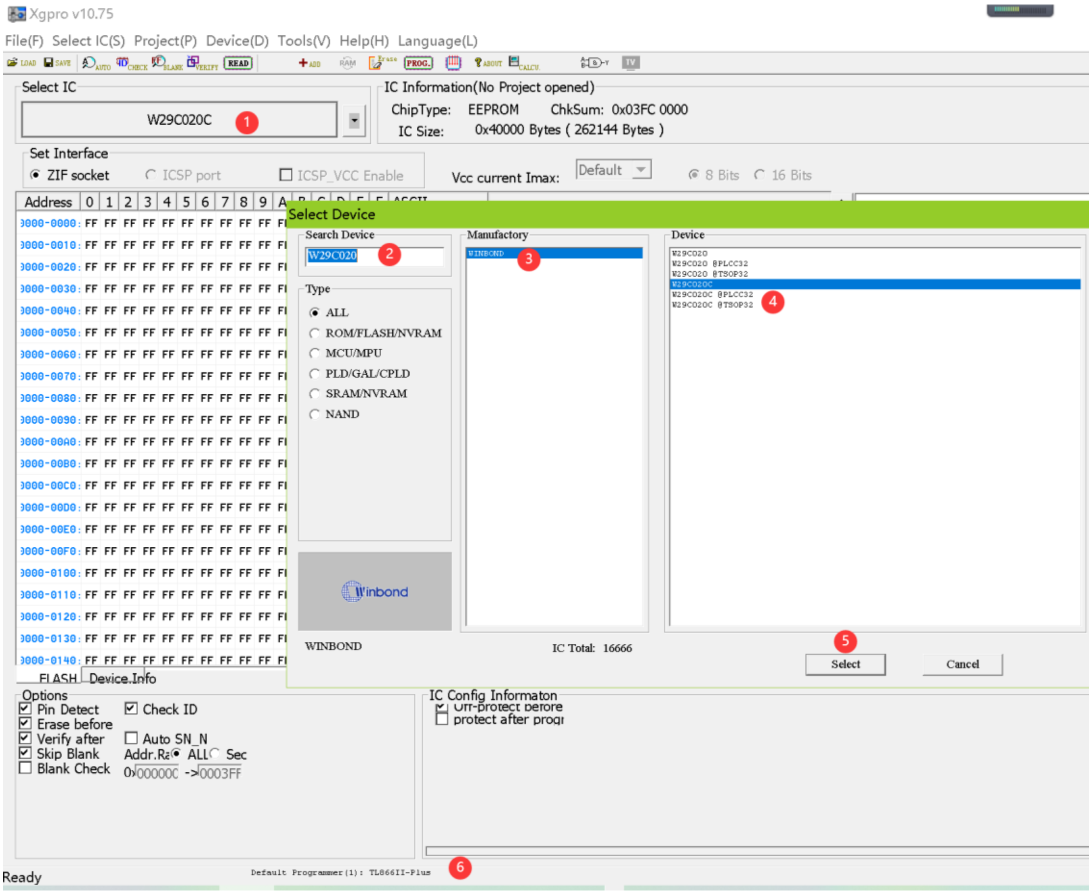
As shown below:

1) Click the **[Select IC]** button, the select Device dialog box pops up.
2) In the search IC, type in the IC model (for example W29C020C)
3) Select the correct manufactory (if there are more than one)

4) Select the correct model. Note: @PLCC32 @TSOP32 refer to the IC package, none: dual-in-line package

5) Click **[Select]** button

6) Note: Show that the programmer is connected


Xgpro v10.75
File(F) Select IC(S) Project(P) Device(D) Tools(V) Help(H) Language(L)
LOAD SAVE AUTO CHECK BLANK VERIFY READ ADD RAM Erase PROG. ABOUT CALCU. TV
1
2
3
4
5
6
Winbond
Select
Cancel
FLASH
Device.Info

Select IC: W29C020C

IC Information(No Project opened)
ChipType: EEPROM ChkSum: 0x03FC 0000
IC Size: 0x40000 Bytes (262144 Bytes)

Set Interface
● ZIF socket ○ ICSP port □ ICSP_VCC Enable Vcc current Imax: Default ● 8 Bits ○ 16 Bits

```
Address  0  1  2  3  4  5  6  7  8  9  A
3000-0000: FF FF FF FF FF FF FF FF FF FF FI
3000-0010: FF FF FF FF FF FF FF FF FF FF FI
3000-0020: FF FF FF FF FF FF FF FF FF FF FI
3000-0030: FF FF FF FF FF FF FF FF FF FF FI
3000-0040: FF FF FF FF FF FF FF FF FF FF FI
3000-0050: FF FF FF FF FF FF FF FF FF FF FI
3000-0060: FF FF FF FF FF FF FF FF FF FF FI
3000-0070: FF FF FF FF FF FF FF FF FF FF FI
3000-0080: FF FF FF FF FF FF FF FF FF FF FI
3000-0090: FF FF FF FF FF FF FF FF FF FF FI
3000-00A0: FF FF FF FF FF FF FF FF FF FF FI
3000-00B0: FF FF FF FF FF FF FF FF FF FF FI
3000-00C0: FF FF FF FF FF FF FF FF FF FF FI
3000-00D0: FF FF FF FF FF FF FF FF FF FF FI
3000-00E0: FF FF FF FF FF FF FF FF FF FF FI
3000-00F0: FF FF FF FF FF FF FF FF FF FF FI
3000-0100: FF FF FF FF FF FF FF FF FF FF FI
3000-0110: FF FF FF FF FF FF FF FF FF FF FI
3000-0120: FF FF FF FF FF FF FF FF FF FF FI
3000-0130: FF FF FF FF FF FF FF FF FF FF FI
3000-0140: FF FF FF FF FF FF FF FF FF FF FI
```

Select Device

Search Device: W29C020

Type
● ALL
○ ROM/FLASH/NVRAM
○ MCU/MPU
○ PLD/GAL/CPLD
○ SRAM/NVRAM
○ NAND

Manufactory: WINBOND

Device:
W29C020
W29C020 @PLCC32
W29C020 @TSOP32
W29C020C
W29C020C @PLCC32
W29C020C @TSOP32

WINBOND IC Total: 16666

Options
☑ Pin Detect ☑ Check ID
☑ Erase before
☑ Verify after □ Auto SN_N
☑ Skip Blank Addr.Ra ● ALL ○ Sec
□ Blank Check 0x000000 ->0003FF

IC Config Informaton
☑ Off-protect before
□ protect after progr

Ready Default Programmer(1): TL866II-Plus

3.2 Read and Save

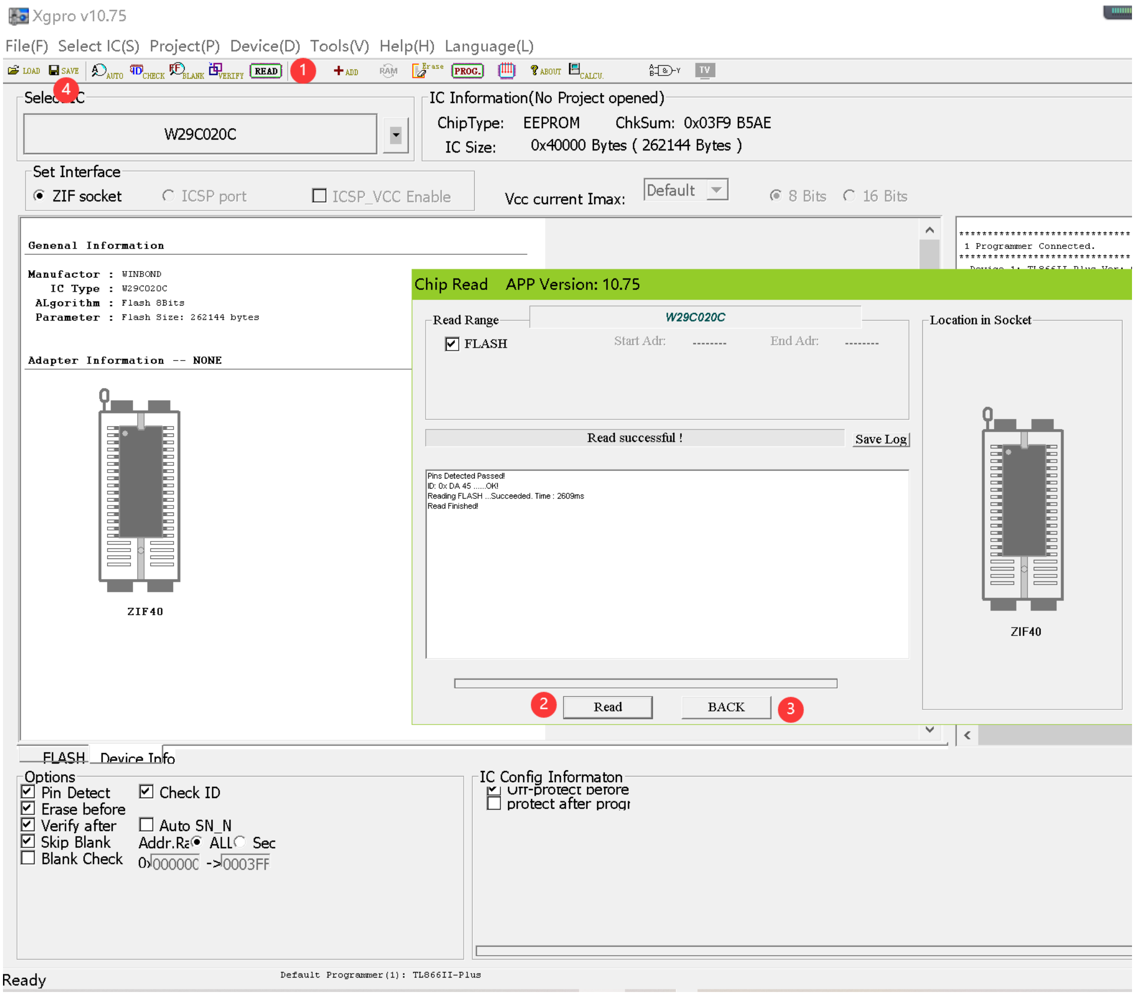
After selecting IC in section 3.1 steps

1) Click the **[Read]** button on the toolbar and pop up the Chip Read dialog box.

2) In the Chip Read dialog box, click the **[Read]** button.

3) Read finished and then click the **[Back]** button.

4) Click **[Save]** button on the toolbar (you can also select "File-Save" in the main menu), type in the file name to save the file.



Select IC: W29C020C

IC Information(No Project opened)
ChipType: EEPROM ChkSum: 0x03F9 B5AE
IC Size: 0x40000 Bytes (262144 Bytes)

Set Interface: ● ZIF socket ○ ICSP port □ ICSP_VCC Enable
Vcc current Imax: Default ● 8 Bits ○ 16 Bits

```
Genenal Information

Manufactor : WINBOND
   IC Type : W29C020C
 ALgorithm : Flash 8Bits
 Parameter : Flash Size: 262144 bytes

Adapter Information -- NONE
```

```
******************************
 1 Programmer Connected.
******************************
```

Chip Read APP Version: 10.75

Read Range: W29C020C
☑ FLASH Start Adr: -------- End Adr: --------

Location in Socket

Read successful !

Save Log

```
Pins Detected Passed!
ID: 0x DA 45 ......OK!
Reading FLASH ...Succeeded. Time : 2609ms
Read Finished!
```

Options
☑ Pin Detect ☑ Check ID
☑ Erase before
☑ Verify after □ Auto SN_N
☑ Skip Blank Addr.Ra ● ALL ○ Sec
□ Blank Check 0x 000000 -> 0003FF

IC Config Informaton
☑ Off-protect before
□ protect after progr

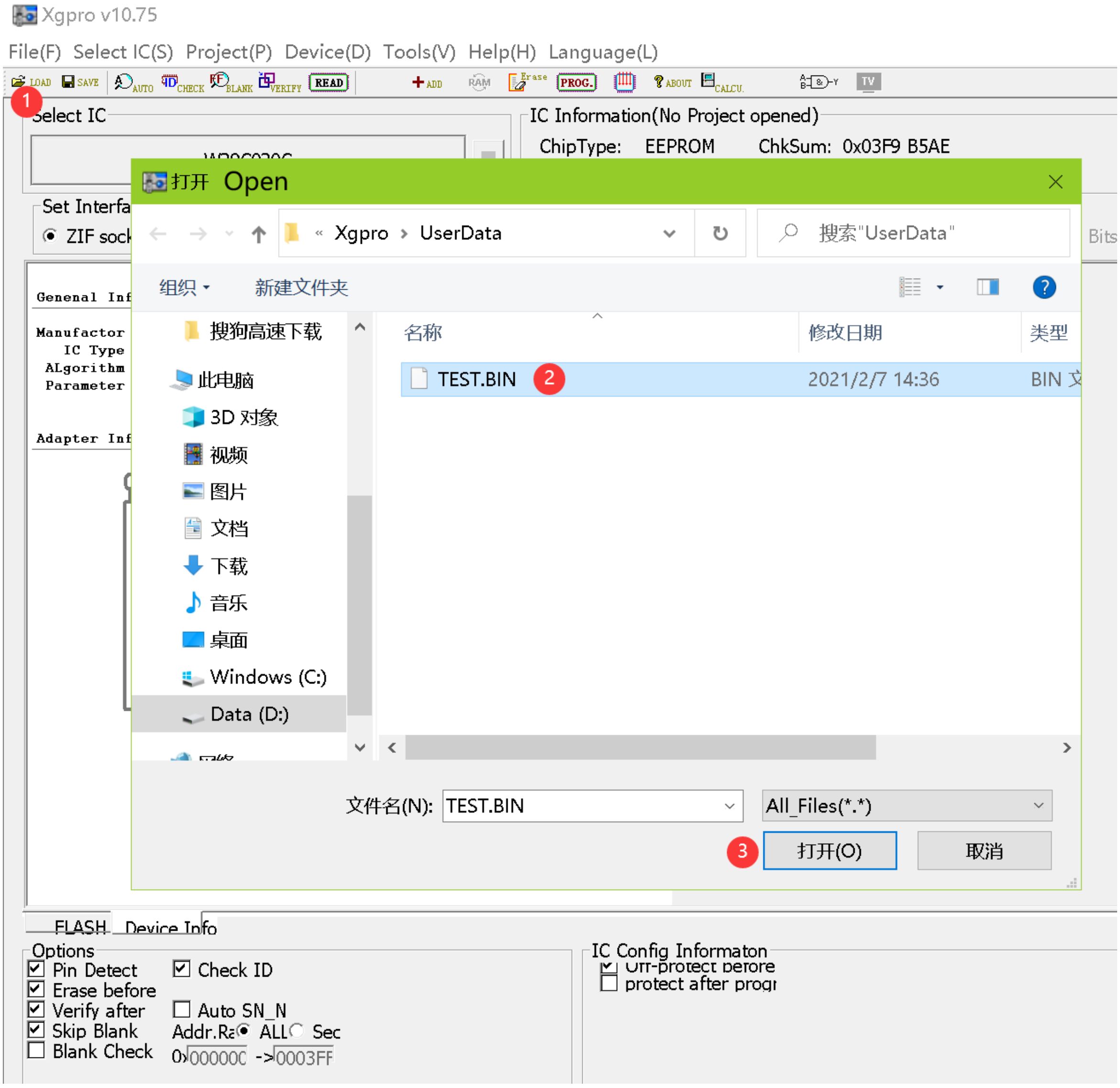
Ready

Default Programmer(1): TL866II-Plus

3.3 Programing a new chip

After completing the steps in section 3.2

1) Click **[Load]** button to bring up the Open File dialog box
2) Select the data file to be written (Test.BIN)
3) Click **[Open]**, the file load options dialog box pops up


Xgpro v10.75
File(F) Select IC(S) Project(P) Device(D) Tools(V) Help(H) Language(L)
LOAD SAVE AUTO CHECK BLANK VERIFY READ ADD RAM Erase PROG. ABOUT CALCU. TV
1
2
3
打开 Open
组织 新建文件夹
搜狗高速下载
此电脑
3D 对象
视频
图片
文档
下载
音乐
桌面
Windows (C:)
Data (D:)
打开(O)
取消
Bits
FLASH Device Info

Select IC

W29C020C

IC Information(No Project opened)

ChipType: EEPROM ChkSum: 0x03F9 B5AE

Set Interfa

● ZIF sock

Genenal Inf

Manufactor
IC Type
ALgorithm
Parameter

Adapter Inf

Xgpro › UserData

搜索"UserData"

名称	修改日期	类型
TEST.BIN	2021/2/7 14:36	BIN 文

文件名(N): TEST.BIN

All_Files(*.*)

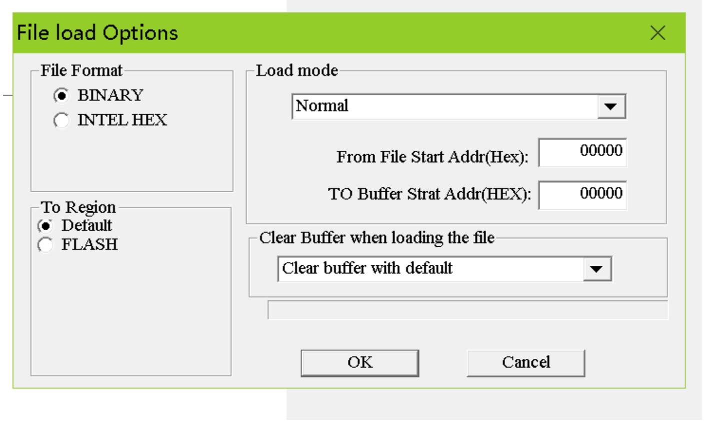
Options

☑ Pin Detect ☑ Check ID
☑ Erase before
☑ Verify after ☐ Auto SN_N
☑ Skip Blank Addr.Ra ● ALL ○ Sec
☐ Blank Check 0x 000000 -> 0003FF

IC Config Informaton

☑ Off-protect before
☐ protect after progr

4) According to the default settings, click **[OK]**, at this time the file content has been loaded to the computer memory buffer.



File Format
● BINARY
○ INTEL HEX

To Region
● Default
○ FLASH

Load mode
Normal

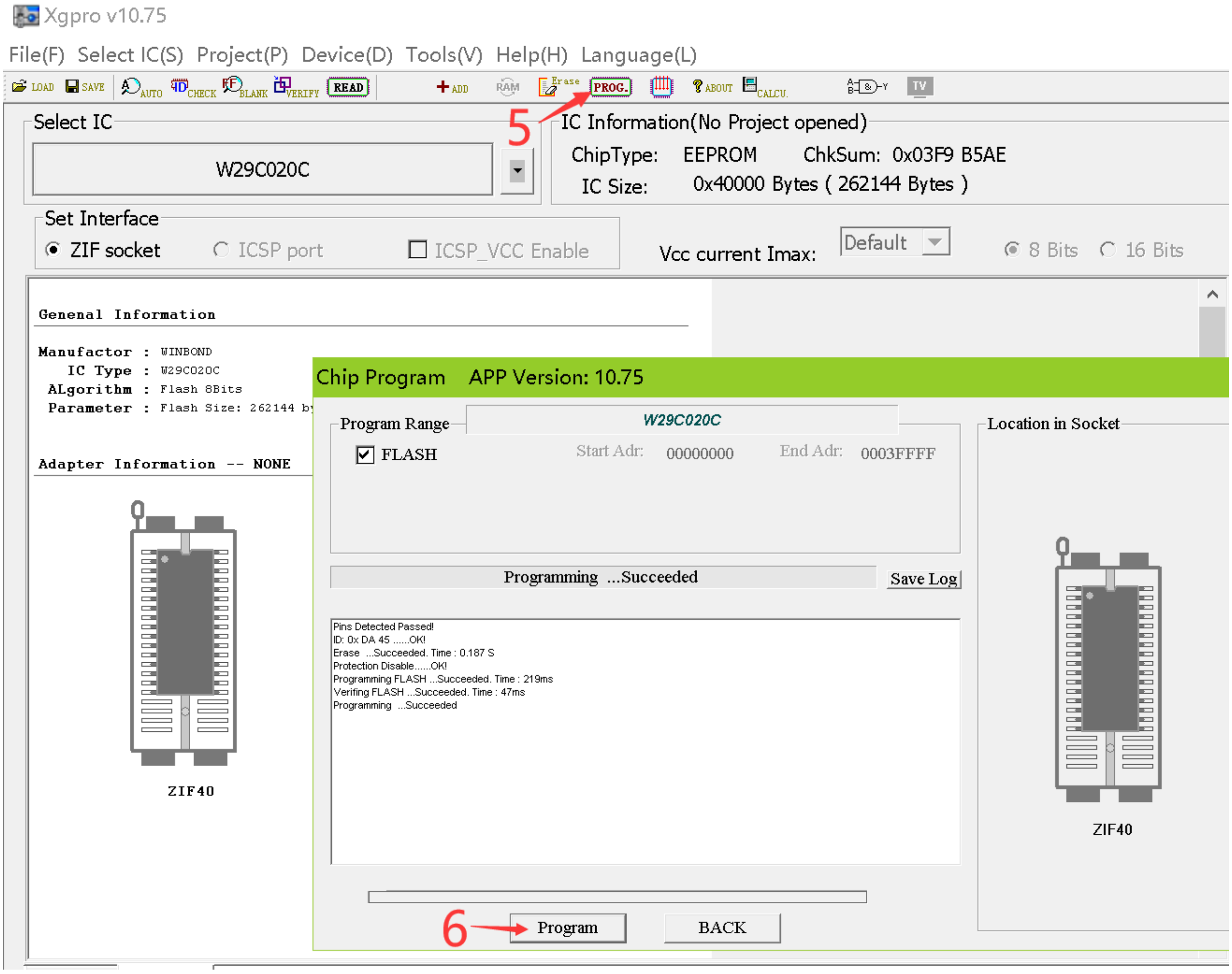
From File Start Addr(Hex): 00000

TO Buffer Strat Addr(HEX): 00000

Clear Buffer when loading the file
Clear buffer with default

5) Click **[PROG]** button, the chip programming dialog box pops up.

6) Click **[Program]** to get started to program the chip. If no error message appears, that means the programming succeeded.



Select IC: W29C020C

IC Information(No Project opened)
ChipType: EEPROM ChkSum: 0x03F9 B5AE
IC Size: 0x40000 Bytes (262144 Bytes)

Set Interface
● ZIF socket ○ ICSP port □ ICSP_VCC Enable
Vcc current Imax: Default
● 8 Bits ○ 16 Bits

```
Genenal Information

Manufactor : WINBOND
   IC Type : W29C020C
 ALgorithm : Flash 8Bits
 Parameter : Flash Size: 262144 by

Adapter Information -- NONE
```

Chip Program APP Version: 10.75

Program Range W29C020C
☑ FLASH Start Adr: 00000000 End Adr: 0003FFFF

Location in Socket

Programming ...Succeeded

```
Pins Detected Passed!
ID: 0x DA 45 ......OK!
Erase  ...Succeeded. Time : 0.187 S
Protection Disable......OK!
Programming FLASH ...Succeeded. Time : 219ms
Verifing FLASH ...Succeeded. Time : 47ms
Programming  ...Succeeded
```

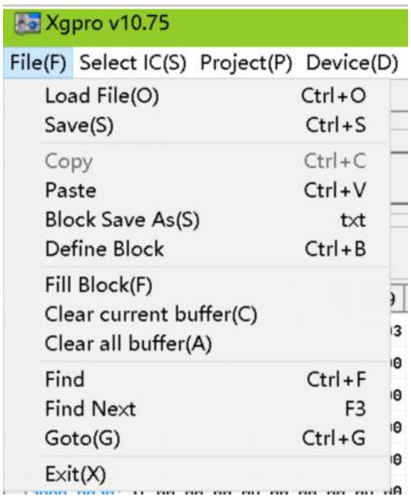
Note: If you need to program more chips, replace the new chip and click [Program] button again to begin a new programming.When programming in batch, you can press **[space bar]** instead of mouse click.

4. Programmer Function Details

4.1 Main Menu Summary

Main Menu Summary	
File	Load File, Save, Find, Goto, Buffer operation, Exit
Select IC	Search and Select IC, 25 Flash Detect, Add IC by user
Project	Open Project, Save Project, Save Project As, Close Project, Project Attrib, Modify Password
Device	Read, Read ID, Verify chip, Verify, Program, Erase chip, Blank Check, Serials number, Testing, Multi Prog., NAND bad block check, logic IC test, TV/LCD Tools
Tools	Calculator, System self-check, Reflash firmware
Help	Help, About, Update On Line
Language	switch languages, English, Chinese, Russian, Polish, German, Spanish, Portuguese, Turkish, Czech
Right click menu Edit buffer	Find the corresponding hexadecimal value or ASCII character string in the buffer, data, copy, paste, fill in, partially save as TXT file

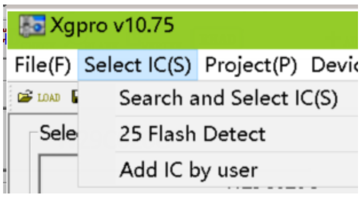
4.2 File Menu



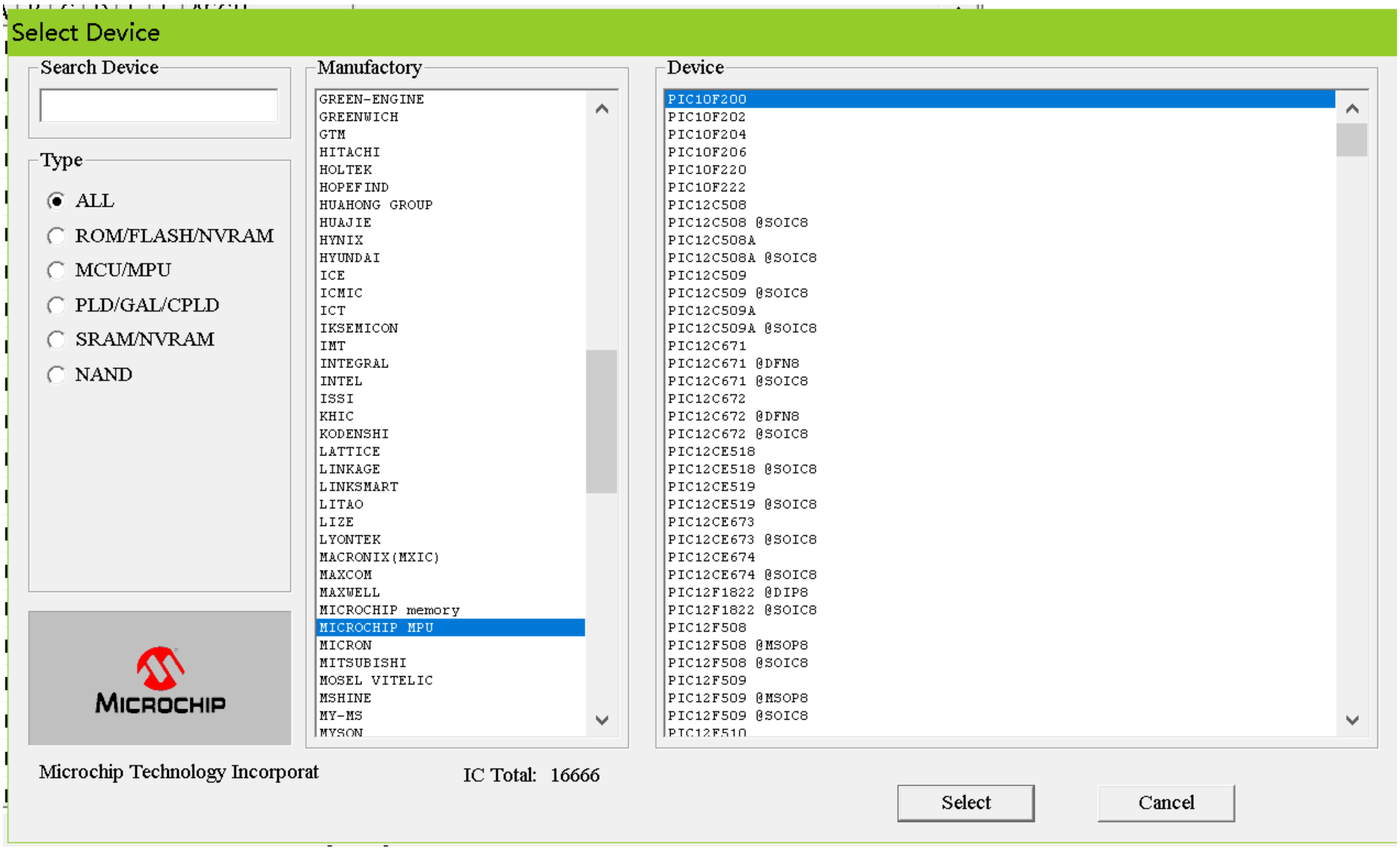
- **Load file**:
  Loads a hex file or a binary file into the specified buffer, the GAL chip loads the file in JED format.
- **Save file**:
  Save the current buffer to HEX file in hex format or BIN file in binary format. For GAL chips, should be saved to JED format file.
- **Note:** For MICROCHIP microcontrollers, when loading HEX files, the HEX files generated by the MICROCHIP IDE development environment are loaded into all buffers. When saving, all buffers can be saved to the HEX file, and it is fully compatible with MICROCHIP IDE development environment.
- Save and open for Bin file, the default is Flash Memory + EEPROM Memory (if present)
  For MICROCHIP microcontroller, Bin saved configuration information (arranged by Microchip address)
- Can use shortcut keys or right-click in the buffer to invoke the menu functions.
- To get more details on Edit Function-->Edit Buffer

4.3 Select IC

The menu is shown below:



4.3.1 Search and select IC

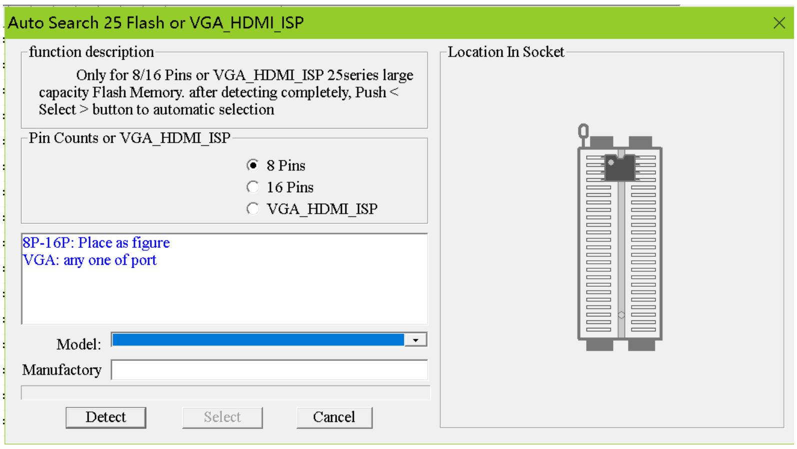


Microchip Technology Incorporat

IC Total: 16666

4.3.2 Auto Detect 25 Flash

25 Flash automatic detect, for 8-pin, 16-pin and VGA (T56) HDMI serial 25 series Nor Flash, note: many chip IDs are the same and results have more than one, so you need to select an accurate model in the results.

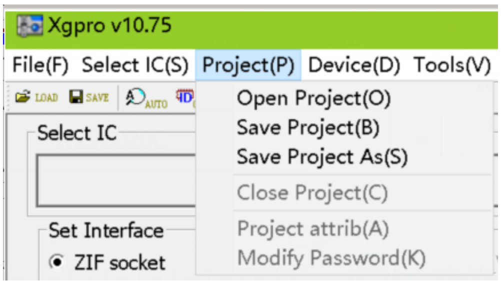


Only for 8/16 Pins or VGA_HDMI_ISP 25series large capacity Flash Memory. after detecting completely, Push < Select > button to automatic selection

8P-16P: Place as figure
VGA: any one of port

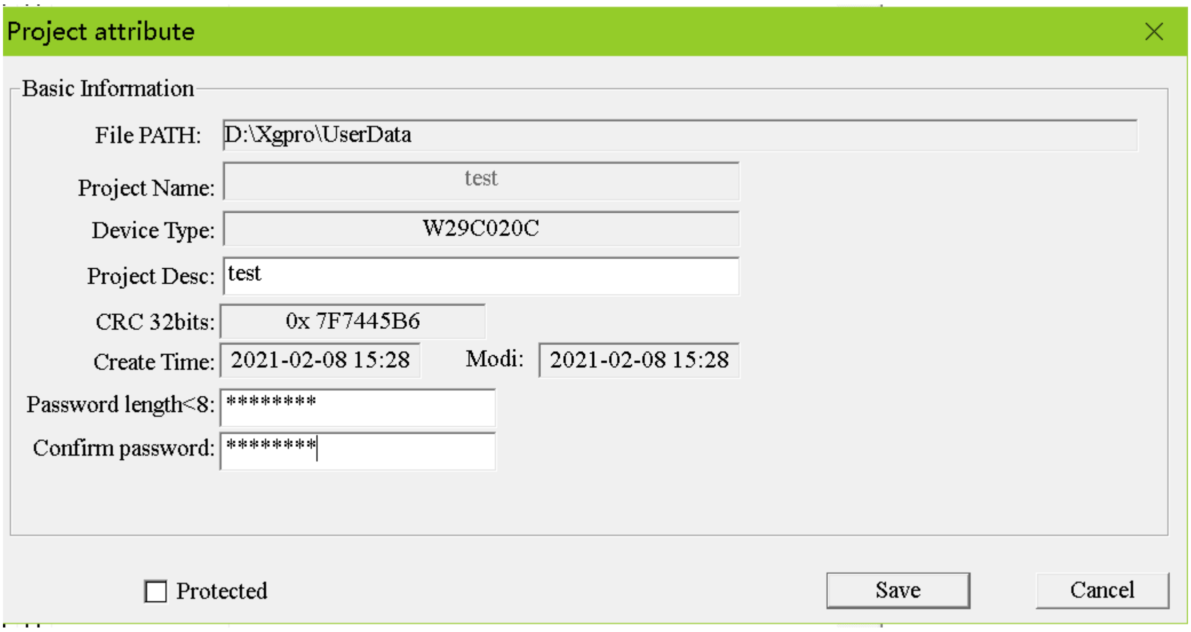
4.4 Project Menu

The project file is created for better batch programming. It is a file that saves all current buffer data and working environment settings, including information about the current device, settings of all operation options, and settings related to auto serials number. It is convenient to use in daily mass production. The project can be set up with password protection, which can protect the security of the product program to a certain extent. The menu is shown below:



Save Project

After setting the required parameters of the chip in the application, select **[Project]** - **[Save-Project]**, that is, a project is created. And the project information is displayed in the information area of the window. Click **[Save Project]**, type in the project name and click OK, then the following project dialog box pops up.


Project attribute

Basic Information

File PATH: D:\Xgpro\UserData

Project Name: test

Device Type: W29C020C

Project Desc: test

CRC 32bits: 0x 7F7445B6

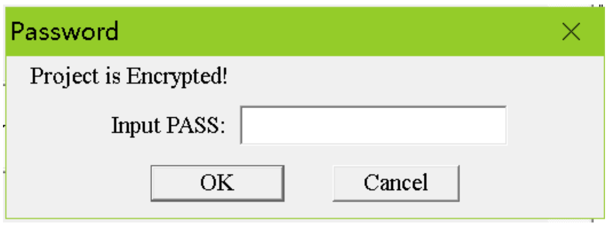
Create Time: 2021-02-08 15:28 Modi: 2021-02-08 15:28

Password length<8: ********

Confirm password: ********

□ Protected

1) Password can be empty, that means the current project is not set with password protection.
2) If the project is set to protection mode, the password must be input. When the option "Protected" is checked, the relevant data of the project will not be changed. Some functions of the programmer are disabled.


Password

Project is Encrypted!

Input PASS:

Open project

Open a current project, select [Project] - [Open Project], select the project name in the window and click OK. If the project has been set password, the following dialog box will pop up. (If no password is set, the project will be opened directly)

Save project As

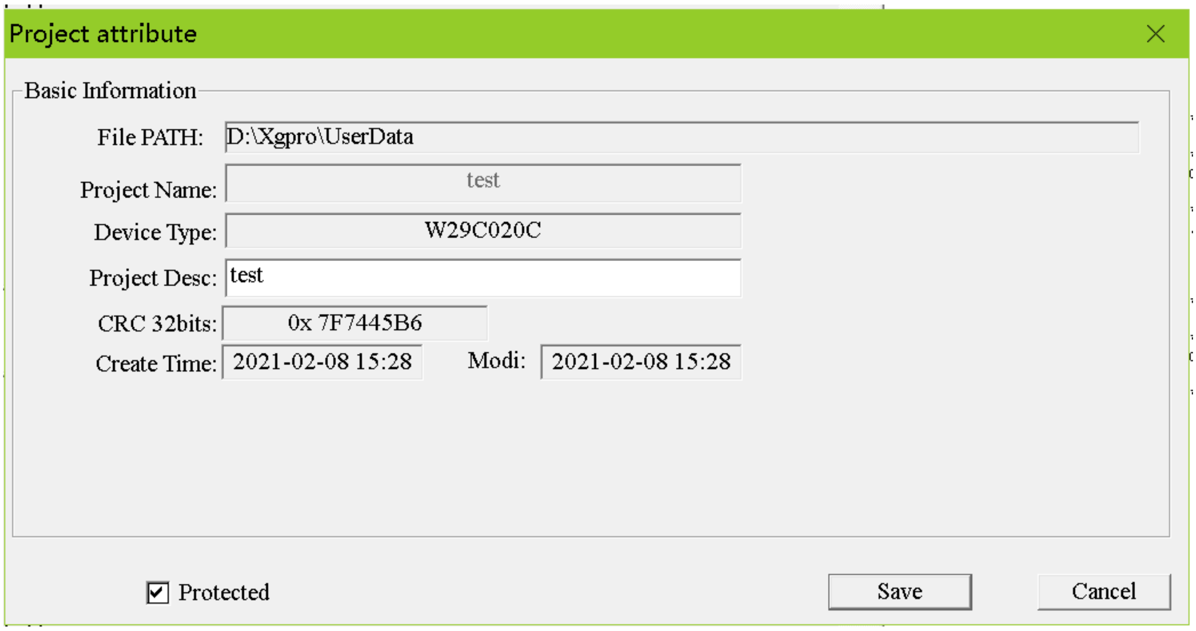
Save the current project as another project name. Refer to the operation of [Save Project].

Close Project

Close the current project. Switch to normal programming mode, close project and clear data in the current buffer.

Project Attrib.

Select [Project] - [Project Attrib] to bring up the project attribute dialog box. The project attribute dialog box can change the project description and project protection mode. You need to input the project password when saving.



Project attribute

Basic Information

File PATH: D:\Xgpro\UserData

Project Name: test

Device Type: W29C020C

Project Desc: test

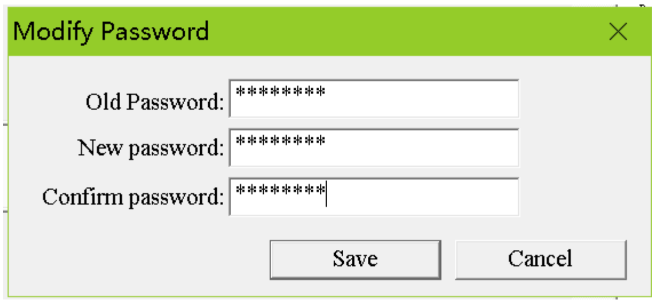
CRC 32bits: 0x 7F7445B6

Create Time: 2021-02-08 15:28 Modi: 2021-02-08 15:28

☑ Protected

Modify Password

Reset project password



Modify Password

Old Password: *********

New password: *********

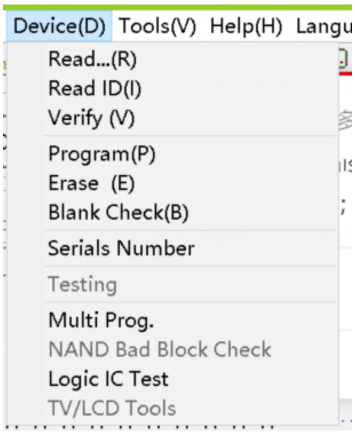
Confirm password: *********

Tips:

1) Accurate engineering descriptions can distinguish from a large number of projects. Engineering models are generally used for mass production.
2) The project file has been encrypted to a certain extent, so the project password can be protected to prevent the project file from being copied. To a certain extent, it can protect the data.
3) When saving or opening project files, all contain 32-bit CRC check information, which avoid corrupting storage files or other uncertainties.

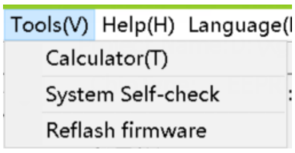
4.5 Device Menu

This menu function is the programming operation of the chips, please see section 4.10 Programmer operation function for details The menu is shown below:



4.6 Tools Menu

The menu is shown below:



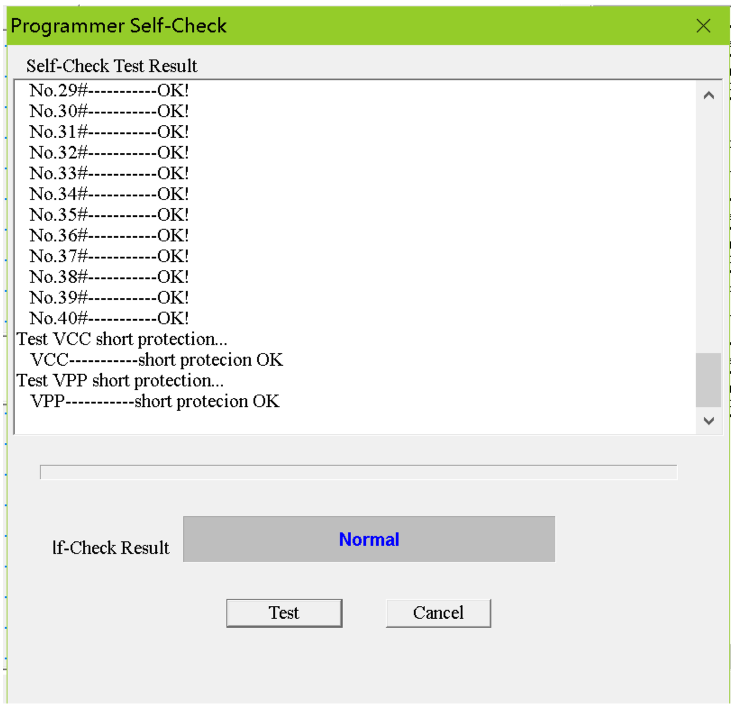
4.6.1 Calculator

This is a calculator that comes with Windows system and it is convenient for calculating various formats when switching to scientific mode.

4.6.2 Self-check programmer hardware.

Note: Before self-checking, please remove the IC and ICSP connecting wires on the ZIF socket, the programmer will apply VPP, VCC, GND to all pins for testing in self check. And test the internal short-circuit over current (apply VPP VCC and GND on each pin at the same time). If the IC is on the ZIF socket, may damage the IC. After the test is completed, the test condition of each power supply is displayed in the dialog box.

4.6.3 self-check on TL866II Plus



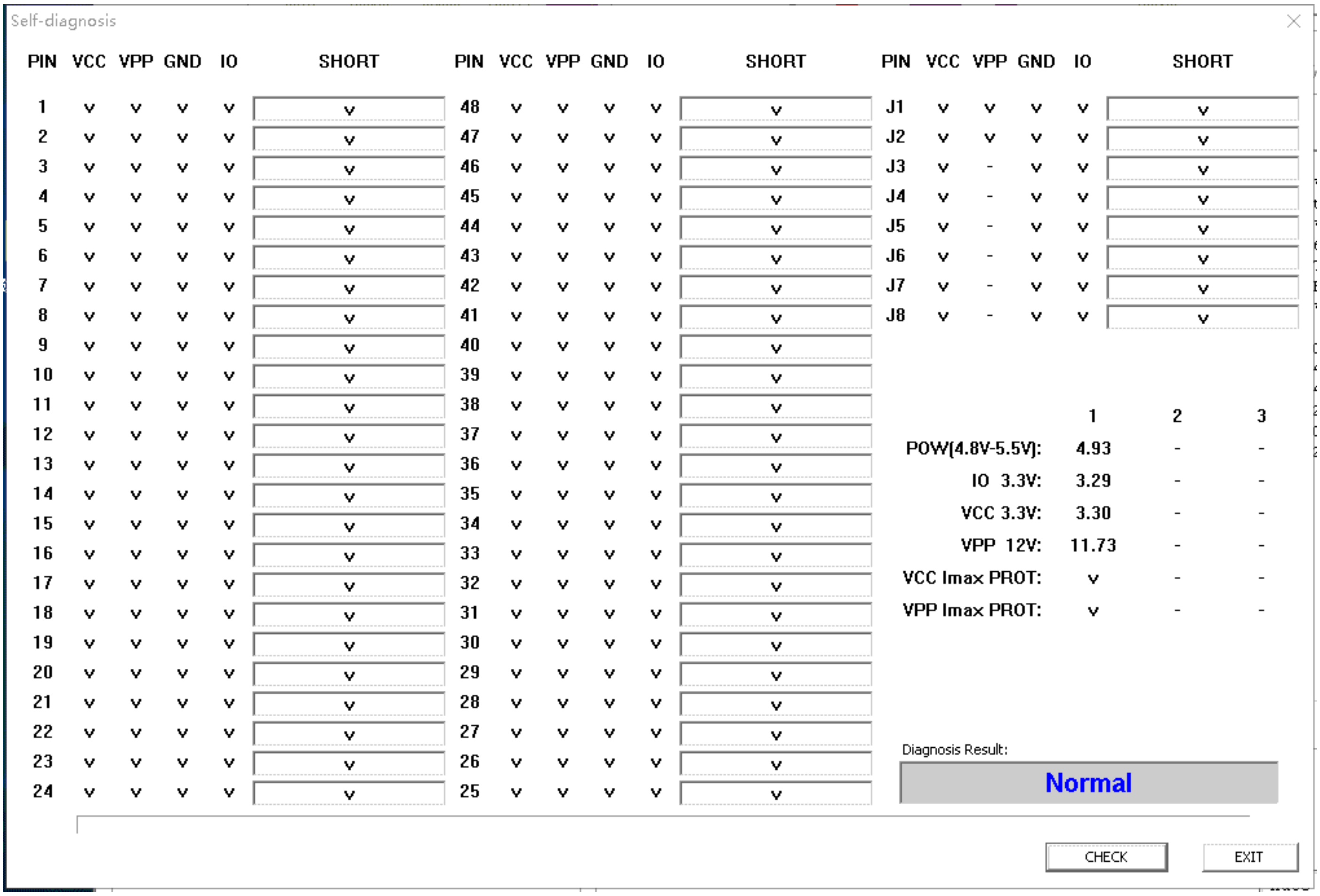
Programmer Self-Check

Self-Check Test Result

```
 No.29#-----------OK!
 No.30#-----------OK!
 No.31#-----------OK!
 No.32#-----------OK!
 No.33#-----------OK!
 No.34#-----------OK!
 No.35#-----------OK!
 No.36#-----------OK!
 No.37#-----------OK!
 No.38#-----------OK!
 No.39#-----------OK!
 No.40#-----------OK!
Test VCC short protection...
  VCC-----------short protecion OK
Test VPP short protection...
  VPP-----------short protecion OK
```

lf-Check Result: Normal

4.6.4 self-check on T56



Self-diagnosis

PIN	VCC	VPP	GND	IO	SHORT	PIN	VCC	VPP	GND	IO	SHORT	PIN	VCC	VPP	GND	IO	SHORT
1	v	v	v	v	v	48	v	v	v	v	v	J1	v	v	v	v	v
2	v	v	v	v	v	47	v	v	v	v	v	J2	v	v	v	v	v
3	v	v	v	v	v	46	v	v	v	v	v	J3	v	-	v	v	v
4	v	v	v	v	v	45	v	v	v	v	v	J4	v	-	v	v	v
5	v	v	v	v	v	44	v	v	v	v	v	J5	v	-	v	v	v
6	v	v	v	v	v	43	v	v	v	v	v	J6	v	-	v	v	v
7	v	v	v	v	v	42	v	v	v	v	v	J7	v	-	v	v	v
8	v	v	v	v	v	41	v	v	v	v	v	J8	v	-	v	v	v
9	v	v	v	v	v	40	v	v	v	v	v						
10	v	v	v	v	v	39	v	v	v	v	v						
11	v	v	v	v	v	38	v	v	v	v	v						
12	v	v	v	v	v	37	v	v	v	v	v						
13	v	v	v	v	v	36	v	v	v	v	v						
14	v	v	v	v	v	35	v	v	v	v	v						
15	v	v	v	v	v	34	v	v	v	v	v						
16	v	v	v	v	v	33	v	v	v	v	v						
17	v	v	v	v	v	32	v	v	v	v	v						
18	v	v	v	v	v	31	v	v	v	v	v						
19	v	v	v	v	v	30	v	v	v	v	v						
20	v	v	v	v	v	29	v	v	v	v	v						
21	v	v	v	v	v	28	v	v	v	v	v						
22	v	v	v	v	v	27	v	v	v	v	v						
23	v	v	v	v	v	26	v	v	v	v	v						
24	v	v	v	v	v	25	v	v	v	v	v						

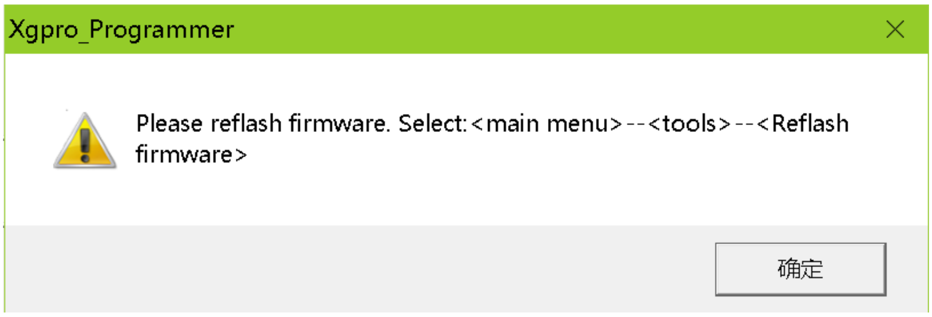
	1	2	3
POW[4.8V-5.5V]:	4.93	-	-
IO 3.3V:	3.29	-	-
VCC 3.3V:	3.30	-	-
VPP 12V:	11.73	-	-
VCC Imax PROT:	v	-	-
VPP Imax PROT:	v	-	-

Diagnosis Result: Normal

4.6.5 **Reflash firmware**

Reflash the firmware of the programmer hardware. The application is upgraded to a new version.

