EAI Software Develop Kit

Preliminary User Guide

Version 1.1

2020-04-14

History

Version	Date	Description	Editor
V1.0	2019/10/21	EAI SDK user guide initial release	Misty
V1.1	2020/04/14	Add compiler and other contents	Misty

Contents

Figures

1. uGelis OS Introduction

uGelis embedded OS is a Edgeless self-developed AIoT operating system, which currently supports running on Edgeless EAI series AI MCU.

EAI series chips are based on ARM cortex M4F processor architecture, supporting armv7-m instruction set architecture and 32-bit address space. The chip adopts multi-core architecture, integrates dual core arm cortex-m4 and CNN convolutional neural network hardware accelerator, and supports offline real-time AI algorithm applications (vision and voice). For details of the chip, please refer to the relevant data sheet and reference manual.

1.1 uGelis Feature

- Ultra lightweight
  - Minimum 8K RAM and 20K ROM
  - Highly configurable
  - High portability
- task management
  - multithreading support
  - priority support
  - multiple scheduling algorithms (FIFO, LIFO, RR)
- Support memory management

  - memory pool
  - memory mapping
  - large memory
  - continuous physical memory

- support multiple network protocols
  - support BLE ,IEEE802.15.4 ,Wifi ,NB-IOT(later),LoRA(later)
  - support IPV4/IPV6
  - support 6LoPAN implementation
  - support TCP/UDP transmission
  - support HTTP,COAP,MQTT
  - support Thread protocol
  - support json format

- support rich components
  - FAT file system
  - USB protocol stack
  - unified drive architecture
  - Power management system
  - Internet of things collaborative framework
  - Java script engine
  - GUI

- Support multiple integrated development environments
  - Linux
  - MinGW
  - GreeIDE
  - Keil
- multiple upgrade methods
  - USB Device
  - USB Host
  - Serial port upgrade
  - SD/MMC upgrade Http upgrade
  - Http upgrade

1.2 uGelis SDK Architecture

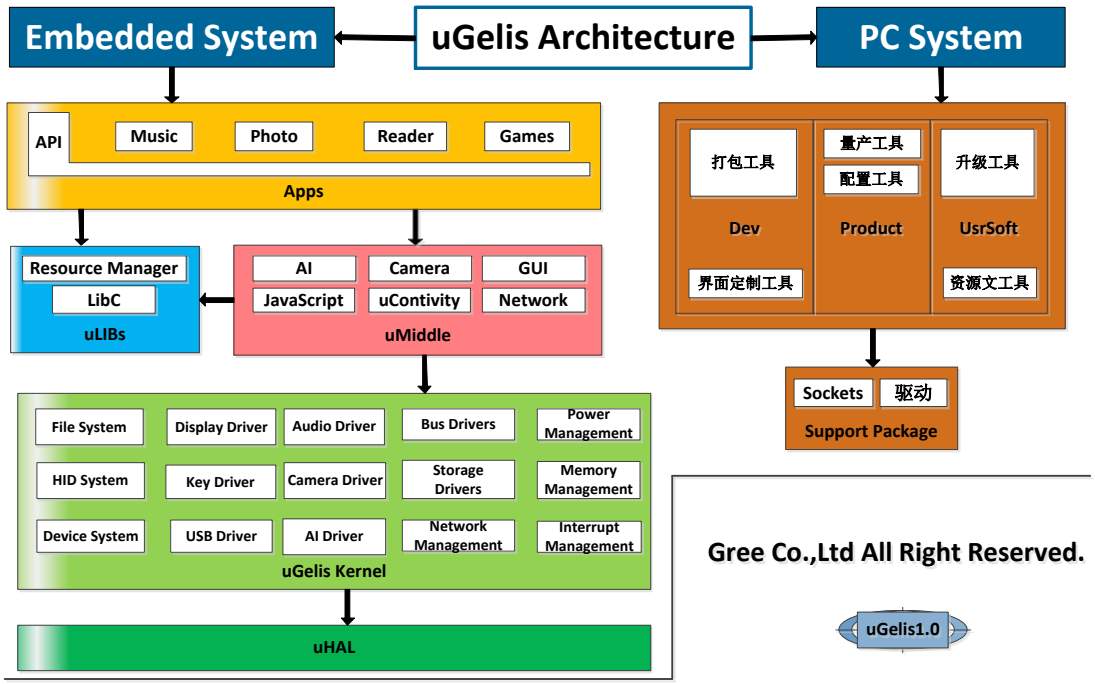


Figure 1 uGelis SDK Architecture

The SDK of Edgeless EAI series AI MCU includes six parts: uHAL, uGelis Kernel, uMiddle, uLibs, App and PC Tools , It is used to simplify and speed up the development of applications on EAI microprocessors. The brief description is as follows:

1) uHAL
2) uGelis Kernel
3) uMiddle
4) uLibs
5) Apps
6) PC Tools

1.3 SDK Description

（1） uHAL

uHAL is a board level support package for EAI series chips. Provides the most basic physical layer driver interface that is independent of OS. The uhal layer supports running directly on keil. It is convenient and fast for porting to other OS, such as UCOS, RT thread, FreeRTOS.

（2） uGelis Kernel

uGelis Kernal includes the system files, management mechanism and various driving mechanisms of ugelis, such as device system, power management, interrupt mechanism, thread synchronization

mechanism, etc. Ugelis kernel is the core of ugelis system, which provides basic and structural functions.

(3) uMiddleware

uMiddleware is a middleware system provided by ugelis, such as IOT interoperability system ucontivity, IOT JavaScript engine, network protocol framework, etc. At present, the middleware system of umiddleware only supports ugelis system, but it can be easily transplanted to other OS systems, such as UCOS, RT thread and FreeRTOS.uMiddleware.

(4) uLibs

uLibs stores the most basic dynamic link shared library of the system, and its function is similar to the DLL file in windows. These shared libraries are required for almost all applications.

(5) Apps

Application development - software developers can develop applications with various functions, such as music, images, documents and games, based on the basic services provided by the ugelis system.

(6) PC Tools

PC Tool refers to some packaging tools, mass production tools, upgrading tools and support packages used on the PC side. Such as Easynet compiler, uGelis Flash, etc

1.4 uGelis Directory Structure

```
├─── ugelis
│    ├─── arch
│    ├─── boards
│    ├─── boot
│    ├─── cmake
│    ├─── CMakeLists.txt
│    ├─── CODEOWNERS
│    ├─── CONTRIBUTING.rst
│    ├─── doc
│    ├─── drivers
│    ├─── dts
│    ├─── export
│    ├─── ext
│    ├─── include         //Ugelis all public header files
│    ├─── Kconfig
│    ├─── Kconfig.ugelis
│    ├─── kelis_example   //Apps
│    ├─── kernel
│    ├─── lib
│    ├─── LICENSE
│    ├─── Makefile
│    ├─── misc
│    ├─── README.rst
│    ├─── samples
│    ├─── scripts
│    ├─── subsys
│    ├─── tests
│    ├─── tools           //tool on pc side
│    ├─── ugelis-env.sh
│    ├─── VERSION
│    ├─── version.h.in
│    ├─── win-ugelis-env.cmd
│    └─── ztest
└─── ulibs
     ├─── hal
     └─── midware
```

Figure 2 uGelis Directory Structure

2. SDK User Guidance

2.1 Preparation

(1) Software and hardware preparation

- PC
- chip hardware platform
- Windows 7 or above
- SDK
- Keil
- Pack files for EAI
- serial software
- hardware related drivers

(2) install Keil

Please install keil above v5.18, because the pack file used by the microprocessor is made of Keil of v5.18. The lower will not recognize the pack file used by the microprocessor. After the installation of Keil, please be sure to register. If you don't register Keil, you can only compile the image with the maximum 32kbyte. If your software is larger than 32kbyte, there will be an error during compilation..

(3) Install Device Family Pack

If you have obtained the pack file of EAI series microprocessors from regular channels (you must obtain the correct version of the pack file, the

higher the general version number, the more chip models supported by the pack file), just double-click the pack file to install, and then click the "next" button until the installation is completed. If the installation is successful, there will be a folder named "gm66xx_dfp" under the directory "XXXX \ arm \ pack \ keil \". XXXX "is the installation path of Keil uVision.

（4） Hardware platform connection

Please confirm that the JLINK driver is installed before connecting.

The connection mode is that PC connects JLINK through USB, and then JLINK connects development board through SWD; serial port connection line connects development board through UART; connect power line to supply power

2.2 Hello World Demonstration

All the demos in the SDK are in the directory of ugelis \ Kelis \ example. Select the subdirectory app to open the Hello world project file.

