Programmer DAS-Memtool: The Ultimate Guide to Reading and Writing Infineon Chips

(Take SAK-TC1796 as an example; it is difficult, and you may not succeed easily.)

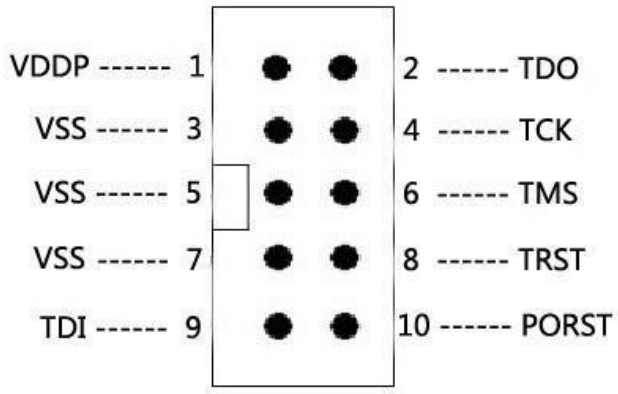
Preparation:

A. Run Infineon-Memtool-DevelopmentTools-x.exe to install the programmer driver and programming software. During the process, prompts for installing the DAS driver and other installations will appear. Continue with the default options until completion (Finish). For any security issues, select “Agree,” “Allow,” or “Always Trust.” The Infineon Memtool read/write software icon will be automatically generated on the desktop.

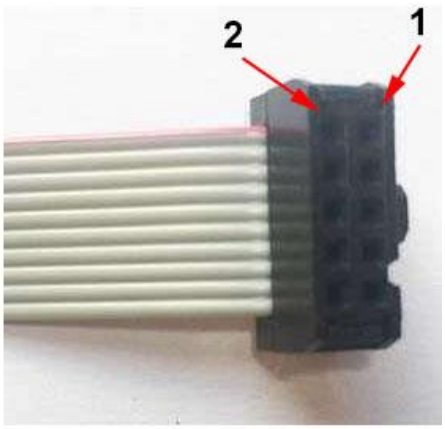
B. Insert the DAS programmer into the USB port. Follow the prompts to automatically complete device recognition and driver installation. During this process, you can “skip obtaining driver software from Windows Update.” After successful installation, the DAS device will be visible in Device Manager.

C. Connect the programmer's 10-pin plug to the target board's programming interface (using header pins, jumper wires, clamps, or probes). Pin definitions for various chip models can be found online, or you can ask the retailer for assistance.

The default JTAG interface mode is defined as follows: (The left figure shows the onboard header pins; the right figure shows the ribbon cable connector, with the small triangle marked on the red side indicating pin 1.) Among these, the signal lines connected to the chip require 8 connections: 1, 3, 9, 2, 4, 6, 8, and 10.



Infineon JTAG Interface Definition



VDDP: Connect to the target chip's operating power supply 3.3V/5V. The programmer board J2 does not output

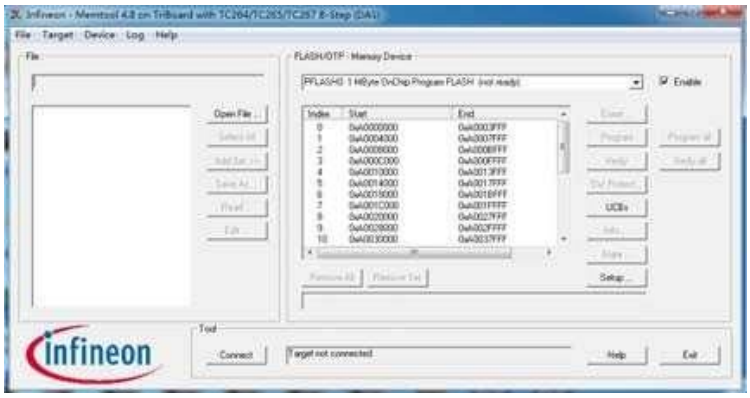
power by default unless it is short-circuited.

Additionally, if using DAP mode, short-circuit the programmer board J1; connect 5 wires: DAP1=TDO, DAP0=TCK, PORST, VDDP, VSS.

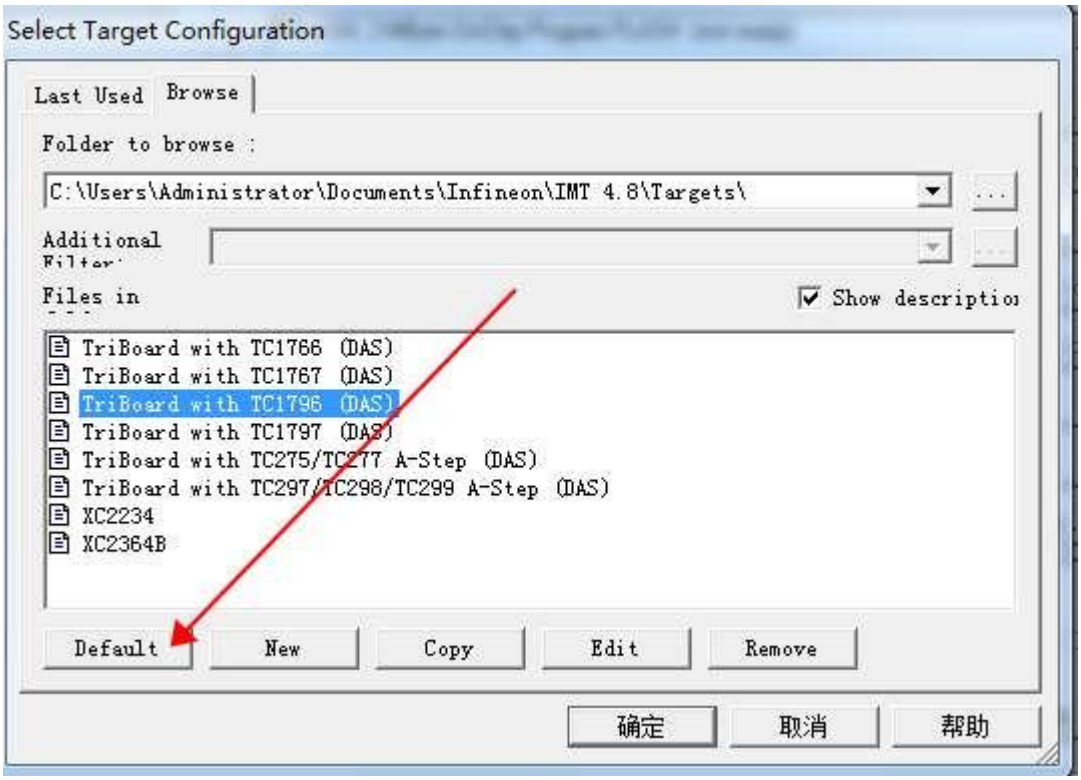
D. Typically, the target board must be powered up and activated using the original power supply method, such as 12V/24V. Then, you should see and ensure that the red power indicator light on the programmer is on.