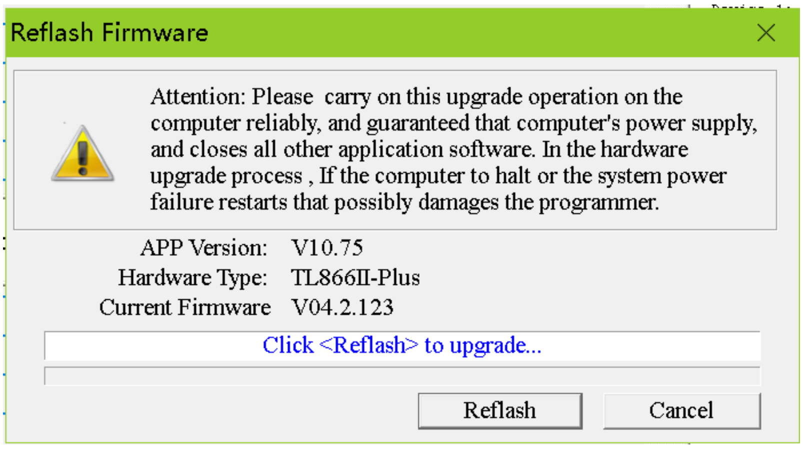
If the program in FLASH needs to be upgraded and reflashed, the application will prompt for a firmware reflash. If no refresh is needed, the reflash function is disabled.



Xgpro_Programmer

Please reflash firmware. Select:<main menu>--<tools>--<Reflash firmware>

Click <Tools-Reflash firmware> to bring up the following dialog box:



Reflash Firmware

Attention: Please carry on this upgrade operation on the computer reliably, and guaranteed that computer's power supply, and closes all other application software. In the hardware upgrade process , If the computer to halt or the system power failure restarts that possibly damages the programmer.

APP Version: V10.75

Hardware Type: TL866II-Plus

Current Firmware V04.2.123

Click <Reflash> to upgrade...

Click <Reflash> button to start, the total reflashing time is about 15 seconds. Click <Cancel> to quit reflashing.

Note:

1. Please do not use an external USB hub when reflashing. Please directly connect the programmer to the USB port of your computer. Using an external hub may not allow the reset upgrade.

2. Close other programs and make sure the power supply of computer is reliable!

4.7 Edit buffer

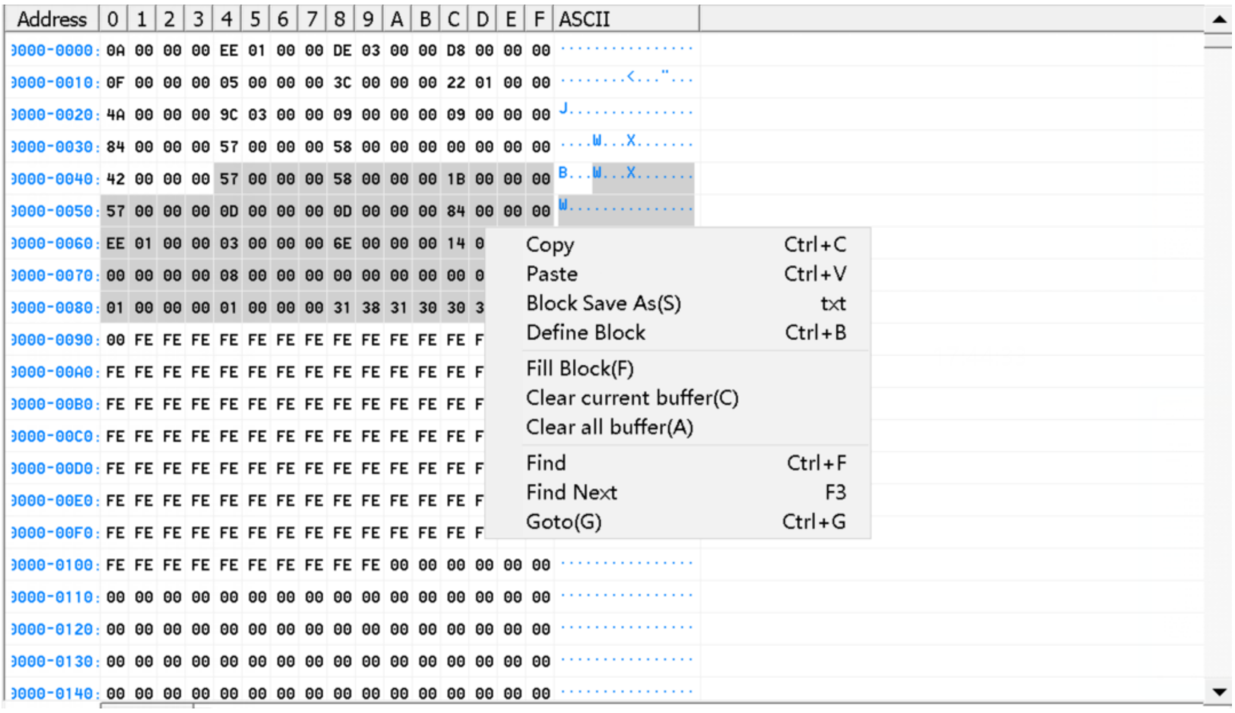
The programmer software can modify the contents in the buffer, the common functions as below:

- Define Block
- Fill Block
- Copy
- Block Save As
- Clear buffer
- Find
- Find Next
- GOTO address

4.7.1 Define Block

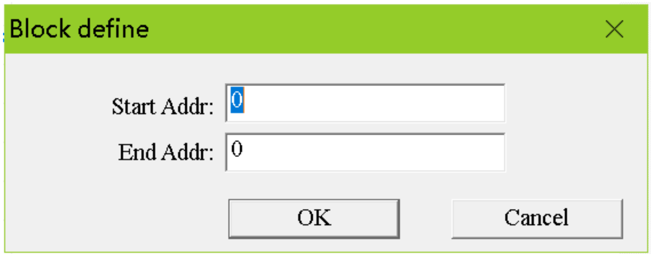
There are two ways to define block:

The first way is: Press the left mouse button in the buffer, move the mouse to select a piece of data and then press the right mouse button, the following menu appears as below:



```
Address    0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  ASCII
0000-0000: 0A 00 00 00 EE 01 00 00 DE 03 00 00 D8 00 00 00 ................
0000-0010: 0F 00 00 00 05 00 00 00 3C 00 00 00 22 01 00 00 ........<..."...
0000-0020: 4A 00 00 00 9C 03 00 00 09 00 00 00 09 00 00 00 J...............
0000-0030: 84 00 00 00 57 00 00 00 58 00 00 00 00 00 00 00 ....W...X.......
0000-0040: 42 00 00 00 57 00 00 00 58 00 00 00 1B 00 00 00 B...W...X.......
0000-0050: 57 00 00 00 0D 00 00 00 0D 00 00 00 84 00 00 00 W...............
0000-0060: EE 01 00 00 03 00 00 00 6E 00 00 00 14 0
0000-0070: 00 00 00 00 08 00 00 00 00 00 00 00 00 0
0000-0080: 01 00 00 00 01 00 00 00 31 38 31 30 30 3
0000-0090: 00 FE FE FE FE FE FE FE FE FE FE FE FE F
0000-00A0: FE FE FE FE FE FE FE FE FE FE FE FE FE F
0000-00B0: FE FE FE FE FE FE FE FE FE FE FE FE FE F
0000-00C0: FE FE FE FE FE FE FE FE FE FE FE FE FE F
0000-00D0: FE FE FE FE FE FE FE FE FE FE FE FE FE F
0000-00E0: FE FE FE FE FE FE FE FE FE FE FE FE FE F
0000-00F0: FE FE FE FE FE FE FE FE FE FE FE FE FE F
0000-0100: FE FE FE FE FE FE FE FE FE FE 00 00 00 00 00 00 ................
0000-0110: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-0120: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-0130: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-0140: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
```

The second way is: in the main menu, click [File] button to select [Define Block], or directly press the shortcut key "CTRL + B" to bring up the following dialog box, type in the start address and end address, and press OK to complete the "Define Block".



Block define

Start Addr: 0

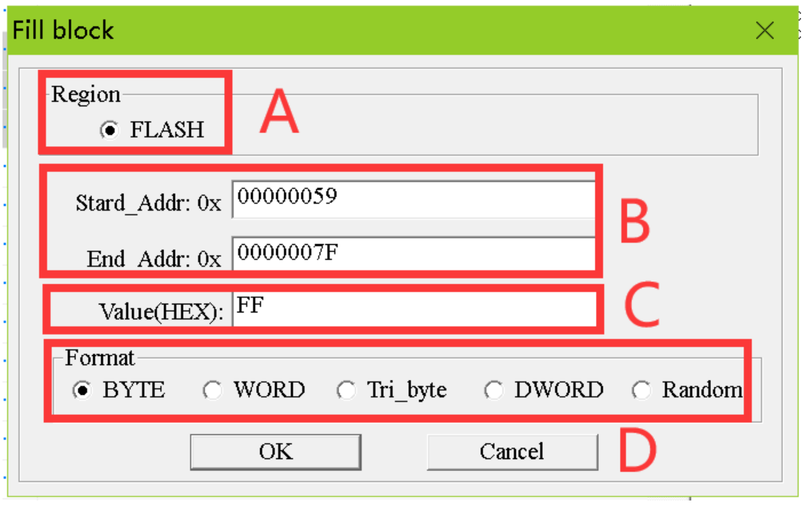
End Addr: 0

Subsequent block operations can be done on the defined data blocks.

Copy block, Fill block, Block save as TXT file

4.7.2 Fill block

“Fill block” is to replace all the data of a continuous address in the program code buffer with a fixed value. Right mouse click to bring up [right click menu], and select [Fill Block], the following dialog box pops up:



Fill block

Region: ● FLASH

Stard_Addr: 0x 00000059

End_Addr: 0x 0000007F

Value(HEX): FF

Format: ● BYTE ○ WORD ○ Tri_byte ○ DWORD ○ Random

A. Buffer Region, the default value is the buffer displayed in the current window.

B. The start address and end address of the block to be filled. The default value is from the start to the end of the defined data block if a block is defined. If no block is defined, it is the full region of the current buffer.

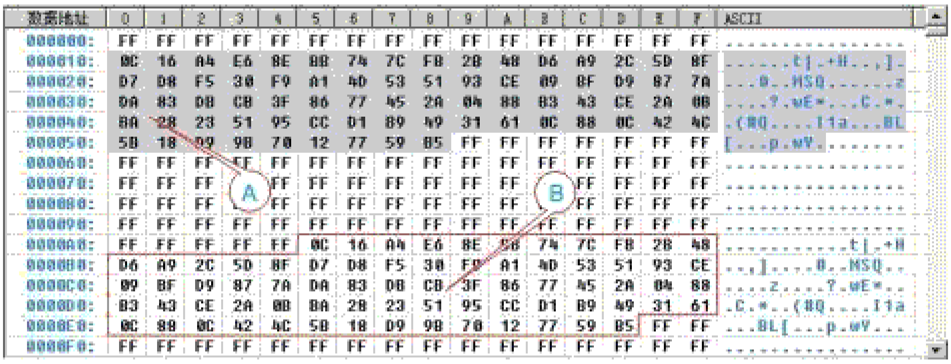
C. Fill value, the maximum allowed is 4 bytes. If the fill format is random, the value is irrelevant.

D. Filling format, default format: If the chip is programmed by 8-bit mode, the default is single-byte mode. If the chip is programmed by 16-bit mode, it is double-byte mode, you can change the filling format you want.

4.7.3 Copy

After defining block, right mouse click to bring up [right click menu] and select [Copy], then the content of the block has been copied to the clipboard of WINDOWS. You can also directly press shortcut key "CTRL+C" to finish copying.

Copy data from area A to area B as below:



```
数据地址  0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  ASCII
000000:  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  ................
000010:  0C 16 A4 E6 8E 88 74 7C F8 28 48 D6 A9 2C 5D 8F  ......t|.+H..,].
000020:  D7 D8 F5 30 F9 A1 4D 53 51 93 CE 09 8F D9 87 7A  ...0..MSQ......z
000030:  DA 83 D8 CB 3F 86 77 45 2A 04 88 83 43 CE 2A 0B  ....?.wE*...C.*.
000040:  8A 28 23 51 95 CC D1 89 49 31 61 0C 88 0C 42 4C  .(#Q....I1a...BL
000050:  5B 18 D9 9B 70 12 77 59 85 FF FF FF FF FF FF FF  [...p.wY........
000060:  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  ................
000070:  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  ................
000080:  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  ................
000090:  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  ................
0000A0:  FF FF FF FF FF 0C 16 A4 E6 8E 88 74 7C F8 28 48  ...........t|.+H
0000B0:  D6 A9 2C 5D 8F D7 D8 F5 30 F9 A1 4D 53 51 93 CE  ..,]....0..MSQ..
0000C0:  09 8F D9 87 7A DA 83 D8 CB 3F 86 77 45 2A 04 88  ....z....?.wE*..
0000D0:  83 43 CE 2A 0B 8A 28 23 51 95 CC D1 89 49 31 61  .C.*..(#Q....I1a
0000E0:  0C 88 0C 42 4C 5B 18 D9 9B 70 12 77 59 85 FF FF  ...BL[...p.wY...
0000F0:  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  ................
```

Note: In order to enhance the speed of pasting, the data format of clipboard is a special binary format for this programmer. So other WINDOW applications cannot directly use the data copied to the clipboard by this program. If you want to copy the data block to other applications, please use the [Block Save As] TXT file function.

Block Save As

You can output the defined data block as a TXT text file, that is convenient for designers to apply when programming. The text format is automatically arranged in 16-bit or 8-bit way. Operation method: After defining block, click the right mouse button to bring up the [right click menu] and select [Save Block As].

4.7.4 Clear Buffer

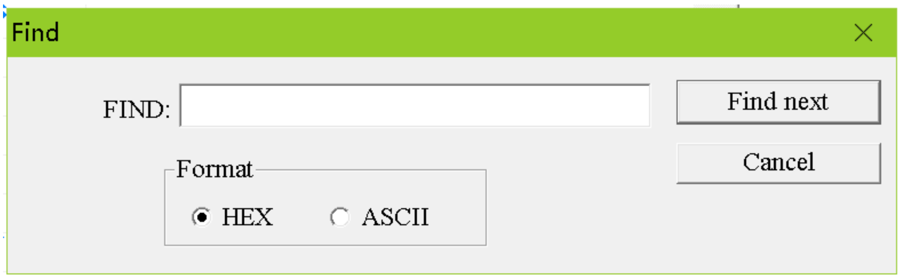
Clear Current Buffer: clear the buffer displayed in the current window of the application.

Clear all buffers: clear all buffers of the chip with default values.

Find, Find Next

Function: Find the HEX or ASCII string in the current buffer.

Operation: Press shortcut key “CTRL+F” or click the [Find] in the right-click menu, the following dialog box pops up:



FIND:

Format

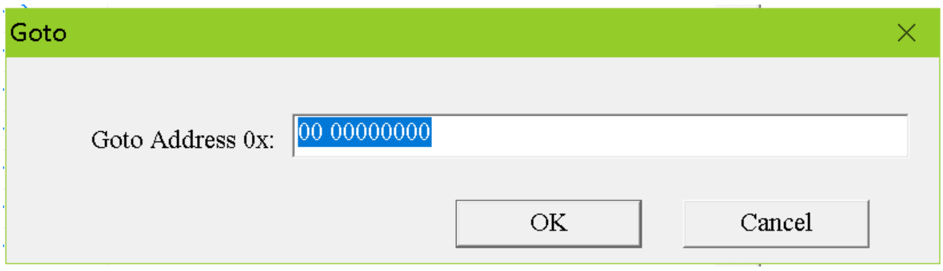
● HEX ○ ASCII

Type in the HEX value or ASCII string you want to find, and finally click the "Find" button in the dialog box. When there is a matching hexadecimal data or string in the program code buffer, the cursor will be located to the address of the first matching content and press F3 to continue finding the next one until the buffer is completely searched.

4.7.5 Goto address

Function: Move the cursor to the address specified in the current buffer.

Operation: Press shortcut key "CTRL+G", or press the right mouse button to bring up the menu and select [Goto], then the following dialog box pops up:



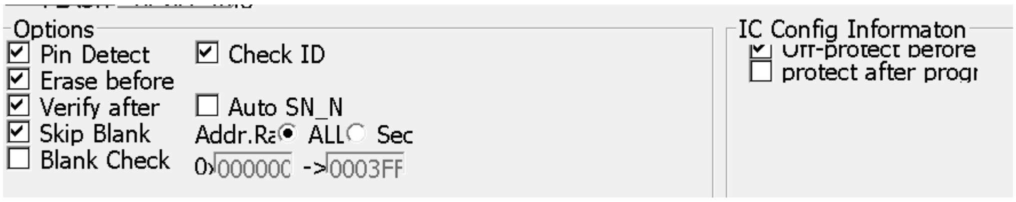
Goto Address 0x: 00 00000000

Type in the address, click OK button and the cursor will move to the address.

4.8 Options

Options of programming setting are shown below. You can set the options

Use default setting for mass production , don't need to modify.



Options

☑ Pin Detect ☑ Check ID

☑ Erase before

☑ Verify after ☐ Auto SN_N

☑ Skip Blank Addr.Ra ● ALL ○ Sec

☐ Blank Check 0x000000 ->0003FF

IC Config Informaton

☑ Off-protect before

☐ protect after progr

4.8.1 Pin Detect

Before operating, the software automatically checks if the chip pins are in good contact. It can avoid programming or reading errors caused by poor pin contact in most cases.

> Special case of pin poor contact
> Because of pin severe oxidation, some individual pins may not be completely disconnected, (for example, it is equivalent to a contact of 3-10K resistance). In this case it is not possible to detect if the contact is good, and errors may occur during programming. So for old chips, the pins should be handled carefully.

For very few chips, the pin detect may fail. You can temporarily uncheck this option "Pin Detect" while ensuring that the pin contact is good. In that case, no pin detection will be performed before reading or writing.

4.8.2 Erase before (programming)

Erase the entire contents of the chip before programming
Some chips do not have erase function, this option will be disabled, for example: most 27Cxxx chips are one time OTP ROM or UVROM. These chips are not erasable or do not have the electrical erase function.
For 24 series, 25 series, EEPROM 93 series chips, this function is also disabled when there is no special erase command. That means these chips do not need to be erased and can be rewritten directly. (If you must erase the data of the chip, you can write 0xFF into the entire memory of the chip)

4.8.3 Verify after (programming)

4.8.4 Skip Blank

The write operation will be skipped for FF data blocks, that can greatly improve the programming efficiency. When this option is checked, the verify operation will also be skipped.

4.8.5 Check ID

Many chips have an internal identification mark, the content of which is generally composed of two or more bytes: generally, the first byte is the Manufacture ID, followed by the chip type or chip capacity. Different chips have different IDs, this option can prevent incorrect chips put on. This option is to check the chip ID before reading or writing. If it is correct, continue; if incorrect, stop. This function is optional. The option is enabled by default.

> For chips of the same type, they have different manufacturers, but may be programmed in the same way. Therefore, for chips that are not in the support list, you can select other manufacturers of similar model chips with the same capacity for programming. Because the ID is different, before programming, just uncheck this option [Check ID]).

4.8.6 Address range

You can set the address range of the programming chip, that can improve the efficiency when mass production.

4.8.7 Blank Check (before programming)

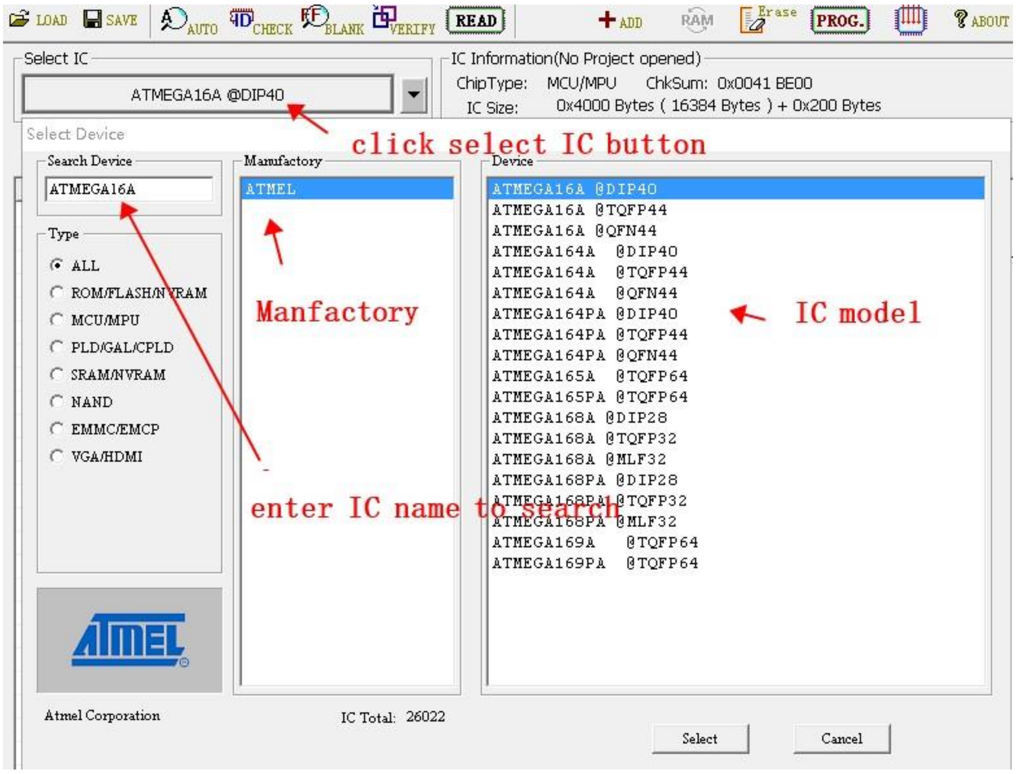
Except for the 27 series UVROM, generally it is not necessary to check this option.

4.8.8 Auto SN_N

Refer to section 4.10.7 Setting the Serials number

4.9 Select IC

Click <Select IC> button in the upper left corner of the software interface



Select IC: ATMEGA16A @DIP40

IC Information(No Project opened)
ChipType: MCU/MPU ChkSum: 0x0041 BE00
IC Size: 0x4000 Bytes (16384 Bytes) + 0x200 Bytes

Select Device

Search Device: ATMEGA16A

Type
- ● ALL
- ○ ROM/FLASH/NVRAM
- ○ MCU/MPU
- ○ PLD/GAL/CPLD
- ○ SRAM/NVRAM
- ○ NAND
- ○ EMMC/EMCP
- ○ VGA/HDMI

Manfactory: ATMEL

Device:

```
ATMEGA16A @DIP40
ATMEGA16A @TQFP44
ATMEGA16A @QFN44
ATMEGA164A  @DIP40
ATMEGA164A  @TQFP44
ATMEGA164A  @QFN44
ATMEGA164PA @DIP40
ATMEGA164PA @TQFP44
ATMEGA164PA @QFN44
ATMEGA165A  @TQFP64
ATMEGA165PA @TQFP64
ATMEGA168A @DIP28
ATMEGA168A @TQFP32
ATMEGA168A @MLF32
ATMEGA168PA @DIP28
ATMEGA168PA @TQFP32
ATMEGA168PA @MLF32
ATMEGA169A   @TQFP64
ATMEGA169PA  @TQFP64
```

Atmel Corporation

IC Total: 26022

Select Cancel

click select IC button

Manfactory

IC model

enter IC name to search

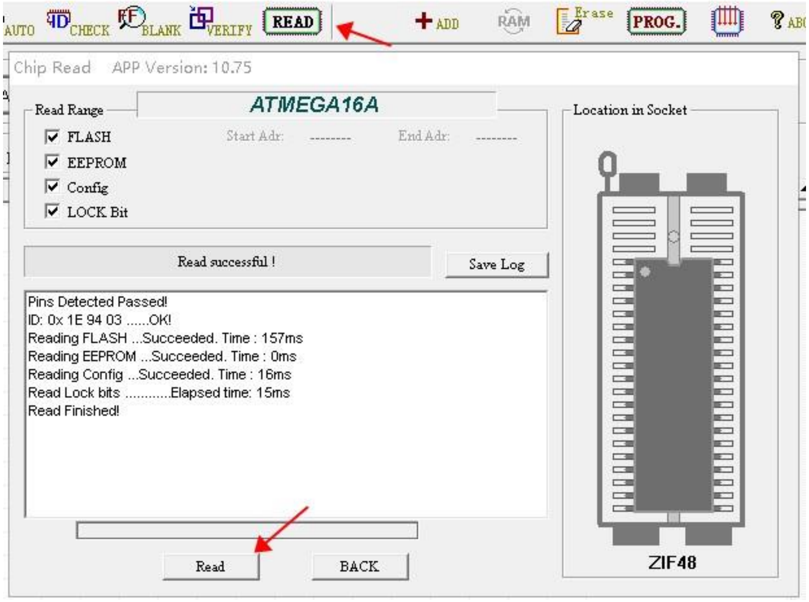
When selecting, the package of chip must be right. Sometimes, different packages of chips have different pin definitions. For the same type of IC, (none Package is DIP package chips.)

if the specific package is not listed, just check the IC data sheet and directly connect the corresponding pins to program.

4.10 Programmer operation function

4.10.1 Read

Read means reading out the data from the chip into the buffer. The reading window may be different from different models of chips. for example, the chip ATMEGA16L, click [Read] button in the toolbar and the following dialog box pops up:



Chip Read APP Version: 10.75

ATMEGA16A

Read Range
- ☑ FLASH
- ☑ EEPROM
- ☑ Config
- ☑ LOCK Bit

Start Adr: -------- End Adr: --------

Location in Socket

Read successful !

Save Log

```
Pins Detected Passed!
ID: 0x 1E 94 03 ......OK!
Reading FLASH ...Succeeded. Time : 157ms
Reading EEPROM ...Succeeded. Time : 0ms
Reading Config ...Succeeded. Time : 16ms
Read Lock bits ...........Elapsed time: 15ms
Read Finished!
```

Read BACK

First of all, set the [Read Range], which includes [FLASH] [EEPROM] [Config.] [LOCK Bit]. do reading operation on the entire chip. Click [Read] button to start reading the data of chip, note: if the chip is encrypted, the read data is invalid.

(for ATMEGA, [LOCK Bit bytes] in config, the datas are readable whether be encrypted or not.) After reading, the reading time of the individual buffers is displayed in the information list box.

Note: generally, the read operation will not report an error. If you want to verify whether the read data is correct, you can use the [Verify] function. Please see the verification instruction for more details. The chip ID value will be checked by default before operation.

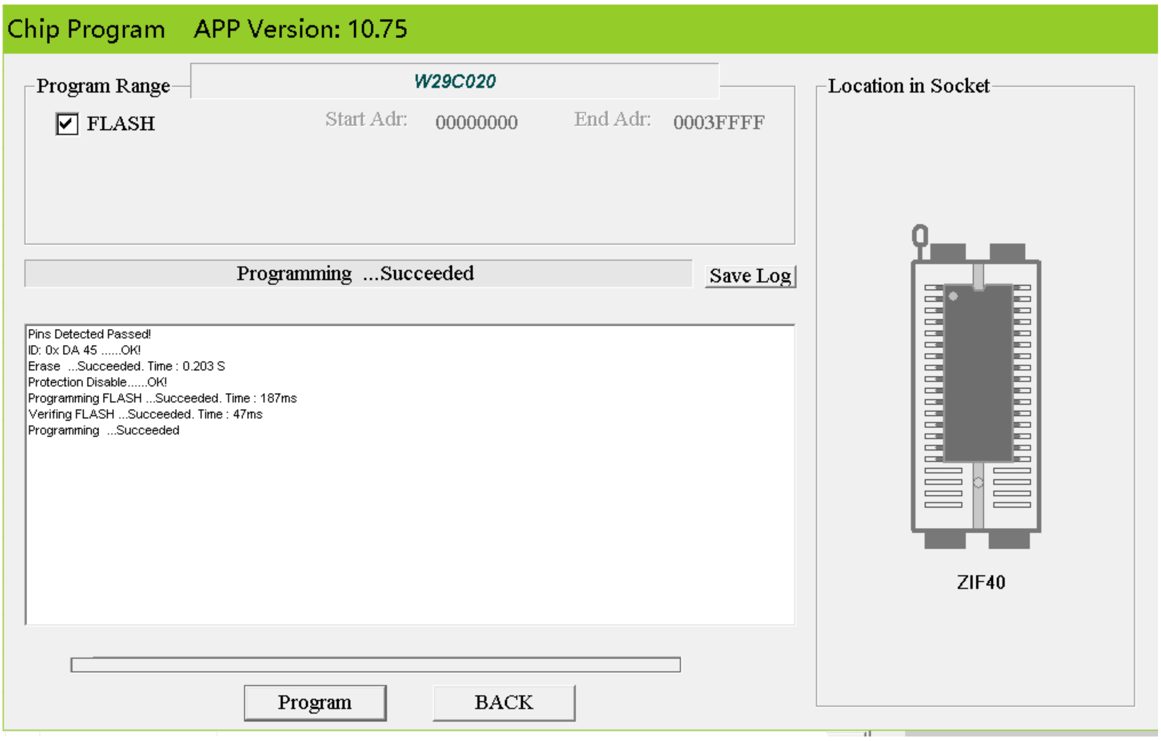
4.10.2 **Program**

Load the data file to program chip

Click **[PROG.]** button in the toolbar, the chip program dialog box pops up as below:

Take W29C020 as an example,

Normal programming is shown below:

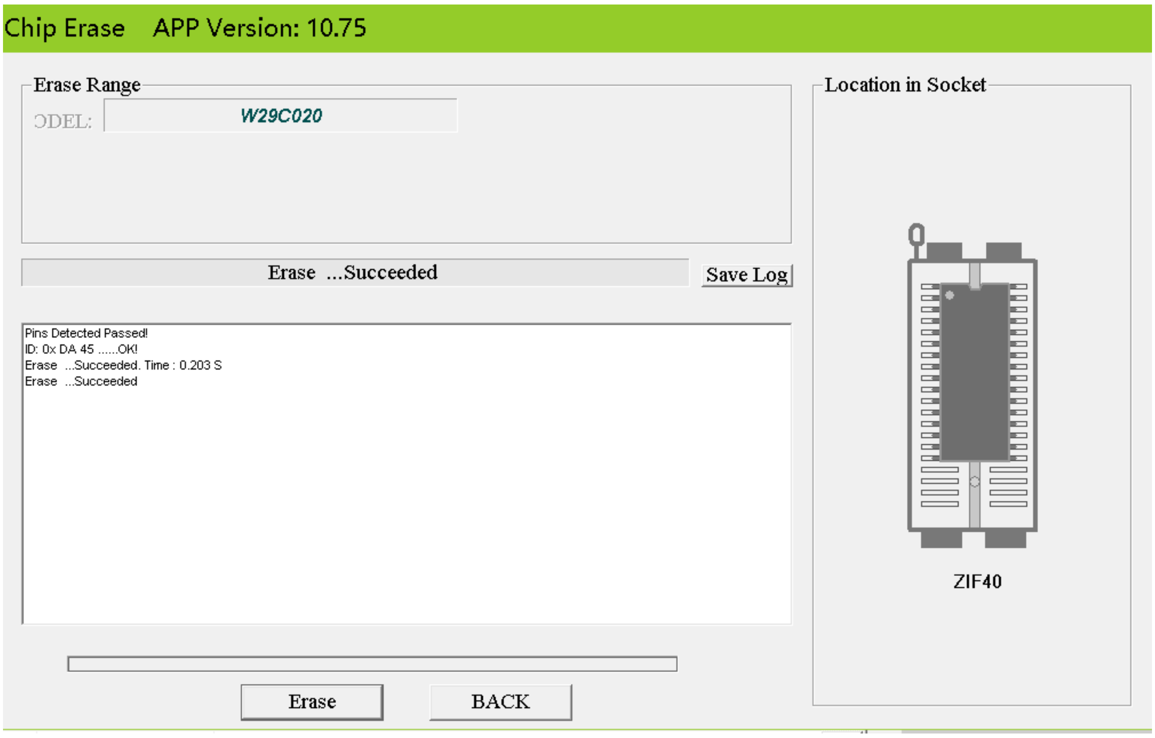


```
Pins Detected Passed!
ID: 0x DA 45 ......OK!
Erase ...Succeeded. Time : 0.203 S
Protection Disable......OK!
Programming FLASH ...Succeeded. Time : 187ms
Verifing FLASH ...Succeeded. Time : 47ms
Programming ...Succeeded
```

- For most of the chips, the programmer will automatically verify during the programming process. Will stop immediately if an error is found.
- If the option [Auto SN_N] is checked, the chip number in the buffer will be automatically incremented according to the setting after programming.
- To ensure the correctness of programming data, it is advisable to check the option **[Verify after]**.
-

4.10.3 **Erase**

For the electrically erasable chip, click [Erase] button in the toolbar to bring up the following dialog box:



Erase ...Succeeded

```
Pins Detected Passed!
ID: 0x DA 45 ......OK!
Erase ...Succeeded. Time : 0.203 S
Erase ...Succeeded
```

The option [Check ID] is checked by default. For few parts of microcontrollers, some chips while it be encrypted and the ID cannot be read out. In that case, please uncheck the option [Check ID], otherwise ID Error will be displayed. Click [Erase] button in the dialog box, after the erase succeeded, the erasing time is displayed.

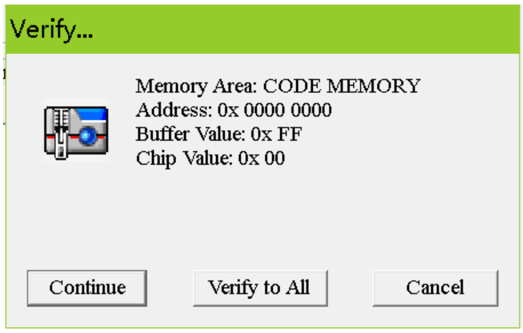
4.10.4 Blank Check

Click [Check Blank] button in the dialog box to start the check blank operation. When it is finished, the elapsed time is displayed. You can select the region of the chip before checking, all region is by default. If the chip is found to be non-empty during the blank check, will stop operation and the address and stored value of the non-empty area will be displayed.

4.10.5 Verify

You can select the region of the chip before verifying, all region is by default. Click [Verify] button in the dialog box to start the operation. If all the data are the same, the elapsed time is displayed when verify finished.

If the chip datas are found to be different during verifying, the following dialog box will pop up:

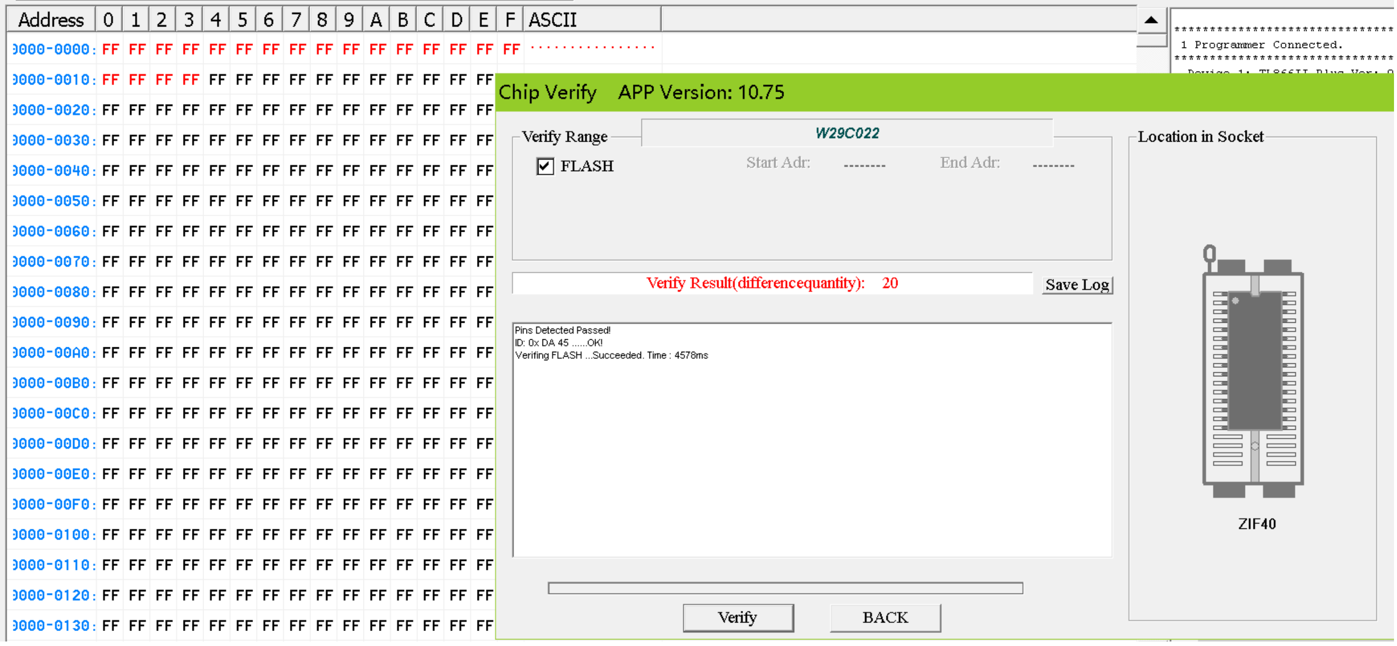


Memory Area: CODE MEMORY
Address: 0x 0000 0000
Buffer Value: 0x FF
Chip Value: 0x 00

Click [Continue]: The same dialog box will pop up again if there are still differences.

Click [Verify to all]: It will compare all the contents of the chip, and prompt will pop up if differences are found during the comparison process, until it is finished. It will prompt how many differences and show the different cells in red font in the buffer (except NAND) , as shown below:

Click [BACK]: It will exit directly without the later verifying.



```
Address | 0| 1| 2| 3| 4| 5| 6| 7| 8| 9| A| B| C| D| E| F| ASCII
0000-0000: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  ................
0000-0010: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0000-0020: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0000-0030: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0000-0040: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0000-0050: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0000-0060: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0000-0070: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0000-0080: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0000-0090: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0000-00A0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0000-00B0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0000-00C0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0000-00D0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0000-00E0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0000-00F0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0000-0100: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0000-0110: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0000-0120: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
0000-0130: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
```

```
********************************
 1 Programmer Connected.
********************************
```

Chip Verify APP Version: 10.75

Verify Range: W29C022

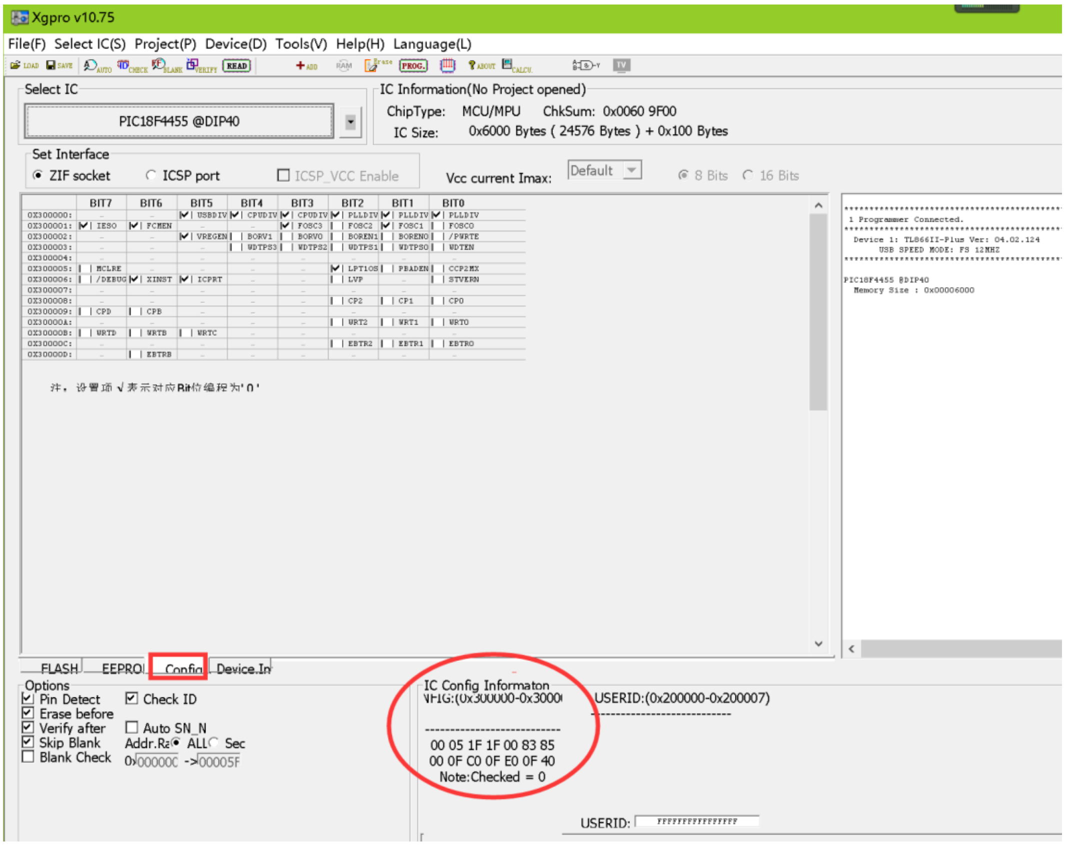
☑ FLASH Start Adr: -------- End Adr: --------

Verify Result(differencequantity): 20

```
Pins Detected Passed!
ID: 0x DA 45 ......OK!
Verifing FLASH ...Succeeded. Time : 4578ms
```

4.10.6 IC Config.

Click [Config.] button, the interface is as below (select IC model: PIC18F4555):



Xgpro v10.75
File(F) Select IC(S) Project(P) Device(D) Tools(V) Help(H) Language(L)
LOAD SAVE AUTO CHECK BLANK VERIFY READ ADD RAM Erase PROG. ABOUT CALCU.
FLASH
EEPROI
Config
Device.In

Select IC: PIC18F4455 @DIP40

IC Information(No Project opened)
ChipType: MCU/MPU ChkSum: 0x0060 9F00
IC Size: 0x6000 Bytes (24576 Bytes) + 0x100 Bytes

Set Interface: ● ZIF socket ○ ICSP port □ ICSP_VCC Enable

Vcc current Imax: Default ● 8 Bits ○ 16 Bits

	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
0X300000:	-	-	☑ USBDIV	☑ CPUDIV	☑ CPUDIV	☑ PLLDIV	☑ PLLDIV	☑ PLLDIV
0X300001:	☑ IESO	☑ FCMEN	-	-	☑ FOSC3	□ FOSC2	☑ FOSC1	□ FOSC0
0X300002:	-	-	☑ VREGEN	□ BORV1	□ BORV0	□ BOREN1	□ BOREN0	□ /PWRTE
0X300003:	-	-	-	□ WDTPS3	□ WDTPS2	□ WDTPS1	□ WDTPS0	□ WDTEN
0X300004:	-	-	-	-	-	-	-	-
0X300005:	□ MCLRE	-	-	-	-	☑ LPT1OS	□ PBADEN	□ CCP2MX
0X300006:	□ /DEBUG	☑ XINST	☑ ICPRT	-	-	□ LVP	-	□ STVERN
0X300007:	-	-	-	-	-	-	-	-
0X300008:	-	-	-	-	-	□ CP2	□ CP1	□ CP0
0X300009:	□ CPD	□ CPB	-	-	-	-	-	-
0X30000A:	-	-	-	-	-	□ WRT2	□ WRT1	□ WRT0
0X30000B:	□ WRTD	□ WRTB	□ WRTC	-	-	-	-	-
0X30000C:	-	-	-	-	-	□ EBTR2	□ EBTR1	□ EBTR0
0X30000D:	-	□ EBTRB	-	-	-	-	-	-

注，设置项√表示对应Bit位编程为'0'

```
******************************************
 1 Programmer Connected.
******************************************
  Device 1: TL866II-Plus Ver: 04.02.124
       USB SPEED MODE: FS 12MHZ
******************************************

PIC18F4455 @DIP40
  Memory Size : 0x00006000
```

Options
☑ Pin Detect ☑ Check ID
☑ Erase before
☑ Verify after □ Auto SN_N
☑ Skip Blank Addr.Ra ● ALL ○ Sec
□ Blank Check 0x 000000 -> 00005F

IC Config Informaton
NFIG:(0x300000-0x3000
----
00 05 1F 1F 00 83 85
00 0F C0 0F E0 0F 40
Note:Checked = 0

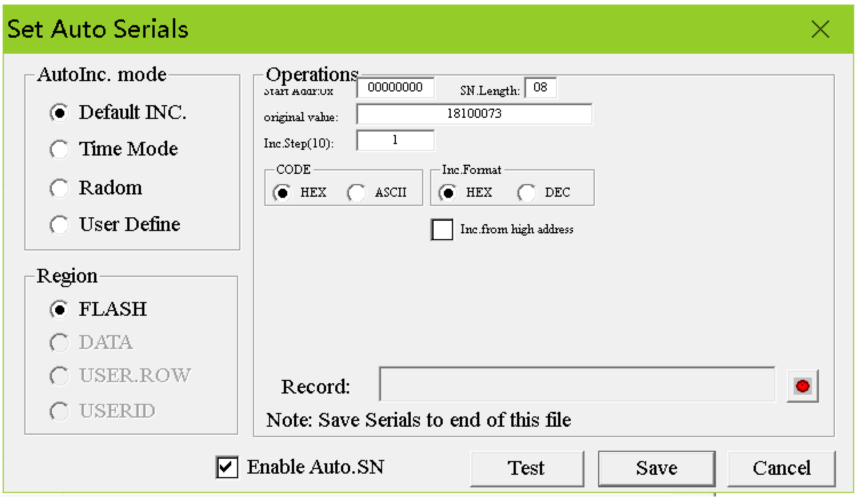
USERID:(0x200000-0x200007)

USERID: FFFFFFFFFFFFFFFF

The value of the config information in the red circle corresponds to the config interface: click on the config interface, the value in the config information will change automatically. The config information varies from chip to chip, so please check the chip's data sheet at first, and configure IC correctly according to your specific needs.IC config, need to understand the role of each bit in the configuration bits

4.10.7 **Serials number**

The chip number setting function is to automatically add a number to a certain memory area of the chip during programming, that can easily program the chip at the factory. You can add time or unique serial numbers and so on, also can use a variety of numbering algorithms. Click <Device>--<Serials Number> in main menu to bring up the following dialog box:



Set Auto Serials

AutoInc. mode
- ● Default INC.
- ○ Time Mode
- ○ Radom
- ○ User Define

Region
- ● FLASH
- ○ DATA
- ○ USER.ROW
- ○ USERID

Operations
- Start Addr:0x 00000000 SN.Length: 08
- original value: 18100073
- Inc.Step(10): 1
- CODE: ● HEX ○ ASCII
- Inc.Format: ● HEX ○ DEC
- □ Inc.from high address

Record:

Note: Save Serials to end of this file

☑ Enable Auto.SN

Firstly, select the automatic numbering algorithm, there are four modes in the dialog box. And then introduce the every mode separately as below:

4.10.7.1 Default INC. (Default Incremental Algorithm)

The setting interface of this algorithm is shown above. The function of this algorithm is to put a number of set length at the place where the set storage area of the start address, and the number increases according to step length at a time. After setting the parameters, you can click <Test> button to test and see the change rule under various parameter settings.