2.2.1 Hello world project demonstration in keil_example

Project execution process:

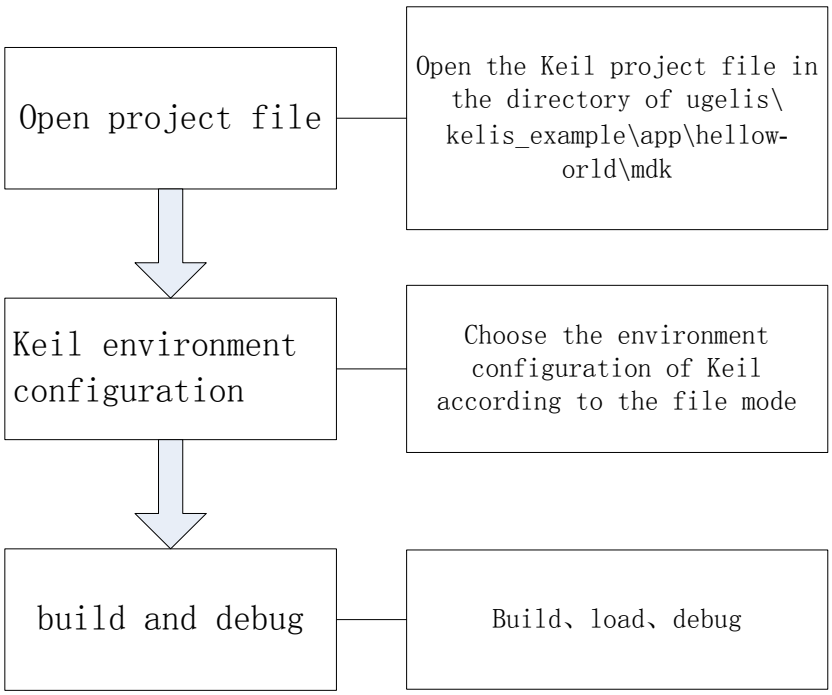


Figure 3 Project execution process

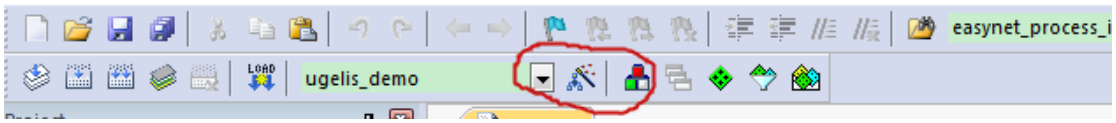
（1） Open project file

Open the Keil project file in the directory of ugelis\kelis_example\app\hellow-orld\mdk ;

Figure 4 Open project file

（2） Keil environment configuration

Step 1:



Step 2:

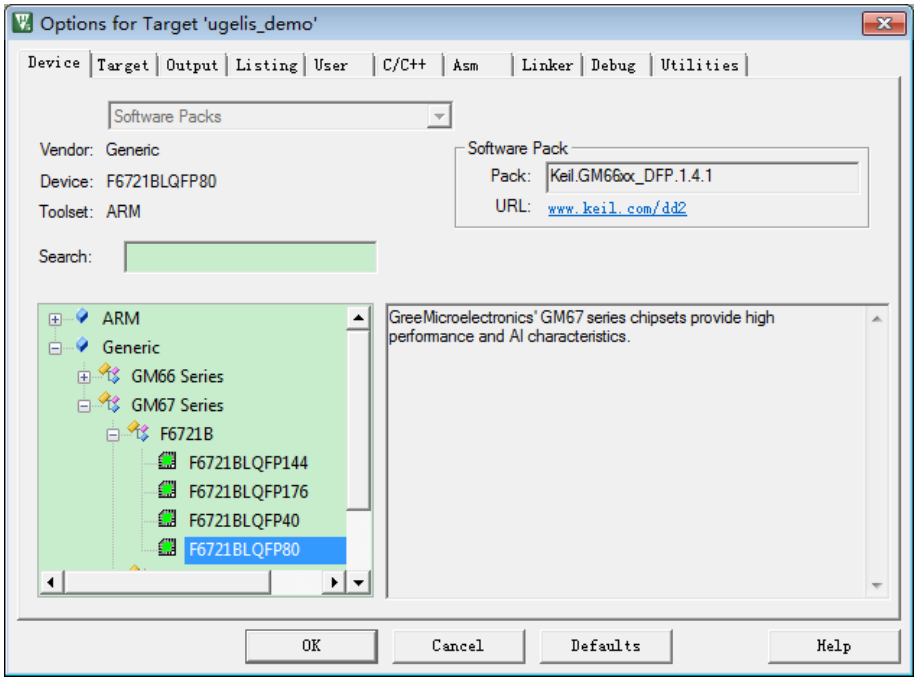


Figure 5 Select target

Note: Please selected according to the die model and PKG, which is F6721B;