Getting started:

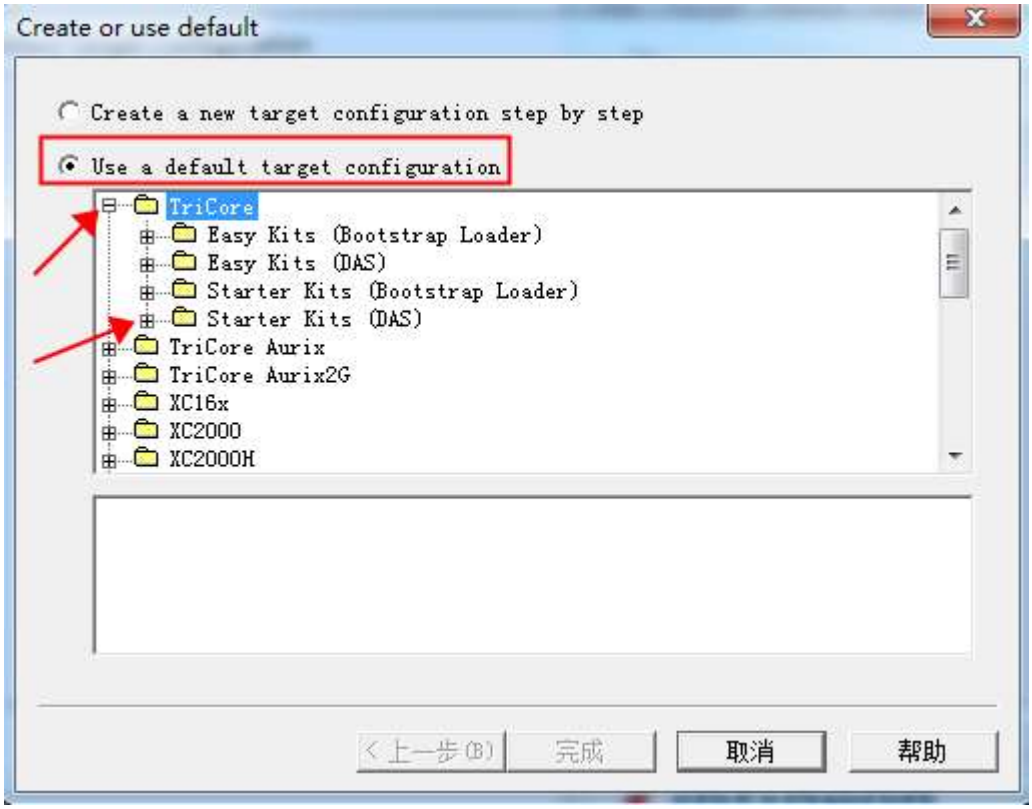
1. Run the software: Find and run the Infineon Memtool read/write software on your desktop to open the following interface:



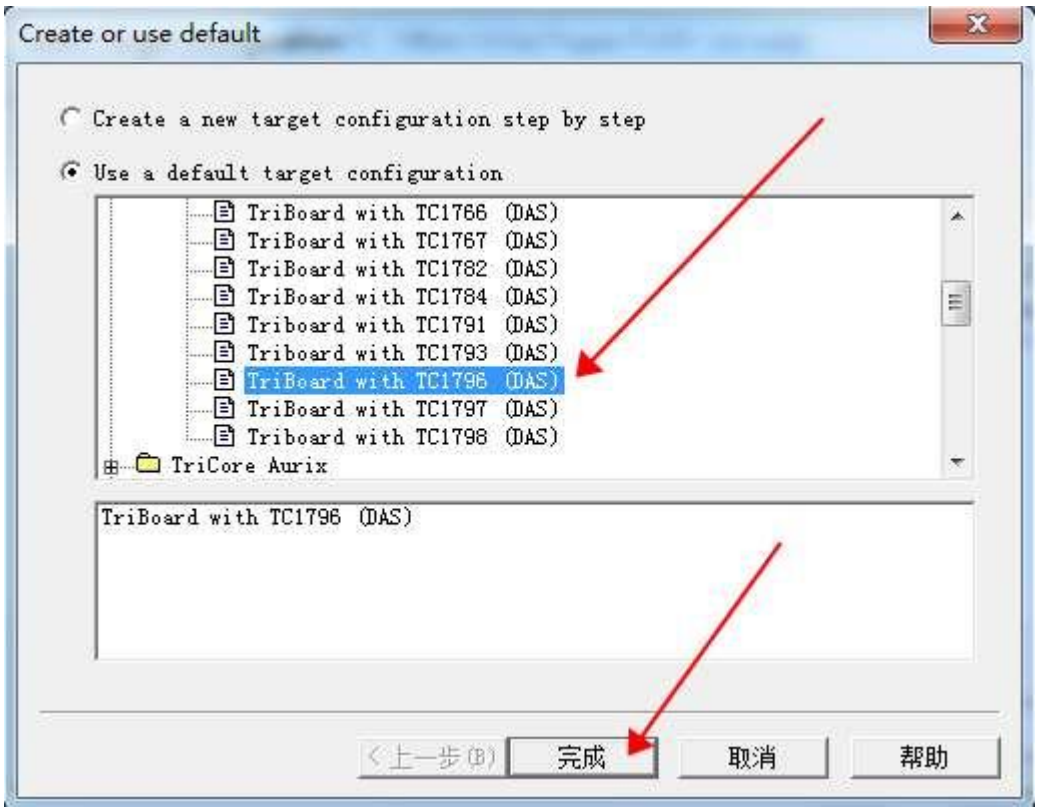
2. Configure target: Select the Target → Change ··· menu, and the following window will appear:



After selecting Default, select the existing configuration by default in the new interface that appears. The TC17xx model is in TriCore, the TC2xx model is in TriCore Aurix, and the same applies to others, all of which must have the DAS identifier.