The initial value and step length of the number can be set in the dialog box.

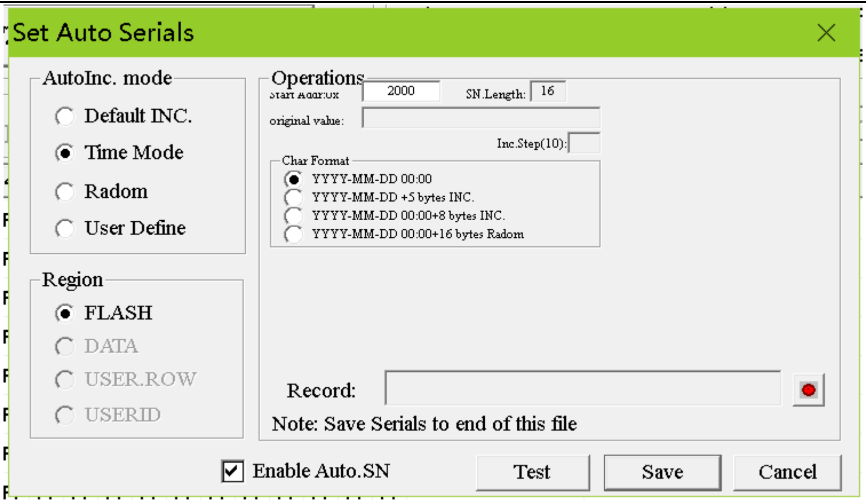
After setting the parameters and testing them well, set the record file. If it is empty, the record file is invalid and the data will not be recorded into the file when programming. If the record file is set, the number you set will be added to the end of the record file after each programming.

Finally, check the option <Enable Auto.SN> and click<Save> button.

4.10.7.2 Time Mode (algorithm)

The interface of the Time Mode dialog box is as below:

at 0x2000, a time data is put in



Set Auto Serials

AutoInc. mode
○ Default INC.
● Time Mode
○ Radom
○ User Define

Region
● FLASH
○ DATA
○ USER.ROW
○ USERID

Operations
Start Addr:0x 2000 SN.Length: 16
original value:
Inc.Step(10):

Char Format
● YYYY-MM-DD 00:00
○ YYYY-MM-DD +5 bytes INC.
○ YYYY-MM-DD 00:00+8 bytes INC.
○ YYYY-MM-DD 00:00+16 bytes Radom

Record:
Note: Save Serials to end of this file

☑ Enable Auto.SN

There are four types of time encoding formats, which are as below:

(1) Date + time Total length of number: 16bytes

The date and time are added at the specified location during programming, the total length is 16bytes.

(2) Date + 5 bytes increment Total length of number: 16 bytes

The increment part can set the initial value and step length.

(3) Date + time + 8 bytes increment Total number length: 24 bytes

The increment part can set the initial value and step length.

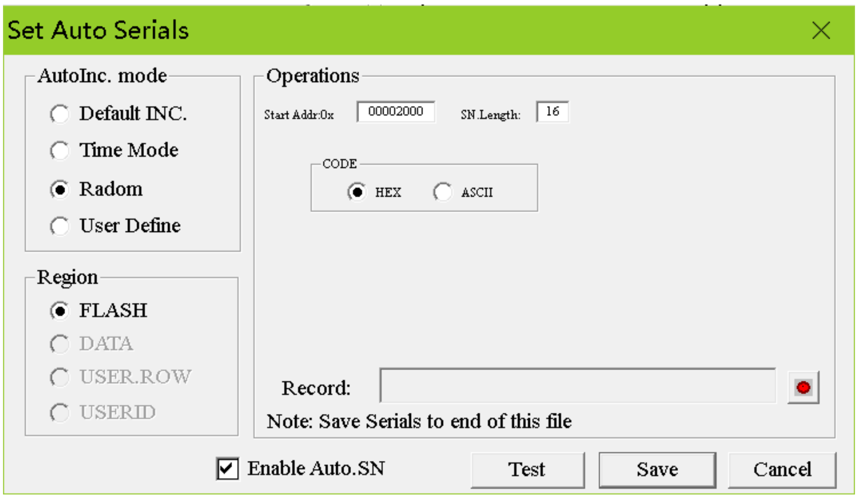
(4) Date + time + 16 bytes of random numbering Total length of numbering: 32 bytes

16 bytes are randomly generated bytes, which can generate a unique serial number of product. In this way, the record file must be set so that the generated code is recorded in the file. Otherwise, you don't know what the product number is.

The next steps are the same as above <Default Incremental Algorithm>

4.10.7.3 Random (Random number algorithm)

The algorithm generates a random number of the specified length, as below: 16 bytes of random data are placed at the start address 0x2000



Set Auto Serials

AutoInc. mode
○ Default INC.
○ Time Mode
● Radom
○ User Define

Region
● FLASH
○ DATA
○ USER.ROW
○ USERID

Operations
Start Addr:0x 00002000 SN.Length: 16

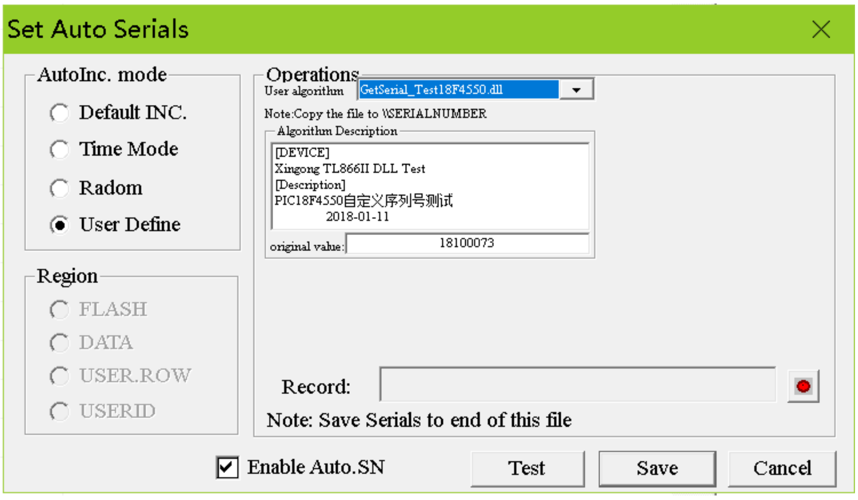
CODE
● HEX ○ ASCII

Record:
Note: Save Serials to end of this file

☑ Enable Auto.SN

4.10.7.4 User Define algorithm

The user-defined algorithm is very flexible, you can set any numbering algorithm, see the catalog for details about how to use it. \\Serialnumber\source_dll in the VC++ source program for detailed instruction. The interface is as below:



Set Auto Serials

AutoInc. mode
○ Default INC.
○ Time Mode
○ Radom
● User Define

Region
○ FLASH
○ DATA
○ USER.ROW
○ USERID

Operations
User algorithm: GetSerial_Test18F4550.dll
Note:Copy the file to \\SERIALNUMBER
Algorithm Description
[DEVICE]
Xingong TL866II DLL Test
[Description]
PIC18F4550自定义序列号测试
2018-01-11
original value: 18100073

Record:
Note: Save Serials to end of this file

☑ Enable Auto.SN

The RC correction byte encryption method for AVR microcontrollers is implemented entirely by calling the DLL dynamic library here. ... \\TAMEGA_LED directory contains all the source programs and test methods. After making the DLL file, you just need to copy it to the application directory of ... \\Serialnumber subdirectory. When setting, choose the algorithm file you made in the user algorithm file list. The following steps are the same as the other algorithms.

Important Notes:

1) After setting the above content, the programmer will not automatically add the number you set to the chip when programming. If need to automatically add the number, you must check the option [Auto SN_N] in the main interface.

2) In the main interface, the condition that you can check the option [Auto SN_N] is that you must firstly set the numbering algorithm and check the option [Enable Auto.SN] in the set auto serials dialog box.

3) After checking the option [Auto SN_N] in the main interface, if you reset the numbering algorithm, the option [Auto SN_N] must be re-checked in the main interface.

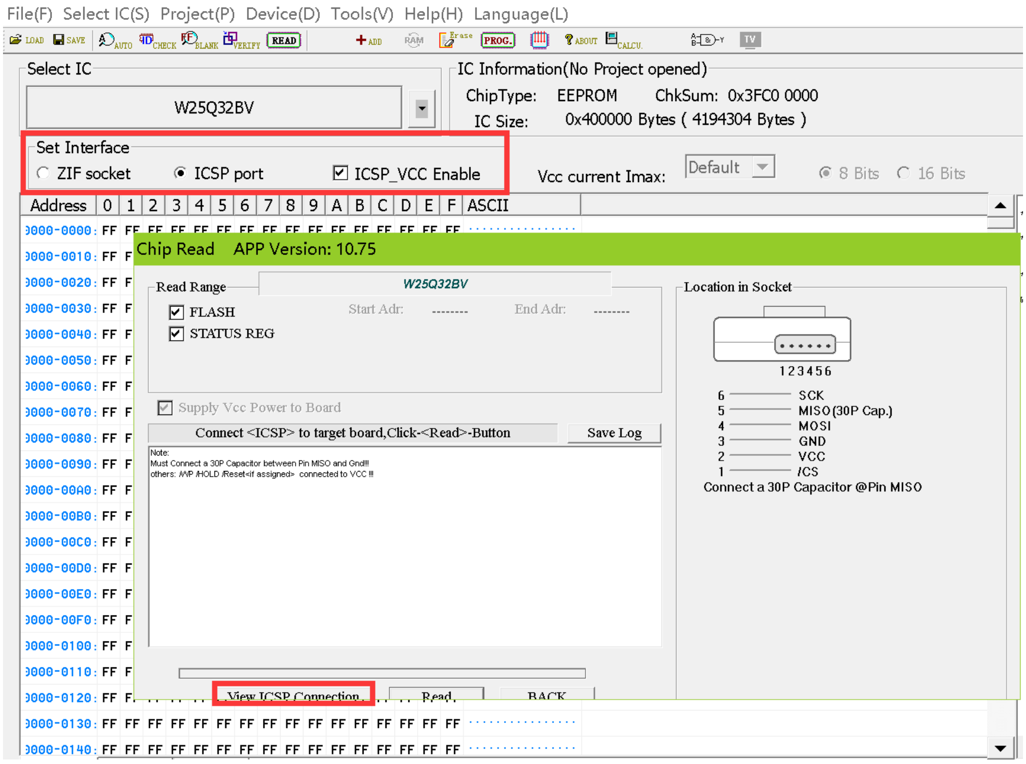
4) If IC model is changed, you must reset the numbering algorithm.

5) In the engineering mode, the algorithm is saved in the project file. You can directly open the project file and automatically transfer to the automatic numbering algorithm. Therefore, it is generally used in the project mode for mass production, it is simple and convenient.

4.11 ICSP Programming

For chips that can be in-circuit programmed serially, the programmer can program the chips on the target board through the unique ICSP interface. Before programming, check the option [ICSP port] in the main interface, at this time the option [ICSP_VCC_Enable] becomes available. The option [ICSP_VCC_Enable] is checked by default, that means the programmer provides VCC power to the target board. Please note: a maximum current that VCC power supply of programmer can provide is 120ma. If the target board needs a larger power supply, please use the power supply from target board itself.

If you do not need power supply from programmer, you can uncheck the option [ICSP_VCC_Enable]. In this case, there will be no VCC power on VCC pins when programming. It is shown as below:


File(F) Select IC(S) Project(P) Device(D) Tools(V) Help(H) Language(L)
LOAD
SAVE
AUTO
CHECK
BLANK
VERIFY
READ
ADD
RAM
Erase
PROG.
ABOUT
CALCU.
TV
Address 0 1 2 3 4 5 6 7 8 9 A B C D E F ASCII
View ICSP Connection
Read
BACK
Save Log
1 2 3 4 5 6

Select IC: W25Q32BV

IC Information(No Project opened)

ChipType: EEPROM ChkSum: 0x3FC0 0000

IC Size: 0x400000 Bytes (4194304 Bytes)

Set Interface: ○ ZIF socket ● ICSP port ☑ ICSP_VCC Enable

Vcc current Imax: Default ● 8 Bits ○ 16 Bits

Chip Read APP Version: 10.75

Read Range: W25Q32BV

☑ FLASH Start Adr: -------- End Adr: --------

☑ STATUS REG

☑ Supply Vcc Power to Board

Connect <ICSP> to target board,Click-<Read>-Button

Note:
Must Connect a 30P Capacitor between Pin MISO and Gnd!!
others: /WP /HOLD /Reset<if assigned> connected to VCC !!!

Location in Socket

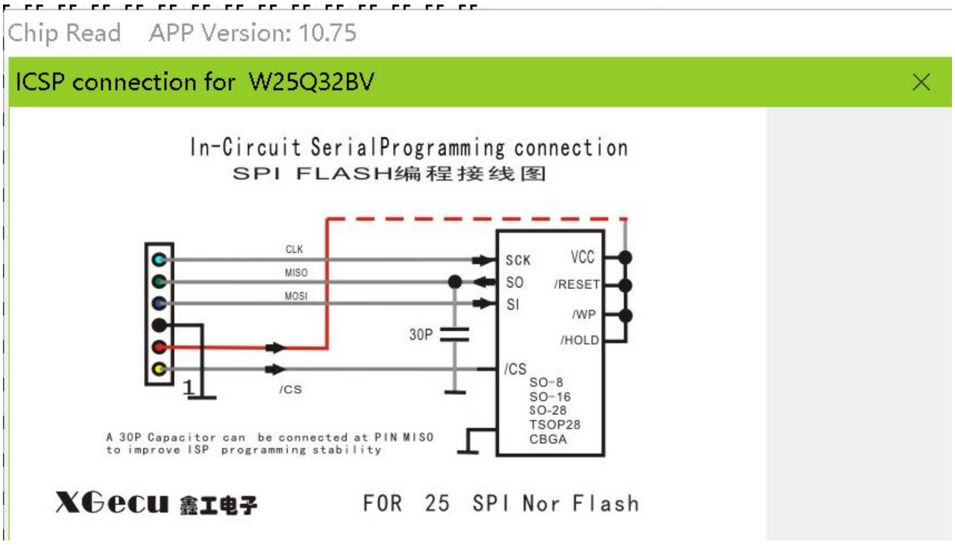
6 SCK
5 MISO(30P Cap.)
4 MOSI
3 GND
2 VCC
1 /CS

Connect a 30P Capacitor @Pin MISO

Next, Read, Erase, Verify, and Program operations will work through the ICSP port. For TL866II, please don't put the chip on the 40PIN universal ZIF socket during operation. The universal ZIF socket of T56 is independent of the ISP port.

Click [Read] button in the toolbar, the following dialog box pops up as below (Please note: the dialog box for read, erase, verify and other operations is the same as write operation):

On the right side, the ICSP port wiring schematic is displayed. For more details, you can click [View ICSP Connection] button.


Chip Read APP Version: 10.75
ICSP connection for W25Q32BV
In-Circuit SerialProgramming connection
SPI FLASH编程接线图
CLK
MISO
MOSI
30P
/CS
1
SCK
SO
SI
/CS
VCC
/RESET
/WP
/HOLD
SO-8
SO-16
SO-28
TSOP28
CBGA
A 30P Capacitor can be connected at PIN MISO to improve ISP programming stability
XGecu 鑫工电子
FOR 25 SPI Nor Flash

ICSP supports the following chips:

1) 24 series 25 series 93 series

2) ATMEL89S51, 52, AVR ATMEGA full series, Note: In parallel for high-voltage programming, ISP for low-voltage programming

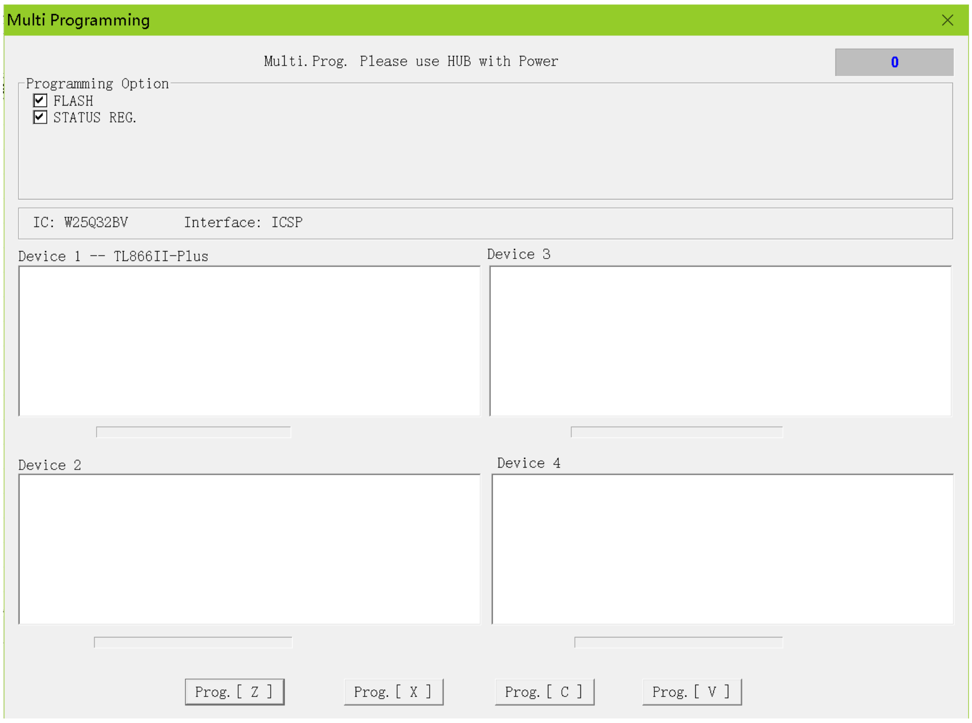
3) MICROCHIP PIC10Fxxx 12Fxxx 16Fxxx 18Fxxx full series.

4) SYNCMOS SM59Dxx SM59Rxx full series

5) SPI NAND, EMMC, VGA_HDMI

4.12 Multi Programming

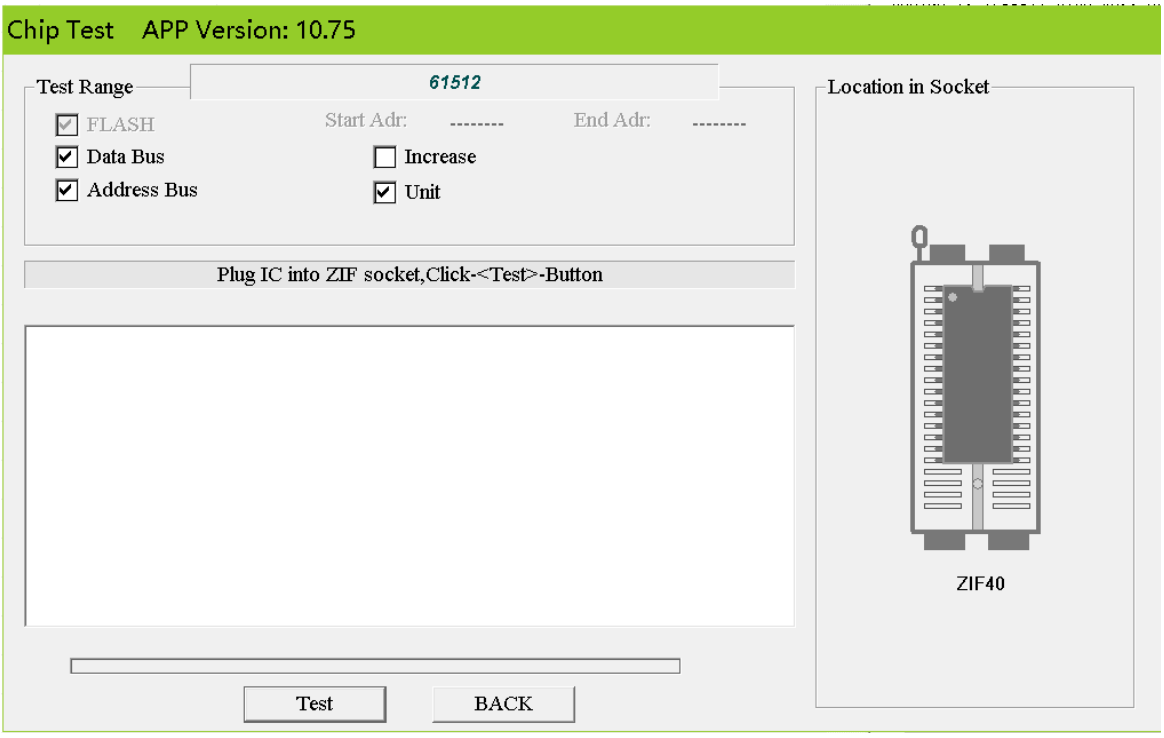
The software support simultaneously working with 4 Programmers on each computer. For USB expansion, please use a high-quality USB hub with power supply. Multi programming interface is as below. The automatic numbering function can be used normally when multi programming.



You can use the shortcut keys [Z] [X] [C] [V] to start the corresponding programmer when multi programming.

4.13 RAM Testing

After selecting the corresponding model of RAM chip, click [Device] - [Testing] button in the toolbar, the following dialog box pops up. TL866II supports, T56 do not support RAM testing temporarily, waiting for the later update.



There are four methods of RAM testing. Through these four methods, you can basically test whether the RAM cell function is normal. General programmer can only read and write the RAM cell and cannot test whether the RAM is good or bad.

For example, if two adjacent address lines are short-circuited, it is impossible to check out the problem by simply reading and writing data from the RAM cell. So only one way to test is not a complete test. Plug chip and click [Testing] button to test RAM.

5. EMMC/EMCP programming

Only T56 can support EMMC EMCP, TL866II cannot support this function.

eMMC (Embedded Multi Media Card) is a standard specification for embedded memory developed by the MMC Association, primarily for products such as cell phones or tablet computers. By integrating a controller into the package, providing a standard interface and managing flash memory, eMMC allows handset manufacturers to focus on other parts of product development and shorten the time to market.

EMCP is a higher-level memory device than EMMC, which combines EMMC and LPDDR into one package, reducing the volume and design for circuit connection.

EMMC/EMCP are available in various BGA packages, general: BGA153/BGA169/BGA162/BGA221/BGA100/BGA254/BGA529 T56 programmer can support them all well.

5.1 EMMC Memory

EMMC flash memory can have up to 8 storage areas, named as BOOT1/BOOT2/RPMB/GPP1/GPP2/GPP3/GPP4/USER. Among them, GPP1-4 are user-defined partitions, which are not configured at the factory.

- BOOT1/BOOT2 is generally used as system boot area and boot data backup partition
- RPMB is a special storage partition, data can be read always, but writing requires a 32-byte authentication key. It must be written through a complex HMAC_SHA256 authentication algorithm, in order to prevent hacking software from illegally modifying the data.
- GPP1-4 are user-definable hardware partitions, which are used in the same way as the USER partition
- USER partition, the largest user data partition
- Partition write protection, GPP1-4 and USER partitions can be written protection in groups, and the write protection modes are temporary write protection, permanent write protection, and power-on write protection. After setting permanent write protection, the protected WPG group can't be erased or modified data permanently.
- EMMC password protection function, if the password is set, the password must be unlocked before accessing the USER

area, otherwise it cannot access.

- ECSD and CSD are configuration registers. The various configurable parameters of EMMC are set in [Device config]. Great part of the registers in ECSD and CSD are read-only registers.
- CID is identification mark of the chip at the factory, which can only be read and cannot be modified.

5.2 EMMC function supported by T56

5.2.1 Analysis of the original chip

This function allows you to get a full overview of the original chip and do operations the way you want.

The content of the original chip is analyzed and the following results are displayed:

- Display chip manufacturer information and life cycle
- Visually display the usage of up to 8 partitions of the chip
- Each partition Displays whether it is write-protected, readable, rewritable or erasable
- Displays whether the whole chip is temporarily write protected or permanently write protected
- Display whether the chip is password protected or not
- Display whether the chip uses 32-bit authentication key written to the RPMB.
- Display other important parameters of ECSD CSD CID

5.2.2 Read/Program function

All functions can be done with one key

- Read CID/CSD/ECSD.
- Clear temporary write protection before programming
- Clear temporary write protection group before programming
- Erase chip before programming
- Blank check before programming
- Program 32-bit authentication key
- Check the chip password
- New partitioning of the chip (partition Config)
- Read and write to BOOT1
- Read and write to BOOT2
- Read and write to RPMB
- Read and write to GPP1
- Read and write to GPP2
- Read and write to GPP3
- Read and write to GPP4
- Read and write on user data (User)
- Setting temporary write protection group
- Setting permanent write protection group
- Setting chip password (or reset password)
- Program ECSD
- Program CSD
- Automatic verification of all the above operations after programming

5.2.3 ISP in-circuit programming

- Ultra-stable ISP in-circuit programming with line length up to 40CM, working frequency 40MHZ, reliable work

- ISP can also use 4-bit wide mode, operation speed increased to 4X faster
- See details on Section 5.9 ISP in-circuit programming example.

5.2.4 EMMC ONE-KEY-GHOST

When reading EMMC, the one-key-ghost project can be automatically generated to realize the one-key copying of the original chip content to the new chip. See details on section 5.8 one-key-ghost example.

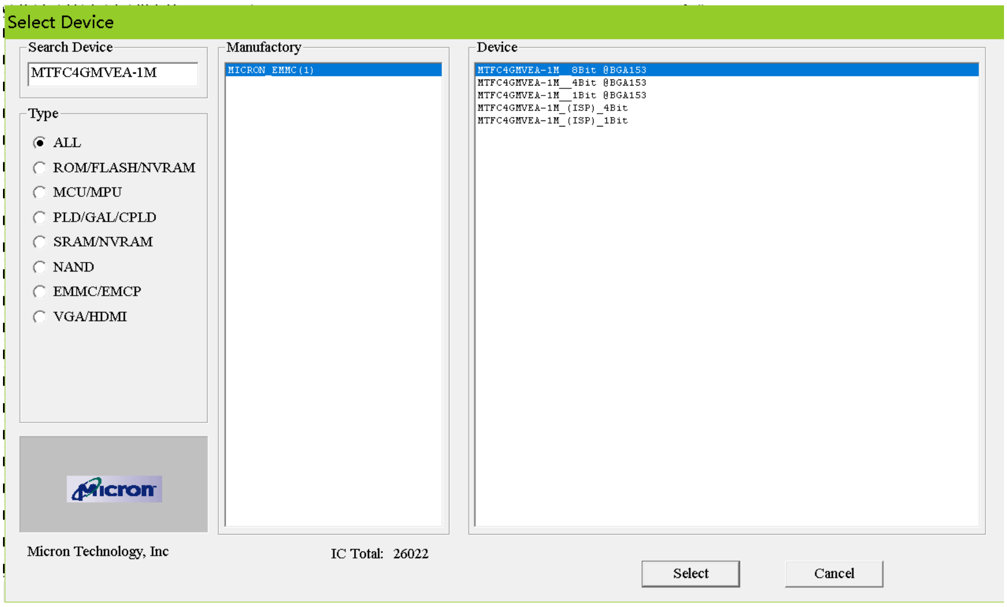
5.2.5 Other EMMC Functions

- The working frequency and working voltage of EMMC can be adjusted manually.
- Data in the user area can be read and written in blocks and multiple files
- If there is a password protection when deleting, perform forced erase
- Password can use ordinary password or SHA1 encrypted password
- The value of ECSD/CSD can be imported from the ECSD_CSD file or manually set by the user
- Data CRC error can be ignored during read operation
- Optional automatic verification after the reading operation is completed.
- There are high-definition pin diagrams and ISP wiring diagrams in the device information. The pin check and locate the BGA pin.
- EMMC has real-time accurate rate display during read and write operations, and the final total time

5.3 EMMC select IC

5.3.1 The usual method to select IC

Type in IC model (for example, MTFC4GMVEA-1M), as shown below (better to select by this method, AUTO may be not work normally on few of chips):



Select Device

Search Device: MTFC4GMVEA-1M

Type:
● ALL
○ ROM/FLASH/NVRAM
○ MCU/MPU
○ PLD/GAL/CPLD
○ SRAM/NVRAM
○ NAND
○ EMMC/EMCP
○ VGA/HDMI

Manufactory: MICRON_EMMC(1)

Device:

```
MTFC4GMVEA-1M__8Bit @BGA153
MTFC4GMVEA-1M__4Bit @BGA153
MTFC4GMVEA-1M__1Bit @BGA153
MTFC4GMVEA-1M_(ISP)_4Bit
MTFC4GMVEA-1M_(ISP)_1Bit
```

Micron Technology, Inc

IC Total: 26022

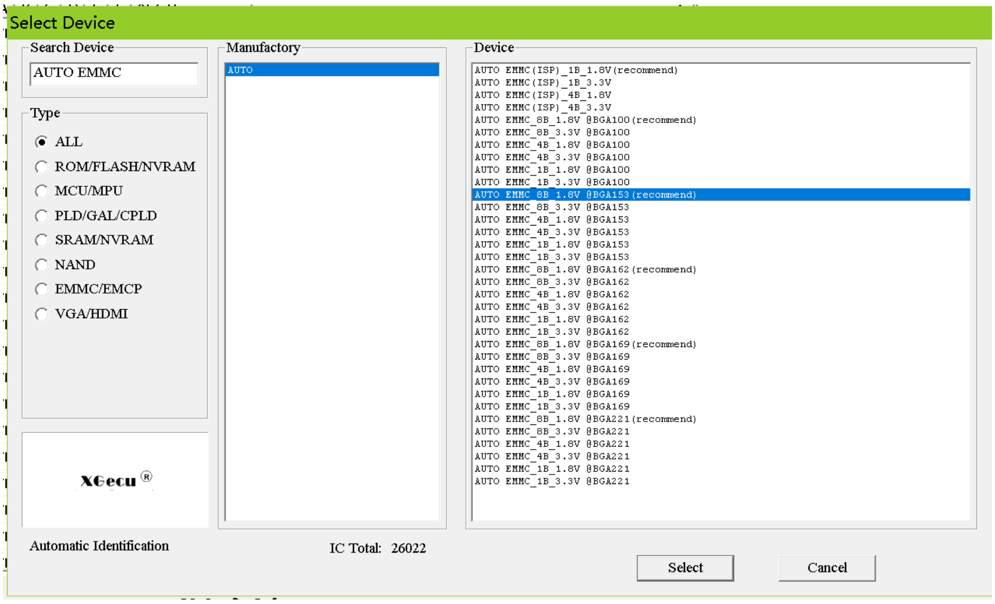
In general, there are 5 options for the same model EMMC:

- 8bit means 8 bits of data bus width is used when programming, and works at the fastest speed.
- 4bit means 4 bits of data bus width is used when programming, it is available when chip pin D7:4 is damaged.
- 1bit means 1 bit data bus width is used when programming, and can be used when chip pin D7:1 is damaged, and is the slowest.
- ISP_4Bit: 4 bits of data bus width is used when programming through ISP. At this time, the programming speed is also faster through ISP.

- ISP_1Bit: 1bit of data bus width is used when programming through ISP.

5.3.2 **Select AUTO_EMMC**

When IC model cannot be found in the list, you can type in "AUTO EMMC" to select the corresponding package as below:



Select Device

Search Device: AUTO EMMC

Type

● ALL
○ ROM/FLASH/NVRAM
○ MCU/MPU
○ PLD/GAL/CPLD
○ SRAM/NVRAM
○ NAND
○ EMMC/EMCP
○ VGA/HDMI

Manufactory: AUTO

Device:

```
AUTO EMMC(ISP)_1B_1.8V(recommend)
AUTO EMMC(ISP)_1B_3.3V
AUTO EMMC(ISP)_4B_1.8V
AUTO EMMC(ISP)_4B_3.3V
AUTO EMMC_8B_1.8V @BGA100(recommend)
AUTO EMMC_8B_3.3V @BGA100
AUTO EMMC_4B_1.8V @BGA100
AUTO EMMC_4B_3.3V @BGA100
AUTO EMMC_1B_1.8V @BGA100
AUTO EMMC_1B_3.3V @BGA100
AUTO EMMC_8B_1.8V @BGA153(recommend)
AUTO EMMC_8B_3.3V @BGA153
AUTO EMMC_4B_1.8V @BGA153
AUTO EMMC_4B_3.3V @BGA153
AUTO EMMC_1B_1.8V @BGA153
AUTO EMMC_1B_3.3V @BGA153
AUTO EMMC_8B_1.8V @BGA162(recommend)
AUTO EMMC_8B_3.3V @BGA162
AUTO EMMC_4B_1.8V @BGA162
AUTO EMMC_4B_3.3V @BGA162
AUTO EMMC_1B_1.8V @BGA162
AUTO EMMC_1B_3.3V @BGA162
AUTO EMMC_8B_1.8V @BGA169(recommend)
AUTO EMMC_8B_3.3V @BGA169
AUTO EMMC_4B_1.8V @BGA169
AUTO EMMC_4B_3.3V @BGA169
AUTO EMMC_1B_1.8V @BGA169
AUTO EMMC_1B_3.3V @BGA169
AUTO EMMC_8B_1.8V @BGA221(recommend)
AUTO EMMC_8B_3.3V @BGA221
AUTO EMMC_4B_1.8V @BGA221
AUTO EMMC_4B_3.3V @BGA221
AUTO EMMC_1B_1.8V @BGA221
AUTO EMMC_1B_3.3V @BGA221
```

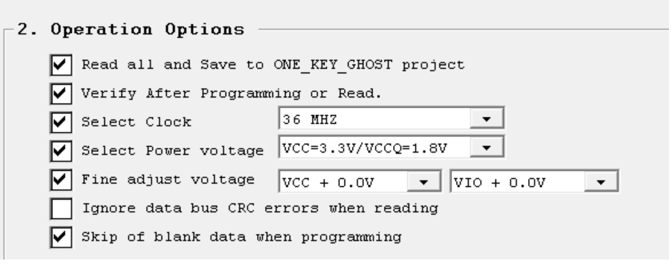
Automatic Identification

IC Total: 26022

- If you program with adapter, select IC of the corresponding package, such as BGA153:
- There are 6 options under BGA153, usually choose the first 8B_1.8V, which represents 8-bit of data bus width is used, IO voltage is 1.8V.
- If you in-circuit program via ISP, usually choose ISP_1B_1.8V (in general, the motherboard IO voltage is 1.8V). If the motherboard IO voltage is 3.3V, choose ISP_1B_3.3V.
- If choose 1.8V, cannot be programmed normally on motherboard, you can try to change to 3.3V for testing.

5.4 EMMC Operation Options

5.4.1 **Programmer Operation Options**



2. Operation Options

☑ Read all and Save to ONE_KEY_GHOST project
☑ Verify After Programming or Read.
☑ Select Clock: 36 MHZ
☑ Select Power voltage: VCC=3.3V/VCCQ=1.8V
☑ Fine adjust voltage: VCC + 0.0V, VIO + 0.0V
☐ Ignore data bus CRC errors when reading
☑ Skip of blank data when programming

- Read all areas and save one-key-ghost project
  while reading, automatically analyze the chip, read all the data. This option will ignore the next (read, program, erase) option in the part of the selection, will automatically select according to the chip content. After reading, in the data folder a project file EMMC_GHOST.MPJ will be automatically generated. If the user wants to clone a chip, just open this ghost project file and directly program it to the new chip.
- Automatically verify after programming or read
  after reading or programming, the data will be automatically verified once.
- EMMC Select Clock
  Select the clock frequency, 36MHZ is default, you can select 40MHZ or 50MHZ while mass production. If the reading of data is unstable, you can also reduce the clock frequency for testing. Maximum 40MHZ via ISP.
- EMMC select Power voltage

Select the IC power voltage, a great part of the chip works at VCC=3.3V/VCCQ=1.8V, with extremely high stability.

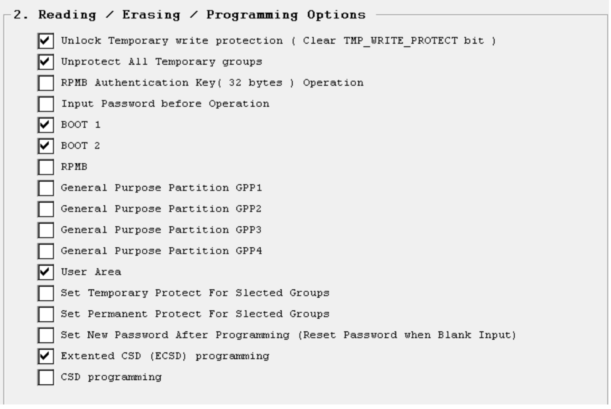
- EMMC fine adjust voltage

  If the reading is unstable, can fine-tuning the voltage. It is possible to get a better stable effect. Users can do various tastings of voltage fine-tuning.
- Ignore data bus CRC errors when reading

  CRC verify error will be ignored when reading, this option is not recommended.
- Skip of blank data when programming

  during the verifying or programming, the blank data is skipped in order to increase the programming speed.

5.4.2 Read, Program, Erase options



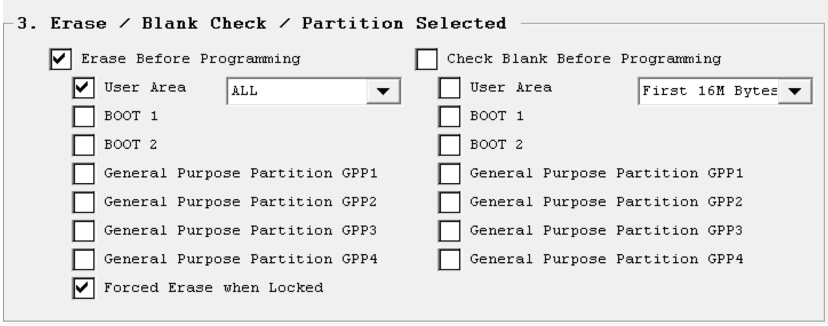
2. Reading / Erasing / Programming Options

☑ Unlock Temporary write protection (Clear TMP_WRITE_PROTECT bit)
☑ Unprotect All Temporary groups
□ RPMB Authentication Key(32 bytes) Operation
□ Input Password before Operation
☑ BOOT 1
☑ BOOT 2
□ RPMB
□ General Purpose Partition GPP1
□ General Purpose Partition GPP2
□ General Purpose Partition GPP3
□ General Purpose Partition GPP4
☑ User Area
□ Set Temporary Protect For Slected Groups
□ Set Permanent Protect For Slected Groups
□ Set New Password After Programming (Reset Password when Blank Input)
☑ Extented CSD (ECSD) programming
□ CSD programming

Select the chip area to read or programmiing. If an item is not selected, no operation will be performed for this item.

The key option RPMB: when programming RPMB, the option [RPMB Authentication Key Operation] must be checked and the correct 32 bytes authentication key must be loaded in the file menu before programming. After the authentication key is written, the chip will use this authentication key permanently and cannot be rewritten with a new authentication key.

5.4.3 Erase, Blank Check option



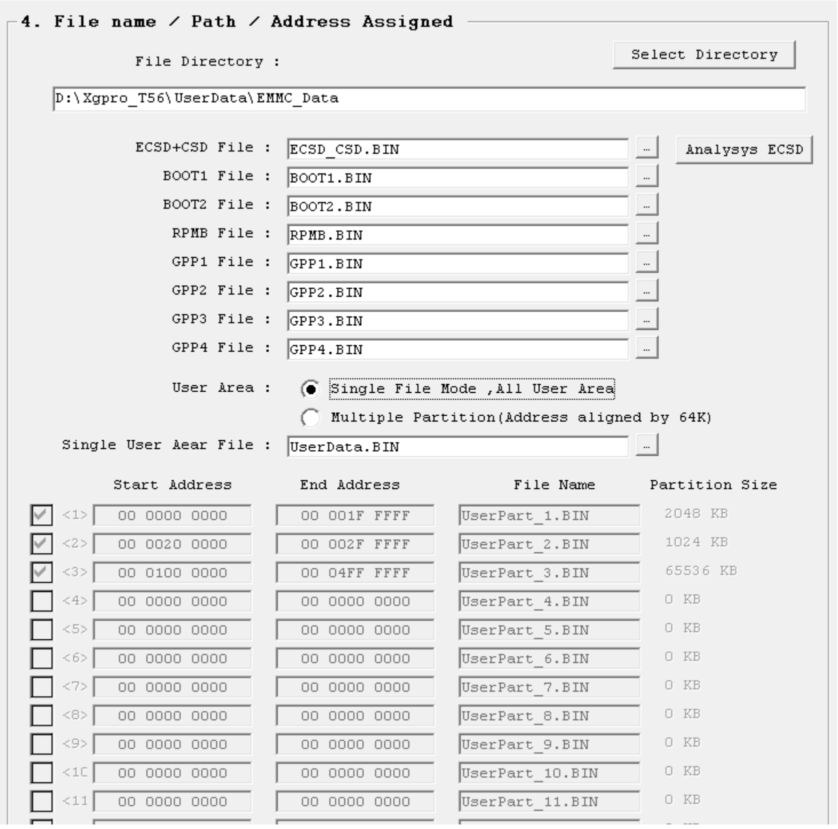
3. Erase / Blank Check / Partition Selected

☑ Erase Before Programming
☑ User Area ALL
□ BOOT 1
□ BOOT 2
□ General Purpose Partition GPP1
□ General Purpose Partition GPP2
□ General Purpose Partition GPP3
□ General Purpose Partition GPP4
☑ Forced Erase when Locked

□ Check Blank Before Programming
□ User Area First 16M Bytes
□ BOOT 1
□ BOOT 2
□ General Purpose Partition GPP1
□ General Purpose Partition GPP2
□ General Purpose Partition GPP3
□ General Purpose Partition GPP4

The erase and blank check operations performed before programming, generally do not need to be changed, just use the default settings. If the chip is new while mass production, the erase can also be unchecked.

5.4.4 File path and file name settings



4. File name / Path / Address Assigned

File Directory : [Select Directory]

D:\Xgpro_T56\UserData\EMMC_Data

ECSD+CSD File : ECSD_CSD.BIN [...] [Analysys ECSD]

BOOT1 File : BOOT1.BIN

BOOT2 File : BOOT2.BIN

RPMB File : RPMB.BIN

GPP1 File : GPP1.BIN

GPP2 File : GPP2.BIN

GPP3 File : GPP3.BIN

GPP4 File : GPP4.BIN

User Area : ● Single File Mode ,All User Area

○ Multiple Partition(Address aligned by 64K)

Single User Aear File : UserData.BIN

		Start Address	End Address	File Name	Partition Size
☑	<1>	00 0000 0000	00 001F FFFF	UserPart_1.BIN	2048 KB
☑	<2>	00 0020 0000	00 002F FFFF	UserPart_2.BIN	1024 KB
☑	<3>	00 0100 0000	00 04FF FFFF	UserPart_3.BIN	65536 KB
□	<4>	00 0000 0000	00 0000 0000	UserPart_4.BIN	0 KB
□	<5>	00 0000 0000	00 0000 0000	UserPart_5.BIN	0 KB
□	<6>	00 0000 0000	00 0000 0000	UserPart_6.BIN	0 KB
□	<7>	00 0000 0000	00 0000 0000	UserPart_7.BIN	0 KB
□	<8>	00 0000 0000	00 0000 0000	UserPart_8.BIN	0 KB
□	<9>	00 0000 0000	00 0000 0000	UserPart_9.BIN	0 KB
□	<1C	00 0000 0000	00 0000 0000	UserPart_10.BIN	0 KB
□	<11	00 0000 0000	00 0000 0000	UserPart_11.BIN	0 KB

- Select the data folder

  set a new folder before reading or programming, the folder name should preferably be device name + IC model for future use, all the data read will be stored under this folder. Including the automatically generated one-key-ghost project files. Of course, the folder name can be changed after reading. When programming, just select the data folder and make sure that all the data files be located under this folder.

- File name, read generally do not need to change the file name.

  If new product development, mass production, select the data file of the corresponding area.

  Note: All files must be set in the same folder.

- Analyze ECSD button

  Check out the information related to ECSD file under the folder.

- If the development of production

  ECSD file may not exist and user can set the relevant configuration in [Device Config].

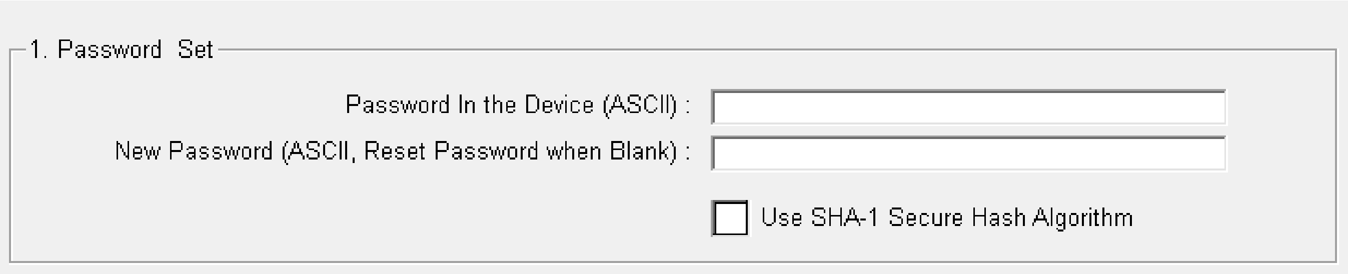
- Blocks mode

  it can be sliced up to 16 blocks, block address must be 64K aligned, 1K=1024 bytes

  Users can write or read the USER Area in blocks, which is convenient for flexible applications when programming in bulk. The blocks mode also has a special function: if the unstable chip has nearly reached the end of its useful life and you want to read the data inside, you can read in blocks. After reading of the all data in the good blocks is completed at one time, then the bad blocks can be read separately for several times (while reading, disable the option "Read all and Save to ONE_KEY-GHOST project").

5.5 EMMC Device Configuration

5.5.1 **EMMC Password Setting**



1. Password Set

Password In the Device (ASCII) :

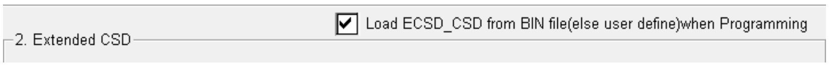
New Password (ASCII, Reset Password when Blank) :

□ Use SHA-1 Secure Hash Algorithm

EMMC can use password protection, that only protects the USER Area, BOOTx/GPPx/RPMB partition, cannot be protected by password.