（3） Setting debug option

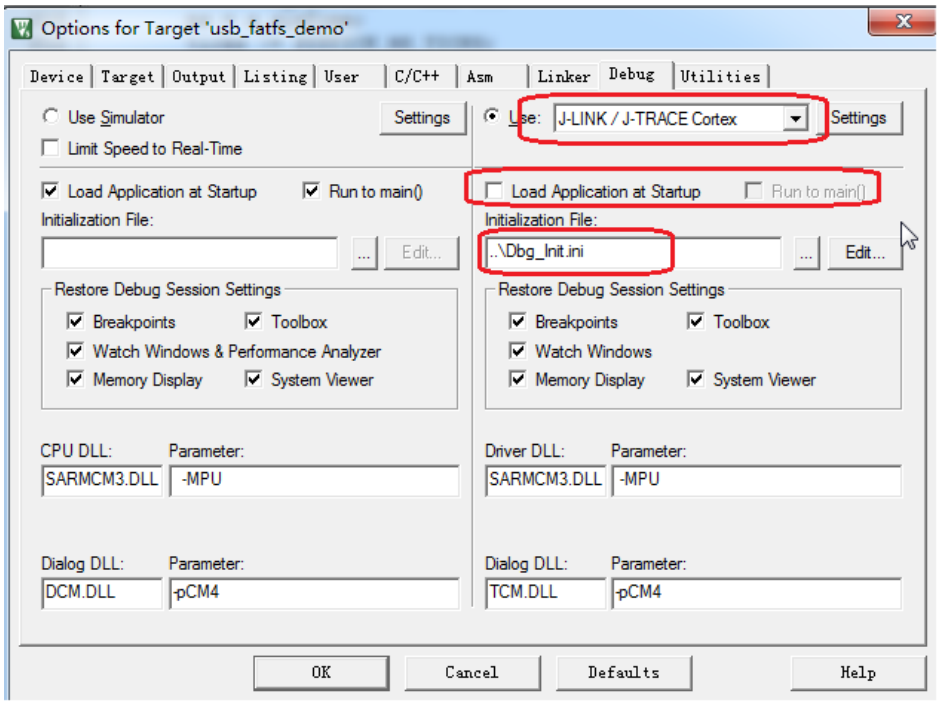


Figure 6 Setting debug option

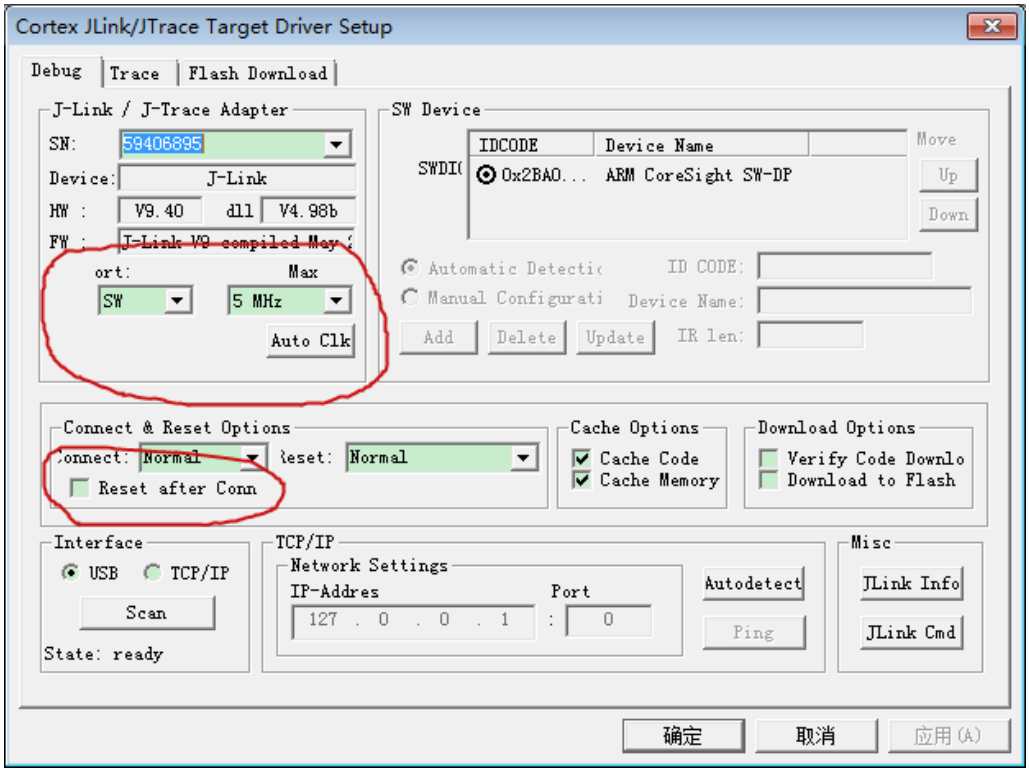


Figure 7 Setting debug option

（4） Setting flash option

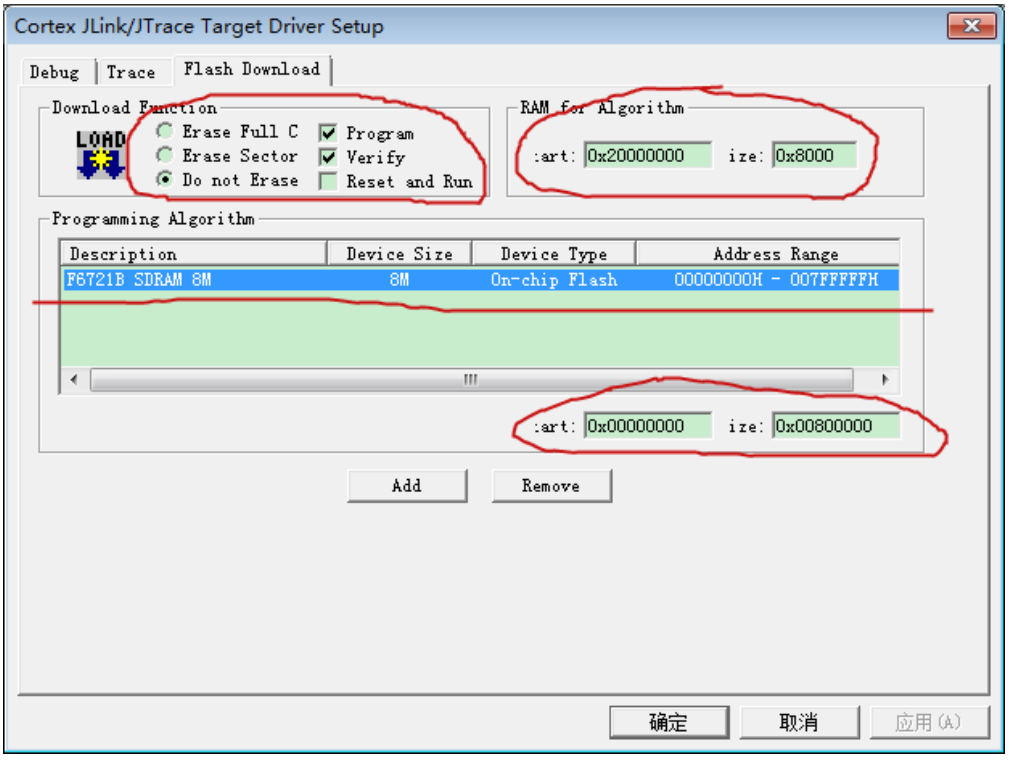


Figure 8 Setting flash option

（5） Build，download，debug

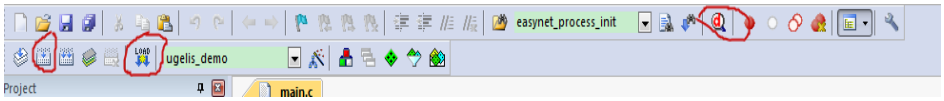


figure 9 keil toolbar

Press the button in the upper left corner of the toolbar  to compile the project，load the generated AXF file in the upper left corner  ，and debug the generated AXF file in the upper right corner  .

2.2.2 New project

- Mode 1: copy projec

  Copy a project file from the above demo directory, and then change it according to the directory structure of the project file with the actual situation.

- mode 2: re create

（1） New directory

Create a new directory under the " \ugelis\kelis_example\" , such as "create_demo".

（2） New project

As shown in the following figure ,open the Keil integrated development environment, click "new uVision project" from the "project" drop-down list, name the project, and click "save" to save.