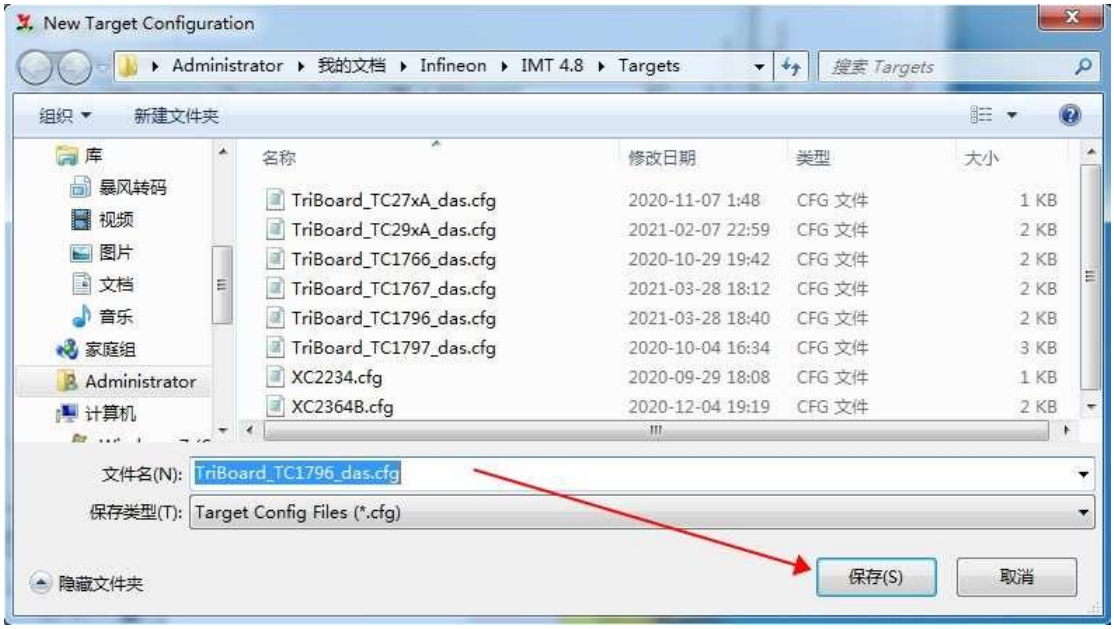


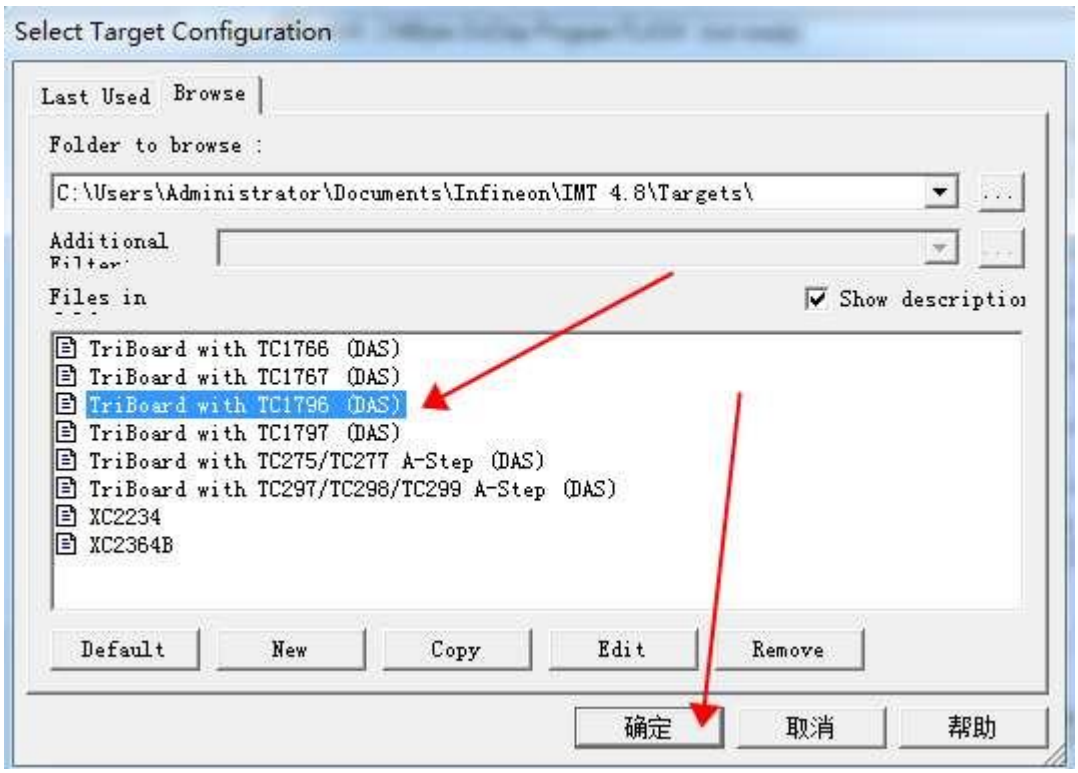
For example, select and click the + sign for the TriCore item, click the + sign for the Starter Kits (DAS) item, find the configuration item for the target model TC1796, and click “Finish,”as shown in the figure below:



A dialog box for saving the configuration file will then appear. Click “Save” at the default

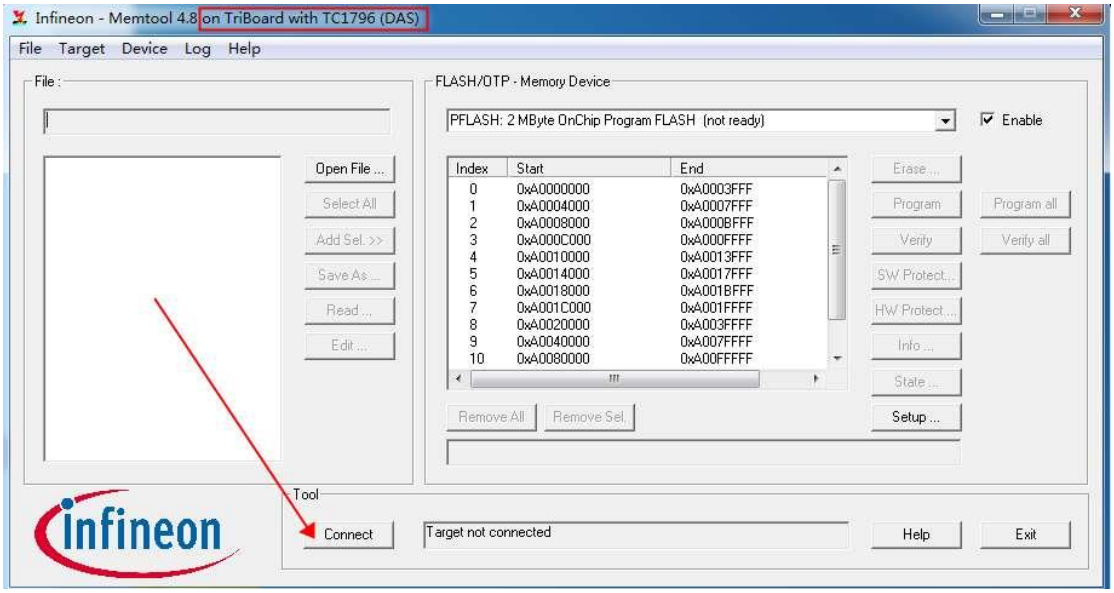
location, and the configuration file will be automatically loaded the next time you enter the software. As shown in the figure below:



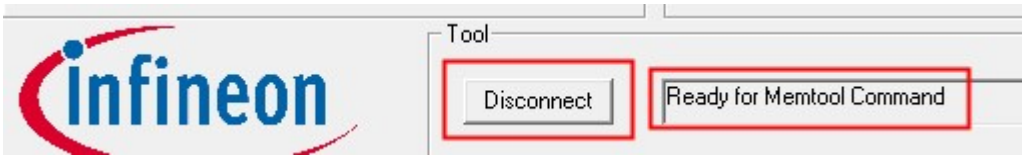


At this point, the configuration of the target is complete! The above configuration is a one-time process, so you don't need to configure it again the next time you enter the software. Simply select the configuration file name and click “OK.” Unless you need to change the target model.