T56 can set normal password and SHA1 password, please refer to EMMC standard document for details.

5.5.2 ECSD Setting



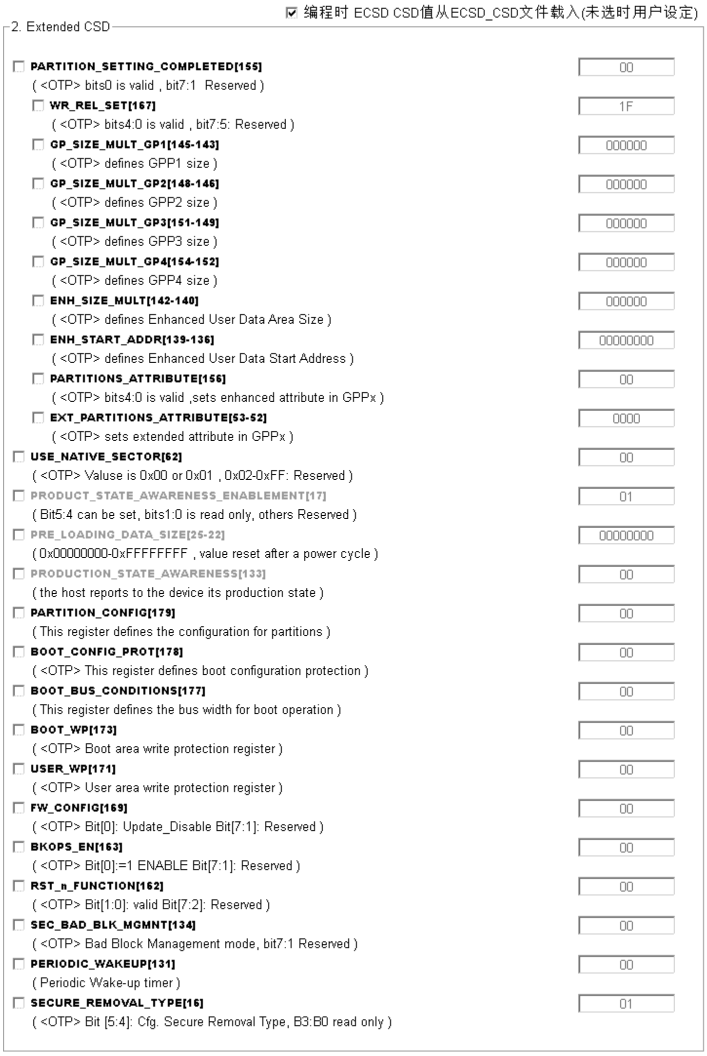
2. Extended CSD

☑ Load ECSD_CSD from BIN file(else user define)when Programming

[ECSD CSD values are loaded from the file when programming] button

When checked, all fields of ECSD and CSD are loaded from the file and the set values in the interface are invalid.

set fields by user while unchecked.



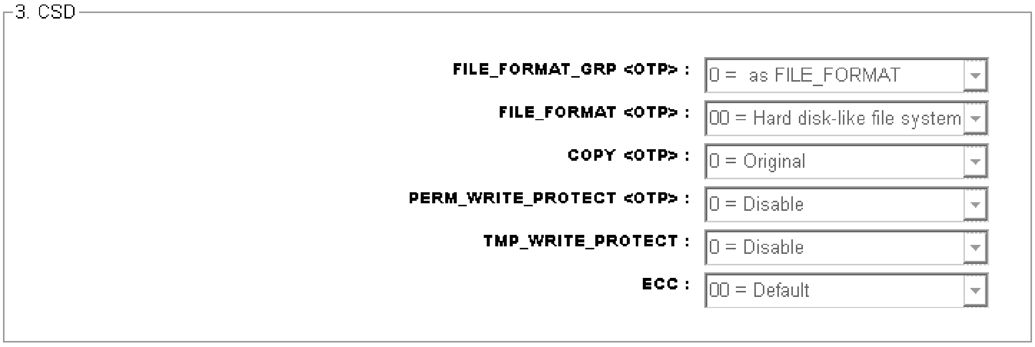
☑ 编程时 ECSD CSD值从ECSD_CSD文件载入(未选时用户设定)

2. Extended CSD

□ **PARTITION_SETTING_COMPLETED[155]** (<OTP> bits0 is valid , bit7:1 Reserved)	00
□ **WR_REL_SET[167]** (<OTP> bits4:0 is valid , bit7:5: Reserved)	1F
□ **GP_SIZE_MULT_GP1[145-143]** (<OTP> defines GPP1 size)	000000
□ **GP_SIZE_MULT_GP2[148-146]** (<OTP> defines GPP2 size)	000000
□ **GP_SIZE_MULT_GP3[151-149]** (<OTP> defines GPP3 size)	000000
□ **GP_SIZE_MULT_GP4[154-152]** (<OTP> defines GPP4 size)	000000
□ **ENH_SIZE_MULT[142-140]** (<OTP> defines Enhanced User Data Area Size)	000000
□ **ENH_START_ADDR[139-136]** (<OTP> defines Enhanced User Data Start Address)	00000000
□ **PARTITIONS_ATTRIBUTE[156]** (<OTP> bits4:0 is valid ,sets enhanced attribute in GPPx)	00
□ **EXT_PARTITIONS_ATTRIBUTE[53-52]** (<OTP> sets extended attribute in GPPx)	0000
□ **USE_NATIVE_SECTOR[62]** (<OTP> Valuse is 0x00 or 0x01 , 0x02-0xFF: Reserved)	00
□ **PRODUCT_STATE_AWARENESS_ENABLEMENT[17]** (Bit5:4 can be set, bits1:0 is read only, others Reserved)	01
□ **PRE_LOADING_DATA_SIZE[25-22]** (0x00000000-0xFFFFFFFF , value reset after a power cycle)	00000000
□ **PRODUCTION_STATE_AWARENESS[133]** (the host reports to the device its production state)	00
□ **PARTITION_CONFIG[179]** (This register defines the configuration for partitions)	00
□ **BOOT_CONFIG_PROT[178]** (<OTP> This register defines boot configuration protection)	00
□ **BOOT_BUS_CONDITIONS[177]** (This register defines the bus width for boot operation)	00
□ **BOOT_WP[173]** (<OTP> Boot area write protection register)	00
□ **USER_WP[171]** (<OTP> User area write protection register)	00
□ **FW_CONFIG[169]** (<OTP> Bit[0]: Update_Disable Bit[7:1]: Reserved)	00
□ **BKOPS_EN[163]** (<OTP> Bit[0]:=1 ENABLE Bit[7:1]: Reserved)	00
□ **RST_n_FUNCTION[162]** (<OTP> Bit[1:0]: valid Bit[7:2]: Reserved)	00
□ **SEC_BAD_BLK_MGMNT[134]** (<OTP> Bad Block Management mode, bit7:1 Reserved)	00
□ **PERIODIC_WAKEUP[131]** (Periodic Wake-up timer)	00
□ **SECURE_REMOVAL_TYPE[16]** (<OTP> Bit [5:4]: Cfg. Secure Removal Type, B3:B0 read only)	01

- All ECSD register fields options, only when checked, the chip will be programmed and verified. Unchecked registers will not be programmed and verified.
- The first major option **[PARTITION SETTING COMPELTED 155],** when checked, the value must be 01, and it's subparts must be all checked, and the subparts must meet the EMMC standard, otherwise they cannot be written. This major option is to set the EMMC partition and partition attributes, the chip is not allowed to repeatedly set the partition size and attributes.
- All registers marked with OTP are written at once, so please set them carefully
- When **[USE_NATIVE_SECTOR 62]** button is checked, the value must be 01, otherwise it is the error.
- All other registers can be checked, or only the non-00 registers be checked
- Note: The above options will be checked automatically when the original chip is read with the one- key-ghost project is generated automatically.

5.5.3 CSD Setting



3. CSD

FILE_FORMAT_GRP <OTP> : 0 = as FILE_FORMAT

FILE_FORMAT <OTP> : 00 = Hard disk-like file system

COPY <OTP> : 0 = Original

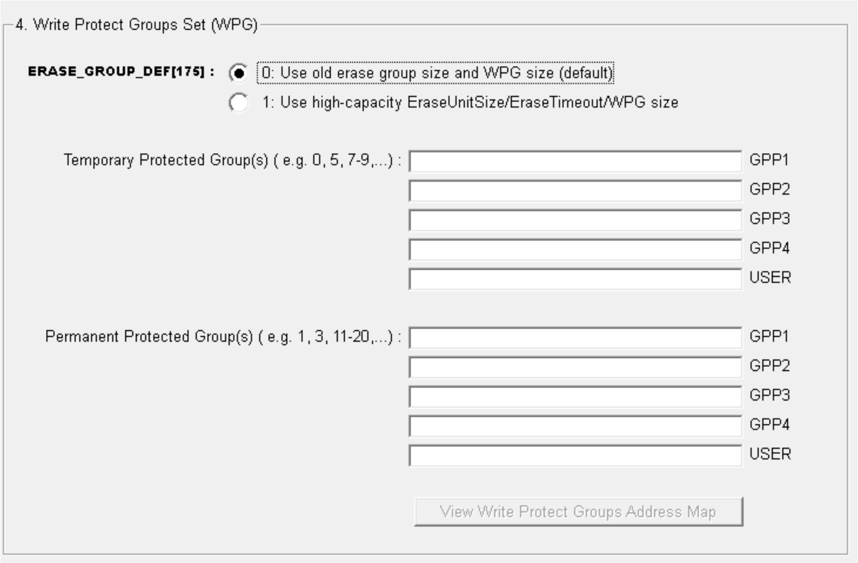
PERM_WRITE_PROTECT <OTP> : 0 = Disable

TMP_WRITE_PROTECT : 0 = Disable

ECC : 00 = Default

- If any option in CSD is non-zero, check the option **[CSD programming]** in EMMC Operation Options (2.Reading/ Erasing/ Programming Options)
- When auto-generating one-key-ghost project, the software will check the all ECSD/CSD options automatically.

5.5.4 Write Protect Groups (WPG)



4. Write Protect Groups Set (WPG)

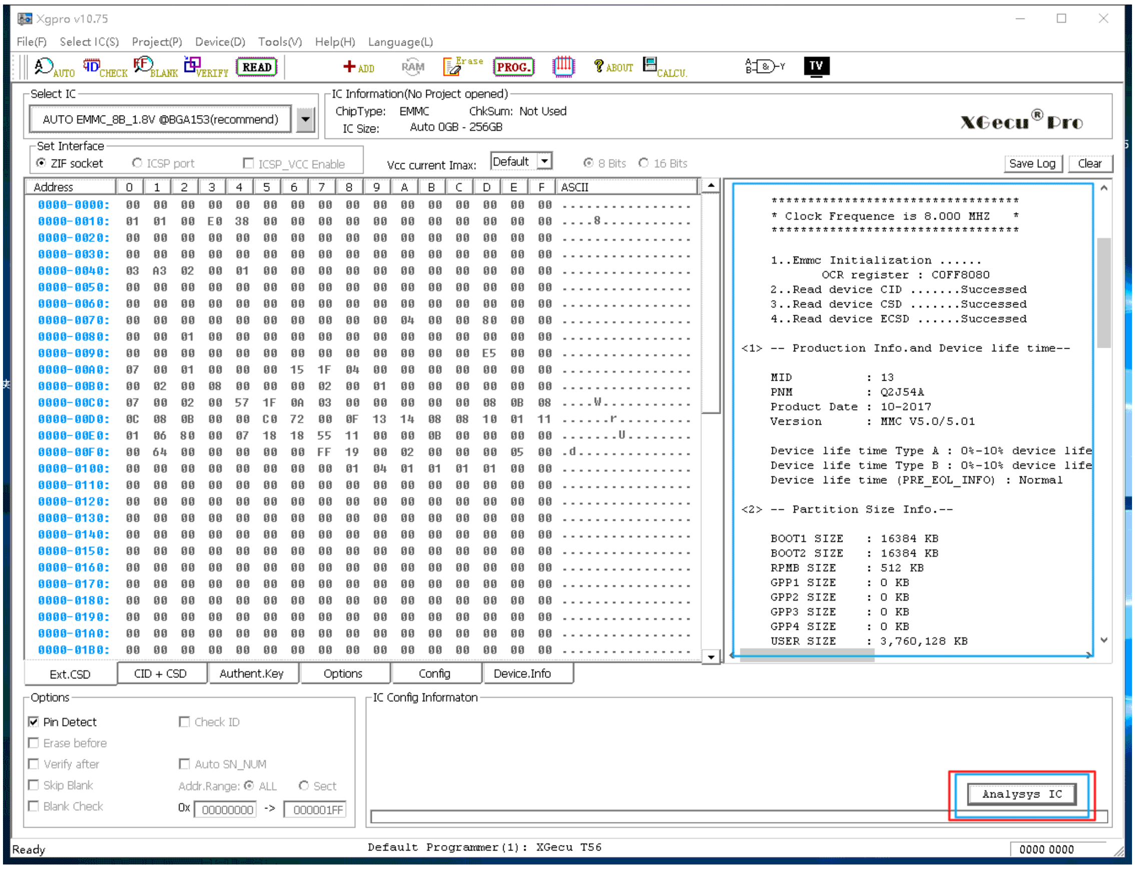
ERASE_GROUP_DEF[175] : ● 0: Use old erase group size and WPG size (default)
○ 1: Use high-capacity EraseUnitSize/EraseTimeout/WPG size

Temporary Protected Group(s) (e.g. 0, 5, 7-9,...) : GPP1, GPP2, GPP3, GPP4, USER

Permanent Protected Group(s) (e.g. 1, 3, 11-20,...) : GPP1, GPP2, GPP3, GPP4, USER

- Write Protect is for the part of chip data, the protection type and protection group are set according to the user's requirements.
- While Analysis of the original chip function: you can check which groups of the original chip was protection set. If you need to set the same write protection group as the original chip, please type in manually.
- When the ECSD file has been loaded, you can check the specific address corresponding to the write protection group.

5.6 EMMC Analysis


Xgpro v10.75
File(F) Select IC(S) Project(P) Device(D) Tools(V) Help(H) Language(L)
AUTO CHECK BLANK VERIFY READ ADD RAM Erase PROG. ABOUT CALCU. TV
XGecu® Pro
Save Log
Clear
Ext.CSD
CID + CSD
Authent.Key
Options
Config
Device.Info
Analysys IC

Select IC: AUTO EMMC_8B_1.8V @BGA153(recommend)

IC Information(No Project opened)
ChipType: EMMC ChkSum: Not Used
IC Size: Auto 0GB - 256GB

Set Interface: ◉ ZIF socket ○ ICSP port □ ICSP_VCC Enable

Vcc current Imax: Default ◉ 8 Bits ○ 16 Bits

```
Address     0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  ASCII
0000-0000: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-0010: 01 01 00 E0 38 00 00 00 00 00 00 00 00 00 00 00 ....8...........
0000-0020: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-0030: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-0040: 03 A3 02 00 01 00 00 00 00 00 00 00 00 00 00 00 ................
0000-0050: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-0060: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-0070: 00 00 00 00 00 00 00 00 00 00 04 00 00 80 00 00 ................
0000-0080: 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-0090: 00 00 00 00 00 00 00 00 00 00 00 00 00 E5 00 00 ................
0000-00A0: 07 00 01 00 00 00 15 1F 04 00 00 00 00 00 00 00 ................
0000-00B0: 00 02 00 08 00 00 00 02 00 01 00 00 00 00 00 00 ................
0000-00C0: 07 00 02 00 57 1F 0A 03 00 00 00 00 00 08 0B 08 ....W...........
0000-00D0: 0C 08 0B 00 00 C0 72 00 0F 13 14 08 08 10 01 11 ......r.........
0000-00E0: 01 06 80 00 07 18 18 55 11 00 00 0B 00 00 00 00 .......U........
0000-00F0: 00 64 00 00 00 00 00 FF 19 00 02 00 00 00 05 00 .d..............
0000-0100: 00 00 00 00 00 00 00 00 01 04 01 01 01 01 00 00 ................
0000-0110: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-0120: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-0130: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-0140: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-0150: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-0160: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-0170: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-0180: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-0190: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-01A0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-01B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
```

```
    ***********************************
    * Clock Frequence is 8.000 MHZ    *
    ***********************************

    1..Emmc Initialization ......
           OCR register : C0FF8080
    2..Read device CID .......Successed
    3..Read device CSD .......Successed
    4..Read device ECSD ......Successed

<1> -- Production Info.and Device life time--

    MID          : 13
    PNM          : Q2J54A
    Product Date : 10-2017
    Version      : MMC V5.0/5.01

    Device life time Type A : 0%-10% device life
    Device life time Type B : 0%-10% device life
    Device life time (PRE_EOL_INFO) : Normal

<2> -- Partition Size Info.--

    BOOT1 SIZE   : 16384 KB
    BOOT2 SIZE   : 16384 KB
    RPMB SIZE    : 512 KB
    GPP1 SIZE    : 0 KB
    GPP2 SIZE    : 0 KB
    GPP3 SIZE    : 0 KB
    GPP4 SIZE    : 0 KB
    USER SIZE    : 3,760,128 KB
```

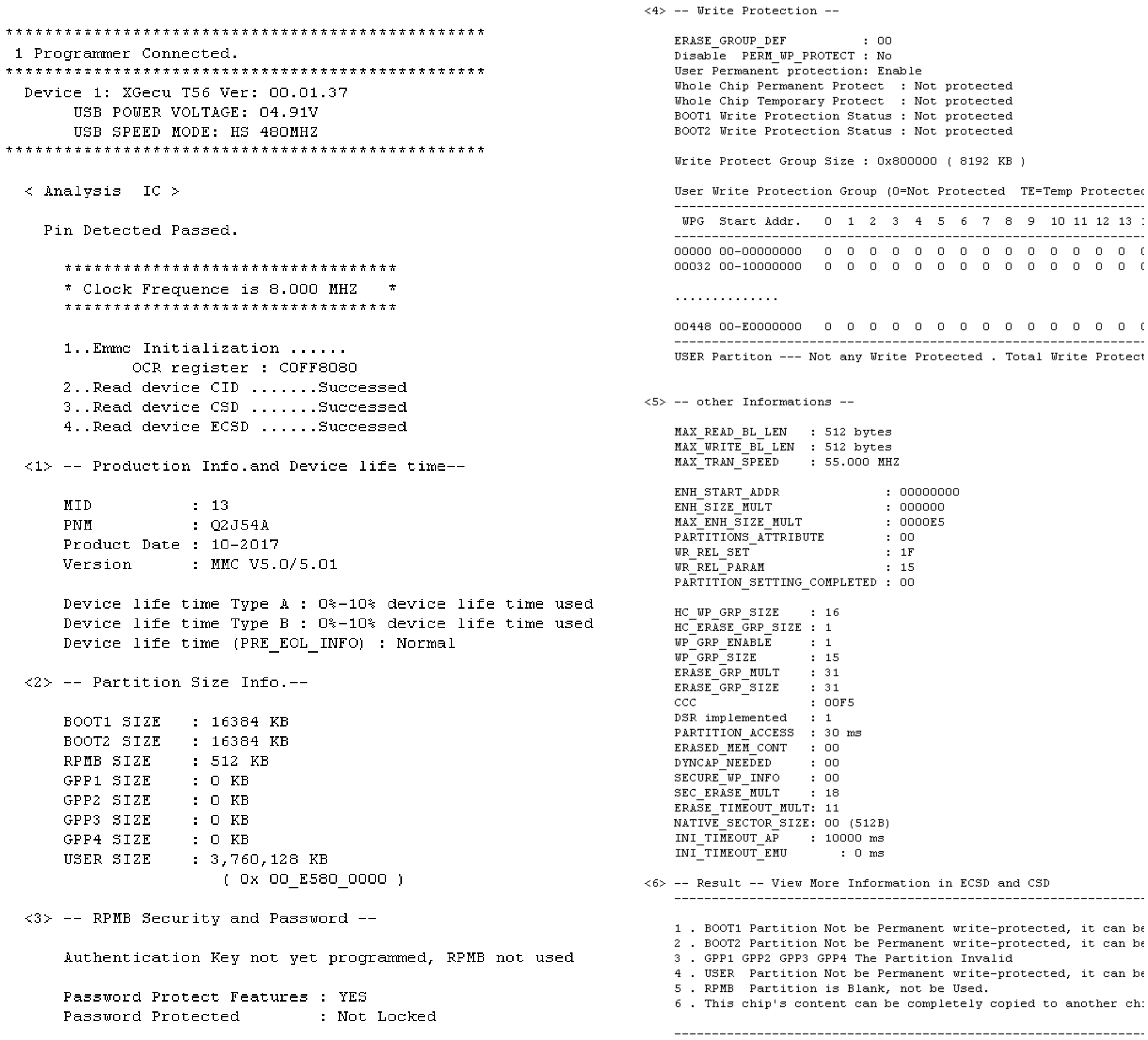
Options
☑ Pin Detect □ Check ID
□ Erase before
□ Verify after □ Auto SN_NUM
□ Skip Blank Addr.Range: ◉ ALL ○ Sect
□ Blank Check 0x 00000000 -> 000001FF

IC Config Informaton

Ready Default Programmer(1): XGecu T56 0000 0000

- Analysis of the original chip is carried out at a lower clock frequency of 8MHZ. You can test whether the connection is normal: If not, try to change the working voltage, or fine-tune the working voltage or select a different bus width for several tests. In that way, for the chips with unstable performance, may be you can get the correct result.
- Connect the adapter or ISP, click button【**Analysis IC**】
- Click button **[SAVE LOG]** to save the analysis result
- From the result, you can get the use condition of the chip
    - Display information about the IC's manufacturing
    - Visually display IC use condition of up to 8 partitions
    - On each partition display whether it is write-protected, readable, rewritable or erasable
    - Display whether the entire chip is temporarily write-protected or permanently write-protected
    - Display if the IC is password protected
    - Display whether the IC uses 32-bit authentication key to write for RPMB.
    - Display other important parameters of ECSD CSD CID

Example: **The IC analysis results are shown below**:



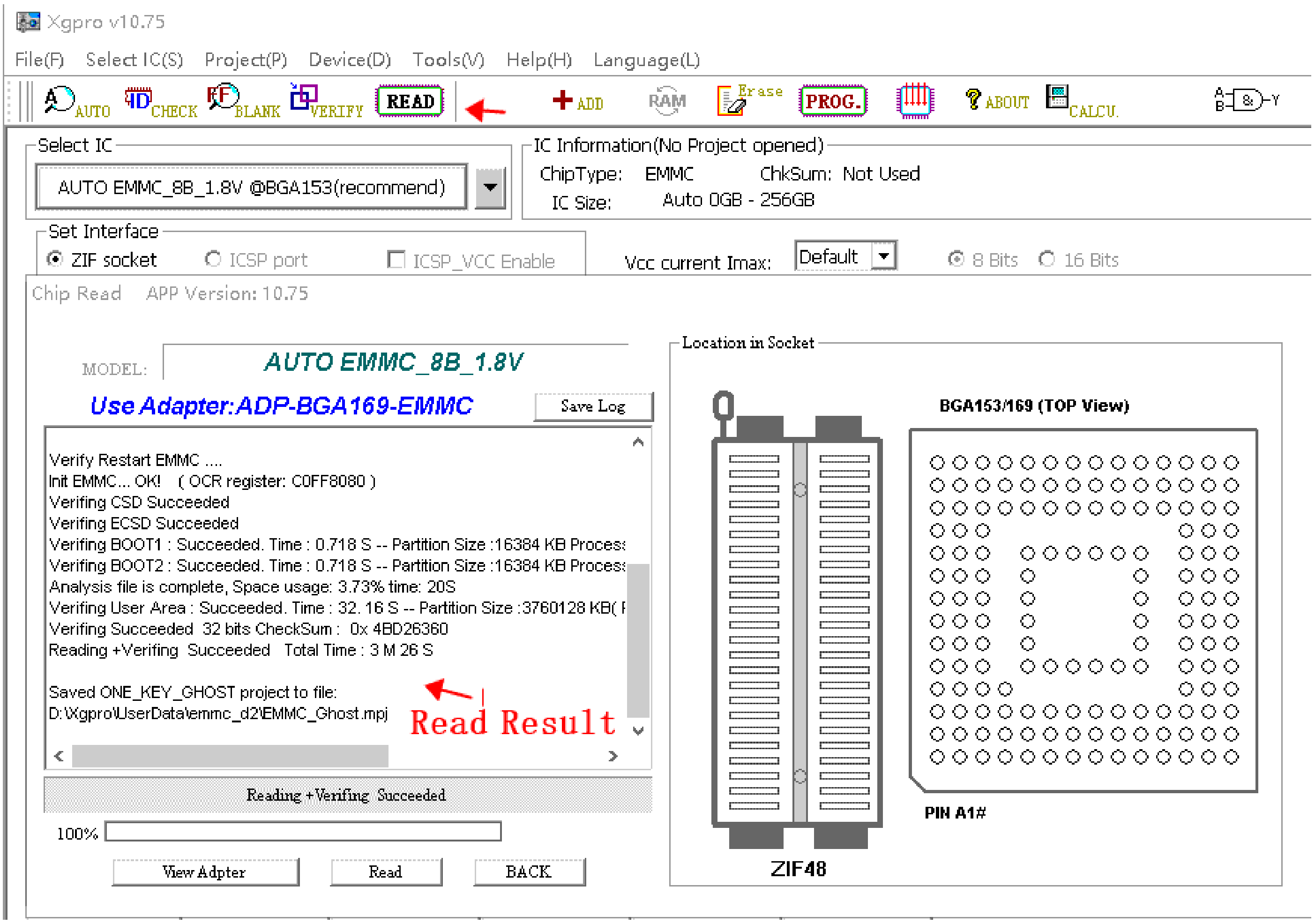
```
*************************************************
 1 Programmer Connected.
*************************************************
  Device 1: XGecu T56 Ver: 00.01.37
       USB POWER VOLTAGE: 04.91V
       USB SPEED MODE: HS 480MHZ
*************************************************

  < Analysis  IC >

    Pin Detected Passed.

      **********************************
      * Clock Frequence is 8.000 MHZ   *
      **********************************

      1..Emmc Initialization ......
             OCR register : COFF8080
      2..Read device CID .......Successed
      3..Read device CSD .......Successed
      4..Read device ECSD ......Successed

  <1> -- Production Info.and Device life time--

      MID          : 13
      PNM          : Q2J54A
      Product Date : 10-2017
      Version      : MMC V5.0/5.01

      Device life time Type A : 0%-10% device life time used
      Device life time Type B : 0%-10% device life time used
      Device life time (PRE_EOL_INFO) : Normal

  <2> -- Partition Size Info.--

      BOOT1 SIZE   : 16384 KB
      BOOT2 SIZE   : 16384 KB
      RPMB SIZE    : 512 KB
      GPP1 SIZE    : 0 KB
      GPP2 SIZE    : 0 KB
      GPP3 SIZE    : 0 KB
      GPP4 SIZE    : 0 KB
      USER SIZE    : 3,760,128 KB
                      ( 0x 00_E580_0000 )

  <3> -- RPMB Security and Password --

      Authentication Key not yet programmed, RPMB not used

      Password Protect Features : YES
      Password Protected        : Not Locked
```

```
<4> -- Write Protection --

    ERASE_GROUP_DEF          : 00
    Disable  PERM_WP_PROTECT : No
    User Permanent protection: Enable
    Whole Chip Permanent Protect  : Not protected
    Whole Chip Temporary Protect  : Not protected
    BOOT1 Write Protection Status : Not protected
    BOOT2 Write Protection Status : Not protected

    Write Protect Group Size : 0x800000 ( 8192 KB )

    User Write Protection Group (0=Not Protected  TE=Temp Protecte
    ---------------------------------------------------------------
     WPG  Start Addr.   0  1  2  3  4  5  6  7  8  9  10 11 12 13
    ---------------------------------------------------------------
    00000 00-00000000   0  0  0  0  0  0  0  0  0  0  0  0  0  0
    00032 00-10000000   0  0  0  0  0  0  0  0  0  0  0  0  0  0

    ..............

    00448 00-E0000000   0  0  0  0  0  0  0  0  0  0  0  0  0  0
    ---------------------------------------------------------------
    USER Partiton --- Not any Write Protected . Total Write Protect


<5> -- other Informations --

    MAX_READ_BL_LEN   : 512 bytes
    MAX_WRITE_BL_LEN  : 512 bytes
    MAX_TRAN_SPEED    : 55.000 MHZ

    ENH_START_ADDR              : 00000000
    ENH_SIZE_MULT               : 000000
    MAX_ENH_SIZE_MULT           : 0000E5
    PARTITIONS_ATTRIBUTE        : 00
    WR_REL_SET                  : 1F
    WR_REL_PARAM                : 15
    PARTITION_SETTING_COMPLETED : 00

    HC_WP_GRP_SIZE    : 16
    HC_ERASE_GRP_SIZE : 1
    WP_GRP_ENABLE     : 1
    WP_GRP_SIZE       : 15
    ERASE_GRP_MULT    : 31
    ERASE_GRP_SIZE    : 31
    CCC               : 00F5
    DSR implemented   : 1
    PARTITION_ACCESS  : 30 ms
    ERASED_MEM_CONT   : 00
    DYNCAP_NEEDED     : 00
    SECURE_WP_INFO    : 00
    SEC_ERASE_MULT    : 18
    ERASE_TIMEOUT_MULT: 11
    NATIVE_SECTOR_SIZE: 00 (512B)
    INI_TIMEOUT_AP    : 10000 ms
    INI_TIMEOUT_EMU       : 0 ms

<6> -- Result -- View More Information in ECSD and CSD
    ---------------------------------------------------------------

    1 . BOOT1 Partition Not be Permanent write-protected, it can be
    2 . BOOT2 Partition Not be Permanent write-protected, it can be
    3 . GPP1 GPP2 GPP3 GPP4 The Partition Invalid
    4 . USER  Partition Not be Permanent write-protected, it can be
    5 . RPMB  Partition is Blank, not be Used.
    6 . This chip's content can be completely copied to another ch

    ---------------------------------------------------------------
```

5.7 Read EMMC

- Select IC (example: JY001_8bits, this is a 4GB IC)
- Set the folder for data storage in [Operation Options] (EMMC_DATA, you can set it arbitrarily)
- Don't need to change any other settings, just click **[Read]** button to start reading.

Reading results as below:



Select IC: AUTO EMMC_8B_1.8V @BGA153(recommend)

IC Information(No Project opened)
ChipType: EMMC ChkSum: Not Used
IC Size: Auto 0GB - 256GB

Set Interface: ● ZIF socket ○ ICSP port □ ICSP_VCC Enable Vcc current Imax: Default ● 8 Bits ○ 16 Bits

Chip Read APP Version: 10.75

MODEL: AUTO EMMC_8B_1.8V

Use Adapter:ADP-BGA169-EMMC

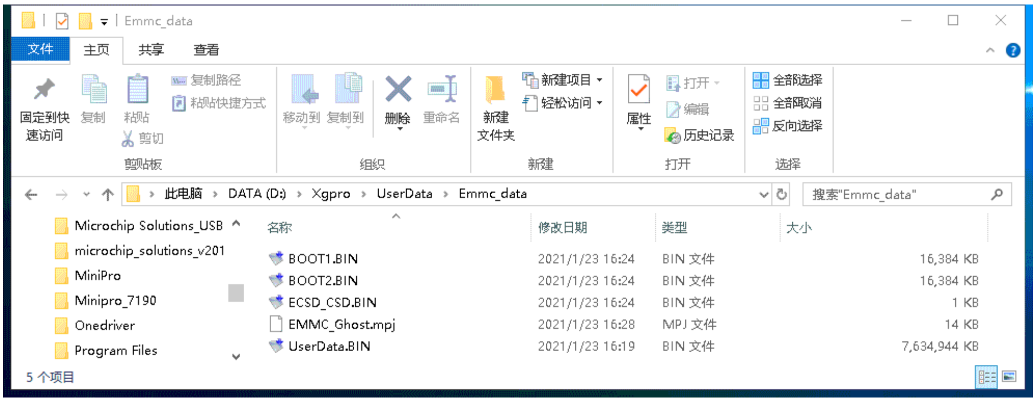
```
Verify Restart EMMC ....
Init EMMC... OK!    ( OCR register: C0FF8080 )
Verifing CSD Succeeded
Verifing ECSD Succeeded
Verifing BOOT1 : Succeeded. Time : 0.718 S -- Partition Size :16384 KB Proces:
Verifing BOOT2 : Succeeded. Time : 0.718 S -- Partition Size :16384 KB Proces:
Analysis file is complete, Space usage: 3.73% time: 20S
Verifing User Area : Succeeded. Time : 32. 16 S -- Partition Size :3760128 KB( F
Verifing Succeeded  32 bits CheckSum :  0x 4BD26360
Reading +Verifing  Succeeded   Total Time : 3 M 26 S

Saved ONE_KEY_GHOST project to file:
D:\Xgpro\UserData\emmc_d2\EMMC_Ghost.mpj
```

Reading +Verifing Succeeded

100%

- 4GB IC, the total time to "Read + Verify" is 4 minutes 28 seconds, this is a more typical time for actual operation for your reference. Generally, the IC of more larger capacity will get the faster reading and writting speed.
- After reading is finished, an EMMC_GHOST.mpj project file is automatically generated as shown above, just open this project when programming.

After reading, the contents of the folder are as below: at least 5 files will be generated, one of which is a one-key-ghost project file. If IC uses RPMB or GPPx, there will be more files.



此电脑 › DATA (D:) › Xgpro › UserData › Emmc_data 搜索"Emmc_data"

名称	修改日期	类型	大小
BOOT1.BIN	2021/1/23 16:24	BIN 文件	16,384 KB
BOOT2.BIN	2021/1/23 16:24	BIN 文件	16,384 KB
ECSD_CSD.BIN	2021/1/23 16:24	BIN 文件	1 KB
EMMC_Ghost.mpj	2021/1/23 16:28	MPJ 文件	14 KB
UserData.BIN	2021/1/23 16:19	BIN 文件	7,634,944 KB

5 个项目

Note: Before reading, it will check the remaining capacity of the disk, if the capacity is not enough, select the data folder in the operation options and save it to another disk

5.8 Program EMMC : one-key-ghost example

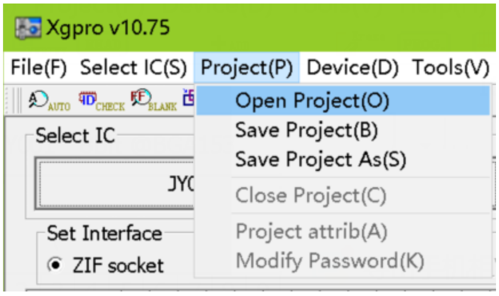
One-key copy of new IC read in the previous section

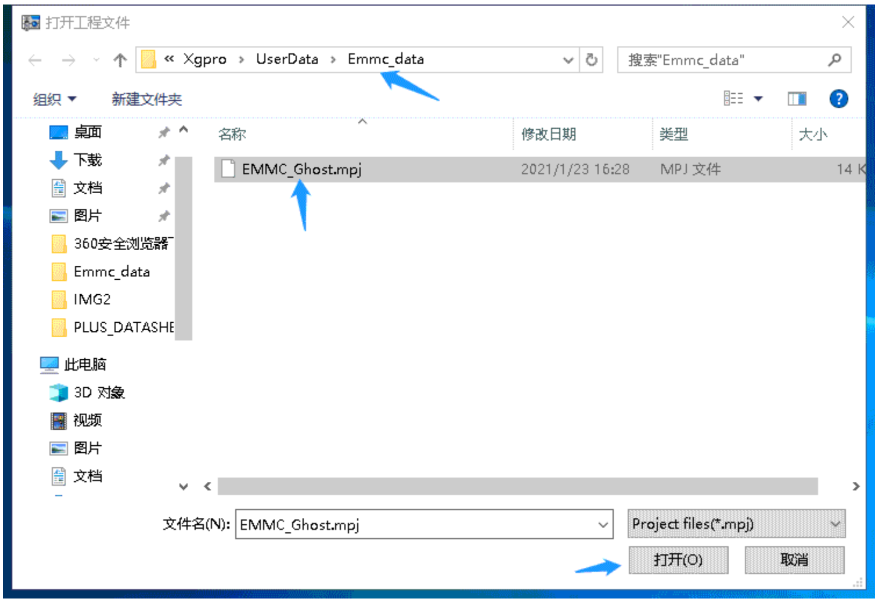
Just open the project file [EMMC_GHOST.mpj] previously read and copy it with one key.

5.8.1 Select IC (for example: JY001_8bits)

How to select IC correctly, you can refer to the previous contents.

5.8.2 Click [Open Project] in Project Menu





名称	修改日期	类型	大小
EMMC_Ghost.mpj	2021/1/23 16:28	MPJ 文件	14 K

文件名(N): EMMC_Ghost.mpj

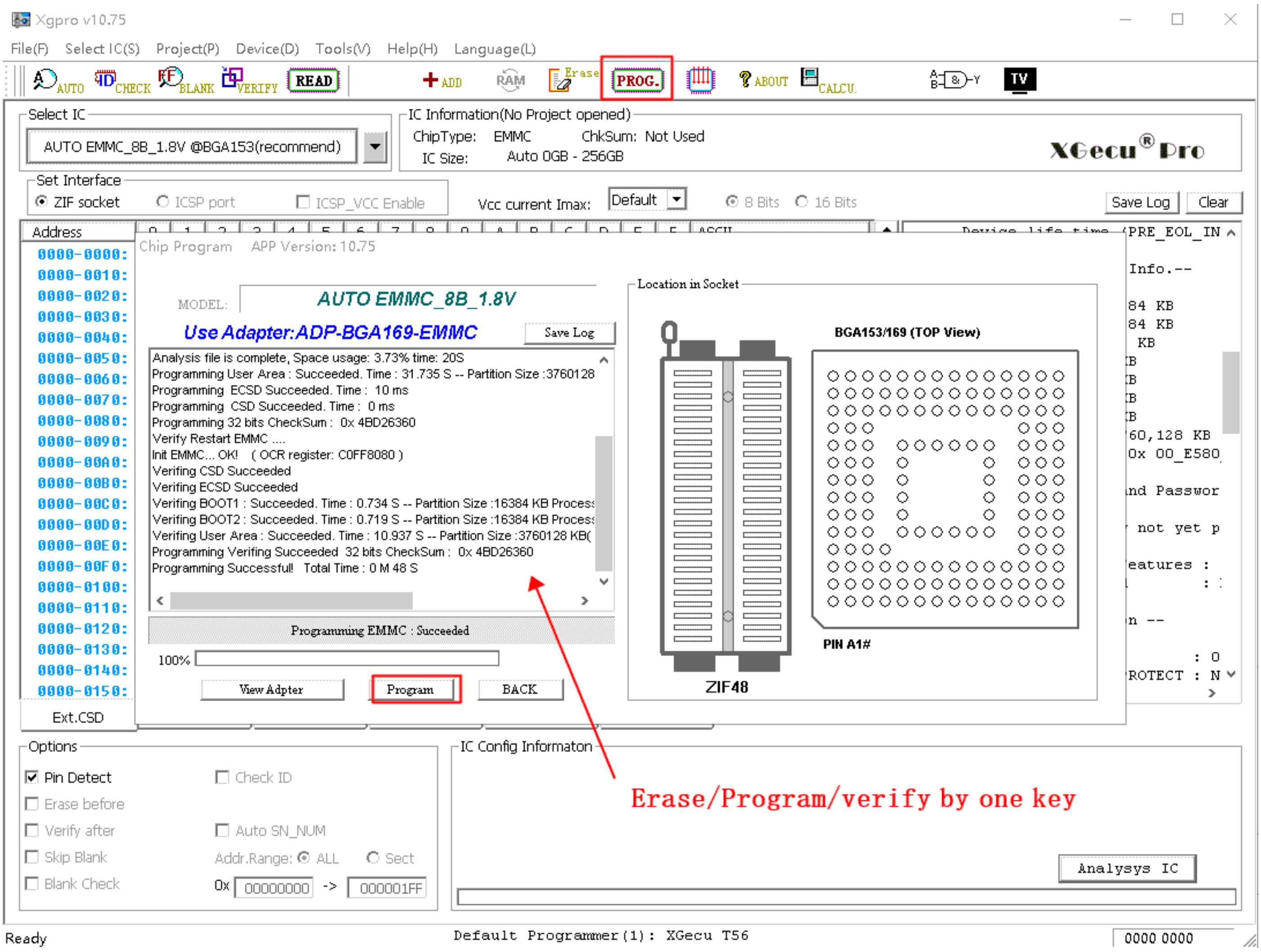
Project files(*.mpj)

After clicking **[Open]**, the project has been loaded

5.8.3 Program new IC (done by one key)

Don't need to make any other settings, at this time the data folder, ECSD, CSD and other operation options have all been set automatically, directly click **[PROG.]** button to copy and program.

Note: For projects that need to program the RPMB partition, please see section 5.8.4 for details



Xgpro v10.75

Select IC: AUTO EMMC_8B_1.8V @BGA153(recommend)

IC Information(No Project opened)
ChipType: EMMC ChkSum: Not Used
IC Size: Auto 0GB - 256GB

Set Interface: ● ZIF socket ○ ICSP port □ ICSP_VCC Enable

Vcc current Imax: Default

Chip Program APP Version: 10.75

MODEL: AUTO EMMC_8B_1.8V

Use Adapter:ADP-BGA169-EMMC

```
Analysis file is complete, Space usage: 3.73% time: 20S
Programming User Area : Succeeded. Time : 31.735 S -- Partition Size :3760128
Programming  ECSD Succeeded. Time :  10 ms
Programming  CSD Succeeded. Time :  0 ms
Programming 32 bits CheckSum :  0x 4BD26360
Verify Restart EMMC ....
Init EMMC... OK!   ( OCR register: C0FF8080 )
Verifing CSD Succeeded
Verifing ECSD Succeeded
Verifing BOOT1 : Succeeded. Time : 0.734 S -- Partition Size :16384 KB Process
Verifing BOOT2 : Succeeded. Time : 0.719 S -- Partition Size :16384 KB Process
Verifing User Area : Succeeded. Time : 10.937 S -- Partition Size :3760128 KB(
Programming Verifing Succeeded  32 bits CheckSum :  0x 4BD26360
Programming Successful!   Total Time : 0 M 48 S
```

Programming EMMC : Succeeded

100%

Options: ☑ Pin Detect □ Erase before □ Verify after □ Skip Blank □ Blank Check □ Check ID □ Auto SN_NUM
Addr.Range: ● ALL ○ Sect
0x 00000000 -> 000001FF

Default Programmer(1): XGecu T56

Erase/Program/verify by one key

Programming process as shown above, "Program + Verify" total time of 0 minutes and 48 seconds, it is faster than reading, because the blank data was skipped when programming.

The new chip copy is completed.

5.8.4 One-key-ghost project notes:

- When programming the project, the capacity of the BOOT, RPMB, and USER partitions of the original chip will be compared with that of the programmed chip. If the capacity of the new chip in any one partition is smaller than the original chip, will stop programming.
- When programming the project, the EMMC version of the original chip and the new chip will be compared. If the version of the new chip is lower than that of the original chip, the programming will be stopped.
- For projects that need to program RPMB partitions, the correct 32-byte authentication key is required. The user can load the authentication key in the File menu after opening the project, and save the project after loading. This authentication key will be used as the new authentication key when programming the RPMB. (Authentication key can also be loaded before reading. Save to project file when saving a project automatically with key.)
- If the authentication key is not loaded, when programming, the all-blank data 0xFF will be used as the authentication key.
- Important Note: authentication key, one chip is only allowed to be programmed once, it cannot be read and cannot be erased. If the authentication key is lost, the RPMB data will be permanently immutable. For specific devices, incorrect authentication key will not pass.
- If new chip’s RPMB has been used when programming, but there is no authentication key, it will prompt key error. At this time,

the RPMB data cannot be programmed. If you have correct authentication key , the RPMB can be programmed normally.

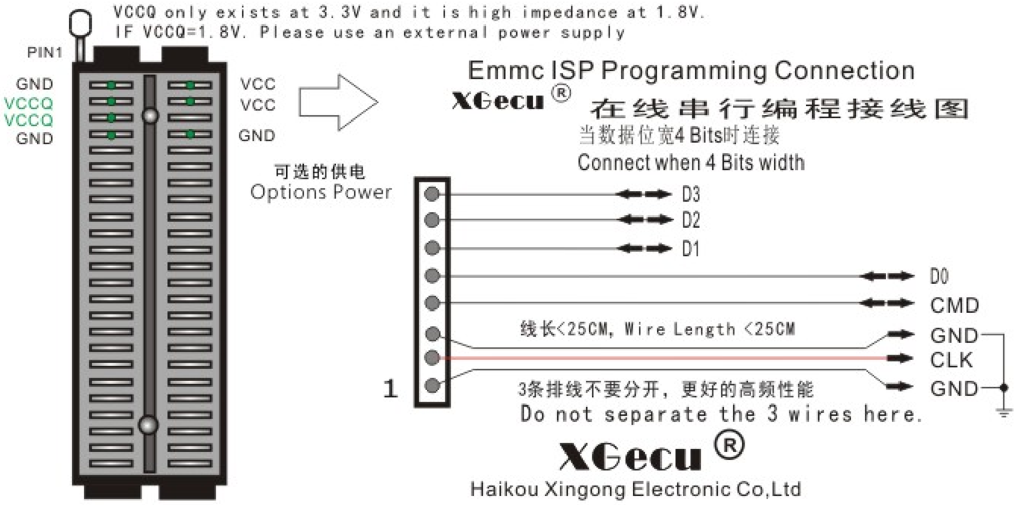
5.9 EMMC ISP in-circuit programming example

EMMC in-circuit programming is an important and convenient application in the home appliance repair industry. T56 programmer through ISP programming, the rate reached to 40Mb / S (single line transmission measured: 4.7M bytes / second), ISP cable length up to 40CM, that is extremely high stability and high speed. For common TV motherboard 4GB IC, the read time is only about 800 seconds.

Note: working frequency setting is 36MHZ by default. Via ISP, it can be set to 40MHZ in most of cases.

5.9.1 **EMMC ISP connection**

ISP Schematic Diagram


VCCQ only exists at 3.3V and it is high impedance at 1.8V.
IF VCCQ=1.8V. Please use an external power supply
PIN1
GND
VCCQ
VCCQ
GND
VCC
VCC
GND
可选的供电
Options Power
Emmc ISP Programming Connection
XGecu ®
在线串行编程接线图
当数据位宽4 Bits时连接
Connect when 4 Bits width
D3
D2
D1
D0
CMD
线长<25CM, Wire Length <25CM
GND
CLK
GND
1
3条排线不要分开，更好的高频性能
Do not separate the 3 wires here.
XGecu ®
Haikou Xingong Electronic Co,Ltd

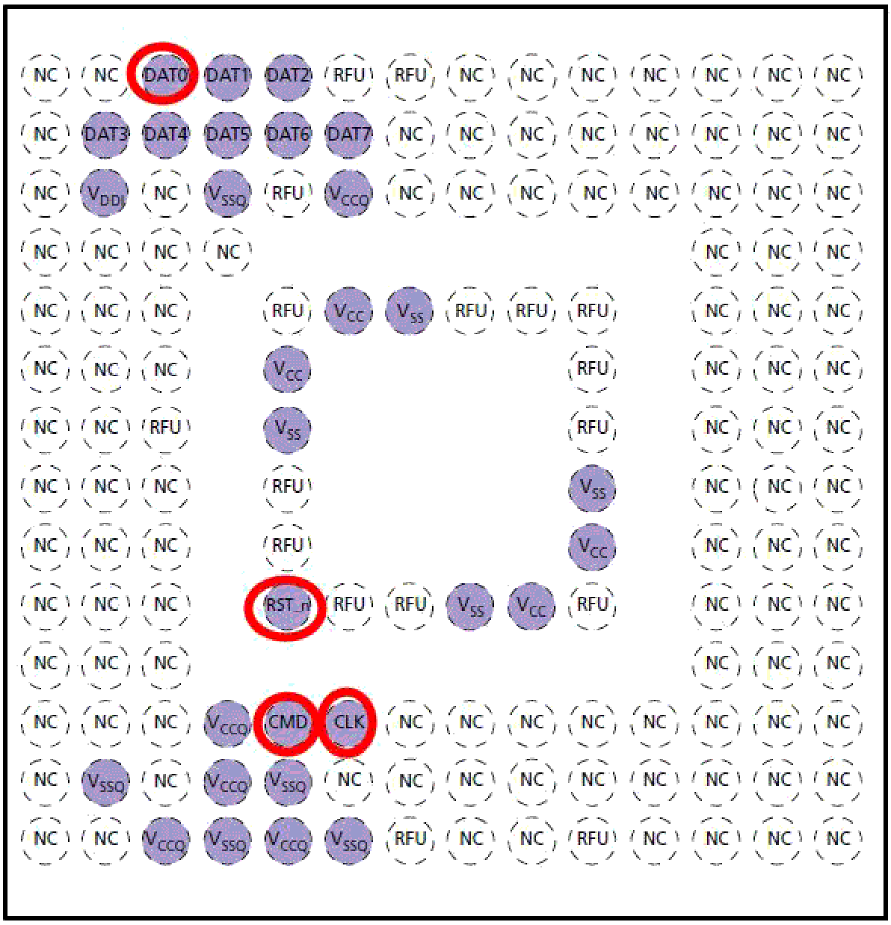
In general, 1bits only need to connect GND / CLK / CMD / D0 to the motherboard, then power the motherboard.