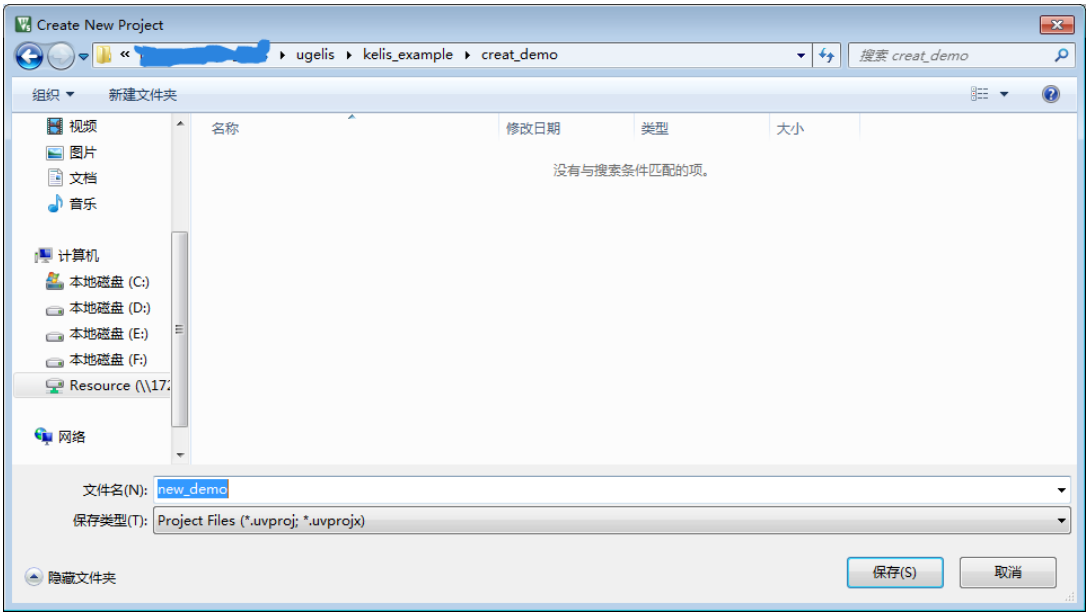


Figure 10 New project

（3） Select device

Choose the right equipment for the project

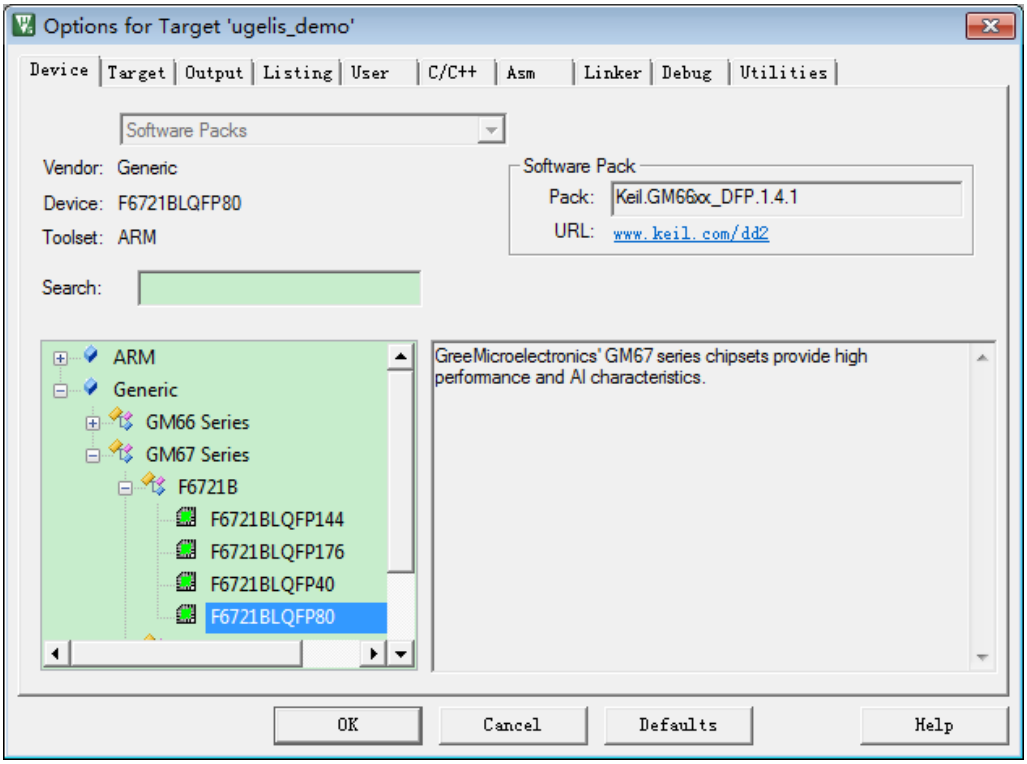


Figure 11 Equipment selection for new project

（4） Select target

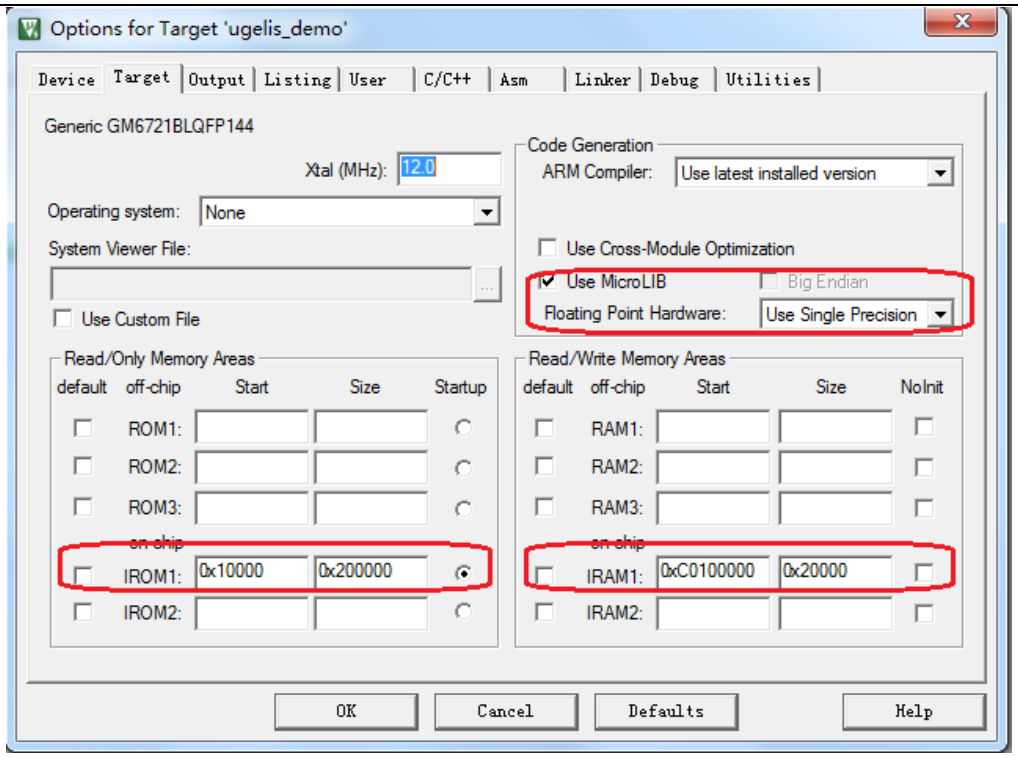


Figure 12 Select target for new project

（5） Set project name and create executable file

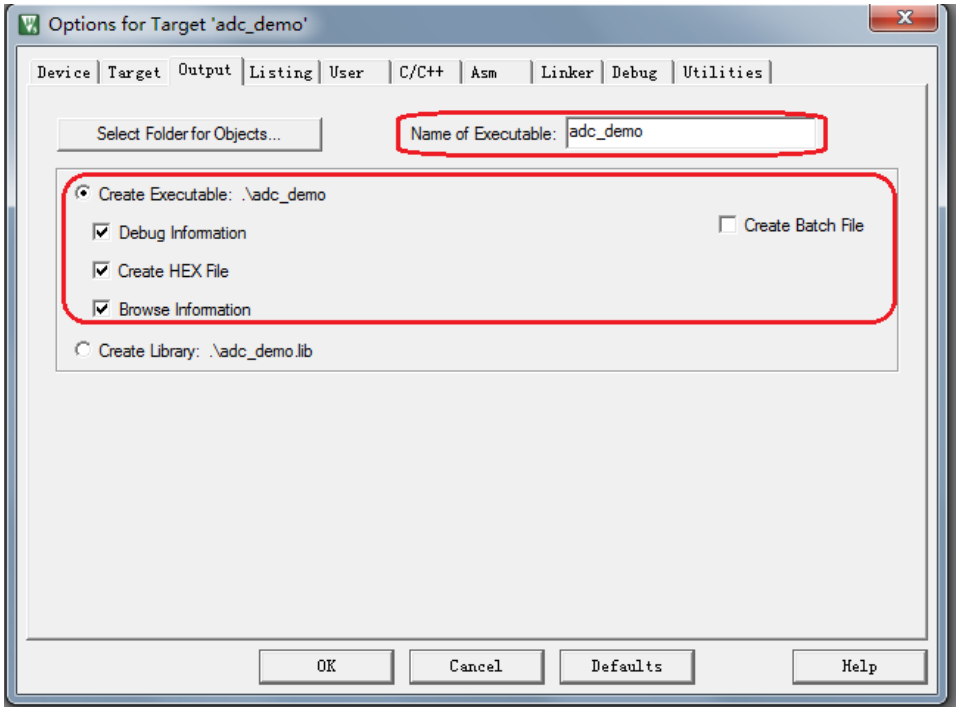


Figure 13 Create executable

（6） Set Include path

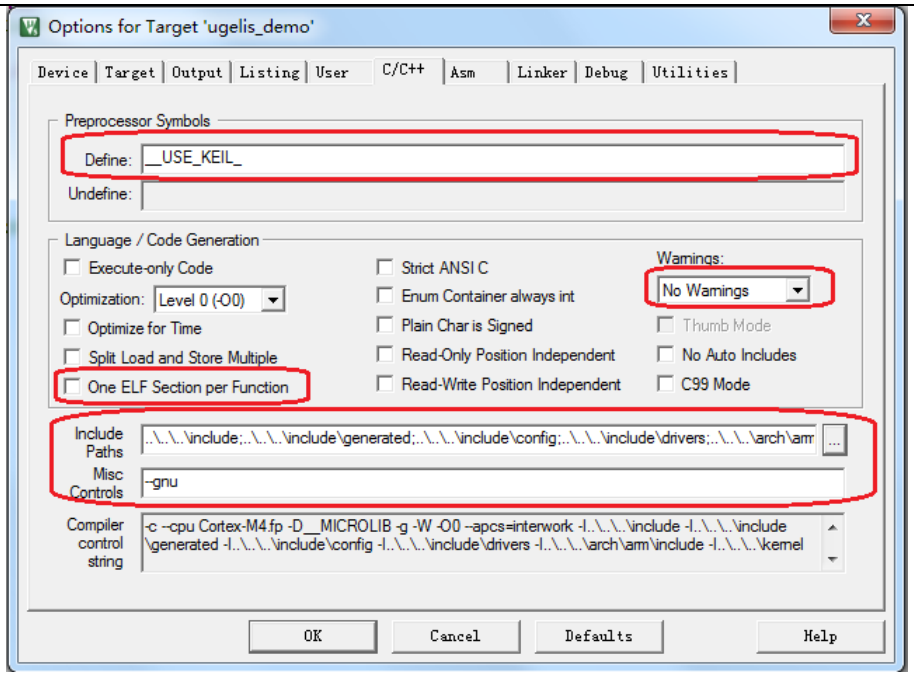


Figure 14 Setting include path

Note：Refer to the engineering setting of kelis_example/adc

（7） Setting Linker options

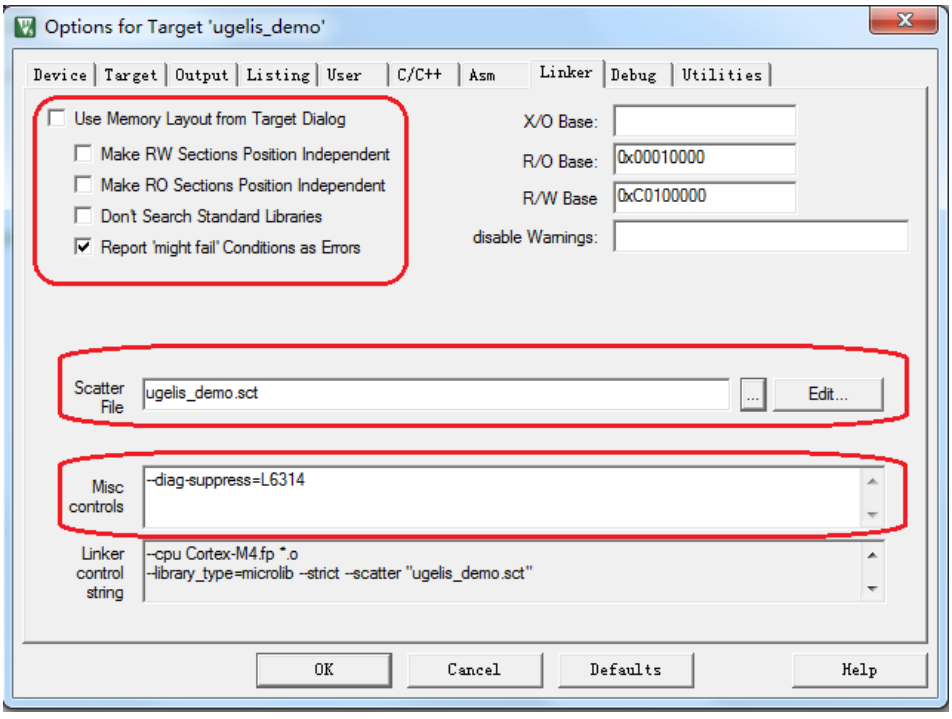


Figure 15 Setting Linker option

（8） Setting Debug option

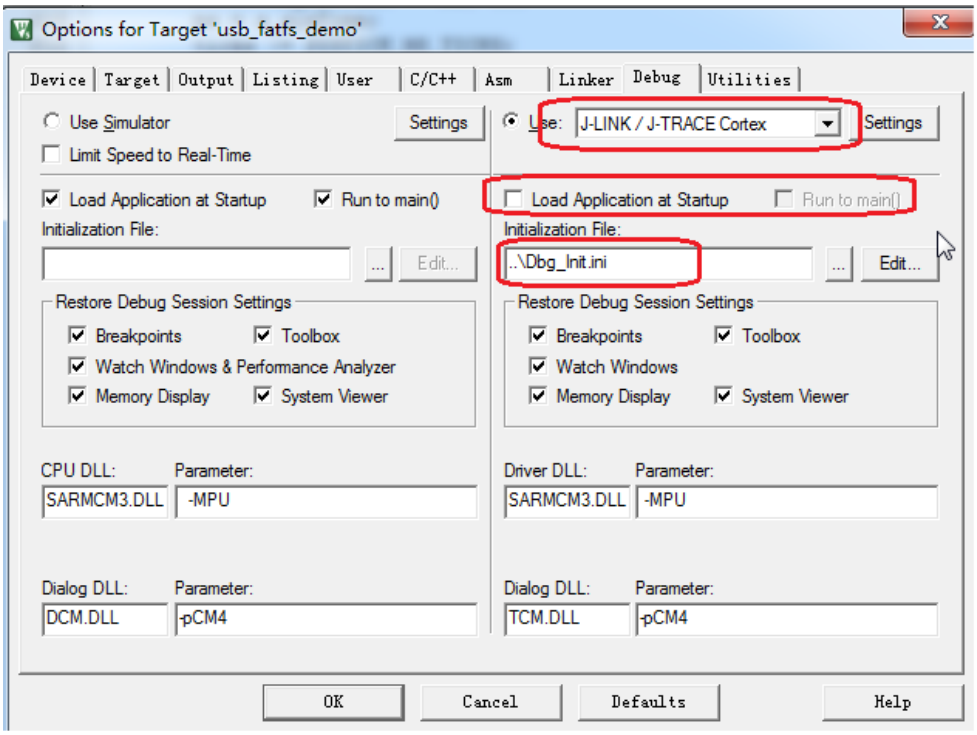


Figure 16 Setting debug option 1

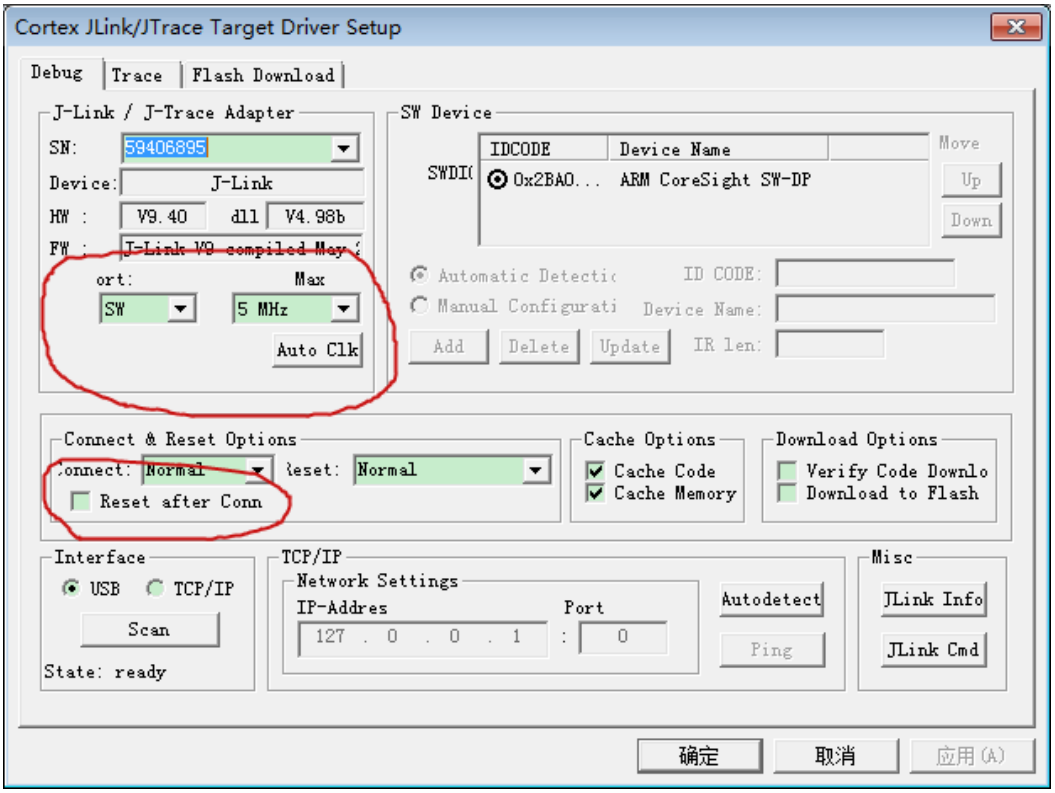


Figure 17 Setting debug option 2

（9） Setting Flash option

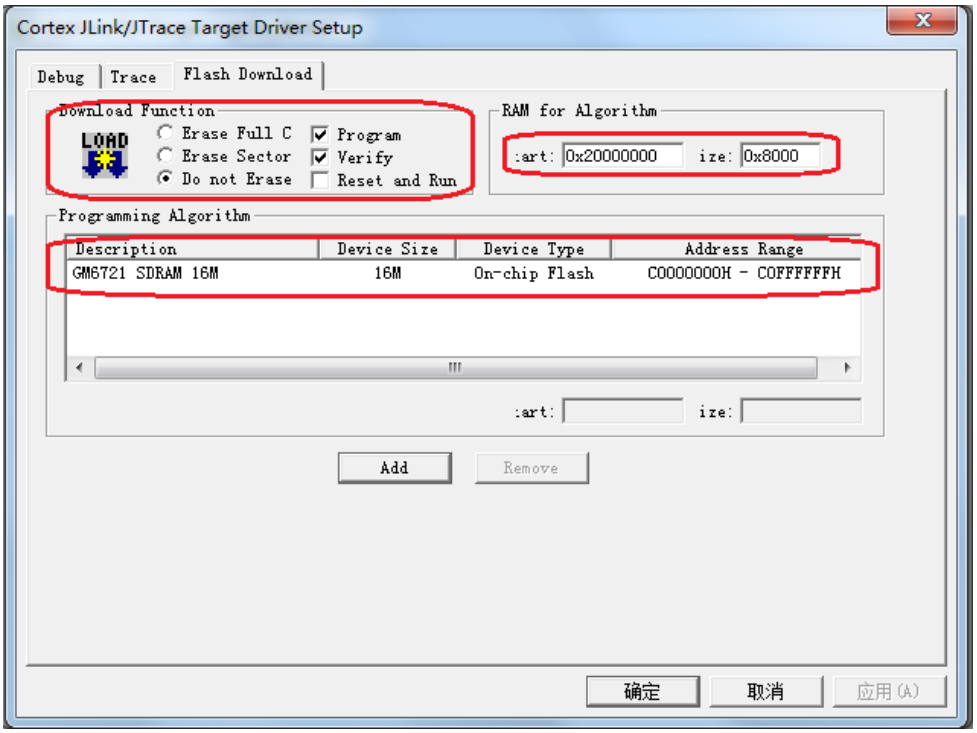


Figure 18 Setting Flash option

（10） Setting Utilities option

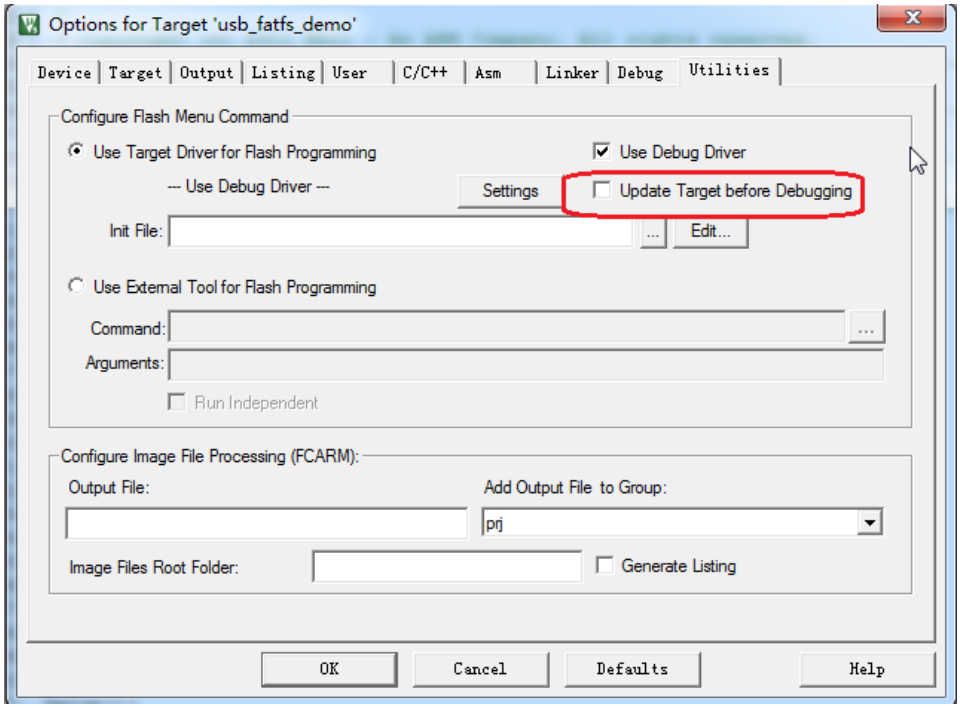


Figure 19 Setting utilities option

3. Flash Burning and Starting

3.1 uGelis Burning

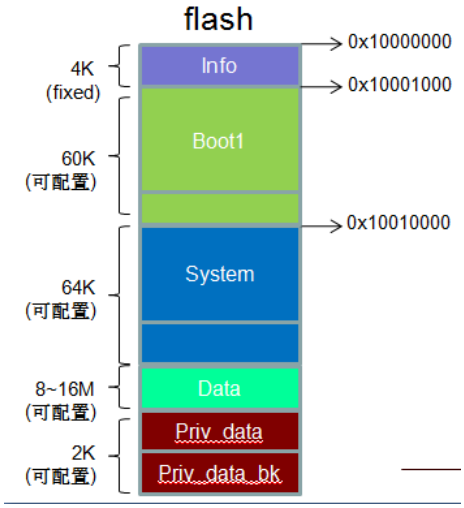


Figure 20 Flash partition diagram

- Flash partition :

Info partition: used to store partition information, fixed at the start of flash, and the partition size is fixed to 4K;

Info partition data format :

Struct Info

{

//total partition number

Uint32_t part_num;

// partition 0: Flash storage location , realsize, address where data is loaded into memory (optional)

Uint32_t info_addr;

Uint32_t info_size;

Uint32_t info_load_addr;

// partition 1: Flash storage location , realsize, address where data is loaded into memory (optional)

Uint32_t boot1_addr;

Uint32_t boot1_size;

Uint32_t boot1_load_addr;

// partition 2: Flash storage location, realsize, address where data is loaded into memory (optional)

Uint32_t system_addr;

Uint32_t system_size;

Uint32_t system_load_addr;