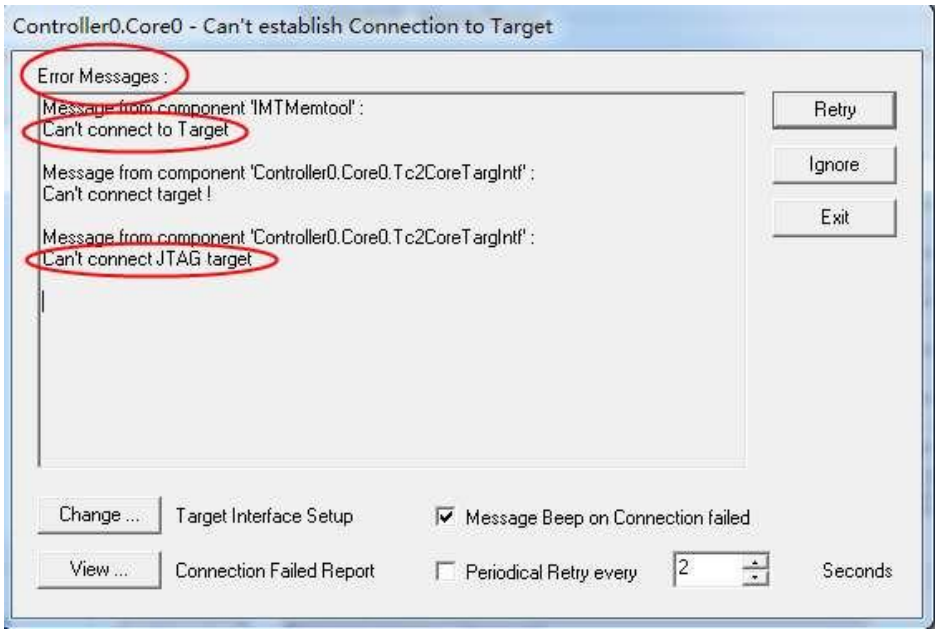
3. Connect to the target: After completing the target configuration in the previous step, you will directly enter the interface shown in the figure below: (The next time you enter the software, you can still select the configured file through Targer→Change ...)



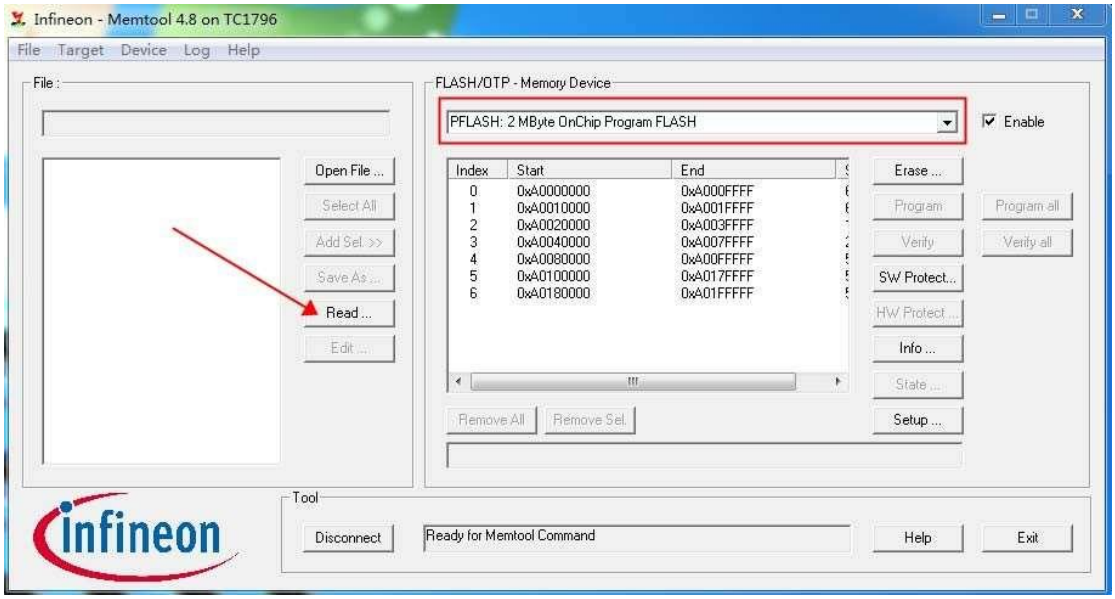
Before starting chip read/write operations, you must first execute Connect to ensure that the device is successfully connected to the target chip: The Connect button on the software interface will change as shown in the figure below, indicating that the connection to the target is successful and you can perform read, erase, write, and other operations.



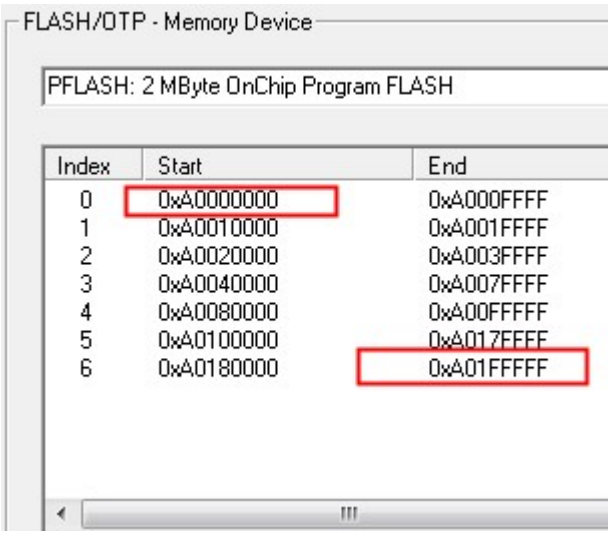
If the connection is unsuccessful, an error message will appear as shown in the figure below. Any incorrect factors will result in a failed connection. In this case, you need to check whether the device driver is working properly, whether the target chip is working, whether the model is selected correctly, whether the hardware interface is connected correctly, whether the chip is encrypted (if encrypted, there is no solution), and so on, until the connection is successful before proceeding to the next step.



4. Reading the chip: After the device successfully connects to the target chip, the Read button will change from gray to black, indicating that reading is possible, as shown in the figure below. The red-framed option indicates that PFLASH is currently selected, with a capacity of 2Mbyte. The FLASH capacity must match the chip's FLASH capacity to ensure all data is read.
PFLASH represents the program data area, which must be read and written; DFLASH or EEPROM stores vehicle information, fault codes, calibration information, and other parameters, which may or may not contain data. The read/write methods for these areas are similar to those for PFLASH, and the red-boxed area allows you to switch between the areas to be read or written.