Notes:

- ISP has two grounds, both grounds need to be connected, and the ground point should be as close to the CLK line as possible.
- CLK line and the two GND lines do not separate, and CLK line should not cross with other lines, CLK line generally has a series of resistors R, you had better remove it.
- Both ends of the crystal of the MCU in PCB, connected to ground, stop the MCU from working.
- Connect the lines, power up the motherboard and check if the RST_n pin of the EMMC is high. If the RST_n voltage is 0, you need to pull up the pin, otherwise EMMC will not work. You had better connect 1K resistor to VCCQ (1.8V or 3.3V). This pin is the reset pin of EMMC.
- When power is supplied through external power supply, the computer and the external power supply's casing are must be reliably connected to the earth (anti-static interference)

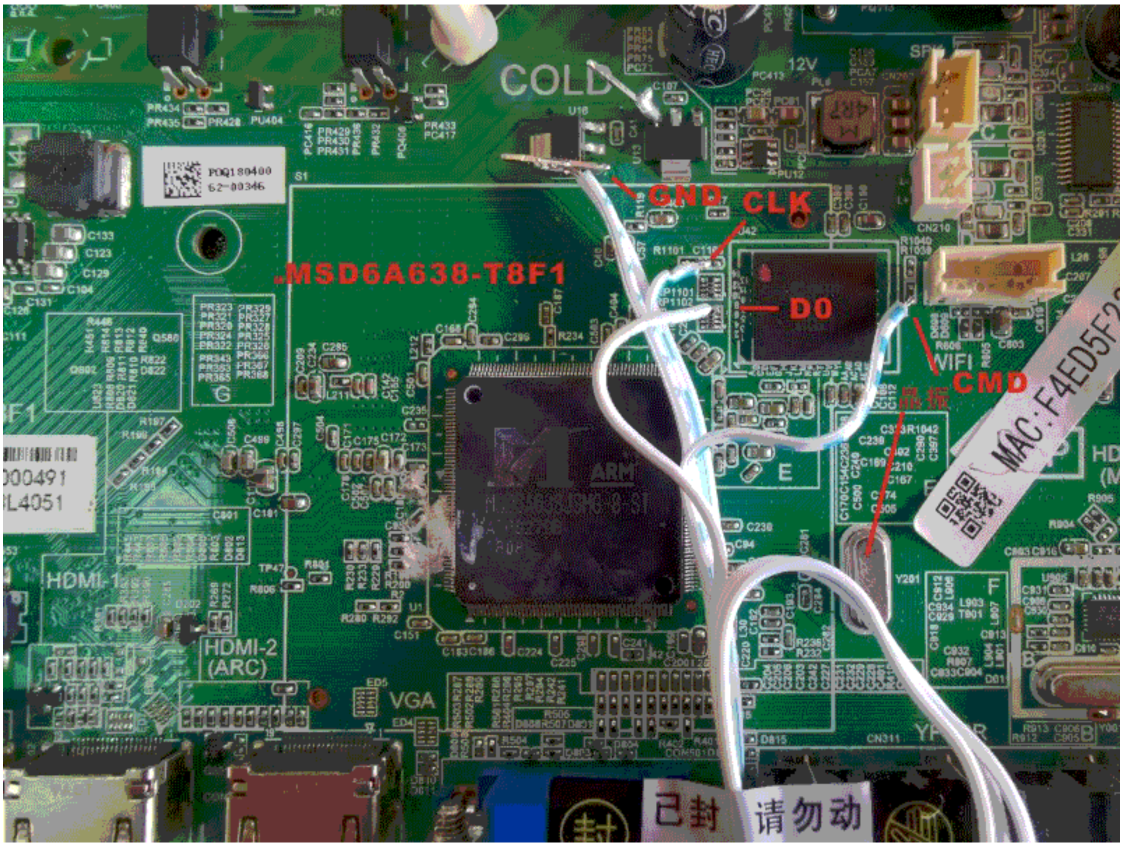
Note: When the ISP is powered by programmer and ISP_3.3V is checked, the VCC and VCCQ pins of programmer will be powered at 3.3V. When ISP_1.8V is checked, the programmer is only for VCC 3.3V, VCCQ high resistance state, VCCQ 1.8V needs external power supply.

EMMC BGA153/169 IC PINS:

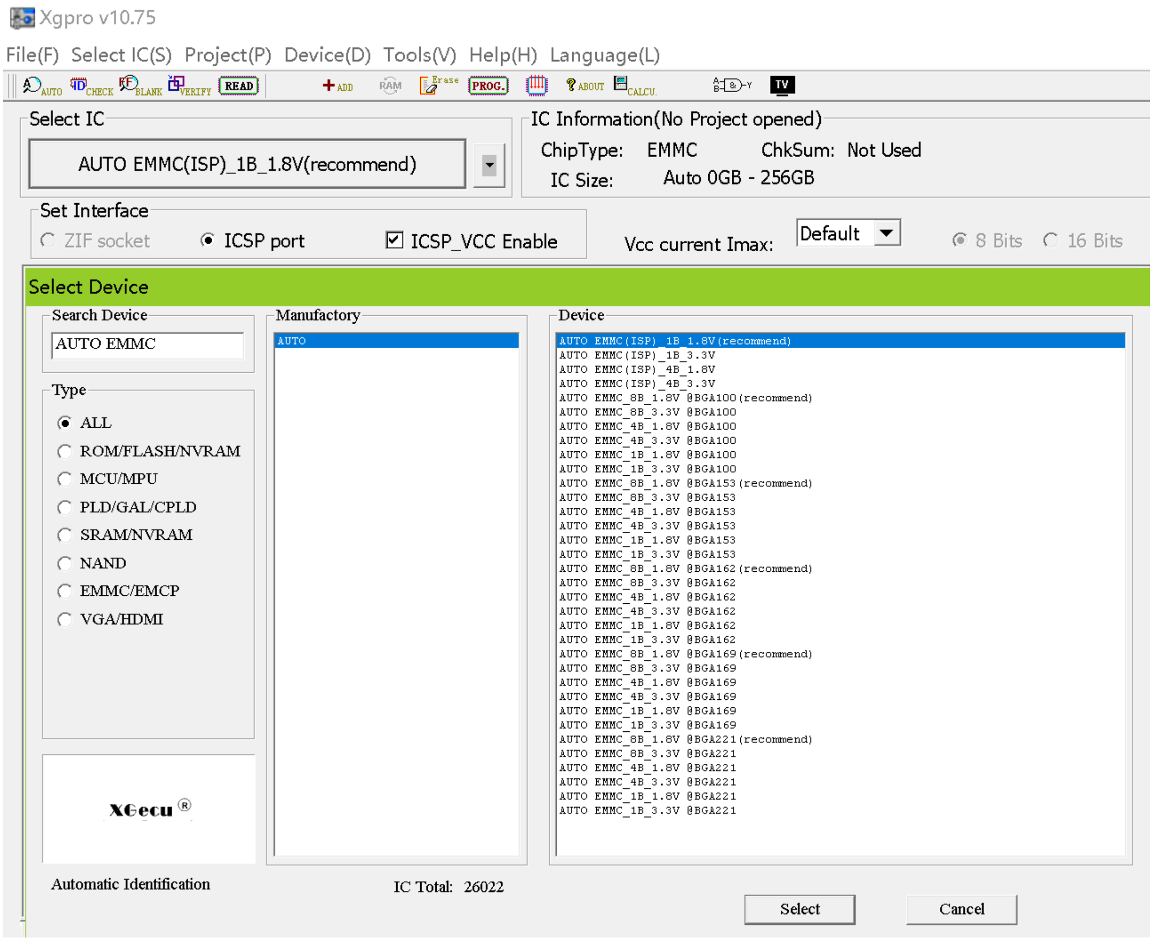


NC	NC	DAT0	DAT1	DAT2	RFU	RFU	NC	NC	NC	NC	NC	NC	NC
NC	DAT3	DAT4	DAT5	DAT6	DAT7	NC	NC	NC	NC	NC	NC	NC	NC
NC	V_{DDi}	NC	V_{SSQ}	RFU	V_{CCQ}	NC	NC	NC	NC	NC	NC	NC	NC
NC	NC	NC	NC								NC	NC	NC
NC	NC	NC		RFU	V_{CC}	V_{SS}	RFU	RFU	RFU		NC	NC	NC
NC	NC	NC		V_{CC}					RFU		NC	NC	NC
NC	NC	RFU		V_{SS}					RFU		NC	NC	NC
NC	NC	NC		RFU					V_{SS}		NC	NC	NC
NC	NC	NC		RFU					V_{CC}		NC	NC	NC
NC	NC	NC		RST_n	RFU	RFU	V_{SS}	V_{CC}	RFU		NC	NC	NC
NC	NC	NC									NC	NC	NC
NC	NC	NC	V_{CCQ}	CMD	CLK	NC	NC	NC	NC	NC	NC	NC	NC
NC	V_{SSQ}	NC	V_{CCQ}	V_{SSQ}	NC	NC	NC	NC	NC	NC	NC	NC	NC
NC	NC	V_{CCQ}	V_{SSQ}	V_{CCQ}	V_{SSQ}	RFU	NC	NC	RFU	NC	NC	NC	NC

The following photo shows the wiring example of the MSD6A638-T8F1 TV board:



5.9.2 **Select IC and Analysis IC**



Select IC: AUTO EMMC(ISP)_1B_1.8V(recommend)

IC Information(No Project opened)
ChipType: EMMC ChkSum: Not Used
IC Size: Auto 0GB - 256GB

Set Interface
○ ZIF socket ● ICSP port ☑ ICSP_VCC Enable
Vcc current Imax: Default ● 8 Bits ○ 16 Bits

Select Device

Search Device: AUTO EMMC

Type
● ALL
○ ROM/FLASH/NVRAM
○ MCU/MPU
○ PLD/GAL/CPLD
○ SRAM/NVRAM
○ NAND
○ EMMC/EMCP
○ VGA/HDMI

Manufactory: AUTO

Device:

```
AUTO EMMC(ISP)_1B_1.8V(recommend)
AUTO EMMC(ISP)_1B_3.3V
AUTO EMMC(ISP)_4B_1.8V
AUTO EMMC(ISP)_4B_3.3V
AUTO EMMC_8B_1.8V @BGA100(recommend)
AUTO EMMC_8B_3.3V @BGA100
AUTO EMMC_4B_1.8V @BGA100
AUTO EMMC_4B_3.3V @BGA100
AUTO EMMC_1B_1.8V @BGA100
AUTO EMMC_1B_3.3V @BGA100
AUTO EMMC_8B_1.8V @BGA153(recommend)
AUTO EMMC_8B_3.3V @BGA153
AUTO EMMC_4B_1.8V @BGA153
AUTO EMMC_4B_3.3V @BGA153
AUTO EMMC_1B_1.8V @BGA153
AUTO EMMC_1B_3.3V @BGA153
AUTO EMMC_8B_1.8V @BGA162(recommend)
AUTO EMMC_8B_3.3V @BGA162
AUTO EMMC_4B_1.8V @BGA162
AUTO EMMC_4B_3.3V @BGA162
AUTO EMMC_1B_1.8V @BGA162
AUTO EMMC_1B_3.3V @BGA162
AUTO EMMC_8B_1.8V @BGA169(recommend)
AUTO EMMC_8B_3.3V @BGA169
AUTO EMMC_4B_1.8V @BGA169
AUTO EMMC_4B_3.3V @BGA169
AUTO EMMC_1B_1.8V @BGA169
AUTO EMMC_1B_3.3V @BGA169
AUTO EMMC_8B_1.8V @BGA221(recommend)
AUTO EMMC_8B_3.3V @BGA221
AUTO EMMC_4B_1.8V @BGA221
AUTO EMMC_4B_3.3V @BGA221
AUTO EMMC_1B_1.8V @BGA221
AUTO EMMC_1B_3.3V @BGA221
```

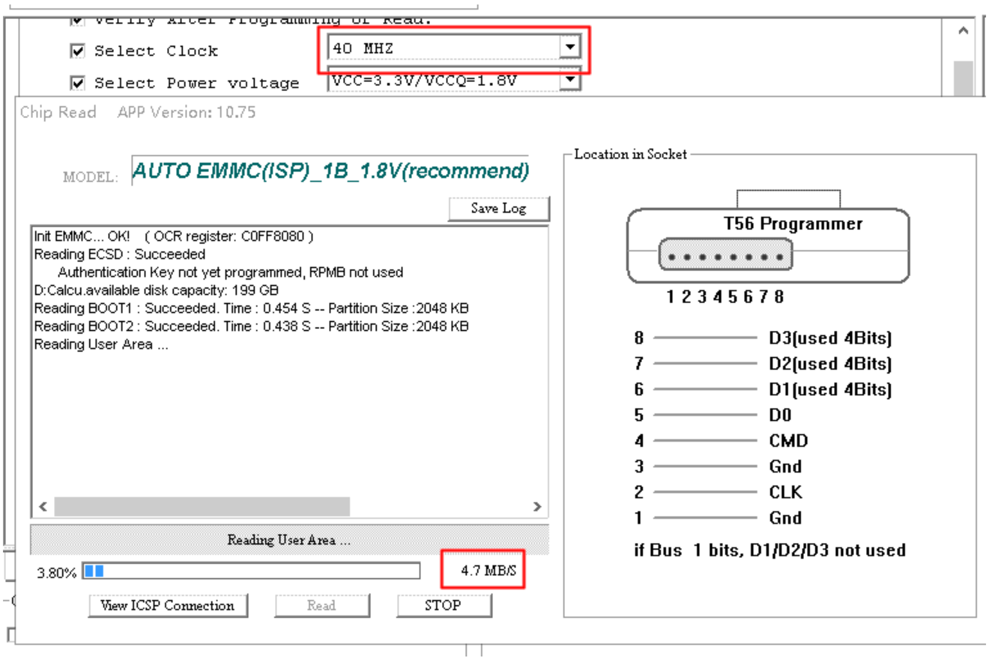
Automatic Identification IC Total: 26022

Select Cancel

- Type in "AUTO EMMC"
- Select AUTO_EMMC(ISP)_1B_1.8V
- Click **<Analyze IC>**button and check if the connection is normal. If there is no problem, will go to the next step

5.9.3 **ISP read data from original IC**

Before reading, set 40MHZ as the operating frequency in the operation options and click button **[Read]** in the toolbar.



☑ Verify After Programming or Read.
☑ Select Clock 40 MHZ
☑ Select Power voltage VCC=3.3V/VCCQ=1.8V

MODEL: AUTO EMMC(ISP)_1B_1.8V(recommend)

```
Init EMMC... OK!   ( OCR register: C0FF8080 )
Reading ECSD : Succeeded
    Authentication Key not yet programmed, RPMB not used
D:Calcu.available disk capacity: 199 GB
Reading BOOT1 : Succeeded. Time : 0.454 S -- Partition Size :2048 KB
Reading BOOT2 : Succeeded. Time : 0.438 S -- Partition Size :2048 KB
Reading User Area ...
```

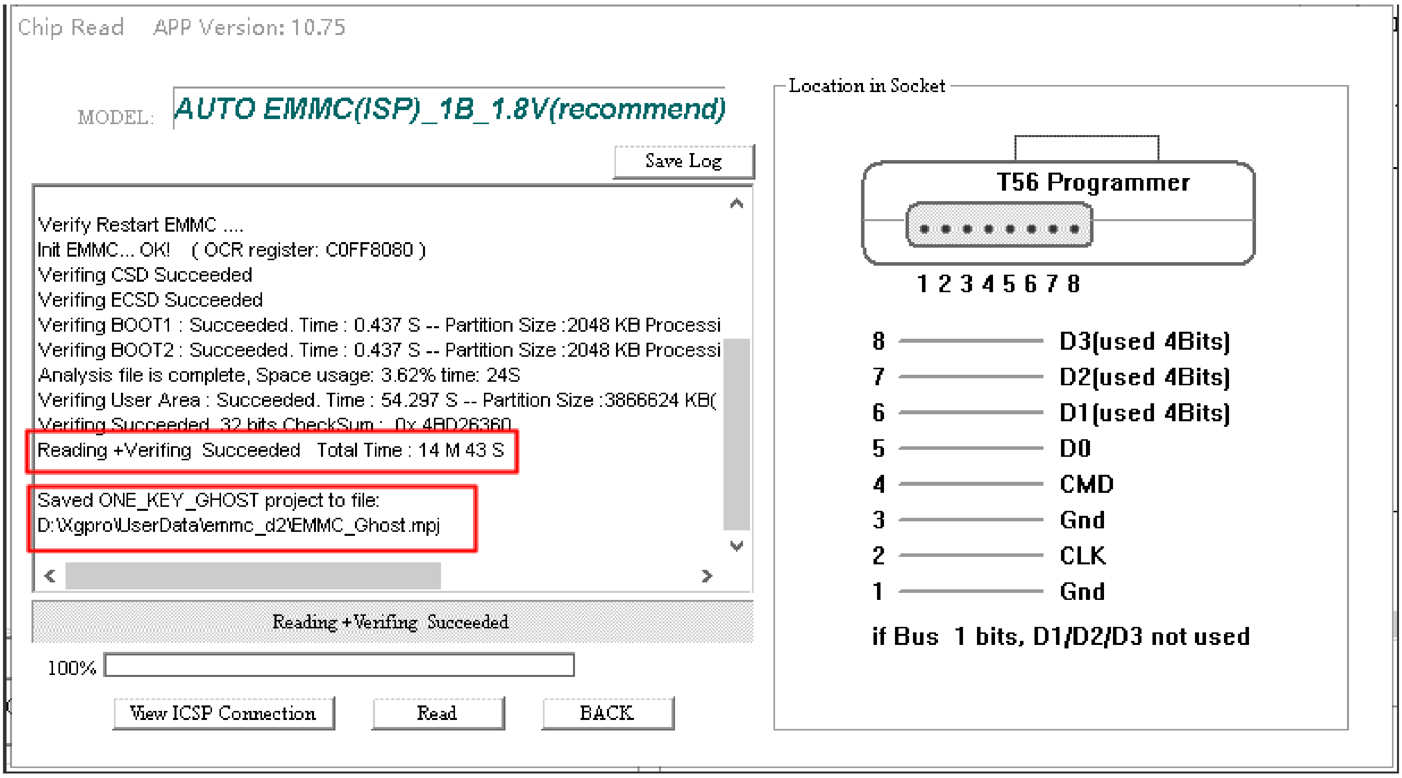
Reading User Area ...

3.80% 4.7 MB/S

8	D3(used 4Bits)
7	D2(used 4Bits)
6	D1(used 4Bits)
5	D0
4	CMD
3	Gnd
2	CLK
1	Gnd

if Bus 1 bits, D1/D2/D3 not used

The reading process is shown in the figure above, the stable reading speed is 4.7MB/s at 40MHZ operating frequency. Reading speed is 4.1MB/s at 36MHZ, if there is the error, frequency can be properly reduced.



```
Verify Restart EMMC ....
Init EMMC... OK!   ( OCR register: C0FF8080 )
Verifing CSD Succeeded
Verifing ECSD Succeeded
Verifing BOOT1 : Succeeded. Time : 0.437 S -- Partition Size :2048 KB Processi
Verifing BOOT2 : Succeeded. Time : 0.437 S -- Partition Size :2048 KB Processi
Analysis file is complete, Space usage: 3.62% time: 24S
Verifing User Area : Succeeded. Time : 54.297 S -- Partition Size :3866624 KB(
Verifing Succeeded  32 bits CheckSum :  0x 4BD26360
Reading +Verifing  Succeeded   Total Time : 14 M 43 S

Saved ONE_KEY_GHOST project to file:
D:\Xgpro\UserData\emmc_d2\EMMC_Ghost.mpj
```

Reading +Verifing Succeeded

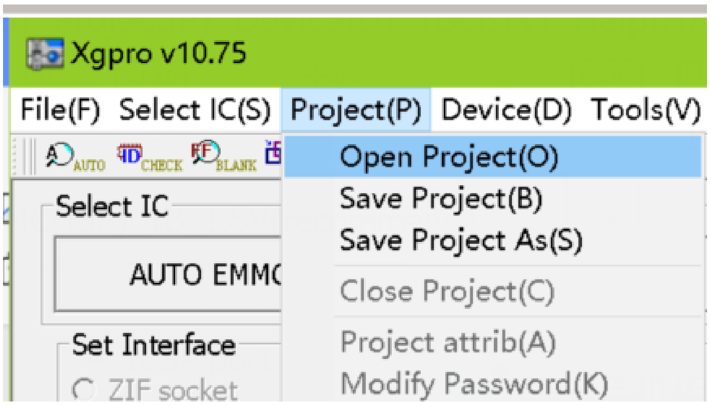
100%

- The total time for "Read + Verify" the 4G IC is less than 900 seconds. The verification time is particularly short because the IC takes up very little space.
- Also saved one-key ghost project file, only need to open the project file when copying, the specific method is the same as in section 5.8.

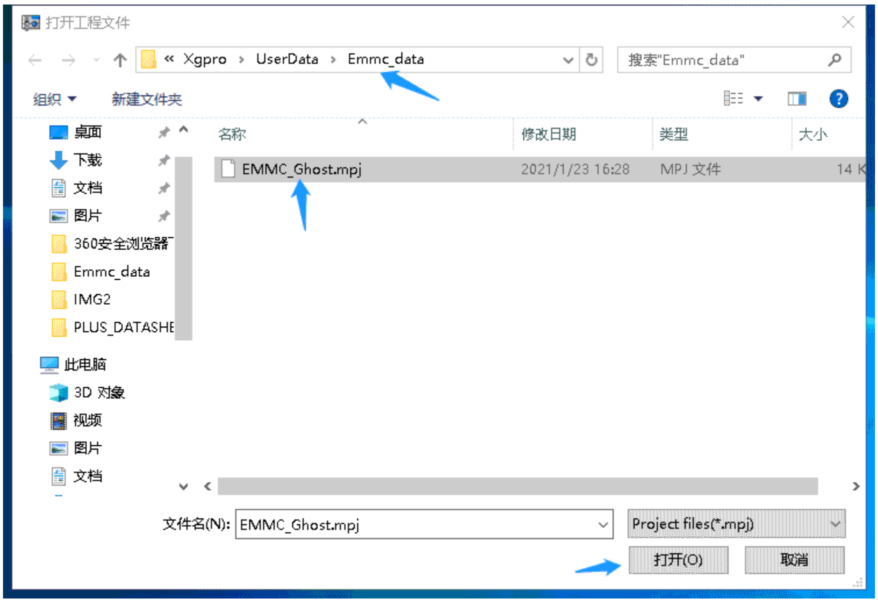
5.9.4 ISP program IC

- *Select IC AUTO_EMMC(ISP)_1B_1.8V
- *Open the one-key clone project file read
  Under normal condition, open the project file and program all the data read from EMMC to finish the copy with one key. If you don't need to program all the data, please uncheck the option that you don't need to write.
  Another method: After selecting IC, do not use the project file, just select the data folder. Check the area to be programmed in the operation options, that is completely set by user
- Click [PROG.] button in the toolbar to start programming IC

Open Project



Open project file: EMMC_GHOST.MPJ



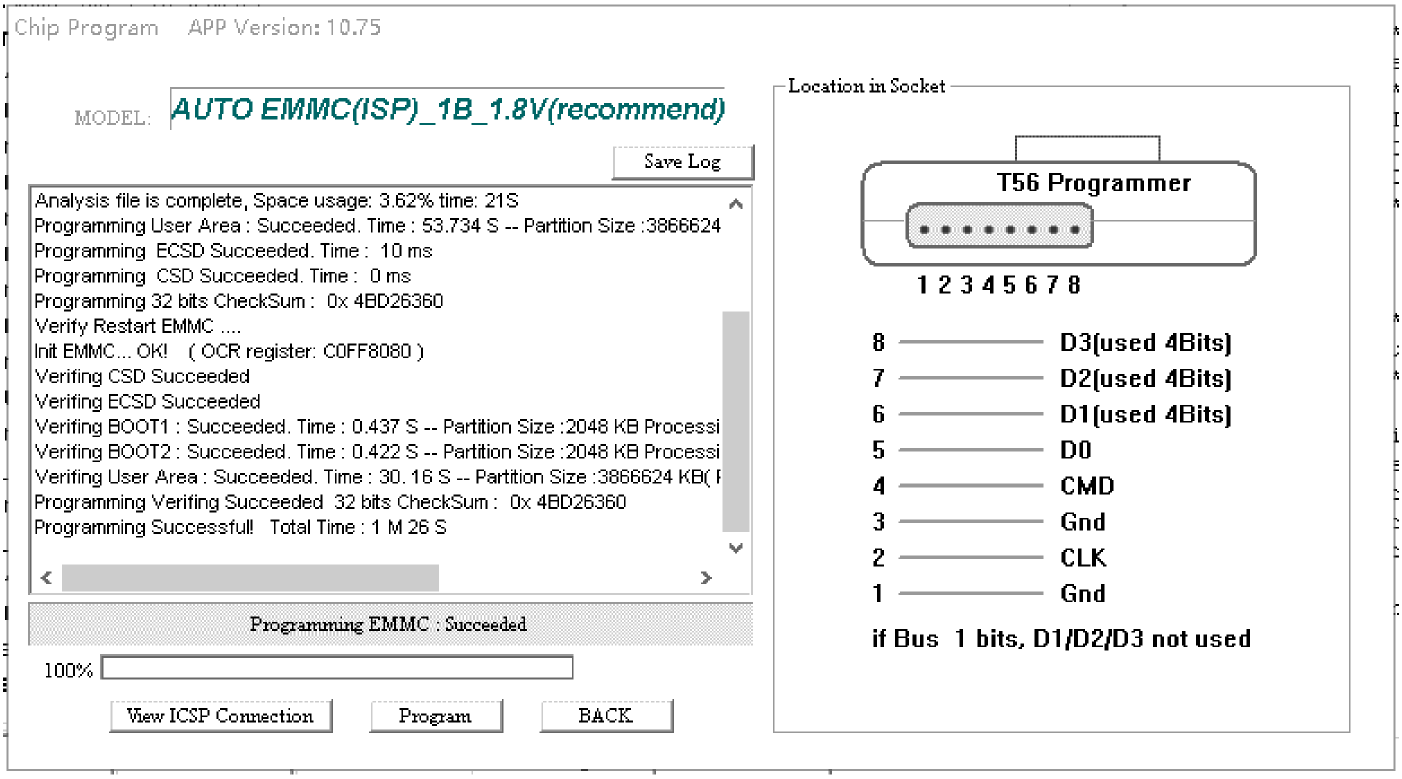
名称	修改日期	类型	大小
EMMC_Ghost.mpj	2021/1/23 16:28	MPJ 文件	14 K

文件名(N): EMMC_Ghost.mpj

Project files(*.mpj)

Click **[PROG.]** on toolbar

The whole programming process, as shown below, Erase, Program and Verify all partitions with one key.



MODEL: *AUTO EMMC(ISP)_1B_1.8V(recommend)*

```
Analysis file is complete, Space usage: 3.62% time: 21S
Programming User Area : Succeeded. Time : 53.734 S -- Partition Size :3866624
Programming  ECSD Succeeded. Time :  10 ms
Programming  CSD Succeeded. Time :  0 ms
Programming 32 bits CheckSum :  0x 4BD26360
Verify Restart EMMC ....
Init EMMC... OK!   ( OCR register: C0FF8080 )
Verifing CSD Succeeded
Verifing ECSD Succeeded
Verifing BOOT1 : Succeeded. Time : 0.437 S -- Partition Size :2048 KB Processi
Verifing BOOT2 : Succeeded. Time : 0.422 S -- Partition Size :2048 KB Processi
Verifing User Area : Succeeded. Time : 30. 16 S -- Partition Size :3866624 KB( F
Programming Verifing Succeeded  32 bits CheckSum :  0x 4BD26360
Programming Successful!   Total Time : 1 M 26 S
```

Programming EMMC : Succeeded

100%

[END of EMMC]

6. NAND/SPI_NAND Programming

T56 supports up to 256G bits NAND flash, supports BGA and TSOP packages and SPI NAND
TL866II only supports TSOP48 chips with less than 8G bits (1G bytes) capacity

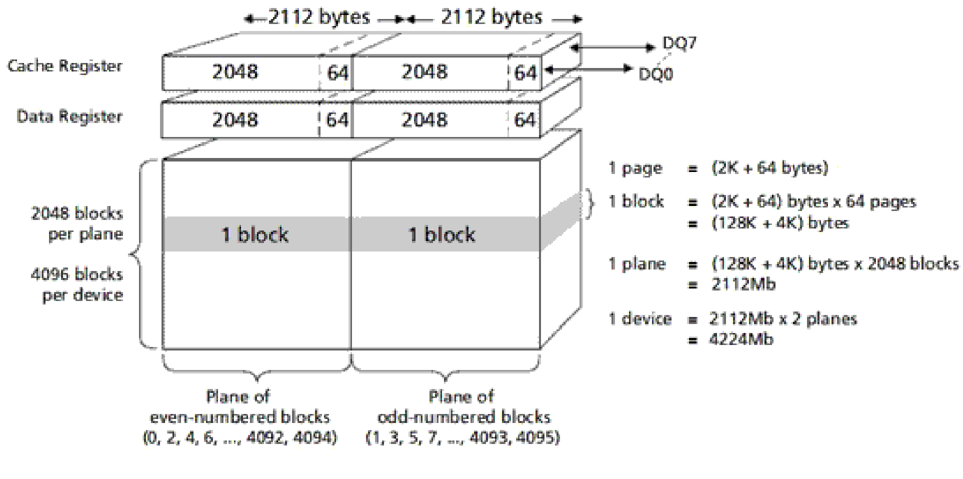
6.1 Characteristics of NAND FLASH

- NOR Flash randomly accesses all memory maps and dedicated interfaces (such as EPROM) address and data lines. But there is no dedicated address line in NAND flash. It is made up of an 8/16-bit wide interface bus that sends commands, addresses and data to internal registers, that provides a more flexible configuration for many original chips. The NAND flash architecture, emphasis on lower cost per bit, higher performance, and disk can easily upgrade through the interface, NAND is lower cost and higher capacity.
- NAND, bad blocks are allowed. Due to the production process of NAND, bad blocks will randomly appear in the factory chip. Bad blocks are initialized at the factory and marked as bad blocks in the special area. If bad blocks appear during use, they also need to be marked.
- Bit flip: If bit flip occurs on critical files, it will cause the system to hang up. Therefore, while using NAND FLASH, algorithms such as ECC/EDC must be used for data correction to ensure reliability.
- There is spare area. Just because NAND FLASH has the above two special features, spare area plays the role of putting bad block flags, ECC values, IC information and file information.
- NAND FLASH bad blocks manage: There are many ways to manage bad blocks in NAND FLASH. Different system providers may choose different management methods of bad block to meet the needs of product development. The software defines three common methods for user to choose from. The software defines three common methods for users to select and use, and enables user-defined ECC algorithms for flexible applications.

6.2 Structure of NAND

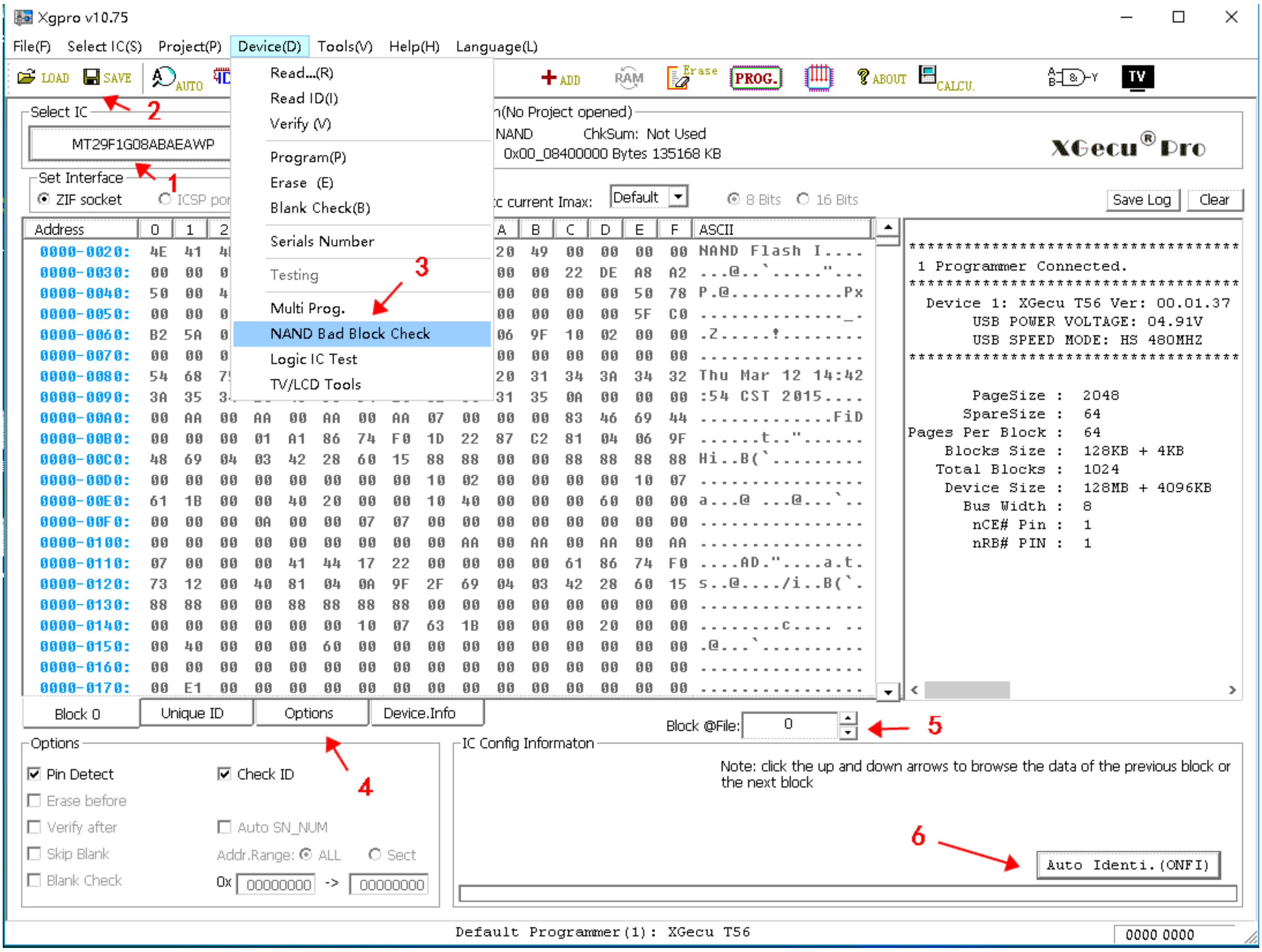
The following figure shows the MT29F4G08ABAEA as an example:
MT29F4G08 has 2 Planes *2048 blocks , the programmer is operated in blocks, index number 0-4095 blocks. Each block has 64 pages , and the processing of each block is separately written to the chip sequentially.
The size of each page: 2048 bytes (Page size) + 64 bytes (Spare size) = 2112 bytes
The total capacity of IC is: 2112 * 64 (Page) * 4096 (Block) = 4224MB.

Figure 9: Array Organization – MT29F4G08 (x8)


2112 bytes
2112 bytes
Cache Register
2048
64
2048
64
DQ7
DQ0
Data Register
2048
64
2048
64
2048 blocks per plane
4096 blocks per device
1 block
1 block
1 page = (2K + 64 bytes)
1 block = (2K + 64) bytes x 64 pages
= (128K + 4K) bytes
1 plane = (128K + 4K) bytes x 2048 blocks
= 2112Mb
1 device = 2112Mb x 2 planes
= 4224Mb
Plane of even-numbered blocks (0, 2, 4, 6, ..., 4092, 4094)
Plane of odd-numbered blocks (1, 3, 5, 7, ..., 4093, 4095)

6.3 NAND Program main interface

After selecting IC, the software interface is as below:



Select IC: MT29F1G08ABAEAWP

Set Interface: ● ZIF socket ○ ICSP port

(No Project opened)
NAND ChkSum: Not Used
0x00_08400000 Bytes 135168 KB

Vcc current Imax: Default ● 8 Bits ○ 16 Bits

```
Address     0  1  2                                        A  B  C  D  E  F  ASCII
0000-0020: 4E 41 4                                        20 49 00 00 00 00 NAND Flash I....
0000-0030: 00 00 0                                        00 00 22 DE A8 A2 ...@..`....."...
0000-0040: 50 00 4                                        00 00 00 00 50 78 P.@...........Px
0000-0050: 00 00 0                                        00 00 00 00 5F C0 ..............._.
0000-0060: B2 5A 0                                        06 9F 10 02 00 00 .Z.....!........
0000-0070: 00 00 0                                        00 00 00 00 00 00 ................
0000-0080: 54 68 7                                        20 31 34 3A 34 32 Thu Mar 12 14:42
0000-0090: 3A 35 3                                        31 35 0A 00 00 00 :54 CST 2015....
0000-00A0: 00 AA 00 AA 00 AA 00 AA 07 00 00 00 83 46 69 44 .............FiD
0000-00B0: 00 00 00 01 A1 86 74 F0 1D 22 87 C2 81 04 06 9F ......t..".......
0000-00C0: 48 69 04 03 42 28 60 15 88 88 00 00 88 88 88 88 Hi..B(`.........
0000-00D0: 00 00 00 00 00 00 00 00 10 02 00 00 00 00 10 07 ................
0000-00E0: 61 1B 00 00 40 20 00 00 10 40 00 00 00 60 00 00 a...@ ...@...`..
0000-00F0: 00 00 00 0A 00 00 07 07 00 00 00 00 00 00 00 00 ................
0000-0100: 00 00 00 00 00 00 00 00 00 AA 00 AA 00 AA 00 AA ................
0000-0110: 07 00 00 00 41 44 17 22 00 00 00 00 61 86 74 F0 ....AD."....a.t.
0000-0120: 73 12 00 40 81 04 0A 9F 2F 69 04 03 42 28 60 15 s..@..../i..B(`.
0000-0130: 88 88 00 00 88 88 88 88 00 00 00 00 00 00 00 00 ................
0000-0140: 00 00 00 00 00 00 10 07 63 1B 00 00 00 20 00 00 ........c.... ..
0000-0150: 00 40 00 00 00 60 00 00 00 00 00 00 00 00 00 00 .@...`..........
0000-0160: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0000-0170: 00 E1 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
```

```
*************************************
 1 Programmer Connected.
*************************************
  Device 1: XGecu T56 Ver: 00.01.37
     USB POWER VOLTAGE: 04.91V
     USB SPEED MODE: HS 480MHZ
*************************************


       PageSize :  2048
      SpareSize :  64
Pages Per Block :  64
    Blocks Size :  128KB + 4KB
   Total Blocks :  1024
    Device Size :  128MB + 4096KB
      Bus Width :  8
       nCE# Pin :  1
       nRB# PIN :  1
```

Options

☑ Pin Detect ☑ Check ID
☐ Erase before
☐ Verify after ☐ Auto SN_NUM
☐ Skip Blank Addr.Range: ● ALL ○ Sect
☐ Blank Check 0x 00000000 -> 00000000

Block @File: 0

IC Config Informaton

Note: click the up and down arrows to browse the data of the previous block or the next block

Default Programmer(1): XGecu T56 0000 0000

1) Click select button to select IC model

2) Load and save: data files to load or save to file after reading

3) **<NAND bad block check>** function in the menu is only valid for standard bad block markers, for a small amount of specific device data, if the bad block marker is not standard, the check result is not accurate, you can ignore it.

4) Configuration and options: configuration of NAND programming method, see the next section for details: configuration and options (for equipment maintenance personnel, generally settings by default, do not need to change)

5) Data block switch: user browse the data buffer by block, after loading file or reading IC, user can use the software to directly browse the data in the set block.

6) ONFI automatic identification of IC parameters: the IC model not listed in the software, for the NAND chip in line with the ONFI standard, the user can use this function to automatically identify the chip parameters, automatically adjust the software parameters after identifying. You can read and write directly, and can be saved as a customized IC model, see Chapter 10 Add customized IC for details.

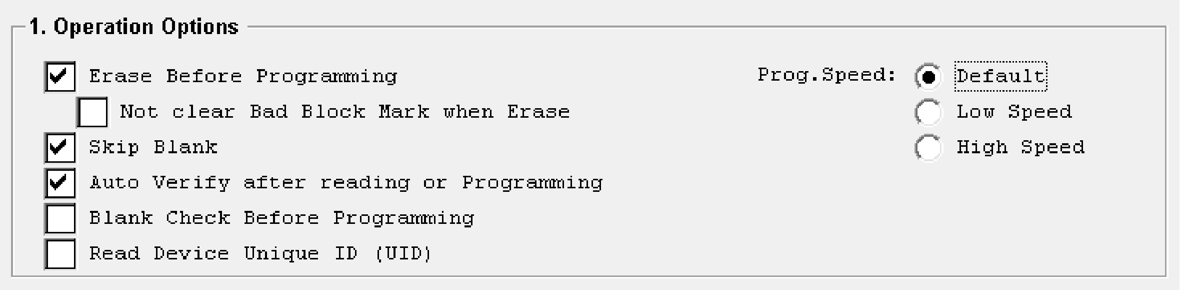
6.4 NAND Configuration and Options

Click **[Options]** button in Figure 4 to enter the programming configuration interface.

6.4.1 **Programming Operation Options**

Settings when reading or programming

Prog. Speed: if an error occurs during reading or programming, please try to select "Low Speed".



1. Operation Options

☑ Erase Before Programming

□ Not clear Bad Block Mark when Erase

☑ Skip Blank

☑ Auto Verify after reading or Programming

□ Blank Check Before Programming

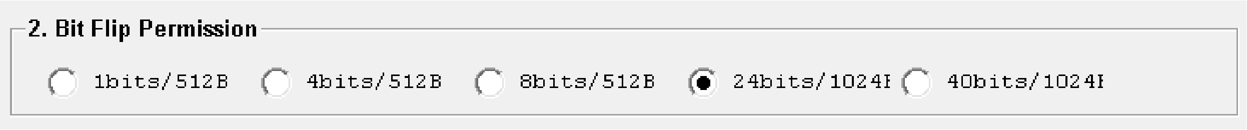
□ Read Device Unique ID (UID)

Prog.Speed: ● Default ○ Low Speed ○ High Speed

6.4.2 **Bit flip Permission**

Set by the required ECC error correction bit, usually set by default, no need to modify.

Note: When reading or writing to the chip, the NAND will be bit flipped, so the data file read may not be exactly the same every time.The file data is valid as long as the auto-verify passed when reading, that means the error data is in allowed range.

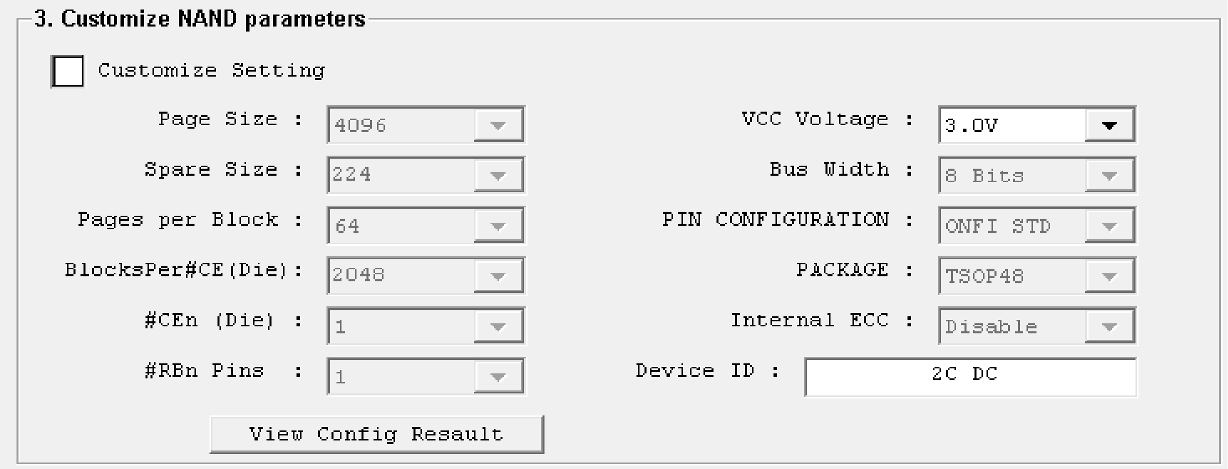


2. Bit Flip Permission

○ 1bits/512B ○ 4bits/512B ○ 8bits/512B ● 24bits/1024I ○ 40bits/1024I

6.4.3 **Customize NAND parameters**

The software allows user to set NAND parameters according to the IC data sheet. Theoretically, user can add any NAND IC by customize setting, see Chapter 10 Adding Custom IC for details.



3. Customize NAND parameters

□ Customize Setting

Page Size : 4096

Spare Size : 224

Pages per Block : 64

BlocksPer#CE(Die): 2048

#CEn (Die) : 1

#RBn Pins : 1

VCC Voltage : 3.0V

Bus Width : 8 Bits

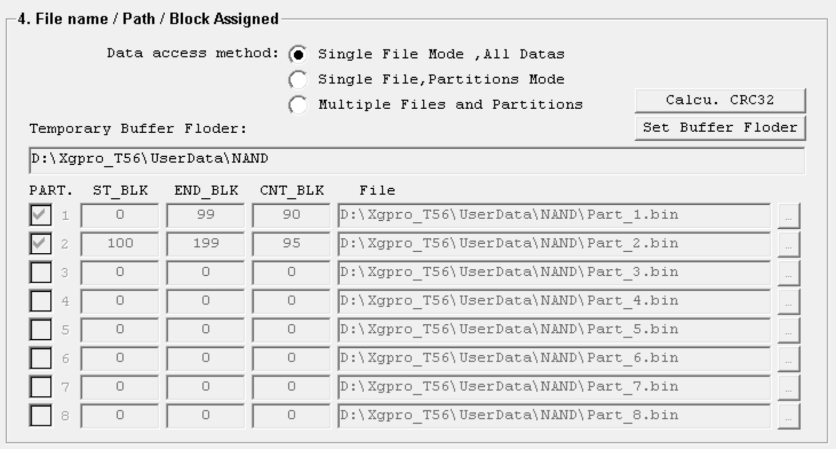
PIN CONFIGURATION : ONFI STD

PACKAGE : TSOP48

Internal ECC : Disable

Device ID : 2C DC

6.4.4 **File name/ Path/ Partition Assigned**



4. File name / Path / Block Assigned

Data access method: ● Single File Mode ,All Datas
○ Single File,Partitions Mode
○ Multiple Files and Partitions

Temporary Buffer Floder:

D:\Xgpro_T56\UserData\NAND

PART.		ST_BLK	END_BLK	CNT_BLK	File
☑	1	0	99	90	D:\Xgpro_T56\UserData\NAND\Part_1.bin
☑	2	100	199	95	D:\Xgpro_T56\UserData\NAND\Part_2.bin
□	3	0	0	0	D:\Xgpro_T56\UserData\NAND\Part_3.bin
□	4	0	0	0	D:\Xgpro_T56\UserData\NAND\Part_4.bin
□	5	0	0	0	D:\Xgpro_T56\UserData\NAND\Part_5.bin
□	6	0	0	0	D:\Xgpro_T56\UserData\NAND\Part_6.bin
□	7	0	0	0	D:\Xgpro_T56\UserData\NAND\Part_7.bin
□	8	0	0	0	D:\Xgpro_T56\UserData\NAND\Part_8.bin

6.4.4.1 Temporary Buffer Folder

The folder where temporary files are stored when a single file is accessed.
NAND files may take up a lot of disk space. Only while the capacity of disk is low, you need to set the folder to another disk, normally no changes are needed.