// partition n: Flash storage location, realsize, address where data is loaded into memory (optional)

...other partitions

<Padding...>

};

Boot1 partition: boot1 data, partition location and size can be configured;

System partition: ugelis data, partition location and size can be configured;

Other partitions: data / priv_data / weights and other data can be stored, and the partition location and size can be configured;

- of which：

Boot0 is stored in OTP. After the chip is powered on, it starts from boot0, which is responsible for loading boot1;

Boot1 is stored in flash and is responsible for loading the system;

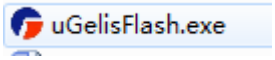
System is stored in flash, which is the main program of ugelis.

3.2 OTP Buring

If you need to power on and start normally, you need to confirm that the OTP program of the chip has been written (the chip that the customer gets has written the OTP program);

If you only need ide to download ugelis program to debug, you do not need to burn OTP.

3.3 uGelisFlash Use

Place the ugelisflash tool in the directory of ugelis \ tools \ ugelisflash, and click the icon 

to start using it.

Ugelisflash is a flash tool, which burns info.bin, boot1.bin, system.bin and some application files to the specified location of flash.

The current root directory of ugelisflash tool has the default partition file config.ini which covers the partition information and the real size of each partition. The ugelisflash tool will automatically generate the info partition file according to config.ini Therefore, if the partition file layout and the partition file size are updated, you need to re burn the info partition. If you need to

customize partition, you can change its contents;

The meaning of each field in Config.ini is as follows：

/* Config.ini */

[PARTITION_COUNT] → total partition number ,

COUNT=9

[PARTITIONX] → partition n , start with 0

NAME=info → partition name

SIZE=0x1000 → partition size

FILE=info.bin → partition file name