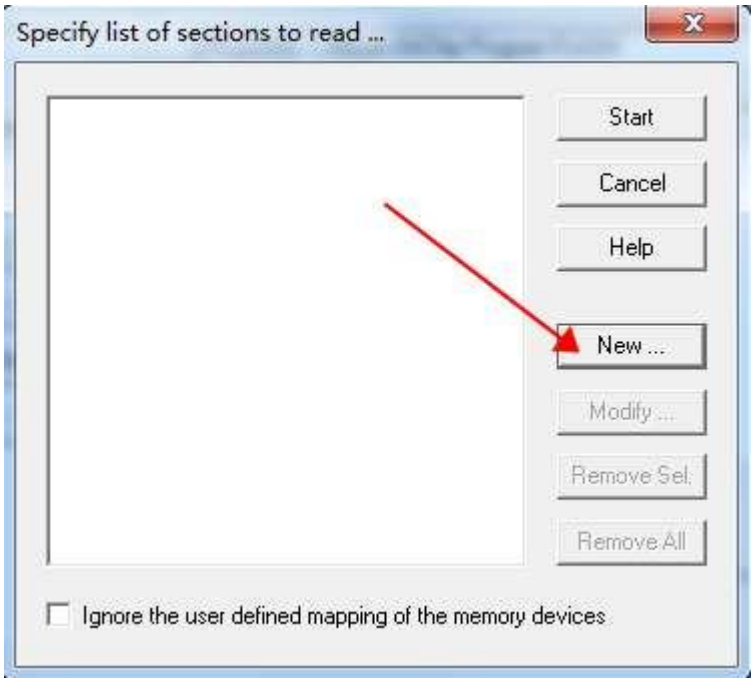
Here, you need to check and remember the start address and end address in the address bar box. In the leftmost row of the figure below, Index 0 has the start address A0000000, and in the rightmost row of the figure below, Index 6 has the end address A01FFFFF. The start address and end address must be set correctly to ensure complete reading.

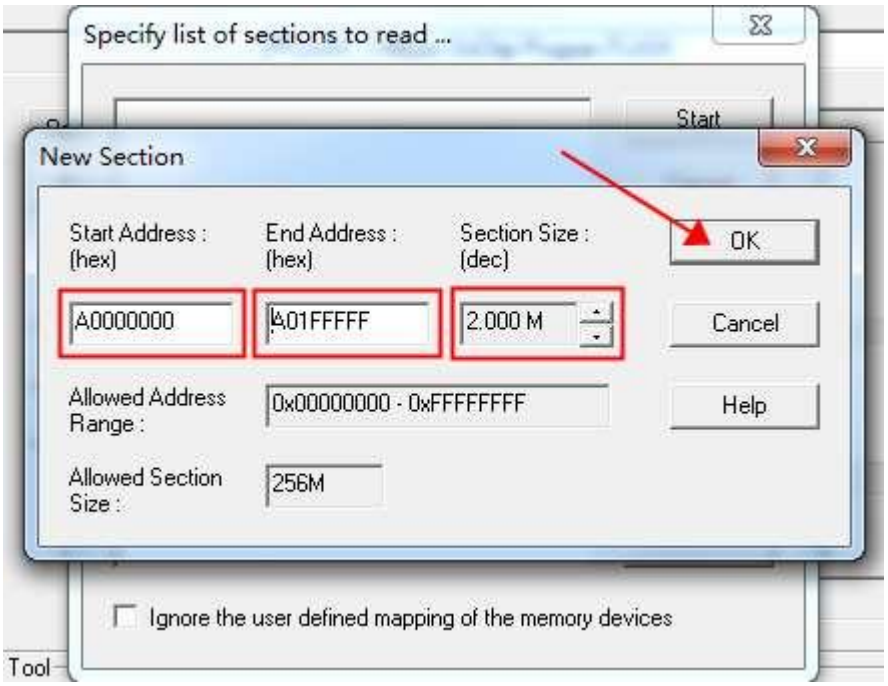


Click the Read button. In the new interface that appears, as shown in the figure below, if the left box is empty, click New to fill in the start address, end address, and size settings. You can also select the address range and click Modify to modify it.

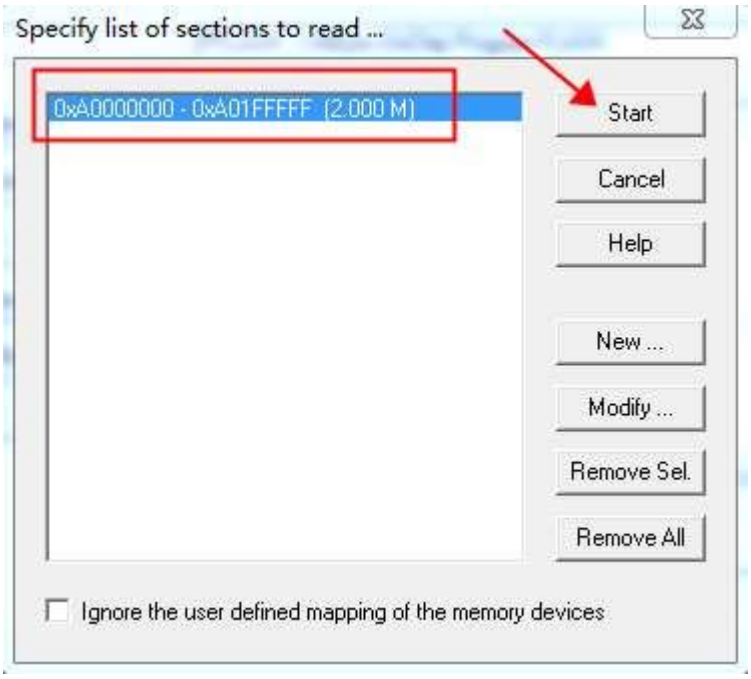
(If the addresses are not continuous, you need to set the addresses in segments by Index,

that is, create another address range and read it out at once with Start.)