6.4.4.2 Data access method

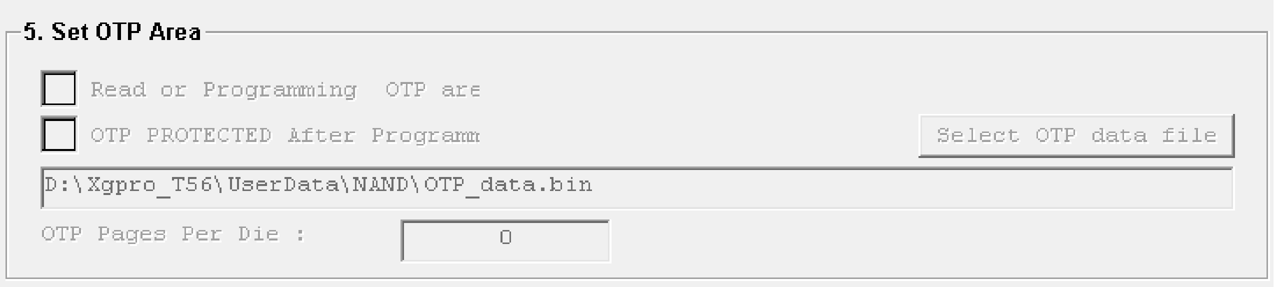
- **Single file Mode, all data**: all data is from a single buffer file when programming, or stored in single buffer file when reading. , generally uses this mode by hard copy.
- **Single file, partitions mode:** data is from a single buffer file (user loaded) when programming, or stored in the buffer file when reading. When programming or reading, the data is handled by partition.
  - PART: Select the partition to use
  - ST_BLK: Start block of partition
  - END_BLK: End block of partition
  - CNT_BLK: The number of data blocks in the file for current partition (the actual number of data blocks be processed)

  Note: The size of the partition (END_BLK-ST_BLK+1) must be larger than CNT_BLK when setting.
  The next partition memory cannot overlap with the previous partition.
- **Multiple files and partitions:** Partition mode is the same as the above, but each partition uses its own file, which does not use the buffer file.
  In mass production, the NAND partition mode is generally used for bad block skipping process, which can be set by user as needed, see Bad Block Management for details.

6.4.5 **Set OTP Area**

Only T56 programmer can support OTP

Some NAND has a special OTP area where data can be written, generally used to store the serial numbers. fixed or unchanging data, etc. OTP is the abbreviation of “One Time Programmable”, that means one-time programmable. Once the data in this area has been programmed, it cannot be programmed again.
When reading the original IC, you can check this option, after reading, check whether the OTP data file data is completely empty. If there is data, OTP data must be written to the new NAND together when copying.



5. Set OTP Area

□ Read or Programming OTP are

□ OTP PROTECTED After Programm

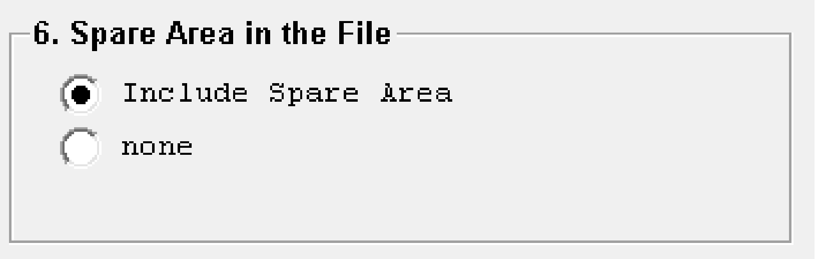
D:\Xgpro_T56\UserData\NAND\OTP_data.bin

OTP Pages Per Die : 0

OTP data file: Disable or enable OTP

OTP pages: refers to the maximum number of pages in this area, only the smaller number of pages can be set.

6.4.6 Spare Area in the File



6. Spare Area in the File

● Include Spare Area

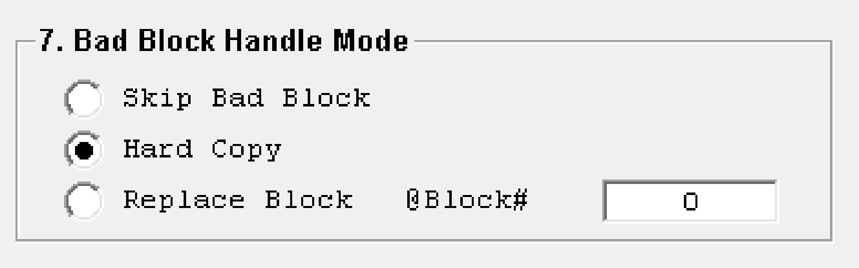
○ none

Due to the special nature of NAND FLASH, the data file is whether it includes the spare data.

If not included: the file will not include the spare area data

6.4.7 Bad Block Manage Mode

3 manage modes for bad block



7. Bad Block Handle Mode

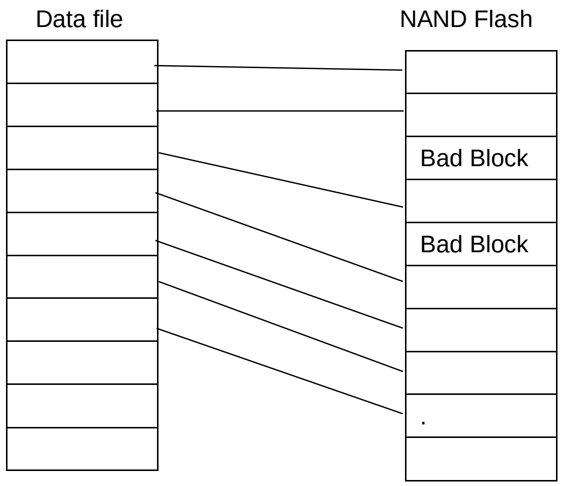
○ Skip Bad Block

● Hard Copy

○ Replace Block @Block# 0

6.4.7.1 Skip Bad Block

If a bad block is found during reading or writing IC, the programmer will skip the bad block. And directly proceed to the next block. as below:



6.4.7.2 Hard Copy

Force data reads or writes regardless of whether the block is marked with a bad block or not. If the verifying displays error during writing, programming will be stopped. This method is generally used for technical maintenance when replacing memory chips.

6.4.7.3 Replace Bad Block

Bad blocks are found to be written to the beginning of a specific block set, that allows the system to logically feel that the memory is still a contiguous area, and is generally used in partitioning mode.
See detailed instructions in the example configuration as below. The number type in the start block (example: 900) is the actual replacement block index that is written to FLASH when the bad block is found.

6.4.7.4 Enable user-defined algorithm file

In the customized algorithm DLL file, before programming user can perform data processing or automatic sequence number calculation for any data in the block, as well as perform ECC algorithms, etc. It is also possible to auto-fill the BBT table with data according to the skipped bad blocks during programming.
For details, please contact us and we can customize it according to your requirements.

6.4.8 Configuration Example 1 (MT29F1G08ABAEA)

Operation method: **single file, partitions mode, skip bad block**

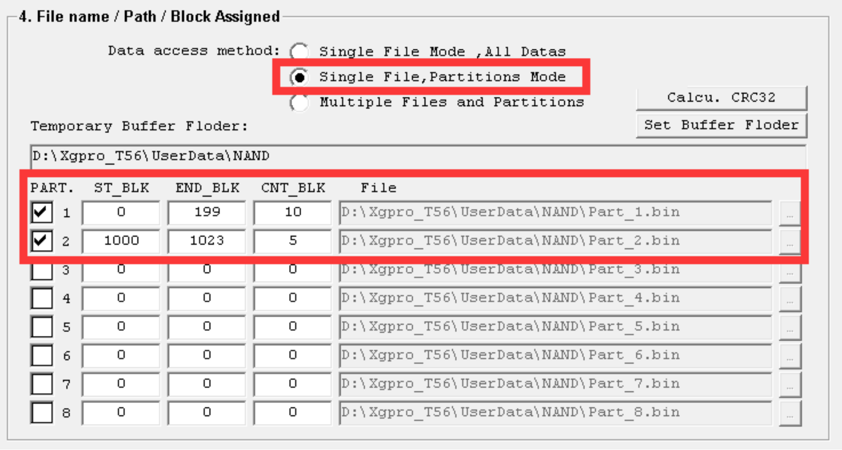
1) program NAND FLASH with partition
2) bad block processing mode: **skip bad block**
3) File data: file includes spare area data.
4) Use 2 partitions:
   The first partition: from 0# to199#, a total of 200 blocks, the actual length of the written data is 10 blocks
   The second partition: from 1000# to 1023#, a total of 24 blocks, the actual length of written data is 5 blocks

After the above settings, you can start programming. The results of the operation are shown in Table 1-2.

If the length of the buffer file is not enough, the programming is terminated.

If the buffer file length exceeds the length of the BLOCK (Total 15 blocks) data to be written, the programming is terminated.

The configuration interface is shown as below:



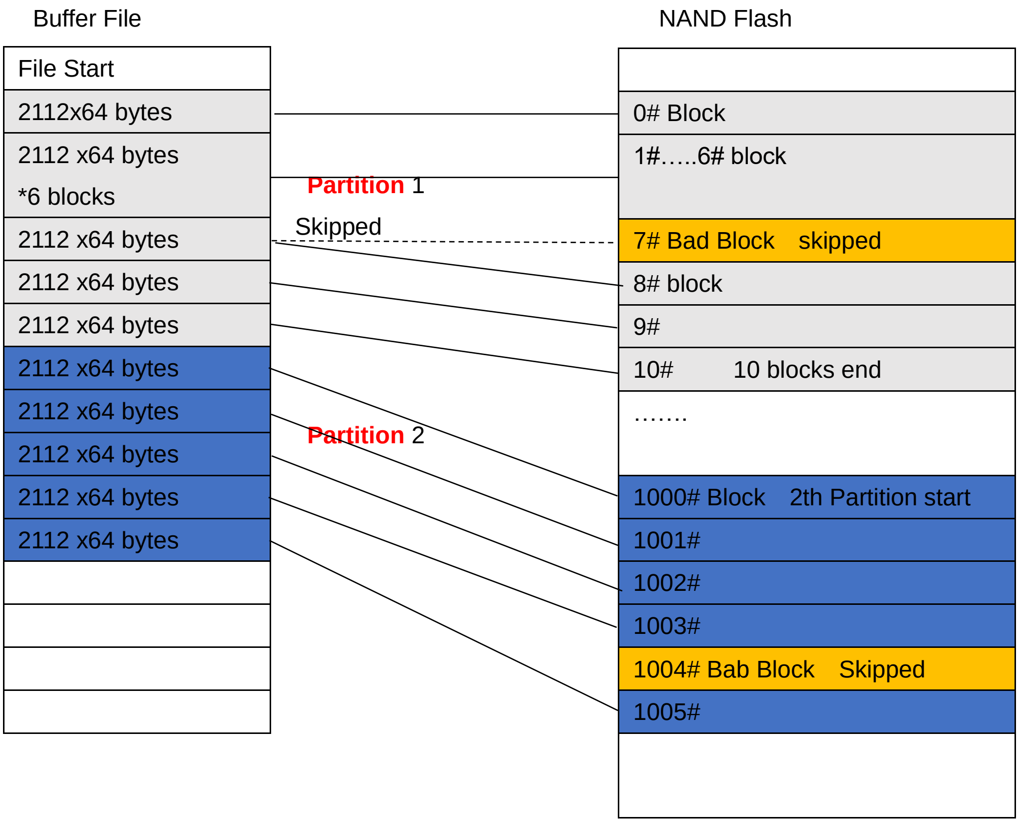
4. File name / Path / Block Assigned

Data access method: ○ Single File Mode ,All Datas
● Single File,Partitions Mode
○ Multiple Files and Partitions

Temporary Buffer Floder:
D:\Xgpro_T56\UserData\NAND

PART.	ST_BLK	END_BLK	CNT_BLK	File
☑ 1	0	199	10	D:\Xgpro_T56\UserData\NAND\Part_1.bin
☑ 2	1000	1023	5	D:\Xgpro_T56\UserData\NAND\Part_2.bin
□ 3	0	0	0	D:\Xgpro_T56\UserData\NAND\Part_3.bin
□ 4	0	0	0	D:\Xgpro_T56\UserData\NAND\Part_4.bin
□ 5	0	0	0	D:\Xgpro_T56\UserData\NAND\Part_5.bin
□ 6	0	0	0	D:\Xgpro_T56\UserData\NAND\Part_6.bin
□ 7	0	0	0	D:\Xgpro_T56\UserData\NAND\Part_7.bin
□ 8	0	0	0	D:\Xgpro_T56\UserData\NAND\Part_8.bin

Table 1-2: Programming Process diagram



Notes:

In this configuration example, when reading or verifying, it is also done by partition

1. Read 10 Blocks of partition 1 and save it to the file
2. Read 5 Blocks of partition 2 and save them to the file
3. Read finish.

By setting partitions flexibly, you can individually read or write data of one or more blocks.

6.4.9 **Configuration Example 2 (K9F1208U0C)**

Operation method: multiple files, partitions mode, replace bad blocks

K9F1208U0C parameter:

IC capacity: (512+16)*32*4096 blocks

Page Size: 512 bytes

Spare size: 16 bytes

Pages Per Block : 32 pages

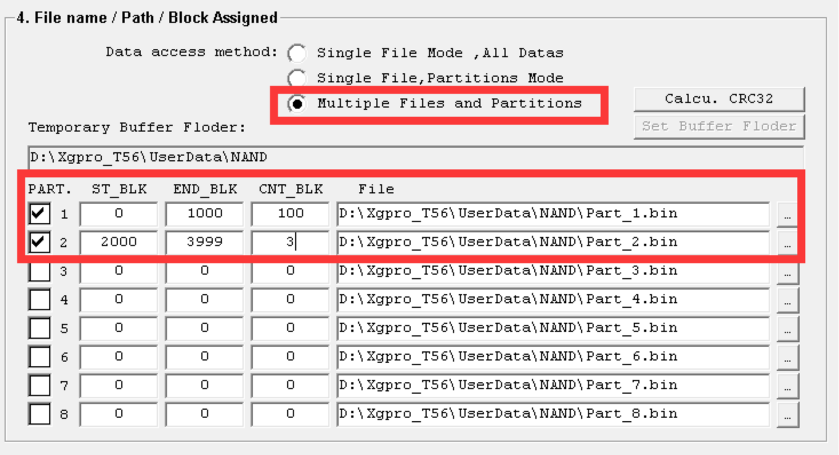
Block Size : 4096

1) Use partition table to write NAND FLASH
2) Bad block handle mode: replace block, if bad blocks are found, write to block 4000# start position
3) File data: includes spare area
4) Use 2 partitions,

   The first partition: from 1# to 1000#, a total of 1000 blocks, the actual length of the written data is 100 blocks

   The second partition: from 2000# to 3999#, a total of 1000 blocks, the actual length of written data is 3 blocks
5) The written data file is: PART_1.BIN Partition 1

   PART_2.BIN Partition 2.
6) Programming process is shown in Table 1-3.



4. File name / Path / Block Assigned

Data access method: ○ Single File Mode ,All Datas

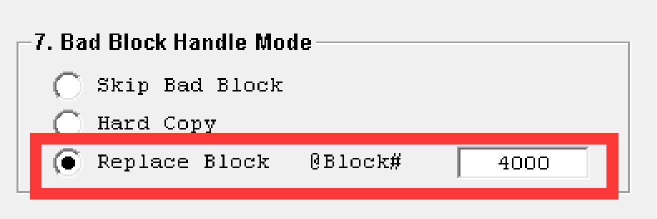
○ Single File,Partitions Mode

● Multiple Files and Partitions

Temporary Buffer Floder:

D:\Xgpro_T56\UserData\NAND

PART.		ST_BLK	END_BLK	CNT_BLK	File
☑	1	0	1000	100	D:\Xgpro_T56\UserData\NAND\Part_1.bin
☑	2	2000	3999	3	D:\Xgpro_T56\UserData\NAND\Part_2.bin
□	3	0	0	0	D:\Xgpro_T56\UserData\NAND\Part_3.bin
□	4	0	0	0	D:\Xgpro_T56\UserData\NAND\Part_4.bin
□	5	0	0	0	D:\Xgpro_T56\UserData\NAND\Part_5.bin
□	6	0	0	0	D:\Xgpro_T56\UserData\NAND\Part_6.bin
□	7	0	0	0	D:\Xgpro_T56\UserData\NAND\Part_7.bin
□	8	0	0	0	D:\Xgpro_T56\UserData\NAND\Part_8.bin

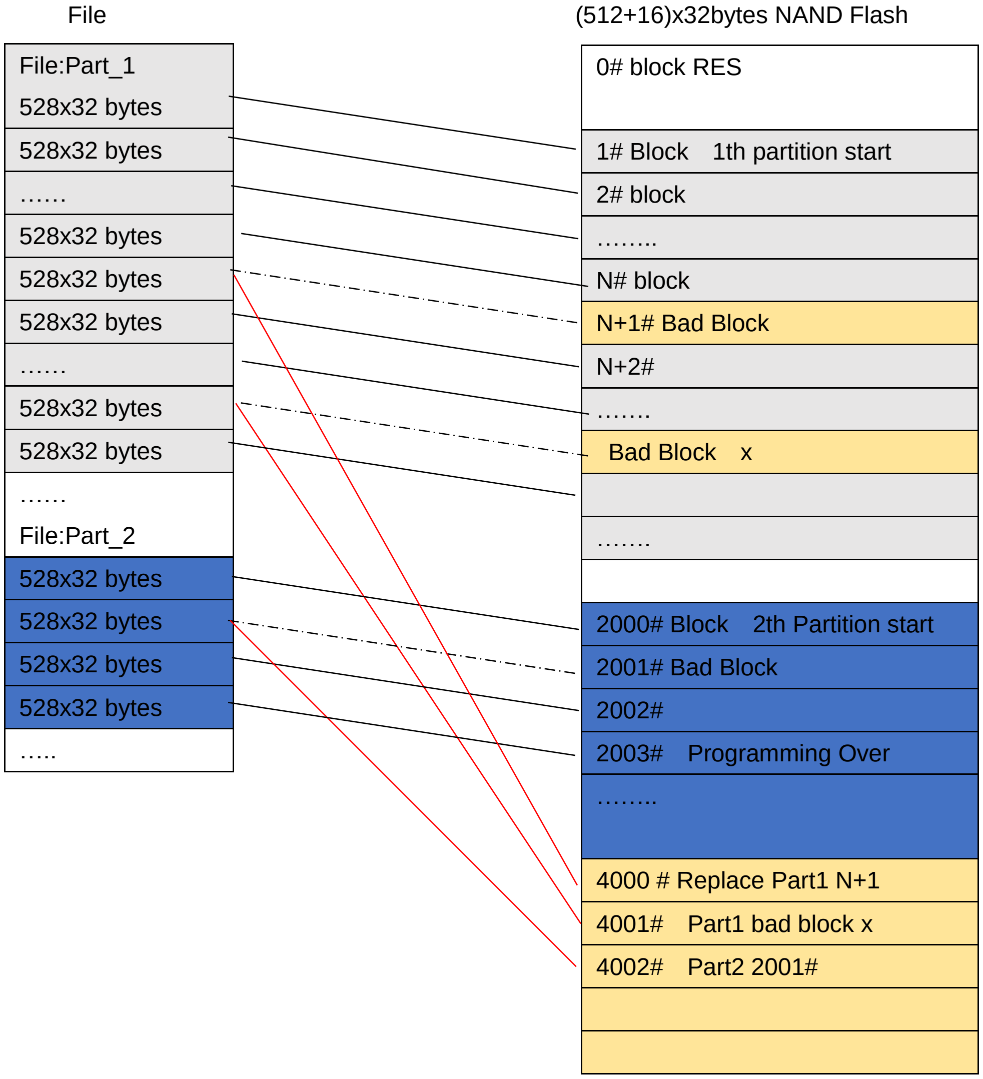


7. Bad Block Handle Mode

○ Skip Bad Block

○ Hard Copy

● Replace Block @Block# 4000



6.5 Program NAND through ISP

ISP programming only supports SPI NAND IC, select X1 mode, check option <ICSP port> to program. ISP programming process is no different from using adapter. Do not separate the three wires, while connect the two GND lines and CLK clock lines.

6.6 Save project file

For all configuration parameters, you can "Save Project" to project file. When use it again, just open the project file and you can directly program. It is suitable for mass production.

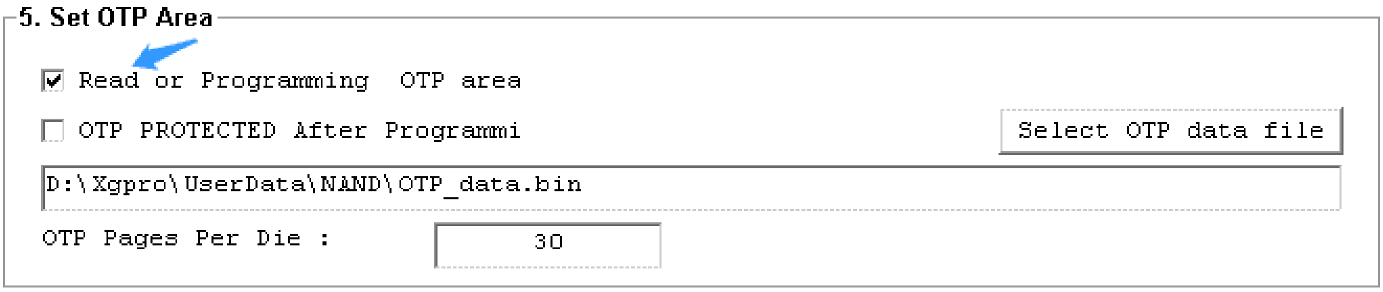
6.7 NAND Flash Copy From Original Chip

6.7.1 Notes:

In repairing equipment, we often need to read out data from an original chip, copy it to another new chip, and then solder it to the board. To ensure that the chip can work properly, users need to pay attention to the following points:

1. Copy chip configuration: Use the default settings of the programmer software.
   Option settings: must use hard copy, the data in the file must include spare area, all other options are not checked by default. When reading by this mode, the programmer can read the entire contents of the chip (including the contents of the bad block) to the file.
2. Some NAND FLASH chips have the unique ID. Embedded system applications may read the unique ID of the chip and perform encryption operations in the program, because the unique ID is set by the chip manufactory at the factory and cannot be changed or copied. In this case, even if you have copied all the contents of the chip correctly (including the OTP contents), it will not work properly after replacing with the new chip.
   For example: MT29F4G08ABAEA has the unique ID (16+16 Bytes) , which can only be read, cannot be rewritten.
   If the application is encrypted by this method, it cannot be replaced with the new chip unless you can modify the software of the embedded system.
3. OTP Area (OTP One-Time Programmable Storage Area)
   OTP Area only exists on some NAND Flash chips, please check the IC data sheet for details.
   For example: MT29F4G08ABAEA has the OTP data of 30 pages (30Pages*2112 bytes). When reading, you should read the OTP content at the same time. Check whether the OTP file is completely empty (FF), if there is OTP data, you need to write OTP at the same time when programming. Only T56 can support reading and writing OTP data.

Read the OTP content, as shown below:



5. Set OTP Area

☑ Read or Programming OTP area

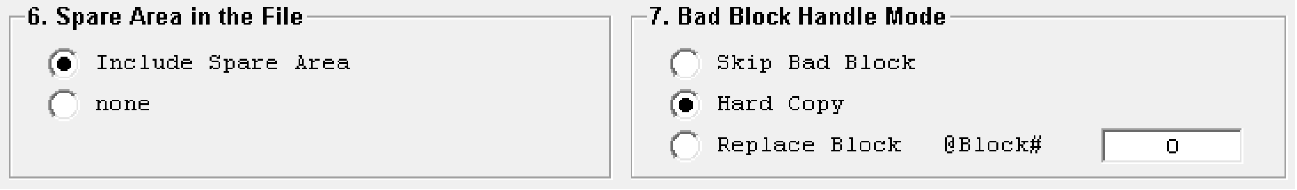
□ OTP PROTECTED After Programmi

D:\Xgpro\UserData\NAND\OTP_data.bin

OTP Pages Per Die : 30

6.7.2 Configuration and options

Default settings: hard copy, single file mode all data, Includes Spare Area (OOB)



6. Spare Area in the File

● Include Spare Area

○ none

7. Bad Block Handle Mode

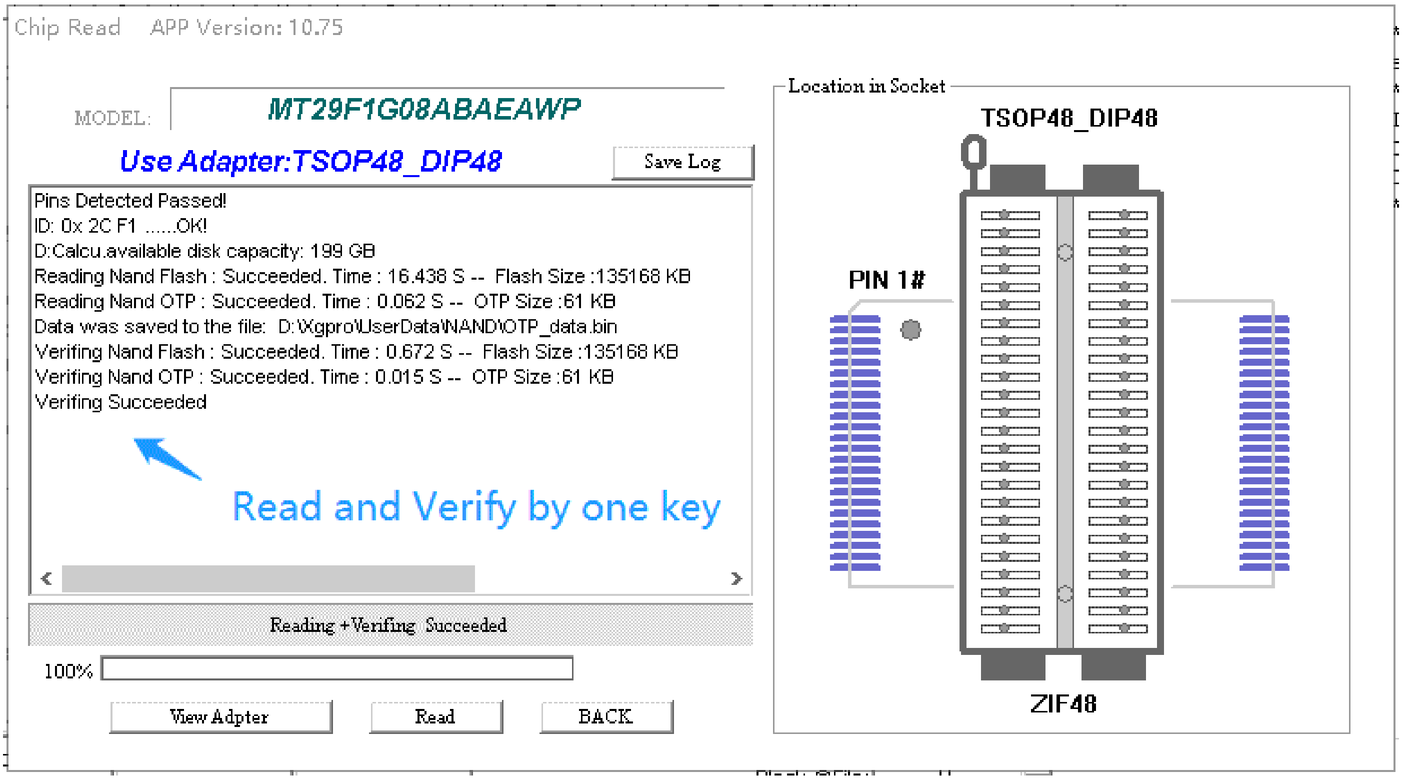
○ Skip Bad Block

● Hard Copy

○ Replace Block @Block# 0

6.7.3 Read the original chip and save

Click [Read] button in the toolbar to start reading the chip, as follows:



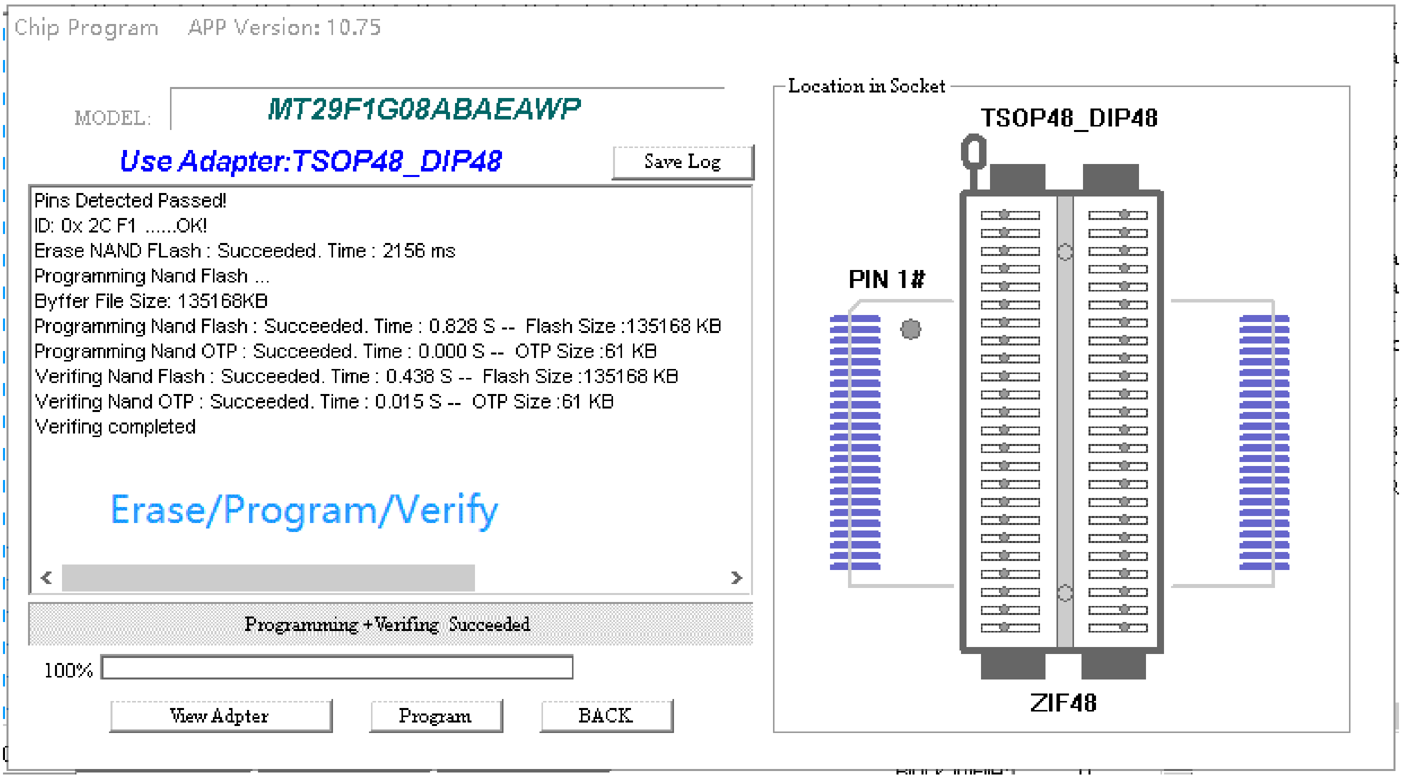
```
Pins Detected Passed!
ID: 0x 2C F1 ......OK!
D:Calcu.available disk capacity: 199 GB
Reading Nand Flash : Succeeded. Time : 16.438 S -- Flash Size :135168 KB
Reading Nand OTP : Succeeded. Time : 0.062 S -- OTP Size :61 KB
Data was saved to the file: D:\Xgpro\UserData\NAND\OTP_data.bin
Verifing Nand Flash : Succeeded. Time : 0.672 S -- Flash Size :135168 KB
Verifing Nand OTP : Succeeded. Time : 0.015 S -- OTP Size :61 KB
Verifing Succeeded
```

Exit the Read dialog box, Click **[Save]** button in the toolbar to save flash to a new file

6.7.4 6. Copy new chip

Selecting correct IC model, and **[Load]** the data file

Put on the new chip to the socket, click **[PROG.]** button in the toolbar, the following interface will pop up: click the **[PROG.]** button to start programming.



```
Pins Detected Passed!
ID: 0x 2C F1 ......OK!
Erase NAND FLash : Succeeded. Time : 2156 ms
Programming Nand Flash ...
Byffer File Size: 135168KB
Programming Nand Flash : Succeeded. Time : 0.828 S -- Flash Size :135168 KB
Programming Nand OTP : Succeeded. Time : 0.000 S -- OTP Size :61 KB
Verifing Nand Flash : Succeeded. Time : 0.438 S -- Flash Size :135168 KB
Verifing Nand OTP : Succeeded. Time : 0.015 S -- OTP Size :61 KB
Verifing completed
```

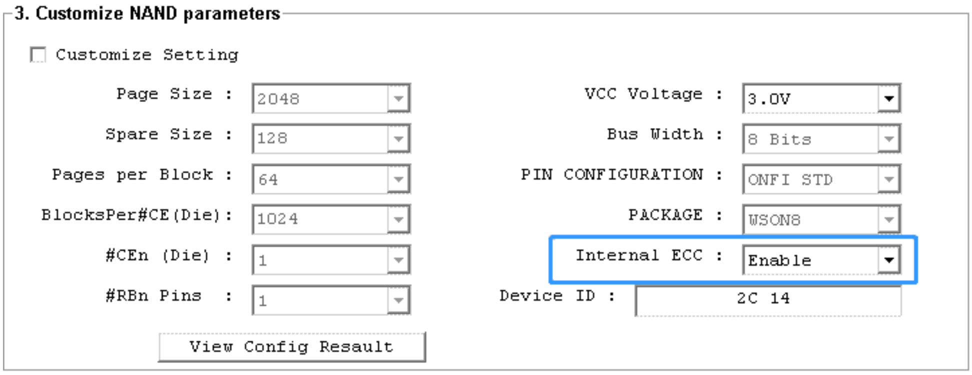
Notes:

1. It is possible that the speed of programming the chip is faster than reading, because the software automatically skips the content of the empty data during programming.
2. In the process of reading, erasing, or programming, if chip has bad blocks, the tip will appear, but it does not affect the programming of the chip.
3. It is not necessary to check the bad block before copying. As long as the verifying is correct and there is no error when programming, the new copied chip will be available. If there is a bad block in the new chip, the software will skip writing if the bad block is empty data when copying. If the new chip has a bad block, and the original chip has data to be written to the location of this bad block, the data can’t be written into this bad block, the programming error will occur. Please replace with another new chip, (the location of the new chip bad block is generally random).
4. Repairers usually do not know how to deal with bad blocks, so only copy directly with hard copy mode.

6.8 NAND internal ECC setting

Notes: (ECC option is disabled for chips without ECC function)

1. All ECC is enabled by default, if you want, you can uncheck the option.
   Enable Internal ECC is the best choice, high reliability, the data read every time is the same, there is no bit flipping.
2. Enable ECC Read + Enable ECC Write, (same reliability as the original chip)
3. Enable ECC Read + Disable ECC write (data may be flipped, less reliable than original chip data)
4. Disable ECC Read + Enable ECC write **(don’t use this method)**
5. Disable ECC Reading + Disable ECC writing, the probability of flipping is twice as high as the third method, and the possibility of error is greater, and the reliability is worse than that of the original chip data.
6. Chip has internal ECC correction, but the device does not use internal ECC correction, Must disable internal ECC when reading or writing. This situation should be rare, if reading and writing via enabled ECC, the chip does not work normally, test with [disable ECC read + disable ECC write]. ECC function switching is shown in the following figure:



3. Customize NAND parameters

☐ Customize Setting

Page Size : 2048

Spare Size : 128

Pages per Block : 64

BlocksPer#CE(Die): 1024

#CEn (Die) : 1

#RBn Pins : 1

VCC Voltage : 3.0V

Bus Width : 8 Bits

PIN CONFIGURATION : ONFI STD

PACKAGE : WSON8

Internal ECC : Enable

Device ID : 2C 14

6.9 Calculation of NAND file address

Calculated in two cases:

- File includes Spare Area
  The location of the data in the file = n Block x (Page Size + Spare Size) x the number of pages per block
- File does not include Spare Area
  Location of data in the file = n Block x Page Size x the number of pages per block

7. VGA_HDMI ISP and Tools

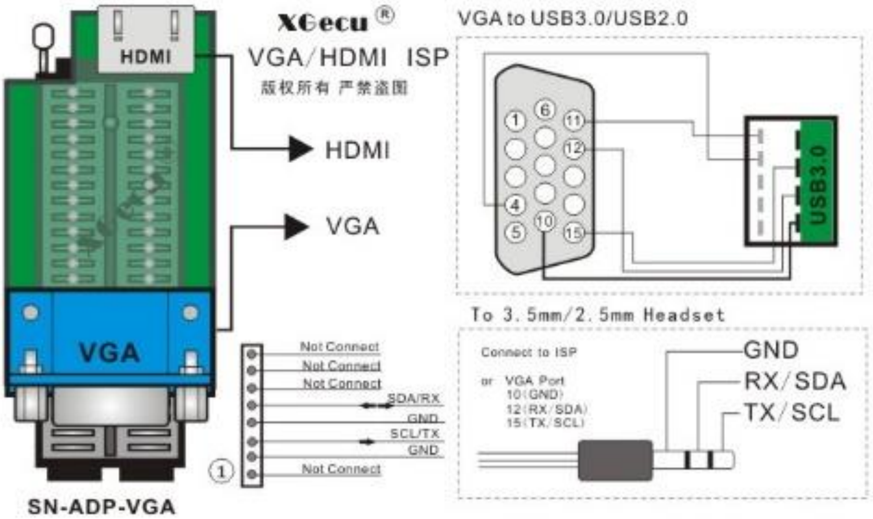
Only T56 programmer can support this function

7.1 VGA_HDMI ISP wiring diagram

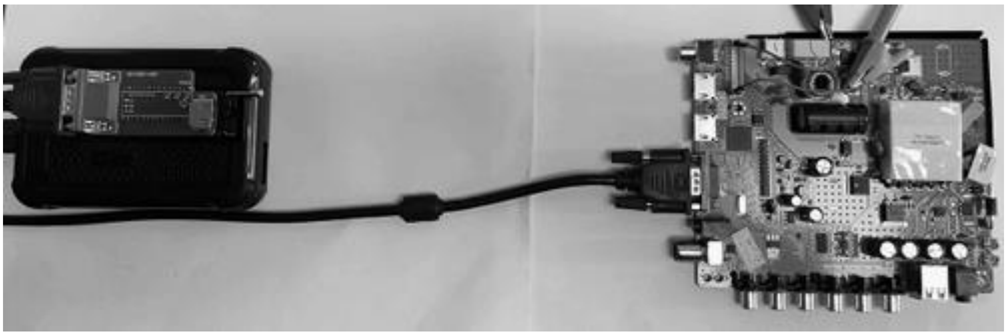
Use the VGA_HDMI adapter to connect a VGA cable or HDMI cable, or ISP cable to the TV motherboard. When in-circuit programming, the TV motherboard is normally required to be in standby mode and to use external power supply.

General wiring diagram as below:

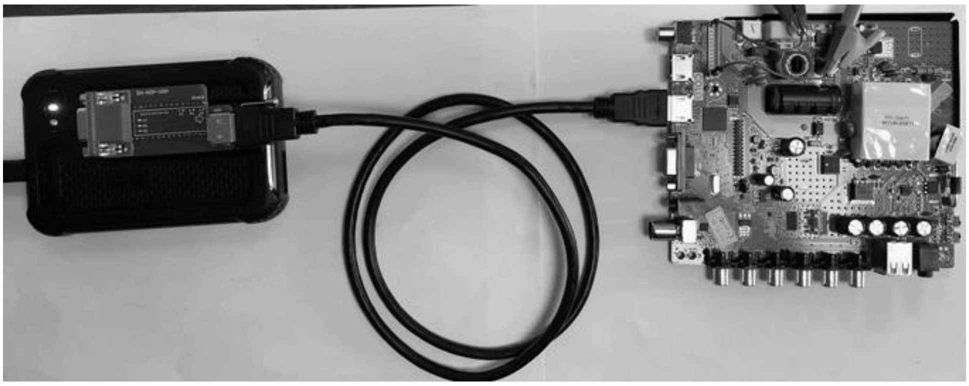
According to the requirements of the motherboard, VGA in-circuit programming can be used in a variety of ways to connect. T56 can automatically recognize the hardware connection, do not need to make any settings in the software.



Connection photo via VGA cable

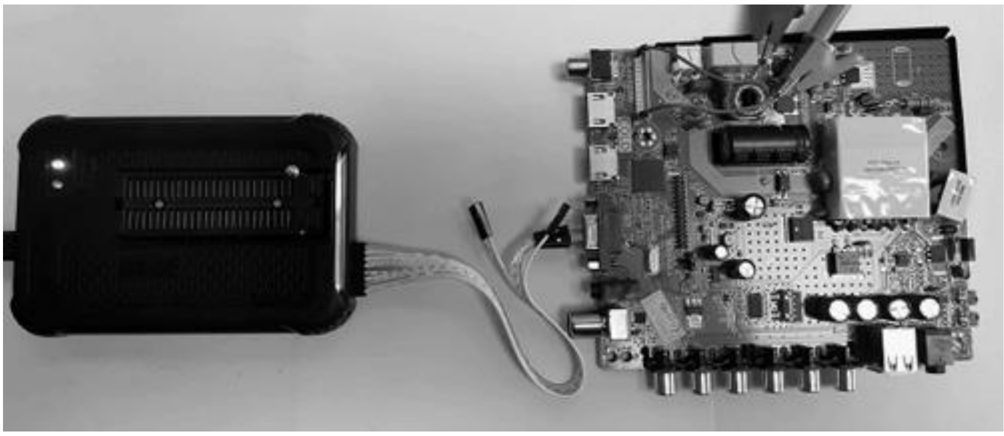


Connection photo via HDMI



Connection photo via ISP:

GND -->VGA 10# or VGA shell / SDA -->VGA 12# / CLK -->VGA 15#



7.2 VGA_HDMI in-circuit programming

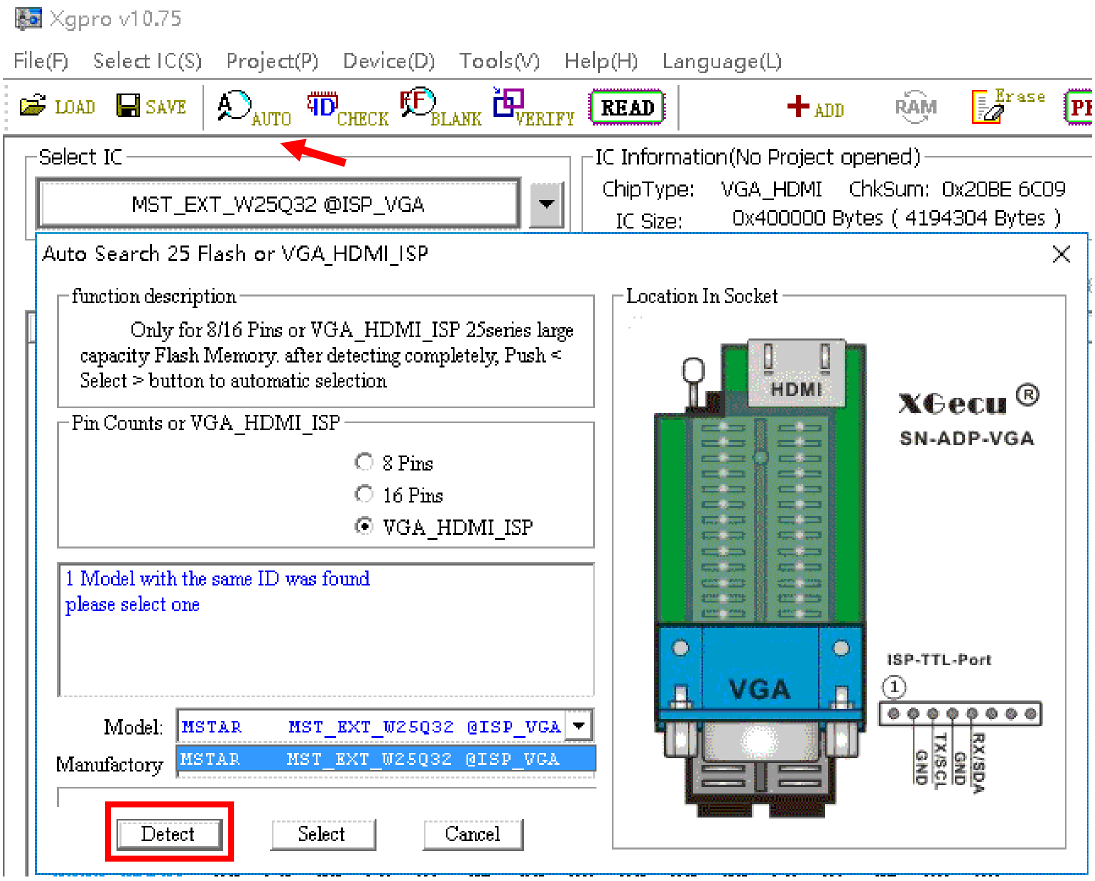
Note: VGA in-circuit programming 25 FLASH, currently only support **MSTAR** part of the motherboard, in the subsequent update will be gradually improved

Connected to the motherboard, power supply to the motherboard, keep TV in standby state

7.2.1 **Auto-identify Chip**

Click **[AUTO]** button, the dialog box will pop up, select **VGA _HDMI_ISP** port.