LOAD_ADDR=0 → address where data is loaded into memory (optional)

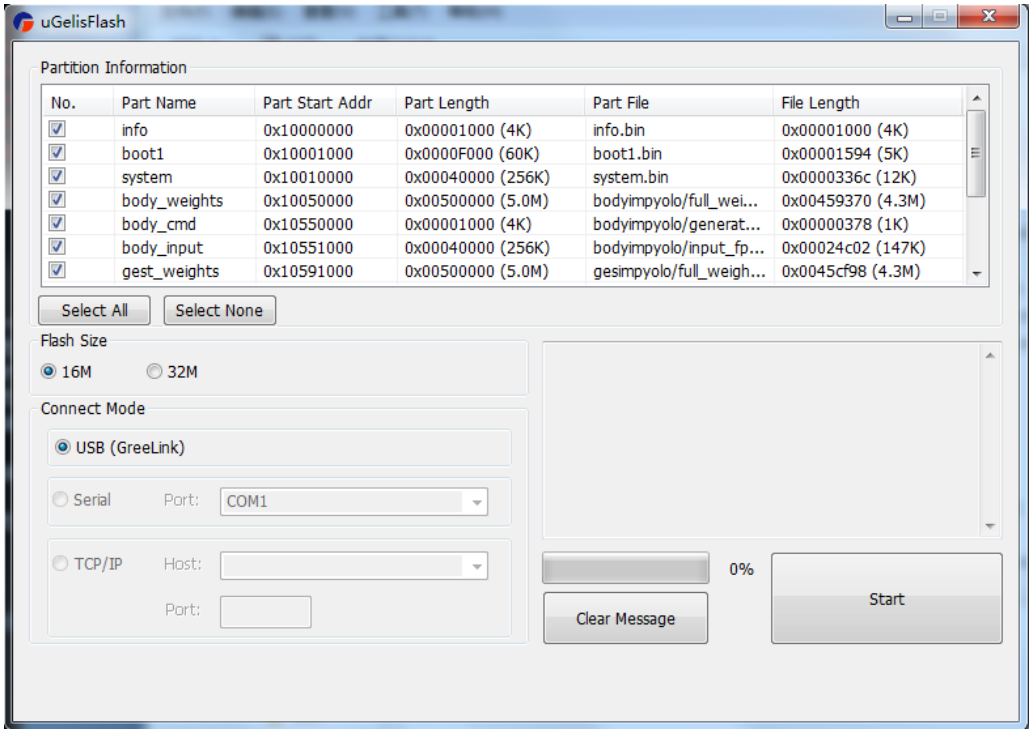


Figure 21 uGelisFlash

Use: configure the config.ini file, open the tool, select the required burn partition, select flash size, and click start. The tool burns the program file to the designated location of flash according to the config file. After using this tool to burn, you can start to complete AI function in keil by calling .OPS file content.

4. Compiler Use

Easynet compiler is a cross platform tool suite for neural network model compilation, forward computing, simulation, demonstration and other functions. Using this tool, it is easier and more efficient to run AI applications on the embedded side.

4.1 Feature

- Support model framework: Keras
- CNN features supported:
  - support convolution kernel sizes 1-7
  - supports a maximum of 512 channels
  - supports Relu、 Leaky Relu、 Batch Norm
  - supports max pooling of 2x2, 3x3
  - 256Kb SRAM (4 banks) and 32KB SRAM (SPAD) accessible externally
- Application:
  - gesture recognition
  - body recognition
  - face recognition
  - voice key

4.2 Compiled Model

4.2.1 Compiler operator

operator	support	kernel_size	stride	channel	padding
Conv	√	1~6	1~3	1024	‘same’ ; 'valid'
		7	2		
BatchNorm	√				
Relu	√				
LeakyRelu	√				
AveragePool	√				
MaxPool	√	2x2; 3x3	2		
Depthwise	√				
FullConnect	√				
Route	√				
Shortcut	√				
Upsample	√				

The compiler supports the compilation of the above operators. The model composed of the above operators can be compiled in theory, such as RESNET network.

4.2.2 Verified network

The following similar networks have been verified by simulation：

·Tinyyolov3

·Tinyyolov2

·Mobilenet

·Mobilenet-ssd

·Posenet

·Insightface

4.3 Function introduction

The compiler is placed in ugelis\tools\compiler\ easynetcompiler00b02-.zip. After decompression, open the compiler directory easynetcompiler00b02, double-click to run env init, double-click to run start_compiler.bat, and the compiler interface pops up.