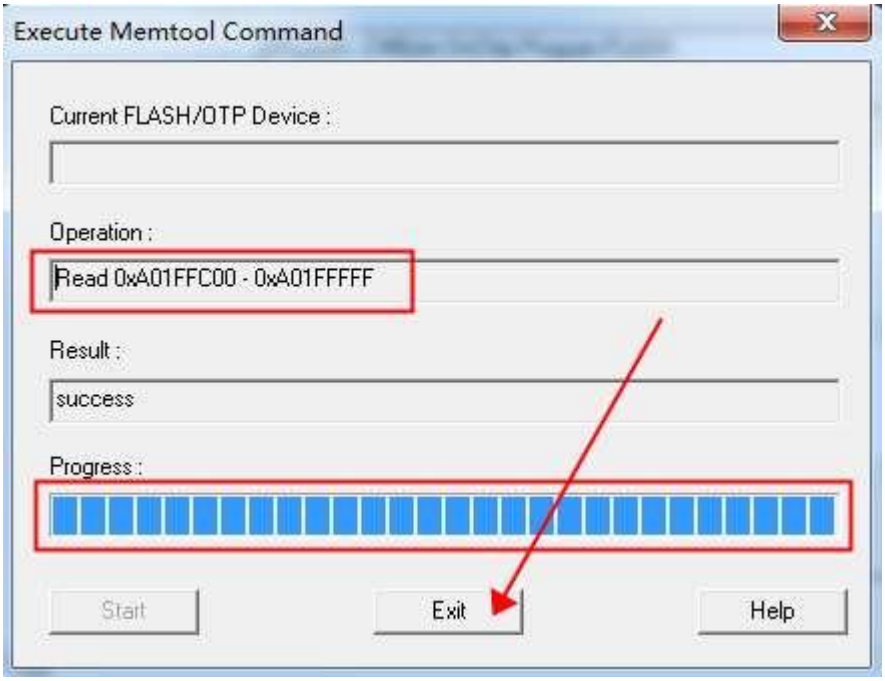




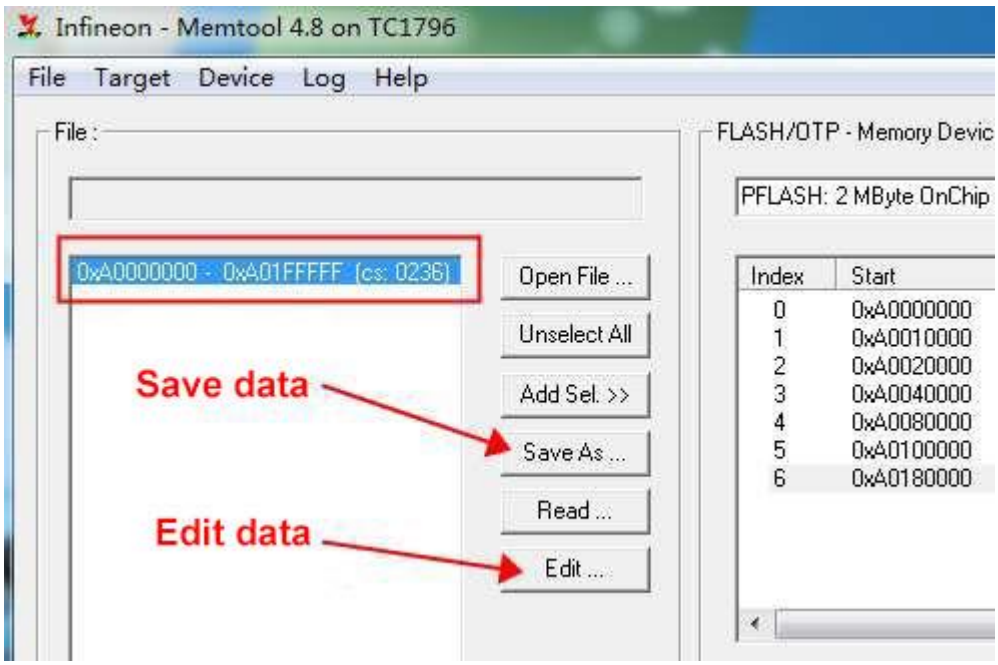
If the address information in the left box below is correct, click Start to begin reading all the address ranges listed in the left box, and a progress bar will appear.



When the prompt shown in the figure below appears, it indicates that the reading was successful. Click Exit to exit, and the chip data reading is complete.

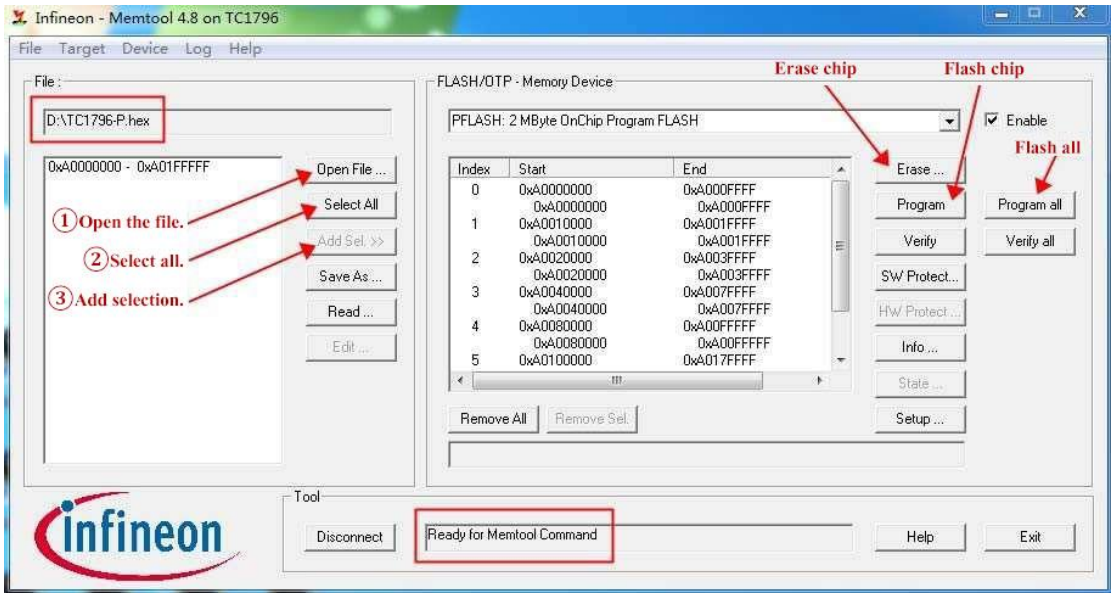


After reading the data, you need to click Save as ... to save the data file in .hex format. The read HEX data can be opened, viewed, and edited using other software, or you can directly click Select All and Edit in this software to view and edit (view only, no editing), as shown in the figure below.

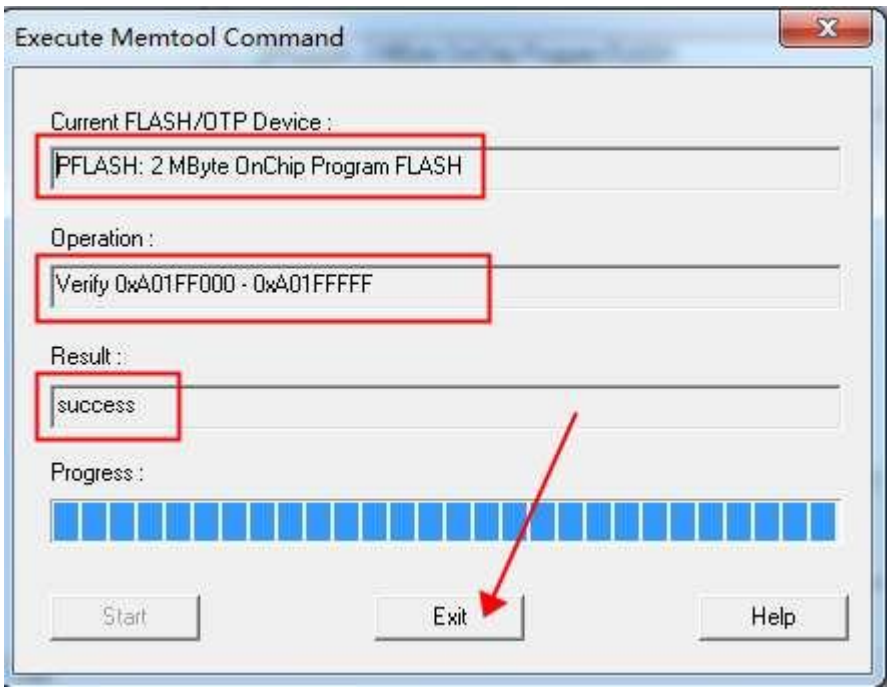


5. Chip programming: It is also necessary to ensure the correctness of the programmer driver, hardware connection, and target board environment. You must first execute Connect to ensure that the device is successfully connected to the target chip.