Click **[Detect]** button to automatically find the chip model and select the chip



Xgpro v10.75
File(F) Select IC(S) Project(P) Device(D) Tools(V) Help(H) Language(L)
LOAD SAVE AUTO ID CHECK BLANK VERIFY READ ADD RAM Erase PI
XGecu ®
SN-ADP-VGA
HDMI
VGA
ISP-TTL-Port
①
GND
TX/SCL
GND
RX/SDA

Select IC: MST_EXT_W25Q32 @ISP_VGA

IC Information(No Project opened)
ChipType: VGA_HDMI ChkSum: 0x20BE 6C09
IC Size: 0x400000 Bytes (4194304 Bytes)

Auto Search 25 Flash or VGA_HDMI_ISP

function description
Only for 8/16 Pins or VGA_HDMI_ISP 25series large capacity Flash Memory. after detecting completely, Push < Select > button to automatic selection

Pin Counts or VGA_HDMI_ISP
○ 8 Pins
○ 16 Pins
● VGA_HDMI_ISP

1 Model with the same ID was found
please select one

Model: MSTAR MST_EXT_W25Q32 @ISP_VGA
Manufactory MSTAR MST_EXT_W25Q32 @ISP_VGA

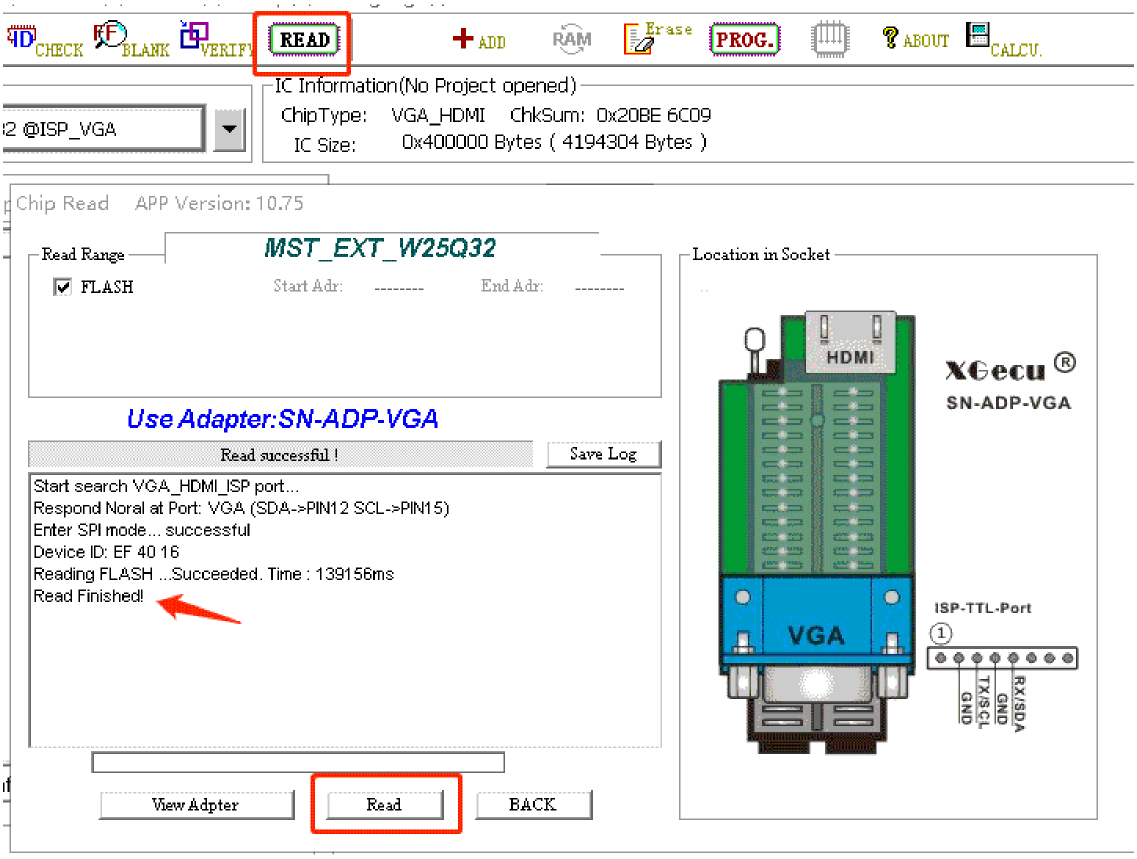
Location In Socket

Detect Select Cancel

7.2.2 In-circuit Read chip data

Click **[Read]** button, the read dialog box pops up, click **[Read]**

When reading, it will automatically find the motherboard communication port and protocol to enter SPI mode.



2 @ISP_VGA

IC Information(No Project opened)
ChipType: VGA_HDMI ChkSum: 0x20BE 6C09
IC Size: 0x400000 Bytes (4194304 Bytes)

Chip Read APP Version: 10.75

Read Range MST_EXT_W25Q32
☑ FLASH Start Adr: -------- End Adr: --------

Use Adapter:SN-ADP-VGA

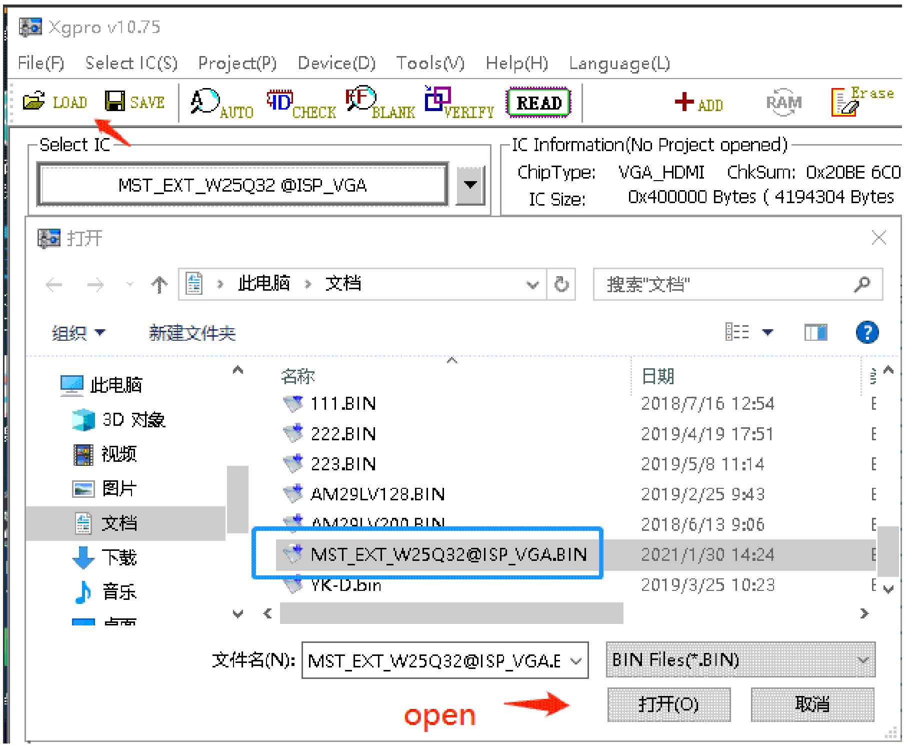
Read successful !

```
Start search VGA_HDMI_ISP port...
Respond Noral at Port: VGA (SDA->PIN12 SCL->PIN15)
Enter SPI mode... successful
Device ID: EF 40 16
Reading FLASH ...Succeeded. Time : 139156ms
Read Finished!
```

Read time 137 seconds, click **[Back],** please **[Save]** the data

7.2.3 VGA in-circuit programming

Click **[LOAD]** in the tool bar to load the data file to be written, as shown in the following figure:



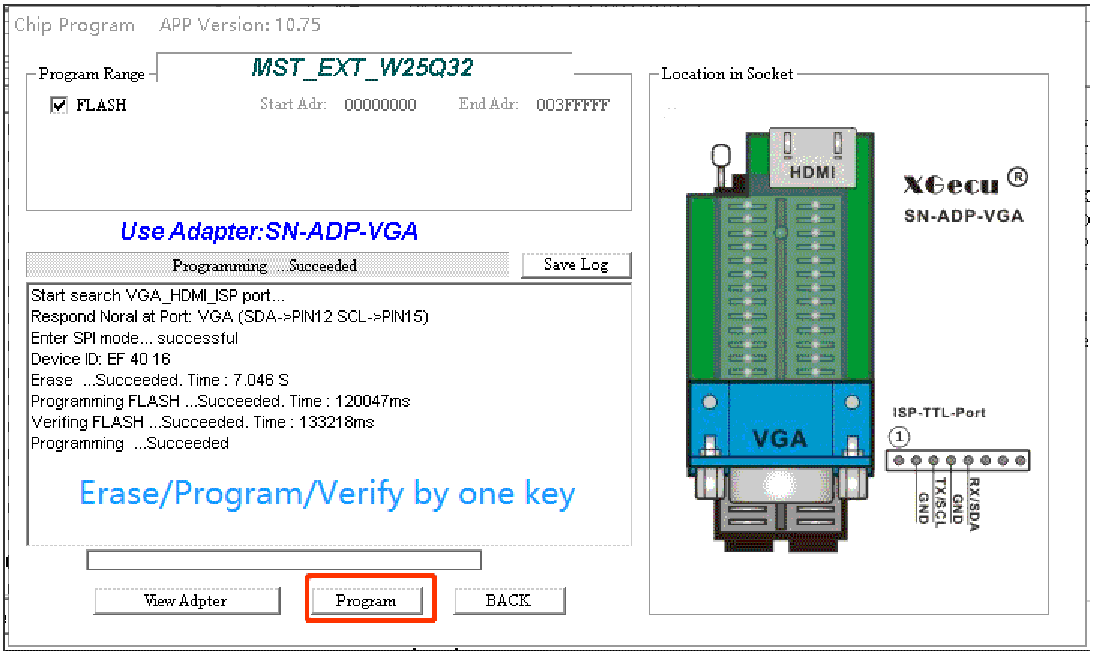
Select IC
MST_EXT_W25Q32 @ISP_VGA

IC Information(No Project opened)
ChipType: VGA_HDMI ChkSum: 0x20BE 6C0
IC Size: 0x400000 Bytes (4194304 Bytes

名称	日期
111.BIN	2018/7/16 12:54
222.BIN	2019/4/19 17:51
223.BIN	2019/5/8 11:14
AM29LV128.BIN	2019/2/25 9:43
AM29LV200.BIN	2018/6/13 9:06
MST_EXT_W25Q32@ISP_VGA.BIN	2021/1/30 14:24
YK-D.bin	2019/3/25 10:23

文件名(N): MST_EXT_W25Q32@ISP_VGA.E
BIN Files(*.BIN)

After loading the file, click **[PROG.]** in the toolbar to bring up the programming dialog box. Click **[Program]** to start programming, as shown in the following figure:



Chip Program APP Version: 10.75

MST_EXT_W25Q32

Program Range

☑ FLASH Start Adr: 00000000 End Adr: 003FFFFF

Use Adapter:SN-ADP-VGA

Programming ...Succeeded

```
Start search VGA_HDMI_ISP port...
Respond Noral at Port: VGA (SDA->PIN12 SCL->PIN15)
Enter SPI mode... successful
Device ID: EF 40 16
Erase  ...Succeeded. Time : 7.046 S
Programming FLASH ...Succeeded. Time : 120047ms
Verifing FLASH ...Succeeded. Time : 133218ms
Programming  ...Succeeded
```

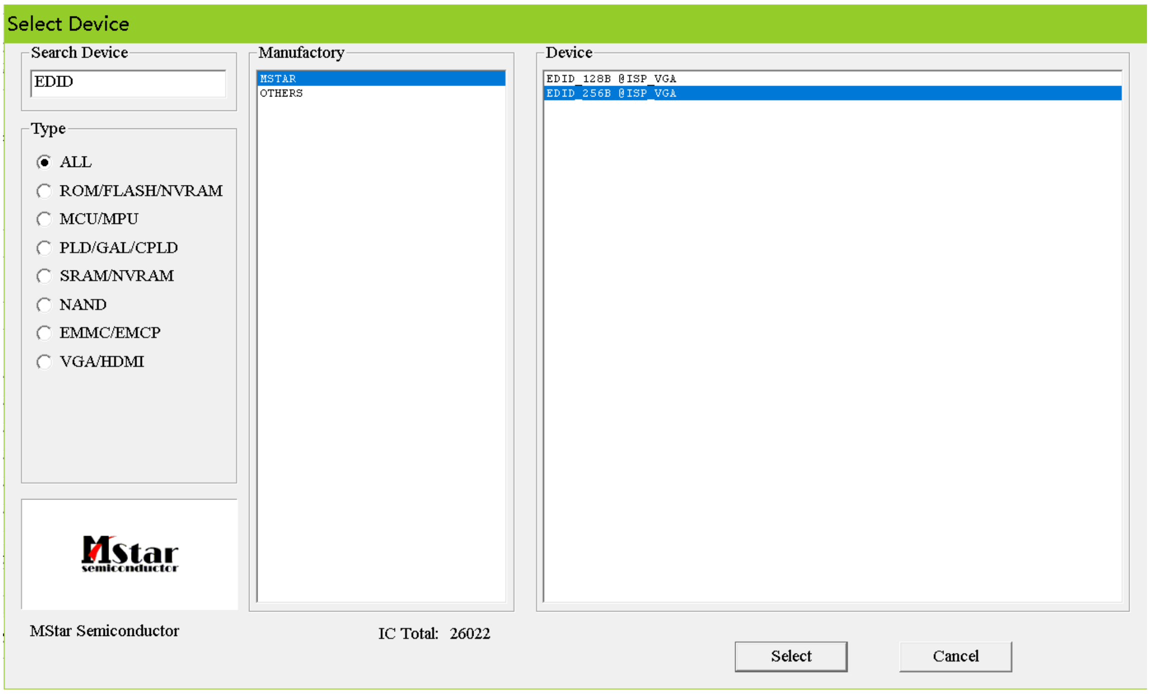
Erase/Program/Verify by one key

The whole programming process is completed

7.3 EDID Programming

Connect TV or monitor, click **[Select IC]** button, in the search device, type in **"EDID"**.

Select **EDID_256B @ISP_VGA**, as shown below:



Select Device

Search Device: EDID

Type

● ALL

○ ROM/FLASH/NVRAM

○ MCU/MPU

○ PLD/GAL/CPLD

○ SRAM/NVRAM

○ NAND

○ EMMC/EMCP

○ VGA/HDMI

Manufactory

- MSTAR
- OTHERS

Device

- EDID_128B @ISP_VGA
- EDID_256B @ISP_VGA

MStar Semiconductor

IC Total: 26022

After selecting, you can read, modify, program and verify the EDID.

The operation method is exactly the same as ordinary FLASH chips, so not describe here.

7.4 TV Tools: Serial Printing

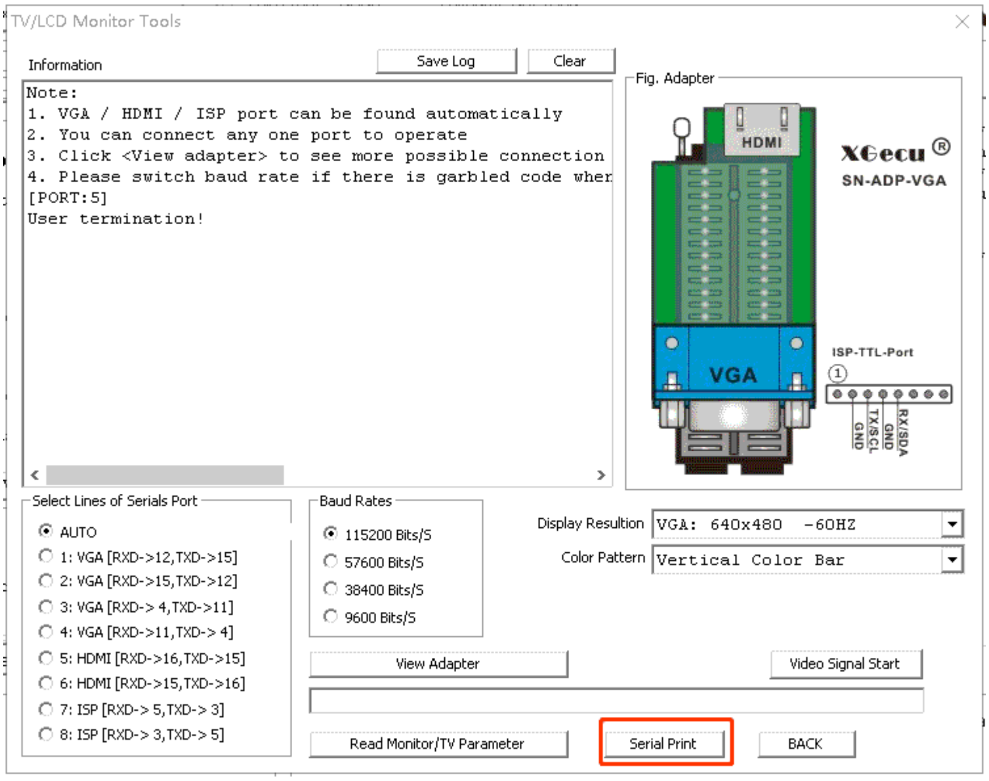
Connect the TV board, click the icon TV in the upper right corner

Click the **[Serial Print]** button and T56 will get started to receive power-on diagnostic information from the TV board.

Turn on the TV power switch. The power-on information will be displayed on the screen, you can click **[Save Log]** to save.

The line sequence is auto found by default, do not need to select it manually.

Baud rate is 115200bps by default, please switch the baud rate if there is a messy code, as follows:



```
Note:
1. VGA / HDMI / ISP port can be found automatically
2. You can connect any one port to operate
3. Click <View adapter> to see more possible connection
4. Please switch baud rate if there is garbled code wher
[PORT:5]
User termination!
```

Select Lines of Serials Port
- ● AUTO
- ○ 1: VGA [RXD->12,TXD->15]
- ○ 2: VGA [RXD->15,TXD->12]
- ○ 3: VGA [RXD-> 4,TXD->11]
- ○ 4: VGA [RXD->11,TXD-> 4]
- ○ 5: HDMI [RXD->16,TXD->15]
- ○ 6: HDMI [RXD->15,TXD->16]
- ○ 7: ISP [RXD-> 5,TXD-> 3]
- ○ 8: ISP [RXD-> 3,TXD-> 5]

Baud Rates
- ● 115200 Bits/S
- ○ 57600 Bits/S
- ○ 38400 Bits/S
- ○ 9600 Bits/S

Display Resultion: VGA: 640x480 -60HZ

Color Pattern: Vertical Color Bar

Click **[STOP]** button to stop receiving messages.

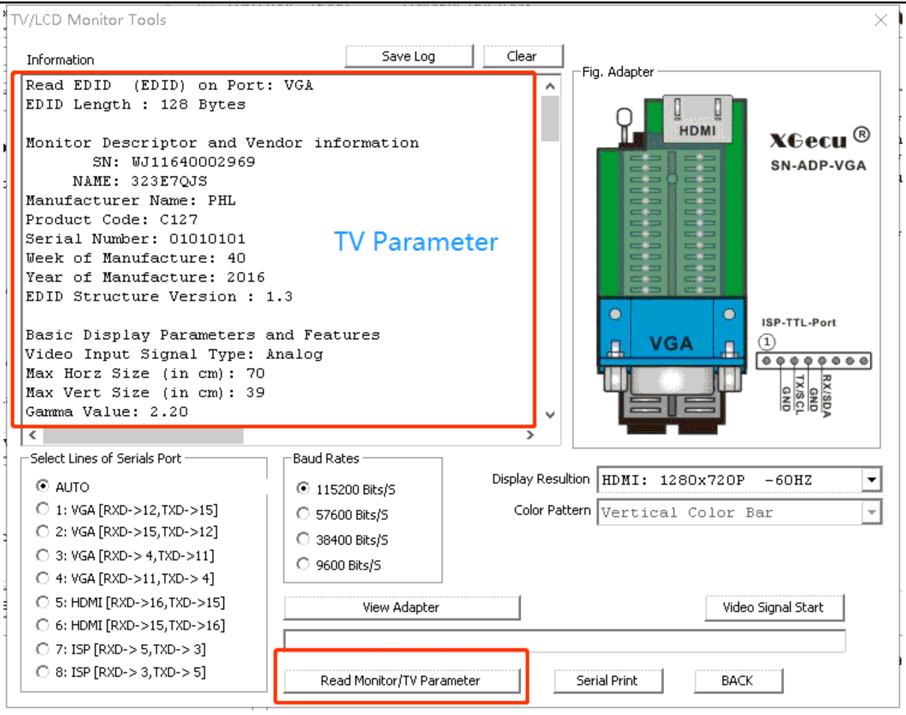
7.5 TV tools: read monitor or TV parameters

Connect TV or monitor, power standby state.

Click on the icon TV in the upper right corner, as below,

click on **[Read Monitor/TV Parameter]**

Show as follow:



```
Read EDID  (EDID) on Port: VGA
EDID Length : 128 Bytes

Monitor Descriptor and Vendor information
       SN: WJ11640002969
     NAME: 323E7QJS
Manufacturer Name: PHL
Product Code: C127
Serial Number: 01010101
Week of Manufacture: 40
Year of Manufacture: 2016
EDID Structure Version : 1.3

Basic Display Parameters and Features
Video Input Signal Type: Analog
Max Horz Size (in cm): 70
Max Vert Size (in cm): 39
Gamma Value: 2.20
```

Select Lines of Serials Port
- ● AUTO
- ○ 1: VGA [RXD->12,TXD->15]
- ○ 2: VGA [RXD->15,TXD->12]
- ○ 3: VGA [RXD-> 4,TXD->11]
- ○ 4: VGA [RXD->11,TXD-> 4]
- ○ 5: HDMI [RXD->16,TXD->15]
- ○ 6: HDMI [RXD->15,TXD->16]
- ○ 7: ISP [RXD-> 5,TXD-> 3]
- ○ 8: ISP [RXD-> 3,TXD-> 5]

Baud Rates
- ● 115200 Bits/S
- ○ 57600 Bits/S
- ○ 38400 Bits/S
- ○ 9600 Bits/S

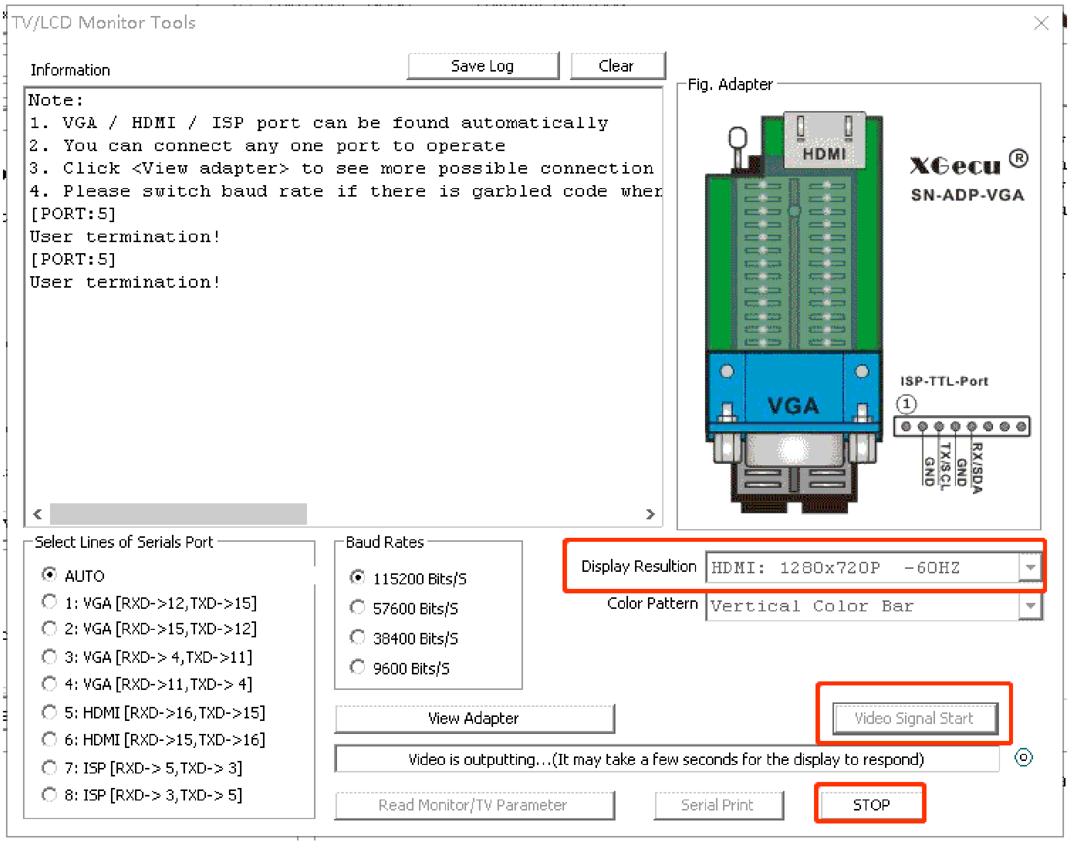
Display Resultion HDMI: 1280x720P -60HZ

Color Pattern Vertical Color Bar

7.6 TV tools: HDMI, VGA video test signal

Video test signal connection, you must use the T56 special SN-ADP-VGA adapter to connect VGA or HDMI

Click the icon TV in the upper right corner of the window, click **[Video Signal Start]** button, various kinds of VGA or HDMI signal will be out.



```
Note:
1. VGA / HDMI / ISP port can be found automatically
2. You can connect any one port to operate
3. Click <View adapter> to see more possible connection
4. Please switch baud rate if there is garbled code wher
[PORT:5]
User termination!
[PORT:5]
User termination!
```

Select Lines of Serials Port
- ● AUTO
- ○ 1: VGA [RXD->12,TXD->15]
- ○ 2: VGA [RXD->15,TXD->12]
- ○ 3: VGA [RXD-> 4,TXD->11]
- ○ 4: VGA [RXD->11,TXD-> 4]
- ○ 5: HDMI [RXD->16,TXD->15]
- ○ 6: HDMI [RXD->15,TXD->16]
- ○ 7: ISP [RXD-> 5,TXD-> 3]
- ○ 8: ISP [RXD-> 3,TXD-> 5]

Baud Rates
- ● 115200 Bits/S
- ○ 57600 Bits/S
- ○ 38400 Bits/S
- ○ 9600 Bits/S

Display Resultion HDMI: 1280x720P -60HZ

Color Pattern Vertical Color Bar

Video is outputting…(It may take a few seconds for the display to respond)

Click **[STOP]** and back

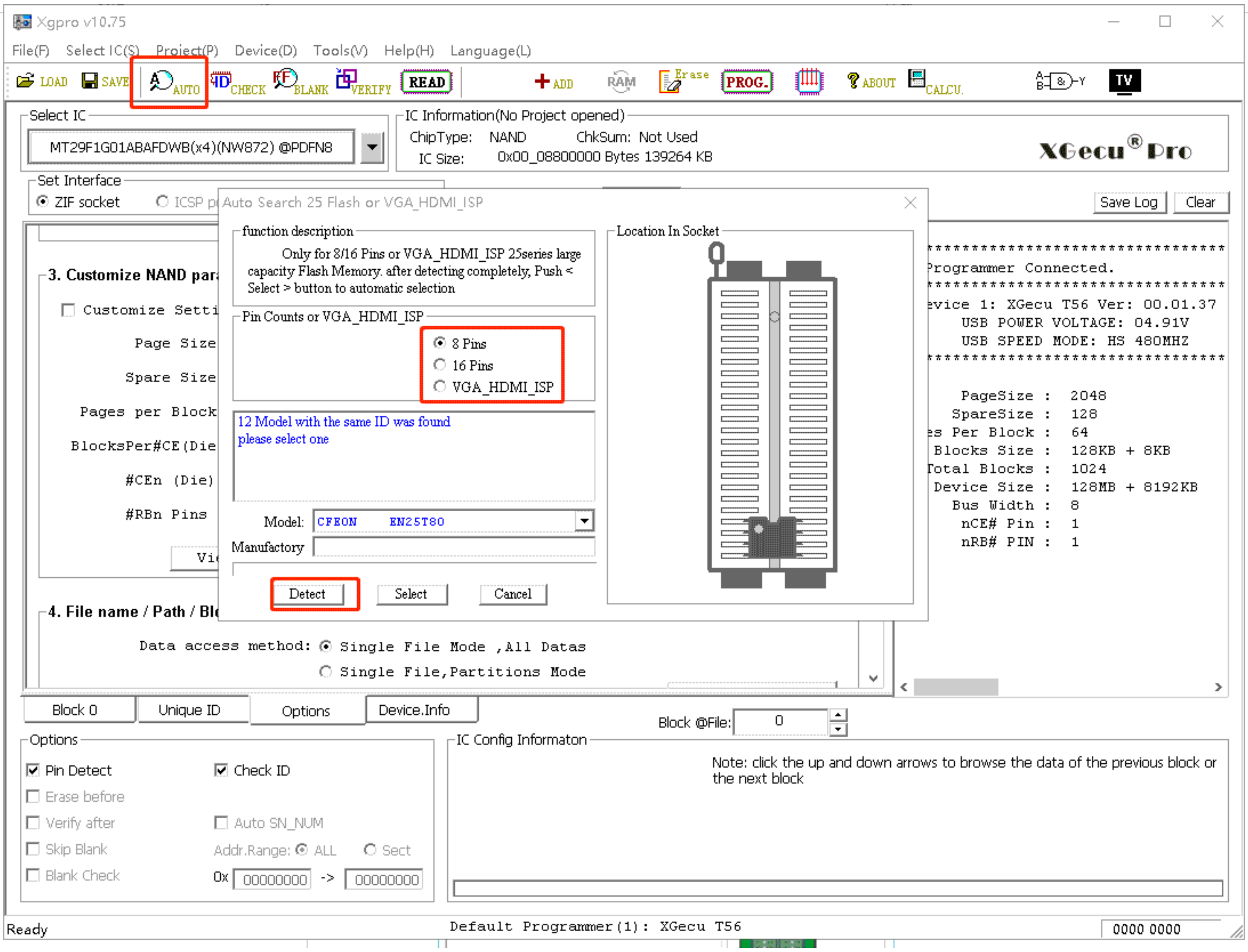
8. Automatic Identification

The automatic-identificatiom is for the SPI 25 series flash. Realize fast selection of 25 norflash chips.

Identification interface: 8-pin and 16-pin chip on the ZIF socket, T56 supports VGA_HDMI ISP interface

8.1 ZIF Socket Identification

Click the **[AUTO]** icon AUTO button in the toolbar, as shown below:


Xgpro v10.75
File(F) Select IC(S) Project(P) Device(D) Tools(V) Help(H) Language(L)
LOAD SAVE AUTO CHECK BLANK VERIFY READ ADD RAM Erase PROG. ABOUT CALCU. TV
XGecu® Pro
Save Log Clear
Detect Select Cancel
Block 0 Unique ID Options Device.Info

Select IC: MT29F1G01ABAFDWB(x4)(NW872) @PDFN8

IC Information(No Project opened)
ChipType: NAND ChkSum: Not Used
IC Size: 0x00_08800000 Bytes 139264 KB

Set Interface: ● ZIF socket ○ ICSP p

Auto Search 25 Flash or VGA_HDMI_ISP

function description: Only for 8/16 Pins or VGA_HDMI_ISP 25series large capacity Flash Memory. after detecting completely, Push < Select > button to automatic selection

Pin Counts or VGA_HDMI_ISP: ● 8 Pins ○ 16 Pins ○ VGA_HDMI_ISP

12 Model with the same ID was found
please select one

Model: CFEON EN25T80
Manufactory:

Location In Socket

3. Customize NAND para
□ Customize Setti
Page Size
Spare Size
Pages per Block
BlocksPer#CE(Die
#CEn (Die)
#RBn Pins

4. File name / Path / Bl
Data access method: ● Single File Mode ,All Datas
○ Single File,Partitions Mode

```
**********************************
Programmer Connected.
**********************************
evice 1: XGecu T56 Ver: 00.01.37
     USB POWER VOLTAGE: 04.91V
     USB SPEED MODE: HS 480MHZ
**********************************

     PageSize :  2048
    SpareSize :  128
es Per Block :  64
  Blocks Size :  128KB + 8KB
Total Blocks :  1024
  Device Size :  128MB + 8192KB
    Bus Width :  8
     nCE# Pin :  1
     nRB# PIN :  1
```

Block @File: 0

Options
☑ Pin Detect ☑ Check ID
□ Erase before
□ Verify after □ Auto SN_NUM
□ Skip Blank Addr.Range: ● ALL ○ Sect
□ Blank Check 0x 00000000 -> 00000000

IC Config Informaton
Note: click the up and down arrows to browse the data of the previous block or the next block

Ready Default Programmer(1): XGecu T56 0000 0000

- Select 8-pin or 16-pin chip
- Click **[Detect],** if the chip is recognized, the list of chip with the same ID will be displayed, select the appropriate one
- Click **[Select]** button in the dialog box, selecting IC is finished, next to reading or programming

8.2 VGA_HDMI In-circuit Identification

Click **[AUTO]** icon AUTO button in the toolbar to check "the VGA_HDMI ISP" interface.

For details, see: Chapter 7 VGA_HDMI ISP in-circuit programming.

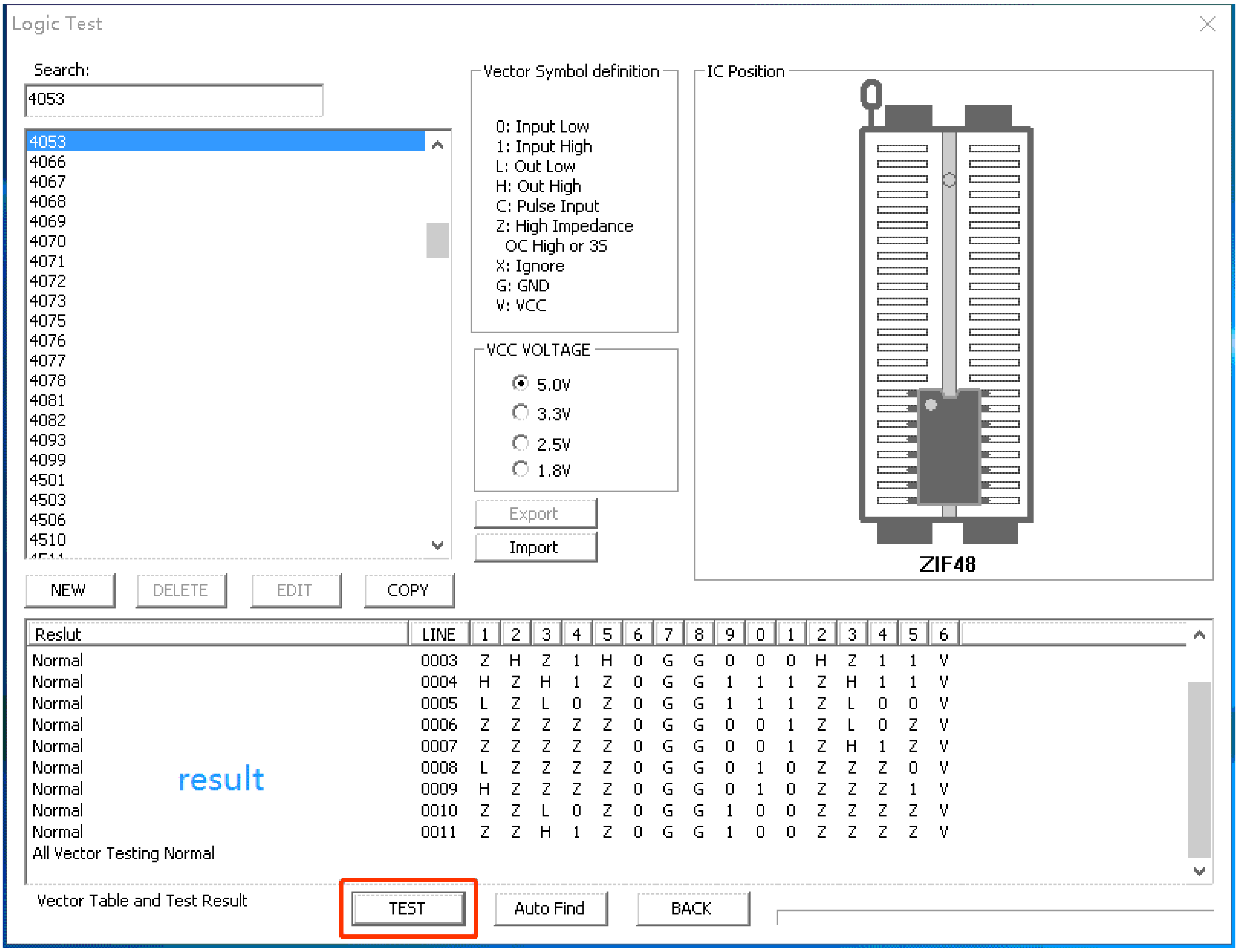
9.Digital Logic IC Testing

Functions:

- Test 54/74 series, CD4xxx digital logic chips, test results positioned to the pin. Adjustable VCC voltage.
- Automatically identify the logic chip.
- User-defined test vectors.

9.1 Test Logic Chip

Click **[Logic Test]** icon in the toolbar to bring up the test dialog box, as below:



Logic Test

Search: 4053

4053
4066
4067
4068
4069
4070
4071
4072
4073
4075
4076
4077
4078
4081
4082
4093
4099
4501
4503
4506
4510

Vector Symbol definition

0: Input Low
1: Input High
L: Out Low
H: Out High
C: Pulse Input
Z: High Impedance
OC High or 3S
X: Ignore
G: GND
V: VCC

VCC VOLTAGE

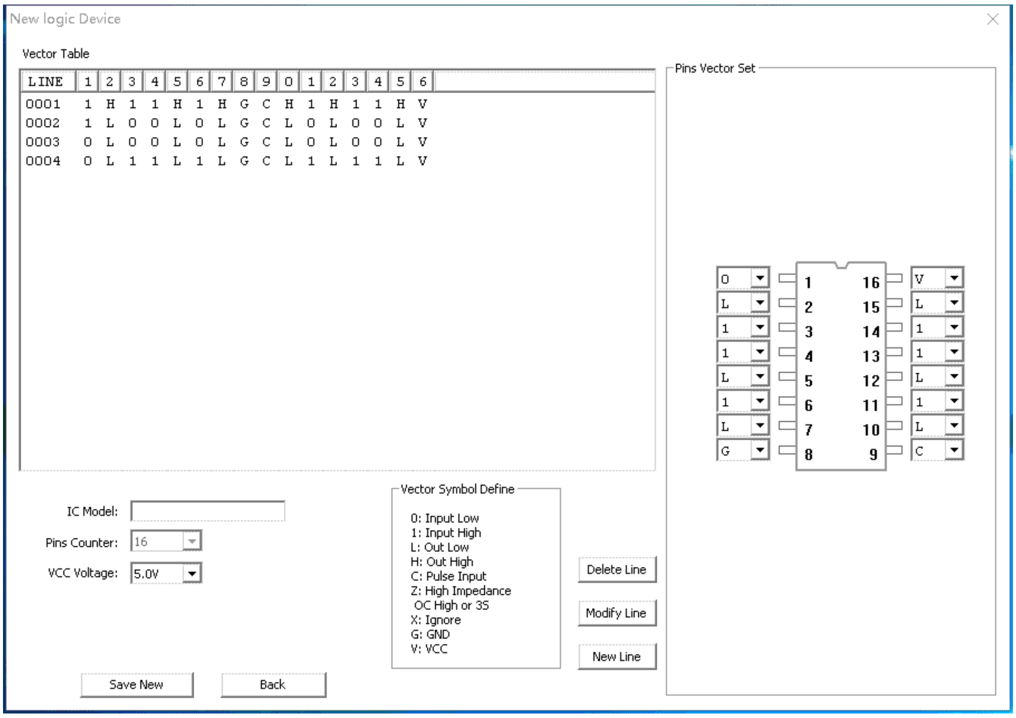
● 5.0V
○ 3.3V
○ 2.5V
○ 1.8V

Reslut	LINE	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
Normal	0003	Z	H	Z	1	H	0	G	G	0	0	0	H	Z	1	1	V
Normal	0004	H	Z	H	1	Z	0	G	G	1	1	1	Z	H	1	1	V
Normal	0005	L	Z	L	0	Z	0	G	G	1	1	1	Z	L	0	0	V
Normal	0006	Z	Z	Z	Z	Z	0	G	G	0	0	1	Z	L	0	Z	V
Normal	0007	Z	Z	Z	Z	Z	0	G	G	0	0	1	Z	H	1	Z	V
Normal	0008	L	Z	Z	Z	Z	0	G	G	0	1	0	Z	Z	Z	0	V
Normal	0009	H	Z	Z	Z	Z	0	G	G	0	1	0	Z	Z	Z	1	V
Normal	0010	Z	Z	L	0	Z	0	G	G	1	0	0	Z	Z	Z	Z	V
Normal	0011	Z	Z	H	1	Z	0	G	G	1	0	0	Z	Z	Z	Z	V
All Vector Testing Normal																	

- After selecting IC model, directly click **[TEST]** button to see the test results of the chip vector
- VCC voltage is variable
- Click the **[NEW]** or **[COPY]** button to add a new customized chip
- The customized chips can be imported or exported for sharing. You can also delete or modify.

9.2 Customize the logic chip

Click **[NEW]** or **[COPY]** button to edit the logic test vector table, As figure shows:



New logic Device

Vector Table

```
LINE  1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6
0001  1 H 1 1 H 1 H G C H 1 H 1 1 H V
0002  1 L 0 0 L 0 L G C L 0 L 0 0 L V
0003  0 L 0 0 L 0 L G C L 0 L 0 0 L V
0004  0 L 1 1 L 1 L G C L 1 L 1 1 L V
```

Pins Vector Set

0	1	16	V
L	2	15	L
1	3	14	1
1	4	13	1
L	5	12	L
1	6	11	1
L	7	10	L
G	8	9	C

IC Model:

Pins Counter: 16

VCC Voltage: 5.0V

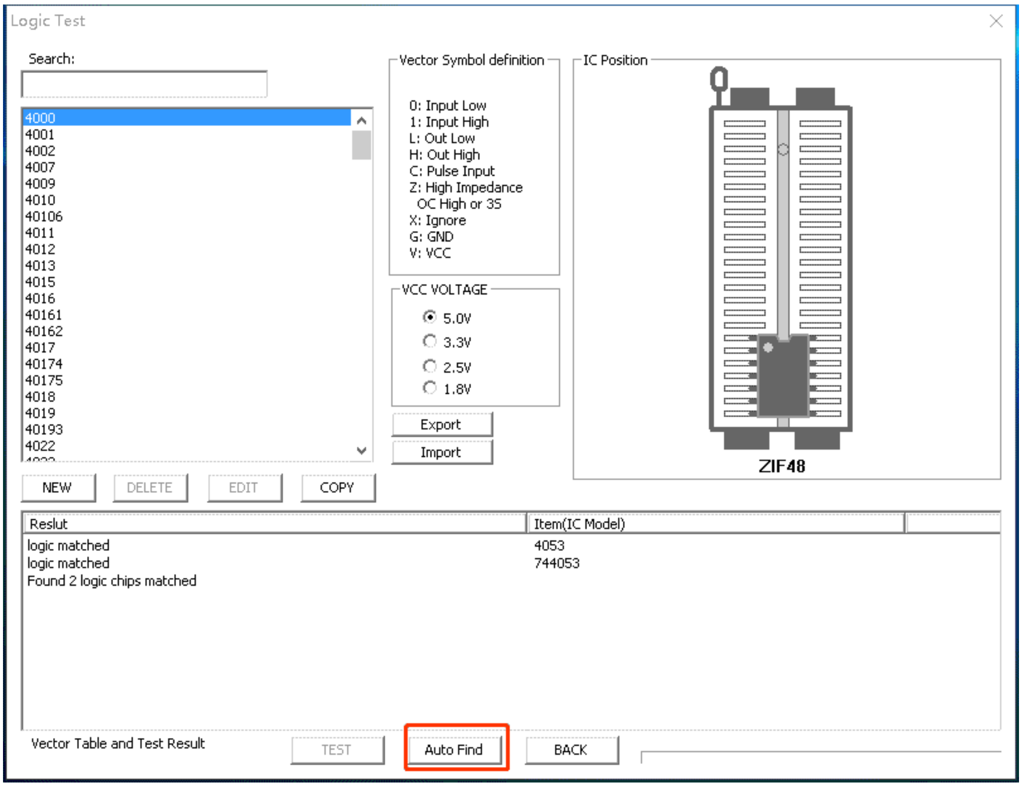
Vector Symbol Define

0: Input Low
1: Input High
L: Out Low
H: Out High
C: Pulse Input
Z: High Impedance
OC High or 3S
X: Ignore
G: GND
V: VCC

Modify and add the test vector table. then type in the new chip model and save the new.

9.3 Logic chip identification

Put the chip in the ZIF socket of the programmer and click **[Auto Find]** , the logic matching chip model will be listed automatically, as below:



Logic Test

Search:

4000
4001
4002
4007
4009
4010
40106
4011
4012
4013
4015
4016
40161
40162
4017
40174
40175
4018
4019
40193
4022

Vector Symbol definition

0: Input Low
1: Input High
L: Out Low
H: Out High
C: Pulse Input
Z: High Impedance
OC High or 3S
X: Ignore
G: GND
V: VCC

VCC VOLTAGE

● 5.0V
○ 3.3V
○ 2.5V
○ 1.8V

IC Position

Reslut	Item(IC Model)
logic matched	4053
logic matched	744053
Found 2 logic chips matched	

Vector Table and Test Result

10. Add customized chips

- For chips that are not in the software support list, there are many chips that may have programming algorithms fully compatible with one of the chips in the list. In this case, you can add them to the custom list after the test is normal, so that you can easily use them later or export the list to share.
- For chips that the software support parameter setting, such as NAND, 27C and other series, various parameters can be set by user. The user can change the parameter settings, add it to user list after getting the programming test to pass.
- New customized chips can also be used as favorites of commonly used chips.

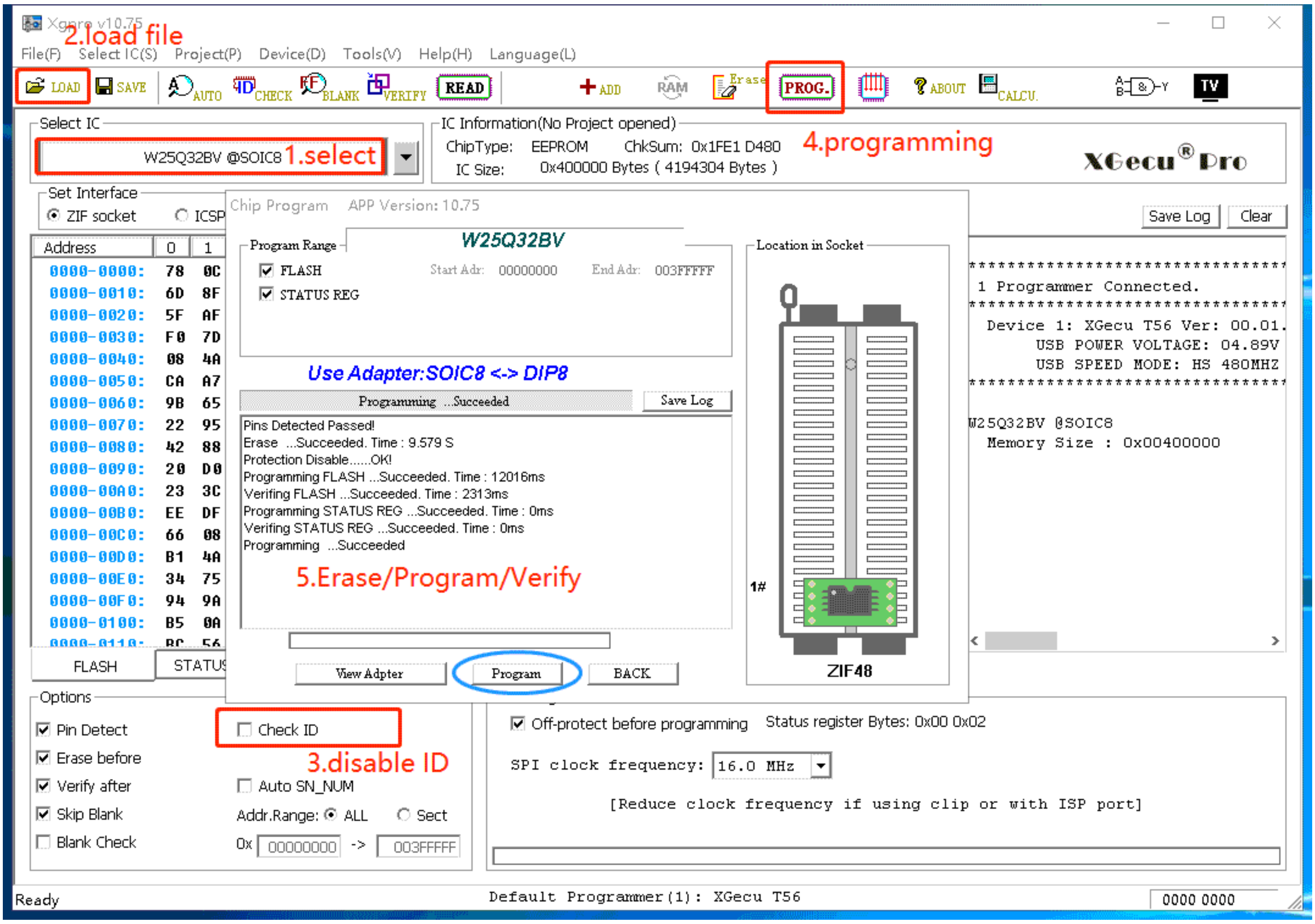
10.1The common method of adding chips

The same type of chips may be the same programming algorithms exactly . But in general due to different manufactories, the identification ID of the chip is different, at this time, just select the compatible chip model, uncheck the option **[check ID]** to read and write for test.