4.3.1 Main Function

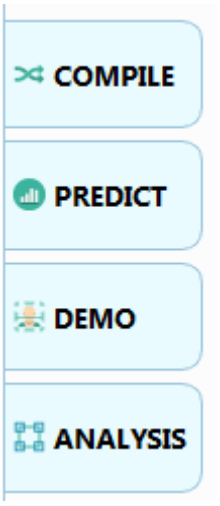


figure 22 major function

- COMPILE

  - compile the model into binary file and copy it to small machine side for use

- PREDICT
  - give the model an input, do forward operation, and view the output results of each layer
- DEMO
  - do Easynet post-processing according to the input, and visually display the demonstration screen
- ANALYSIS
  - parse operation to help debugging

4.3.2 Compile

Compiler input

Keras model compilation: easynetcomputer supports the conversion of keras to Easynet model, HDF5 can be converted into .cfg and .wt files that can be executed by compile.

Compiler output

The binary storage location of the compiled output is output / seq_ai.ops, which contains the command sequence (cmds) and the weight after quantization. After the compilation is successful, the seq_ai.ops is written to the EAI chip, loaded, parsed and run by the program to complete the corresponding CNN network automatic operation.

Interface operation

compile the keras model:

(1) Select keras in type;

(2) Select the existing keras model, the .H5 file;

(3) Support level by level cross layer optimization compilation: select compile level, recommend level 1, use level 0 when level 1 compilation is not successful, do not recommend Level 2;

(4) Select Yolo according to the model requirements, and set the parameters in the purple box according to your own model. If Yolo is checked, there will be op-yolo operator in the final ops;

(5) Start compile;

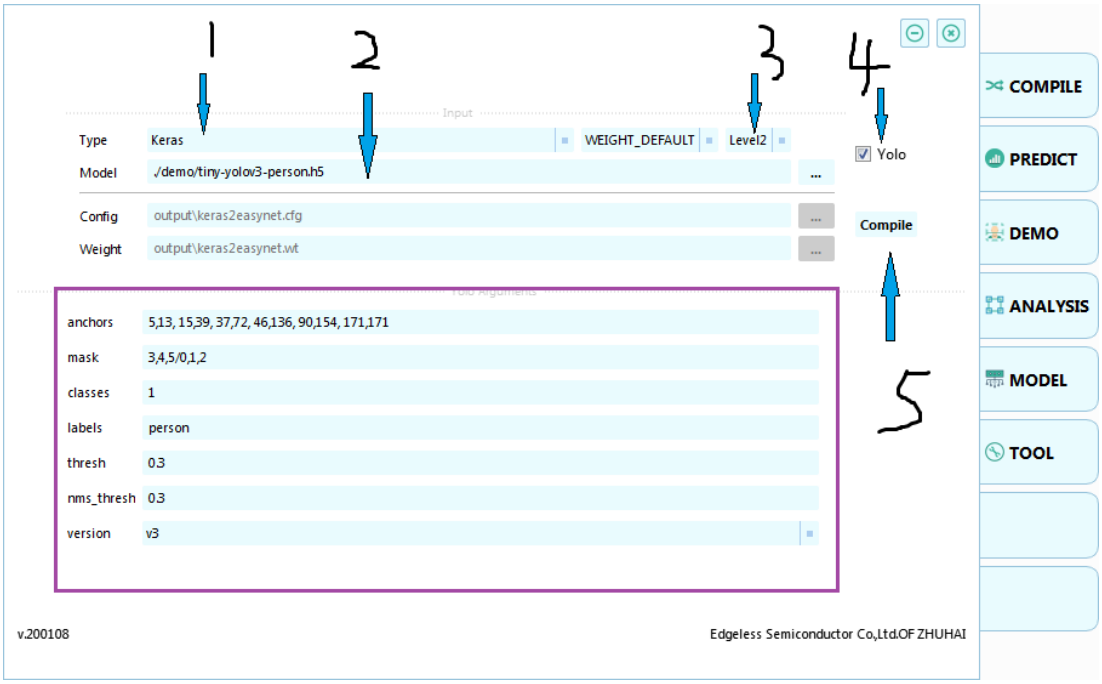


Figure 23 Compile GUI

(6) The compiler will first automatically convert the keras model file to

Easynet's .cfg and.wt

（7） Wait for compilation to complete prompt box ;

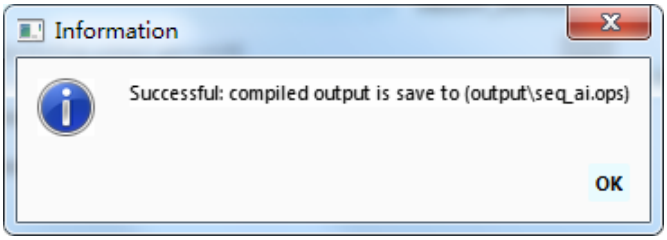


Figure 24 Compilation to succeed

The compiled file is seq_ai.ops, which is stored in the output directory

4.3.3 predict

Predict input

The input format is .in file supported by Easynet.

You can also select a picture file directly. The compiler will automatically convert the picture to an. In file

Predict output

The output result is the calculation result of each layer of neural network. By selecting different layers on the interface, you can switch to the corresponding output result.

Interface operation

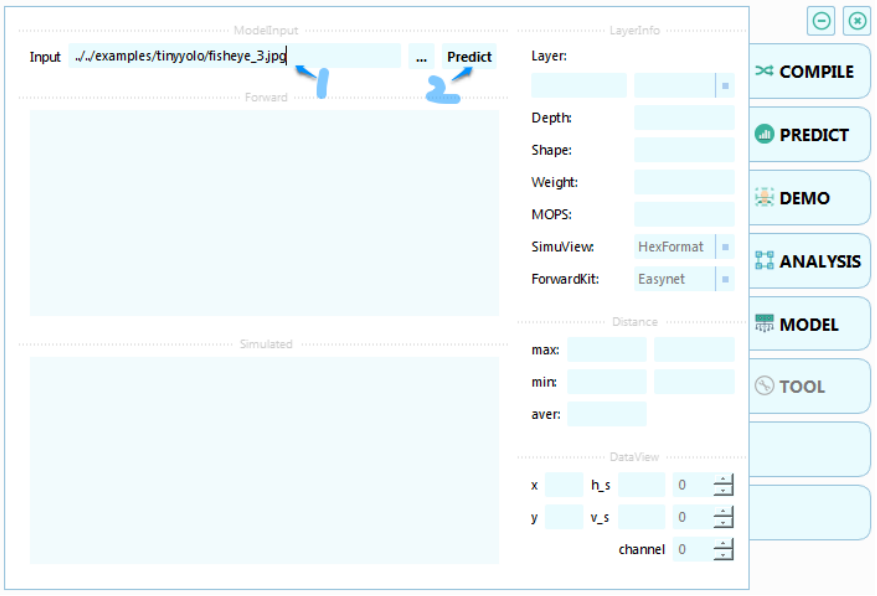


Figure 25 Predict

The output result is as follows

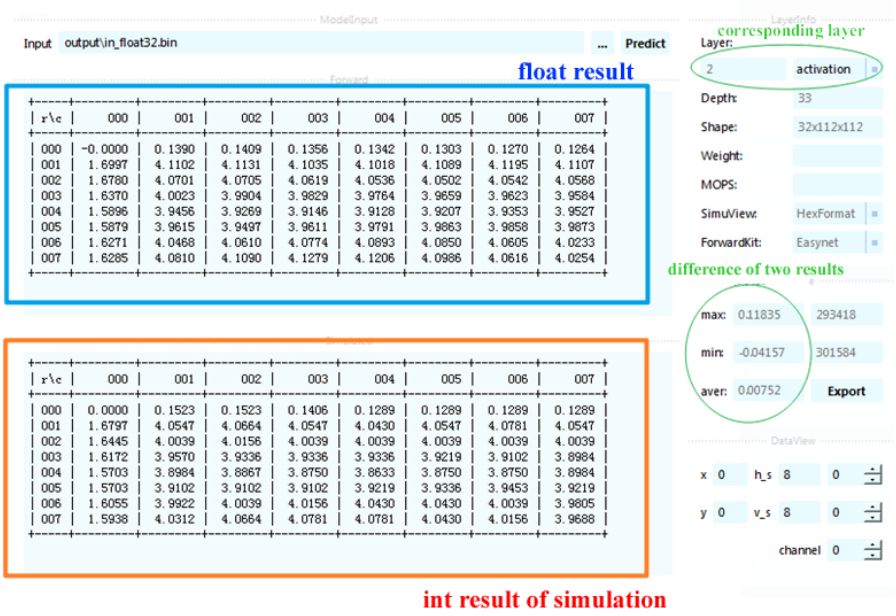


Figure 26 Predict result

- each layer can be selected for printout, layer 0 is selected herelayer

  Conv layer is used as data printing layer

- Depth represents the depth of the input network model

- Shape represents the output size of this layer

As shown in the figure above, on the PC side, the forward calculation result of floating-point number and the integer calculation result of KDP simulation (displayed in floating-point).