Using the PFLASH region as an example, click ① Open File to select and open the HEX-formatted data file you just saved or an existing one; then perform ② Select All and ③ Add Sel, as shown in the figure below. After that, you can proceed with the chip erase (Erase) and chip programming (Program) operations. (Bin files can also be opened via File → Open Bin File…)

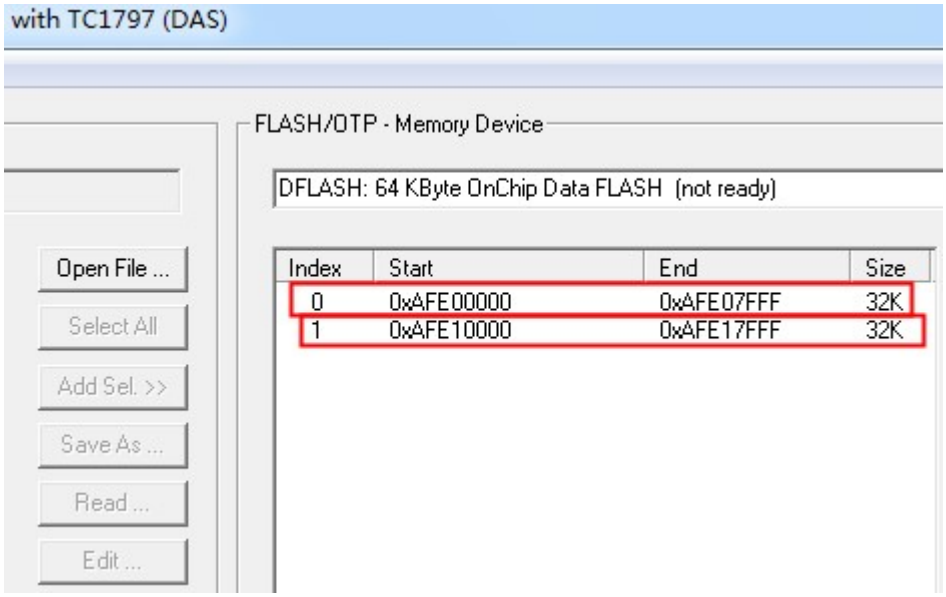


By directly selecting “Program,” the current selected PFLASH will be automatically erased, written, and verified, with progress prompts displayed until the following message appears, indicating that the PFLASH data has been successfully written and downloaded. Click ‘Exit’ to close. (The “Write All” button is not necessary.)

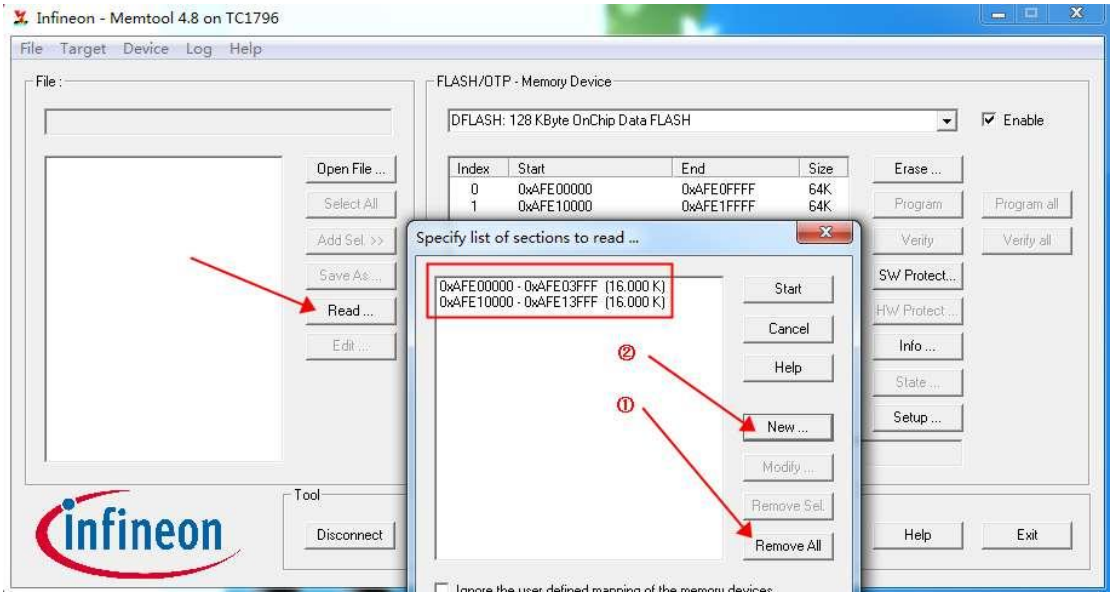


Okay, PFLASH zone read/write complete! (It is indeed cumbersome, but as long as you can read and write, the problem can always be solved.) Other settings or operations not mentioned in this tutorial can be left at their default values, ignored, or figured out on your own.

If read/write operations on the DFLASH region are required, the procedure is similar to that described earlier for PFLASH. For example, first select the DFLASH read/write target, verify the start address and end address, and note that the address range may vary depending on the model. The addresses may also be non-contiguous, such as in the TC797 model, where the DFLASH read/write ranges must be set in blocks (Block 0: AFE00000 – AFE07FFF, Block 1: AFE10000 – AFE17FFF), as shown in the figure below.



Still select Read, first use Remove ALL to remove the original address range items in the box that may be incorrect, making it empty, and then select New to set the new address range (usually New into 2 blocks). The 2 blocks of DFLASH address ranges that have been set should be the same as the addresses displayed in the previous interface, and then you can operate Start.



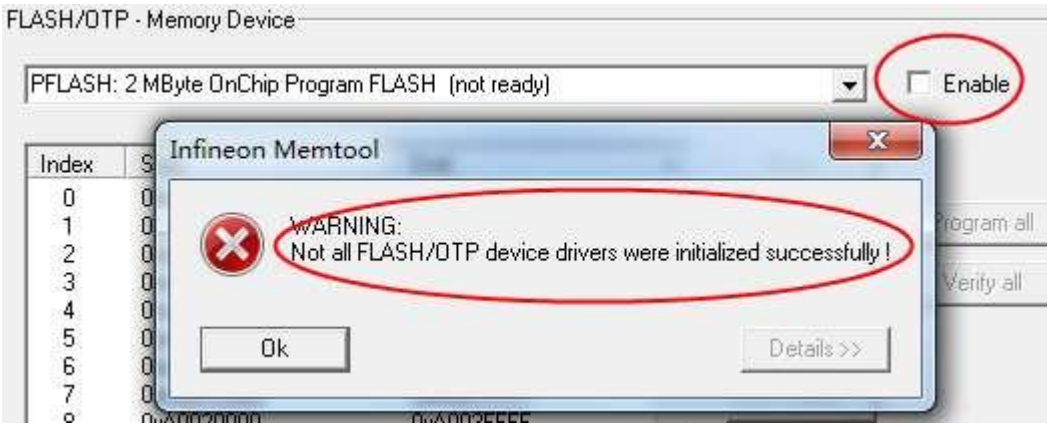
Read DFLASH.

OK. It's quite complicated, but I wish you success!