10.1.1 Test algorithm

For example: XM25QH32B (assume, this chip is not in the support list)

We can select the same type of chip W25Q32BV for testing, as below:



Select IC: W25Q32BV @SOIC8

IC Information(No Project opened)
ChipType: EEPROM ChkSum: 0x1FE1 D480
IC Size: 0x400000 Bytes (4194304 Bytes)

Set Interface: ● ZIF socket ○ ICSP

Chip Program APP Version: 10.75

W25Q32BV

Program Range
☑ FLASH Start Adr: 00000000 End Adr: 003FFFFF
☑ STATUS REG

Use Adapter:SOIC8 <-> DIP8

Programming ...Succeeded

```
Pins Detected Passed!
Erase ...Succeeded. Time : 9.579 S
Protection Disable......OK!
Programming FLASH ...Succeeded. Time : 12016ms
Verifing FLASH ...Succeeded. Time : 2313ms
Programming STATUS REG ...Succeeded. Time : 0ms
Verifing STATUS REG ...Succeeded. Time : 0ms
Programming ...Succeeded
```

```
Address         0  1
0000-0000:     78 0C
0000-0010:     6D 8F
0000-0020:     5F AF
0000-0030:     F0 7D
0000-0040:     08 4A
0000-0050:     CA A7
0000-0060:     9B 65
0000-0070:     22 95
0000-0080:     42 88
0000-0090:     20 D0
0000-00A0:     23 3C
0000-00B0:     EE DF
0000-00C0:     66 08
0000-00D0:     B1 4A
0000-00E0:     34 75
0000-00F0:     94 9A
0000-0100:     B5 0A
```

```
***********************************
 1 Programmer Connected.
***********************************
  Device 1: XGecu T56 Ver: 00.01.
       USB POWER VOLTAGE: 04.89V
       USB SPEED MODE: HS 480MHZ
***********************************

W25Q32BV @SOIC8
  Memory Size : 0x00400000
```

Options
☑ Pin Detect ☐ Check ID
☑ Erase before
☑ Verify after ☐ Auto SN_NUM
☑ Skip Blank Addr.Range: ● ALL ○ Sect
☐ Blank Check 0x 00000000 -> 003FFFFF

☑ Off-protect before programming Status register Bytes: 0x00 0x02

SPI clock frequency: 16.0 MHz

[Reduce clock frequency if using clip or with ISP port]

Ready Default Programmer(1): XGecu T56 0000 0000

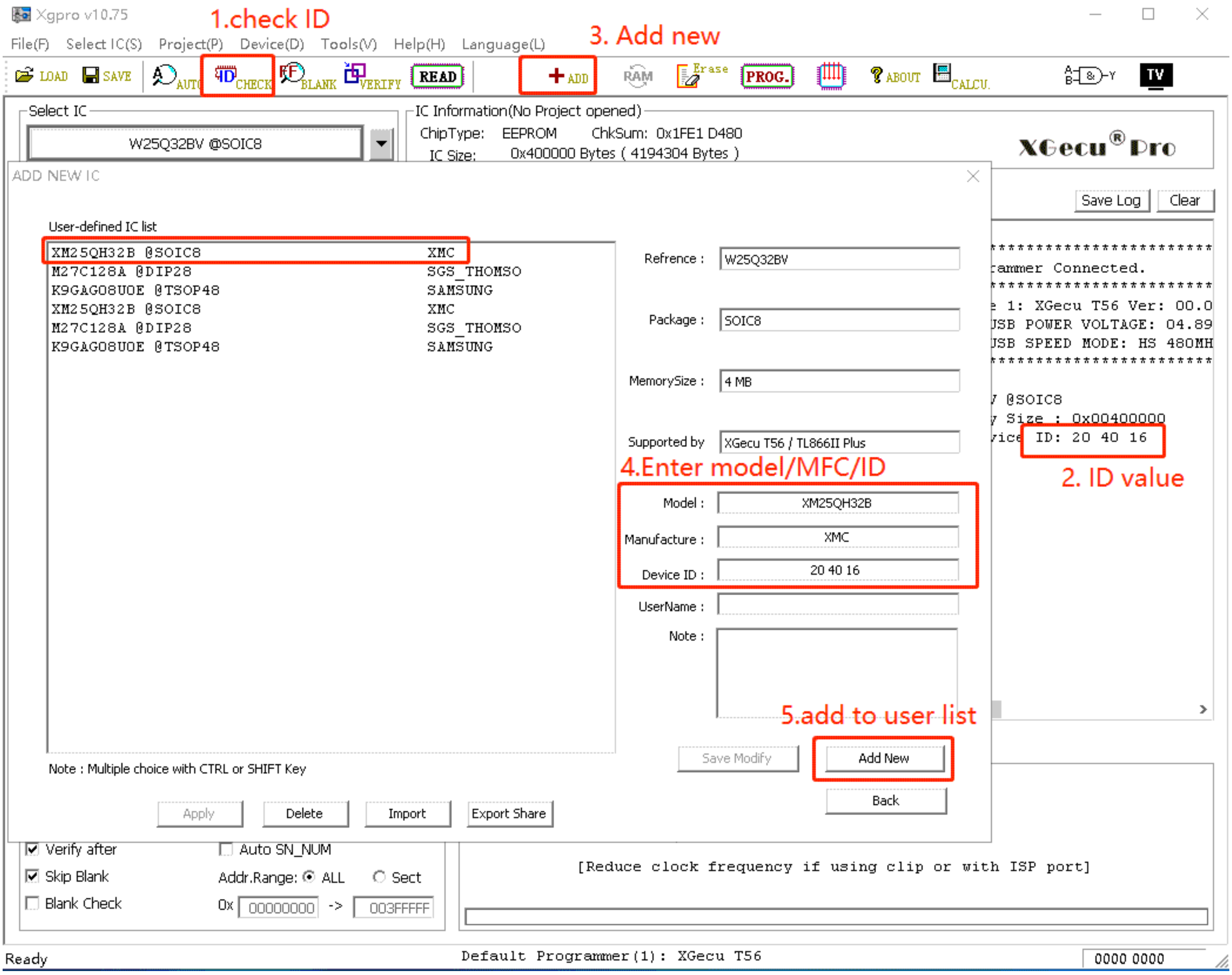
As the above, the test is completed, that means the chip is programmed correctly by selecting W25Q32B.

STEP:

1) Select IC W25Q32BV
2) Load a data file
3) Uncheck the option **[check ID]**
4) program for testing
5) Erase, write and verify completed, if testing is correct, you can go to the next step. If it failed, you can select another similar chips for testing.

10.1.2 **Read ID and add the chip model**

As shown below:


Xgpro v10.75
1.check ID
3. Add new
File(F) Select IC(S) Project(P) Device(D) Tools(V) Help(H) Language(L)
LOAD SAVE AUTO ID CHECK BLANK VERIFY READ ADD RAM Erase PROG. ABOUT CALCU. TV
Select IC
IC Information(No Project opened)
XGecu® Pro
ADD NEW IC
User-defined IC list
Save Log Clear
4.Enter model/MFC/ID
2. ID value
5.add to user list
Note : Multiple choice with CTRL or SHIFT Key
Save Modify
Add New
Back
Apply Delete Import Export Share
Ready

W25Q32BV @SOIC8

ChipType: EEPROM ChkSum: 0x1FE1 D480
IC Size: 0x400000 Bytes (4194304 Bytes)

```
XM25QH32B @SOIC8          XMC
M27C128A @DIP28           SGS_THOMSO
K9GAG08U0E @TSOP48        SAMSUNG
XM25QH32B @SOIC8          XMC
M27C128A @DIP28           SGS_THOMSO
K9GAG08U0E @TSOP48        SAMSUNG
```

Refrence : W25Q32BV

Package : SOIC8

MemorySize : 4 MB

Supported by XGecu T56 / TL866II Plus

Model : XM25QH32B

Manufacture : XMC

Device ID : 20 40 16

UserName :

Note :

```
************************
rammer Connected.
************************
e 1: XGecu T56 Ver: 00.0
USB POWER VOLTAGE: 04.89
USB SPEED MODE: HS 480MH
************************
V @SOIC8
y Size : 0x00400000
vice ID: 20 40 16
```

☑ Verify after ☐ Auto SN_NUM
☑ Skip Blank Addr.Range: ● ALL ○ Sect
☐ Blank Check 0x 00000000 -> 003FFFFF

[Reduce clock frequency if using clip or with ISP port]

Default Programmer(1): XGecu T56 0000 0000

STEP:

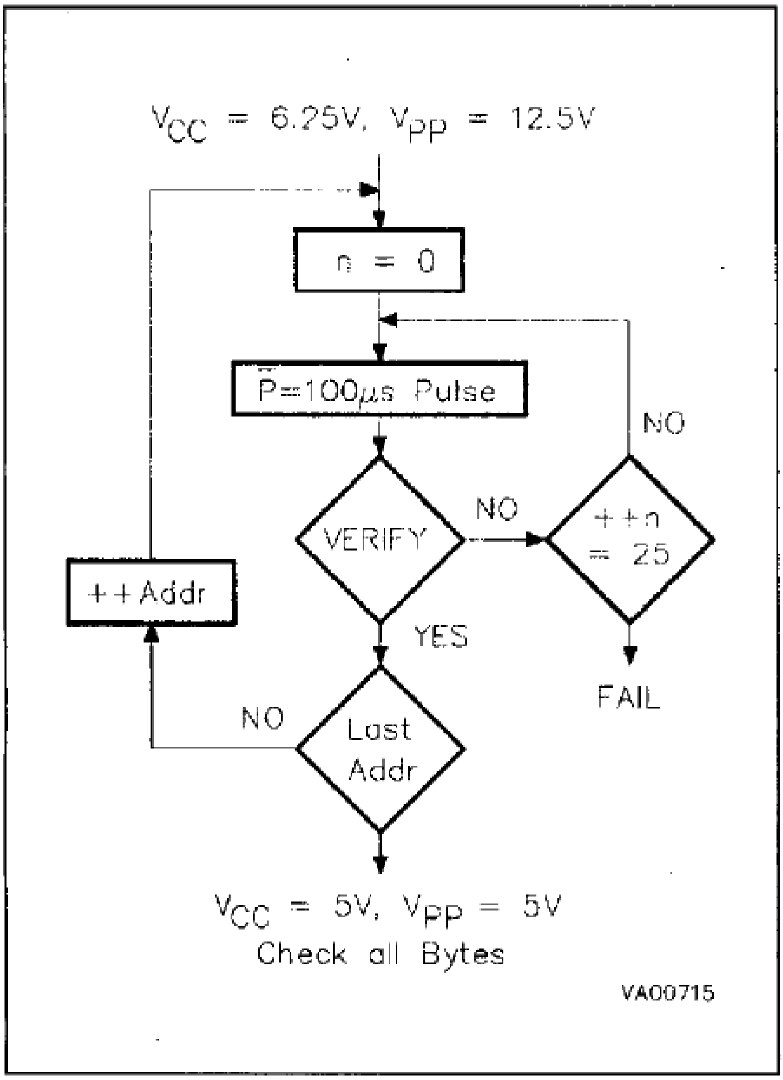
1) Click **[Check ID]** to read the current chip ID
2) Display the current ID value
3) Click **[ADD]** in the toolbar
4) Type in the chip model, manufactory, the chip ID read, and other information
5) Click **[Add New]**, a new model will be added in the user list
6) Just double click the chip model in the custom list to use it.

10.2 Add 27 series device

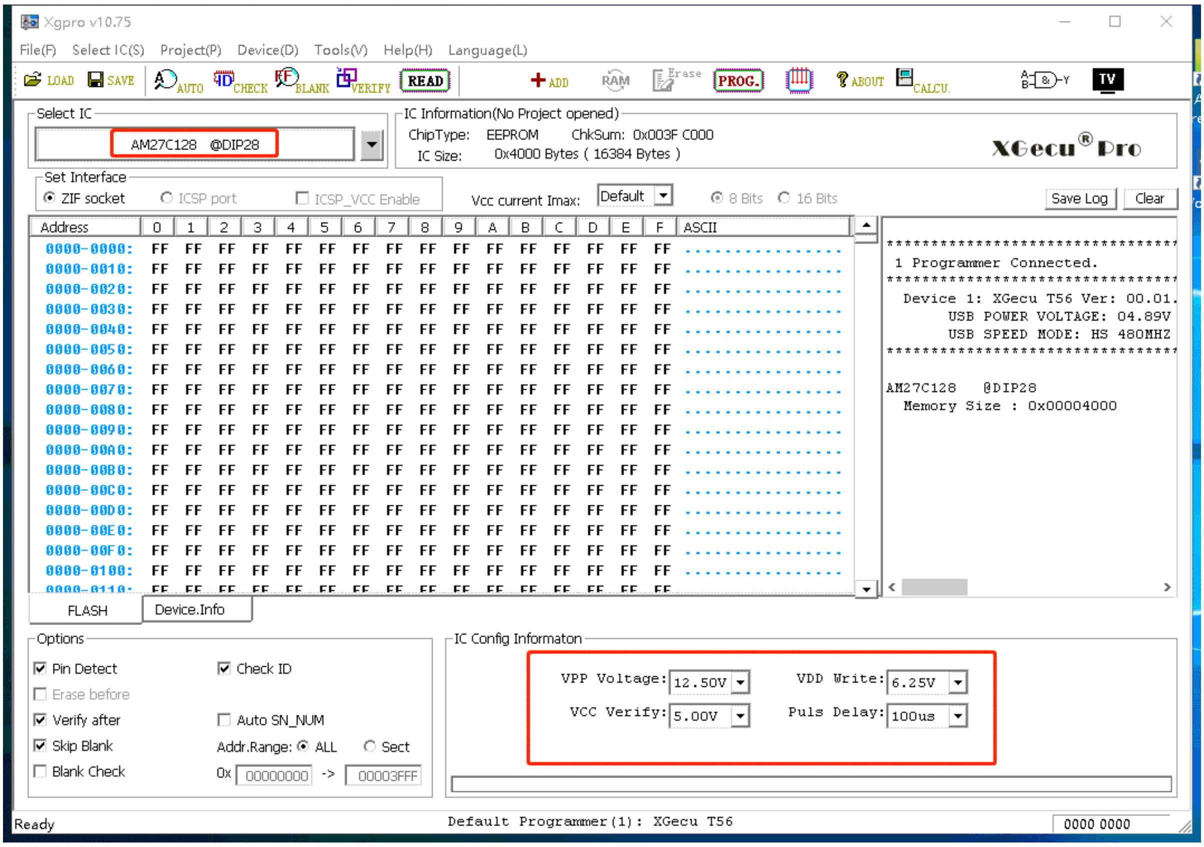
Example: M27C128A is not in the supported list, the basic method of adding chips is the same as before, more reliable method of adding is as below:

1. Select IC: Type in “27C128” to find, select AM27C128 @DIP28 Note: the chip package and chip capacity must be the same.
2. Download the M27C128A IC data set from the Internet
   From the data sheet, you can find:
   Chip ID: 20 0A, VCC=6.25V when programming, VCC=5V when verifying and reading, VPP=12.50V for programming, pulse time is 100us

Figure 7. Programming Flowchart



3. According to the parameters in data sheet, modify the chip parameters as shown below:


Xgpro v10.75
File(F) Select IC(S) Project(P) Device(D) Tools(V) Help(H) Language(L)
LOAD SAVE AUTO CHECK BLANK VERIFY READ ADD RAM Erase PROG. ABOUT CALCU. TV
XGecu® Pro
FLASH
Device.Info
Save Log
Clear

Select IC: AM27C128 @DIP28

IC Information(No Project opened)
ChipType: EEPROM ChkSum: 0x003F C000
IC Size: 0x4000 Bytes (16384 Bytes)

Set Interface: ● ZIF socket ○ ICSP port □ ICSP_VCC Enable

Vcc current Imax: Default ● 8 Bits ○ 16 Bits

```
Address    0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  ASCII
0000-0000: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0010: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0020: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0030: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0040: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0050: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0060: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0070: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0080: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0090: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-00A0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-00B0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-00C0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-00D0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-00E0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-00F0: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0100: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ................
0000-0110: FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
```

```
*********************************
 1 Programmer Connected.
*********************************
  Device 1: XGecu T56 Ver: 00.01.
        USB POWER VOLTAGE: 04.89V
        USB SPEED MODE: HS 480MHZ
*********************************

AM27C128    @DIP28
  Memory Size : 0x00004000
```

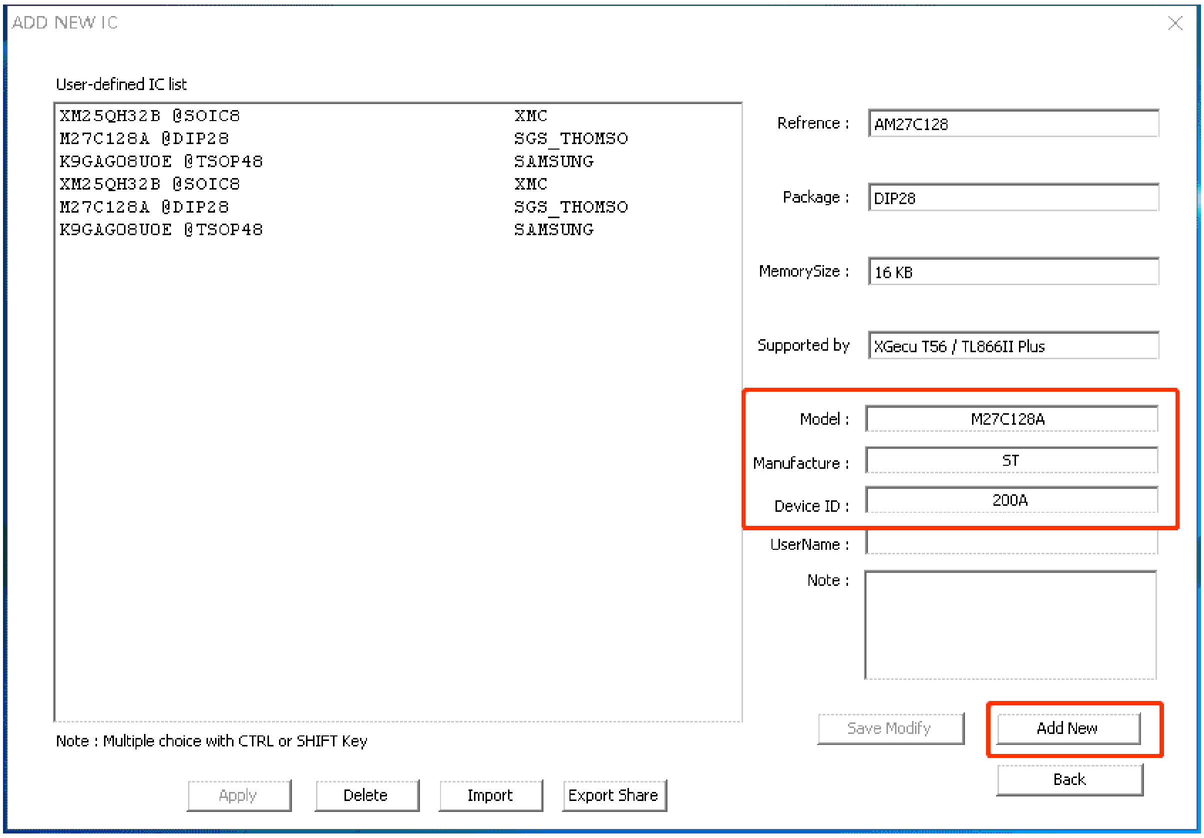
Options:
☑ Pin Detect ☑ Check ID
□ Erase before
☑ Verify after □ Auto SN_NUM
☑ Skip Blank Addr.Range: ● ALL ○ Sect
□ Blank Check 0x 00000000 -> 00003FFF

IC Config Informaton:
VPP Voltage: 12.50V VDD Write: 6.25V
VCC Verify: 5.00V Puls Delay: 100us

Ready Default Programmer(1): XGecu T56 0000 0000

4. Click **[ADD]** button in the toolbar

In the following position, type in the chip model: M27C128A Manufactory: SGS-THOMSON Chip ID: 20 0A, as shown below: click **[Add new]** button



User-defined IC list

```
XM25QH32B @SOIC8        XMC
M27C128A @DIP28         SGS_THOMSO
K9GAG08U0E @TSOP48      SAMSUNG
XM25QH32B @SOIC8        XMC
M27C128A @DIP28         SGS_THOMSO
K9GAG08U0E @TSOP48      SAMSUNG
```

Note : Multiple choice with CTRL or SHIFT Key

Refrence : AM27C128
Package : DIP28
MemorySize : 16 KB
Supported by XGecu T56 / TL866II Plus
Model : M27C128A
Manufacture : ST
Device ID : 200A
UserName :
Note :

5. Double click the new chip to apply using it.

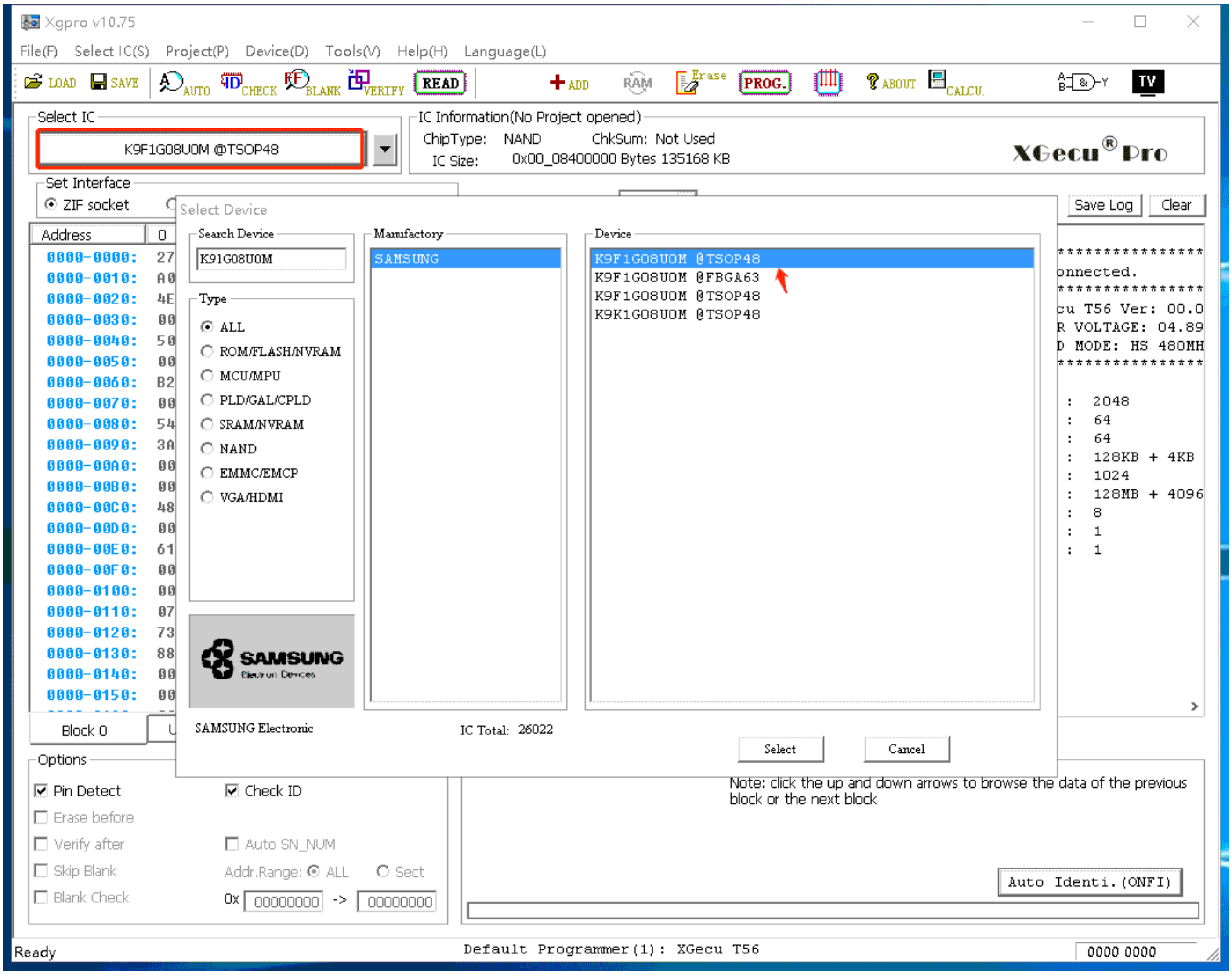
10.3 Add NAND chips

For NAND chips, theoretically you can customize all NAND chips. For ONFI standard chips, you can also automatically identify and add to the custom list. For non-ONFI standard chips, you can add it according to the IC data sheet.

10.3.1 **Select a NAND chip of any type with the same package**

K9GAG08U0E TSOP48 package chip as an example to add custom.

Arbitrarily select a K91G08U0M @TSOP48 (you can select any other TSOP48 NAND chip)


Xgpro v10.75
File(F) Select IC(S) Project(P) Device(D) Tools(V) Help(H) Language(L)
LOAD SAVE AUTO CHECK BLANK VERIFY READ ADD RAM Erase PROG. ABOUT CALCU. TV
XGecu® Pro
Save Log Clear
Select Device
Select Cancel
Auto Identi.(ONFI)

Select IC: K9F1G08U0M @TSOP48

IC Information(No Project opened)
ChipType: NAND ChkSum: Not Used
IC Size: 0x00_08400000 Bytes 135168 KB

Set Interface: ● ZIF socket

Search Device: K91G08U0M

Type:
● ALL
○ ROM/FLASH/NVRAM
○ MCU/MPU
○ PLD/GAL/CPLD
○ SRAM/NVRAM
○ NAND
○ EMMC/EMCP
○ VGA/HDMI

Manufactory: SAMSUNG

Device:
K9F1G08U0M @TSOP48
K9F1G08U0M @FBGA63
K9F1G08U0M @TSOP48
K9K1G08U0M @TSOP48

SAMSUNG Electronic   IC Total: 26022

```
Address     0
0000-0000:  27
0000-0010:  A0
0000-0020:  4E
0000-0030:  00
0000-0040:  50
0000-0050:  00
0000-0060:  B2
0000-0070:  00
0000-0080:  54
0000-0090:  3A
0000-00A0:  00
0000-00B0:  00
0000-00C0:  48
0000-00D0:  00
0000-00E0:  61
0000-00F0:  00
0000-0100:  00
0000-0110:  07
0000-0120:  73
0000-0130:  88
0000-0140:  00
0000-0150:  00
```

Block 0

```
*****************
onnected.
*****************
cu T56 Ver: 00.0
R VOLTAGE: 04.89
D MODE: HS 480MH
*****************

 :  2048
 :  64
 :  64
 :  128KB + 4KB
 :  1024
 :  128MB + 4096
 :  8
 :  1
 :  1
```

Options
☑ Pin Detect ☑ Check ID
☐ Erase before
☐ Verify after ☐ Auto SN_NUM
☐ Skip Blank Addr.Range: ● ALL ○ Sect
☐ Blank Check 0x 00000000 -> 00000000

Note: click the up and down arrows to browse the data of the previous block or the next block

Ready   Default Programmer(1): XGecu T56   0000 0000

10.3.2 **Check IC data sheet to find the corresponding parameters**

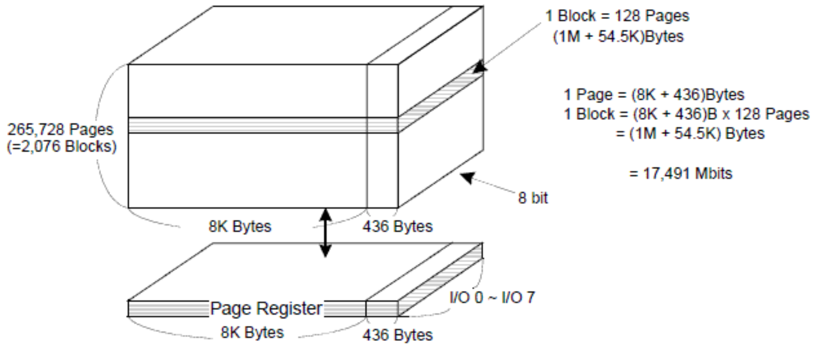


Figure 2. K9GAG08U0E Array Organization

From the above figure you can see,

K9GAG08U0E Parameter:

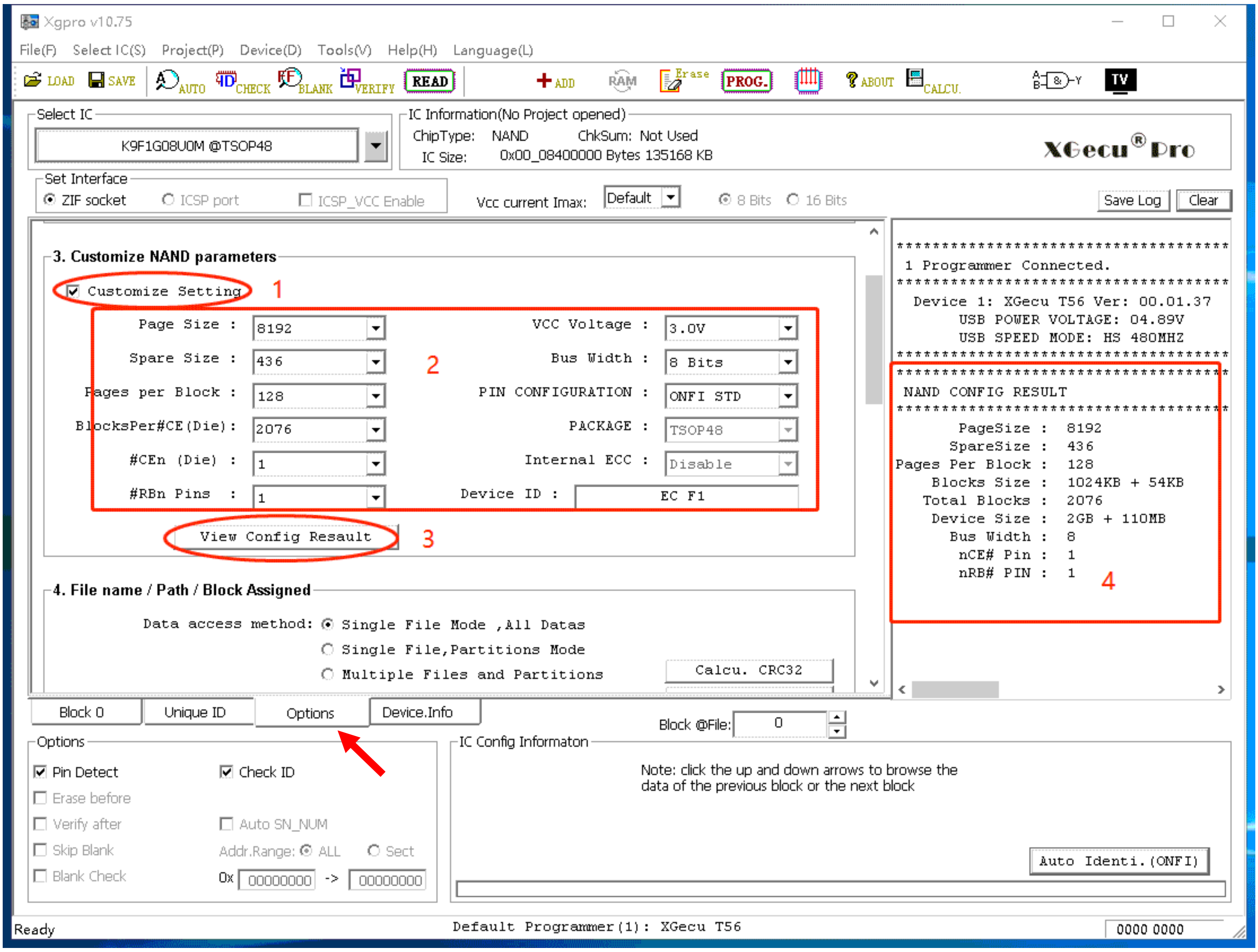
Page Size 8K (8192) Spare Size=436 Number of pages per block: 128 pages, total number of blocks 2076 blocks

In addition, from data sheet you can find, the operating voltage is 3.3V 1CE pin, 1 RB pin, 8-bit data width

ID: EC D5 84 72 (only read the first 4 bytes)

10.3.3 Set NAND parameters

Click [Options] to set as below **(3. Customize NAND parameters)**:


Xgpro v10.75
File(F) Select IC(S) Project(P) Device(D) Tools(V) Help(H) Language(L)
LOAD SAVE AUTO CHECK BLANK VERIFY READ ADD RAM Erase PROG. ABOUT CALCU. TV
XGecu® Pro
Save Log Clear
Block 0 Unique ID Options Device.Info
Auto Identi.(ONFI)
Calcu. CRC32
1
2
3
4

Select IC: K9F1G08U0M @TSOP48

IC Information(No Project opened)
ChipType: NAND ChkSum: Not Used
IC Size: 0x00_08400000 Bytes 135168 KB

Set Interface: ● ZIF socket ○ ICSP port □ ICSP_VCC Enable

Vcc current Imax: Default ● 8 Bits ○ 16 Bits

3. Customize NAND parameters

☑ Customize Setting

Page Size : 8192
Spare Size : 436
Pages per Block : 128
BlocksPer#CE(Die): 2076
#CEn (Die) : 1
#RBn Pins : 1
VCC Voltage : 3.0V
Bus Width : 8 Bits
PIN CONFIGURATION : ONFI STD
PACKAGE : TSOP48
Internal ECC : Disable
Device ID : EC F1

View Config Resault

4. File name / Path / Block Assigned

Data access method: ● Single File Mode ,All Datas
○ Single File,Partitions Mode
○ Multiple Files and Partitions

```
**************************************
 1 Programmer Connected.
**************************************
  Device 1: XGecu T56 Ver: 00.01.37
       USB POWER VOLTAGE: 04.89V
       USB SPEED MODE: HS 480MHZ
**************************************
**************************************
 NAND CONFIG RESULT
**************************************
        PageSize :  8192
       SpareSize :  436
Pages Per Block :  128
     Blocks Size :  1024KB + 54KB
    Total Blocks :  2076
     Device Size :  2GB + 110MB
       Bus Width :  8
        nCE# Pin :  1
        nRB# PIN :  1
```

Options
☑ Pin Detect ☑ Check ID
□ Erase before
□ Verify after □ Auto SN_NUM
□ Skip Blank Addr.Range: ● ALL ○ Sect
□ Blank Check 0x 00000000 -> 00000000

Block @File: 0

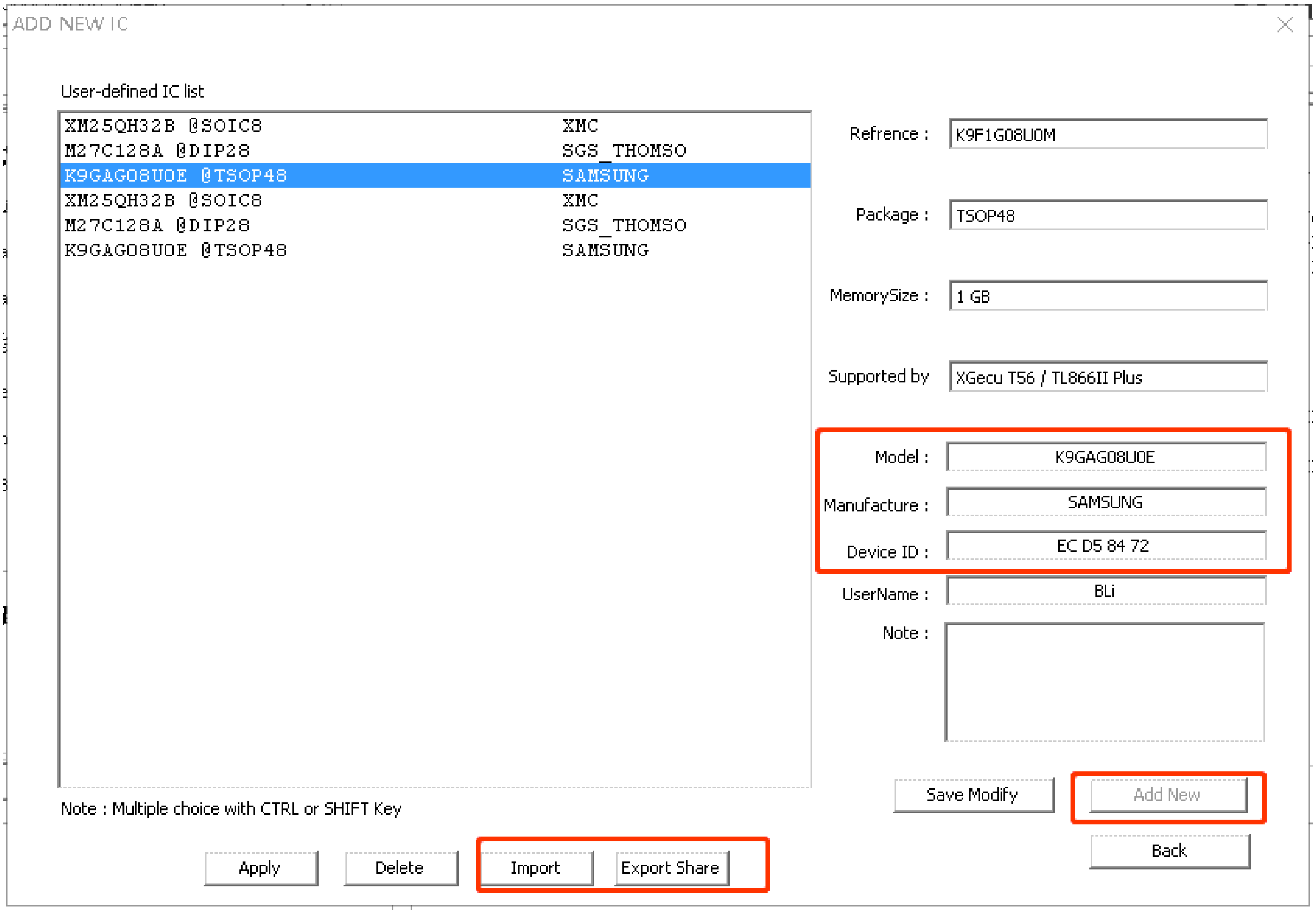
IC Config Informaton
Note: click the up and down arrows to browse the data of the previous block or the next block

Ready Default Programmer(1): XGecu T56 0000 0000

1. Check option **[customize setting]**
2. Set value parameter according to IC data sheet
3. Click **[View Config Result]** and you can view the final result in the right information column **[4]**.

10.3.3 **Add to User List**

After the parameters are set, add new list in the same way as before, click the **[ADD]** button + ADD in the toolbar to enter the ADD NEW IC dialog box, as below:


ADD NEW IC

User-defined IC list

XM25QH32B @SOIC8	XMC
M27C128A @DIP28	SGS_THOMSO
K9GAG08U0E @TSOP48	SAMSUNG
XM25QH32B @SOIC8	XMC
M27C128A @DIP28	SGS_THOMSO
K9GAG08U0E @TSOP48	SAMSUNG

Refrence : K9F1G08U0M

Package : TSOP48

MemorySize : 1 GB

Supported by XGecu T56 / TL866II Plus

Model : K9GAG08U0E

Manufacture : SAMSUNG

Device ID : EC D5 84 72

UserName : BLi

Note :

Note : Multiple choice with CTRL or SHIFT Key

Respectively type in the IC name, manufacturer and device ID. And click **[Add New]** to add to the list.

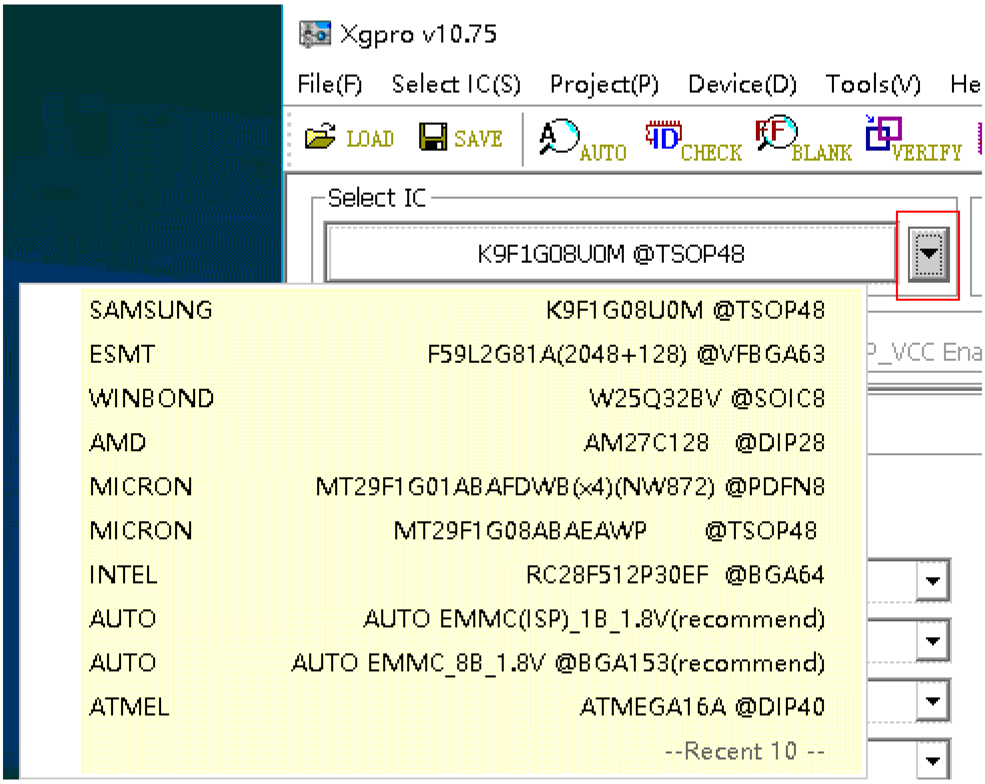
Double click the added model, you can program the IC. If the read/write is normal, it means the addition is correct.

10.4 Import and Export Share

- For customized chips, user can export the database for sharing, just send the exported data file to the third party, who uses the import function to import the list.
- Export data, use CTRL or SHIFT key + left mouse button to multi-select and export multiple chips at one time. The extension of exported file is xxx.ULI.

10.5 Commonly-used chip favorites

For commonly-used chips, there is a drop-down list of the last 10 chips to the right of the **[Select IC button]**:


Xgpro v10.75
File(F) Select IC(S) Project(P) Device(D) Tools(V) He
LOAD SAVE AUTO CHECK BLANK VERIFY
Select IC
K9F1G08U0M @TSOP48
P_VCC Ena

SAMSUNG	K9F1G08U0M @TSOP48
ESMT	F59L2G81A(2048+128) @VFBGA63
WINBOND	W25Q32BV @SOIC8
AMD	AM27C128 @DIP28
MICRON	MT29F1G01ABAFDWB(x4)(NW872) @PDFN8
MICRON	MT29F1G08ABAEAWP @TSOP48
INTEL	RC28F512P30EF @BGA64
AUTO	AUTO EMMC(ISP)_1B_1.8V(recommend)
AUTO	AUTO EMMC_8B_1.8V @BGA153(recommend)
ATMEL	ATMEGA16A @DIP40
	--Recent 10 --

If it is not enough to use, you can add to this custom list.

Selection method:

After selecting IC, directly click **[+ADD]** button in the toolbar, without modifying the content, directly add to the custom list. Next time, user can select the IC in the custom list. User can select the IC model more quickly.

11. Appendix

11.1Disclaimer

(1) Products that have been repaired, modified by any organization or individual not officially authorized by us are not covered by the warranty.

(2) The ZIF socket of the product and other man-made damages are not covered by the warranty.

(3) Joint and several extended liability caused by hardware failure or software defects.

Special Declare

Due to the variety of chips supported by the programmer and the complexity of the software system, errors or defects are inevitable. In mass production, especially for OTP one-time programming chips, you must make sure that the programming is correct before mass production. We are not responsible for collateral expansion losses due to hardware failure or software defects. If you find any problems or have valuable opinions about our products, please contact us.

11.2 Warranty

Within one year from the date of purchase, if there is a failure during normal operations, can repair for free.

11.3 Contact Us

As the software is constantly updated, the manual content is for reference only and is subject to the product application.
The latest version software can be downloaded from http://www.xgecu.com/en.

Name: Haikou Xingong Electronic Co.,Ltd.
Address: 4th Floor, Jingrun Hotel, No.80, Qiuhai Avenue West,
Haikou, Hainan Province, China, Postal Code: 570311
EMIAL: 532007590@QQ.com
TEL: 0898-68681816

11.4 FAQ (Frequently Asked Questions)

1) Device unavailable or error

It is usually because the USB driver is not installed, or the driver is prohibited by antivirus software, at this time the RUN light on the USB programmer flashes rapidly.
Solution: Refer to section 2.1 and re-install the application software.

2) Error when updating firmware, unable to refresh

It is usually caused by using a USB hub. Please do not use an external HUB when refreshing. Please directly connect the programmer to the USB port of your computer to refresh. Restart the computer to test. If it still cannot solve the problem, please try to do the refresh operation on another computer.

3) Poor pin contact

Situation 1: Old chips, often there are some tips of poor pin contact, need to handle with pins carefully.
Situation 2: If using an adapter, please use a multimeter to measure whether the adapter pins are normal.
Situation 3: Individual pins of individual chips may not support pin contact check. In this case, you can temporarily uncheck the option **<Pin Detect>** in the lower left corner of the main interface.
Note: Pin detect can only check if the pin is on, cannot check the contact resistance.
If the contact resistance is too large, especially for NAND high-speed chips, it will also cause working errors. So be sure to clean the adapter, please do not easily trust any clip-type connection.
Note: Pin detect can only check if the pin is on, but cannot detect the contact resistance. If the contact resistance is too large, especially for NAND high-speed chips, it will also cause working errors. So make sure to clean the adapter and please do not easily trust any clip-type connection. After clamping, hard twist the adapter so that there is a small amount of frictional sliding between cooper sheets of the ZIF socket and pin headers or between the pins of integrated circuit, to make a good contact.

4) Check ID error

Many chips have an internal chip identification mark (Identification). The content generally has two or more bytes, generally the first byte is the Manufacturer ID, followed by the chip type or chip capacity. Different chip has the different ID and the function of this option is to prevent incorrect chips put in. This option is to check the chip ID before reading or writing. If it is correct, continue. If ID incorrect, abort. This function is optional. The option is checked by default.

> For the same type of chip, different manufactory may program in the same way. For chips not supported in the list, you can select same capacity same type ICs of other manufactories to program. Because ID is different, before programming, uncheck the option <Check ID>.

For few parts of microcontrollers, the chip ID cannot be read after chip is encrypted. At this time, the option <Check ID> should be unchecked in the main interface, otherwise check ID error will appear.

5) Programming for chips 27C series VPP voltage 21V or 25V

T56 directly supports up to 25V.
TL866II only supports programming voltage VPP up to 18V. For earlier 27C series ROMs, chips with VPP voltage of 21V and 25V are no longer supported for writing, but can be read. If used, you can replace with the same type of chips that their programming voltage are 13V or 12V. For example, 27C32 chip, its programming voltage is 21V or 13V. The chip can be directly replaced.
Please do not use an external power supply to raise the voltage for forced programming. When the chip is damaged, the 21V/25V high voltage may run back into the programmer and there will be the hardware damage.