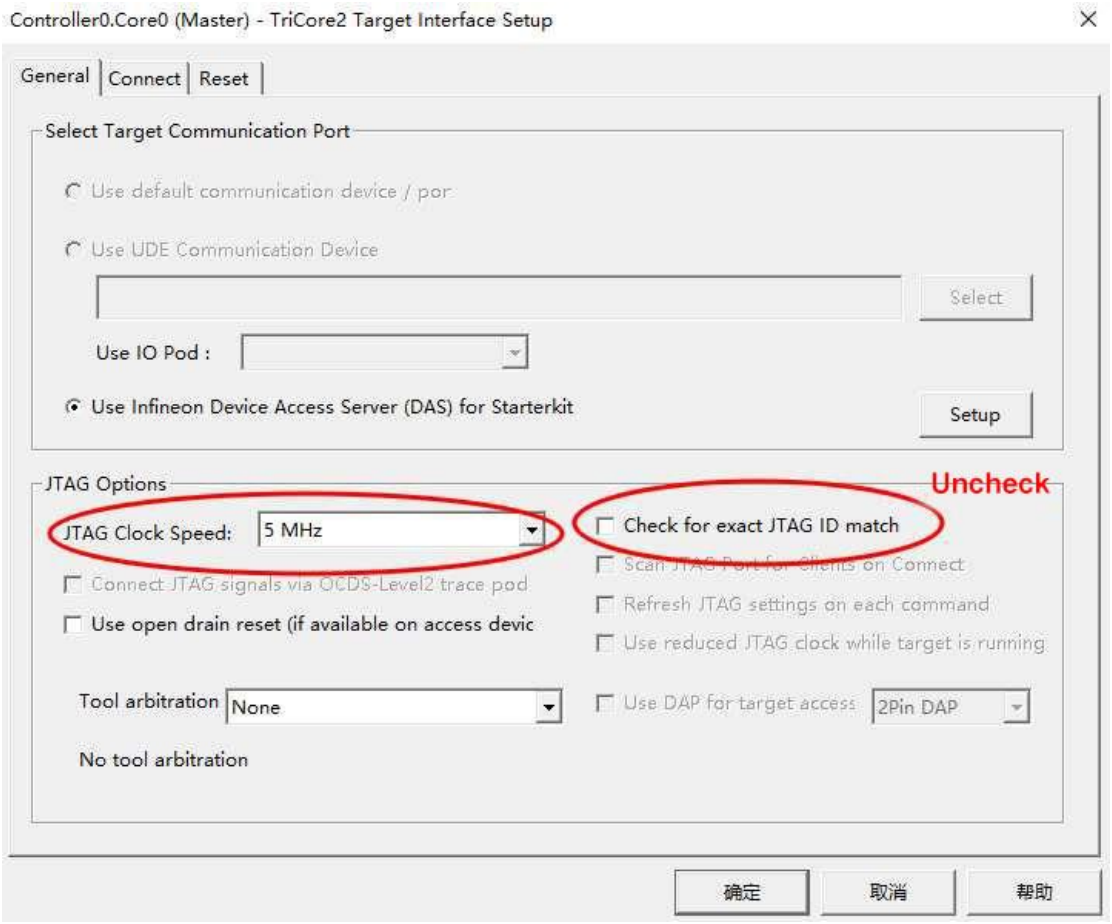
Frequently Asked Questions:

1. If you are programming a brand-new blank chip or cannot connect, you need to configure the Boot mode to internal Flash boot mode to establish a connection. refer to the chip data sheet for the HWCFG definition. For example, you may need to temporarily set the TC17xx pins P0.6/P0.7 or TC2xx pins P10.5/P10.6 to a high level, while the TC1766/TC1796 is more complex.

2. If the error message shown in the figure below appears when connecting the chip to Connect, it is usually because the PORST signal is affected and needs to be isolated; it may also be a model selection or configuration issue (the key point is that the JTAG Clock in Target →Setup should be set to 5MHz or lower, not 10MHz). If an error occurs during writing, you can also try lowering the speed.

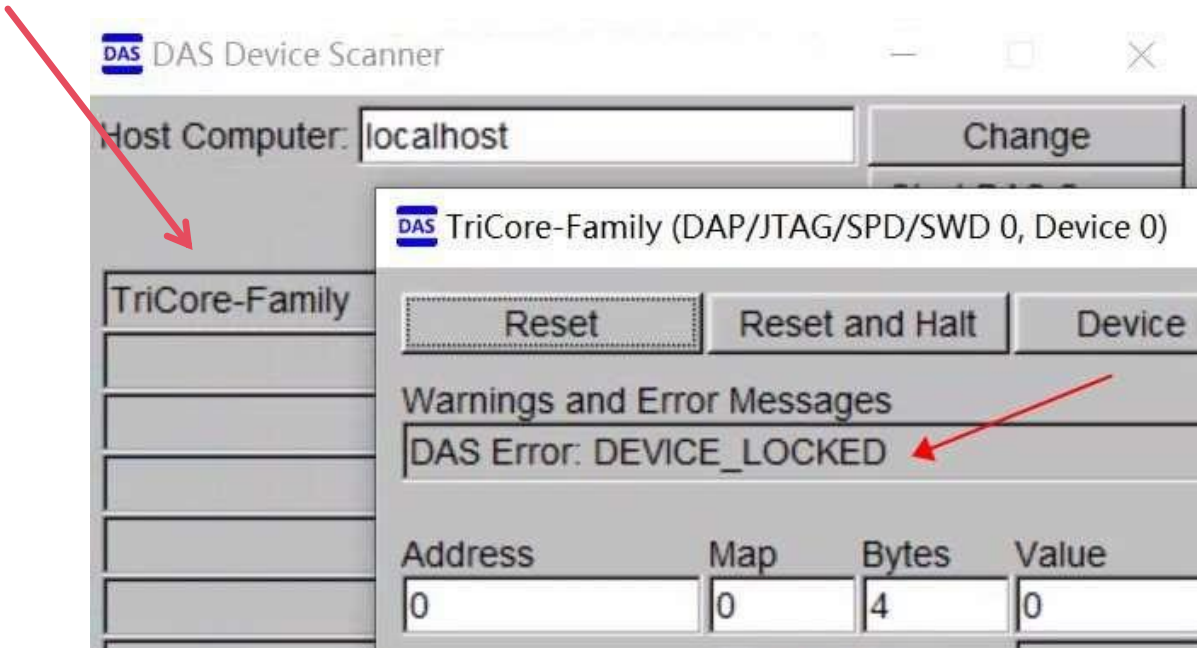


3. For models TC275/277/297, the configuration is different. Target→ Setup should be set as follows to work:



4. If you cannot connect to the target chip, you can first use the background scanning software to confirm whether the power supply and interface connections are correct. Locate and run C:\Program Files\DAS64\clients\das_device_scanner.exe，→Start DAS Server →Install Servers → Start. Return to the initial interface and click Start Client. If the chip category is identified as TriCore-Family, it indicates that the power supply and interface connections are correct. Otherwise, further

troubleshooting is required. If a “LOCKED” message appears, it may indicate that the chip is either locked or that the Boot pin configuration is incorrect. (If using a 32-bit system, the scanning software is located in the \DAS\ folder.)



5. If the above steps fail repeatedly, you will need to remove the target chip and place it on our programming socket + adapter, which will ensure a successful connection and read/write operation (except for encrypted chips).