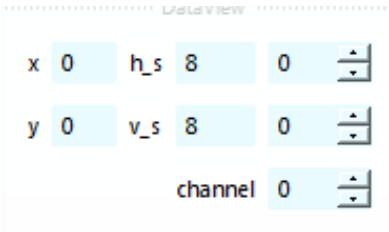


Figure 27 Detailed explanation results

- x indicates the starting position of the x-direction printed value in the output value of this channel，h_s indicates the number of lines printed. It is easy to display here and 8 is selected; ;
- y indicates the starting position of the y-direction printed value in the output value of this channel，v_s indicates the printed column, which is easy to display and 8 is selected ;
- channel indicates the number of channels output.

4.3.4 demo

The current demo only supports Yolo post-processing.

Demo input

The input has been determined during the compilation and prediction phase without additional input.

Demo output

Output the original image and corresponding result box and text result.

Interface operation

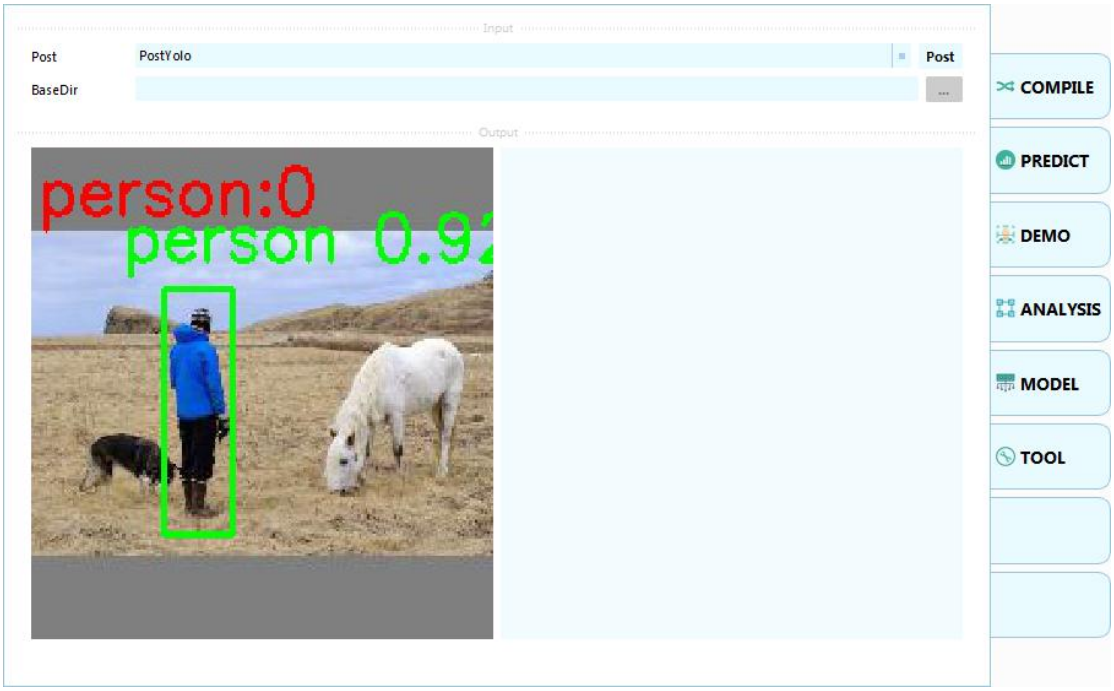


figure 28 demo interface

4.3.5 analysis

Analysis input

Input has been determined during compilation, no additional input is required.

Analysis input

The output is the model structure and specific parameters of the network model on the EAI chip.

Interface operation

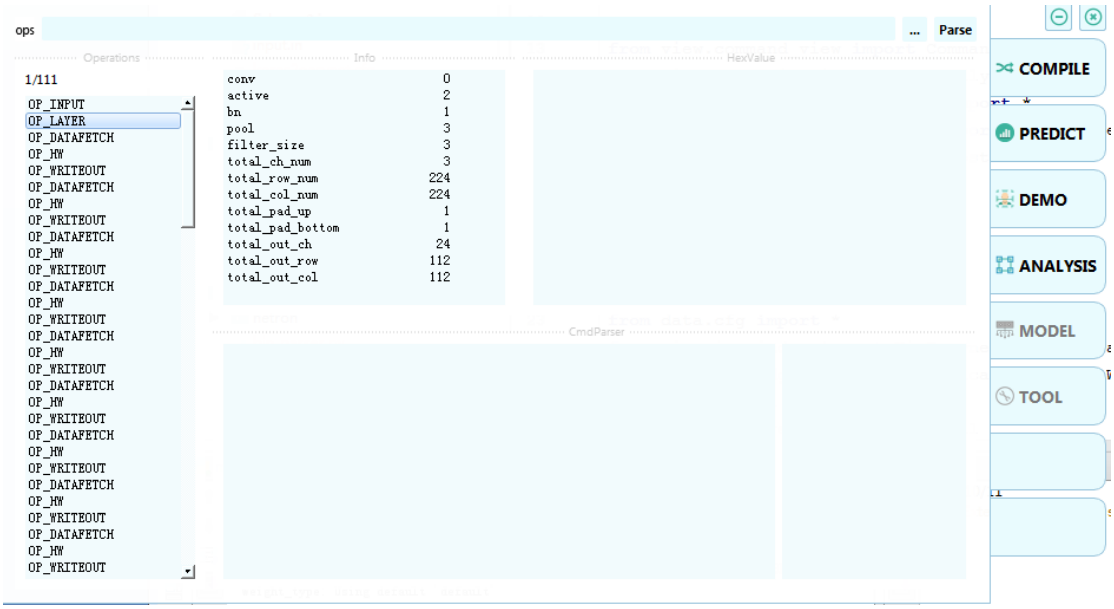


Figure 29 Analysis result interface

Click each level to see the specific operation or parameters of this level. When debugging the program, it can help improve the efficiency of the R & D personnel.

5. AI Demo User Guide

a) Prepare the hardware and software work platform

JLINK connect the development board after connecting the computer ,then power on

b) Demo burning

Copy config.ini in directory ugelis\tools\uGelisFlash_f6721b\images\ai to ugelis\tools\uGelisFlash_f6721b，open ugelisflash，click start to burn file.After burning, power off and power on the demo to start.

c) Keil debug

First, burn the weight to flash through uGelisflash (if you have done demo burning, you can skip this step, and the weight only needs to be burned once). Click flash_all.bat in ugelis\kelis_example\ai_example\ai directory to open ugelisflash, configure the config.ini, and click start to burn

Then open the project file in ugelis\kelis_example\ai_example\ai\mdk_80 directory, click compile, download, debug in sequence to enter debug mode, set breakpoint, and click SRART to start debugging program.

The following figure will appear during the first download. Select No:

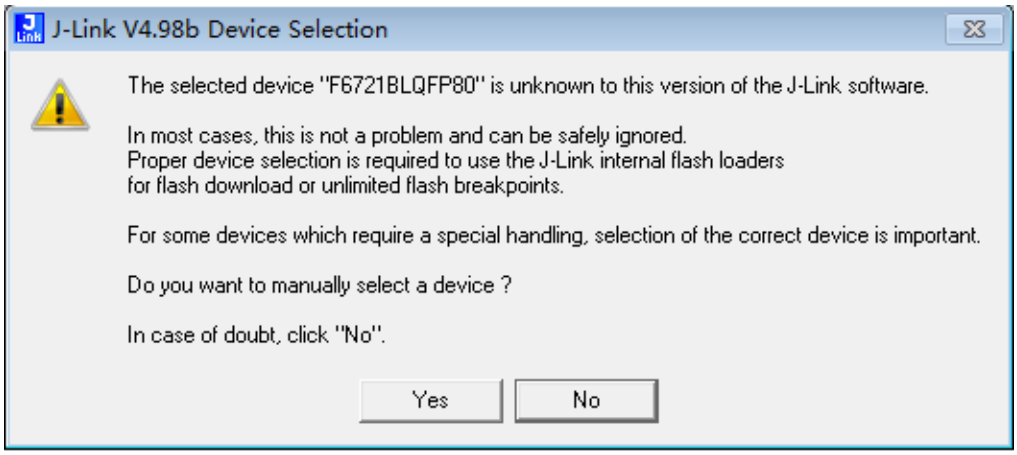


Figure 30 Download Block diagram

If you have modified the program and want to burn the demo to flash again, you can click CCC in the flash_app.batdirectory to open ugelisflash, configure the config.ini, and click start to burn.

6. Question and Answer

Q1 : ugliesflash Burning Filed

Answer :

a : Try several times more, power off and on quickly, then click start immediately

b : Confirm whether J-Link is connected correctly and open J-Link commande. The following figure shows the display of J-Link commande in different situations ;

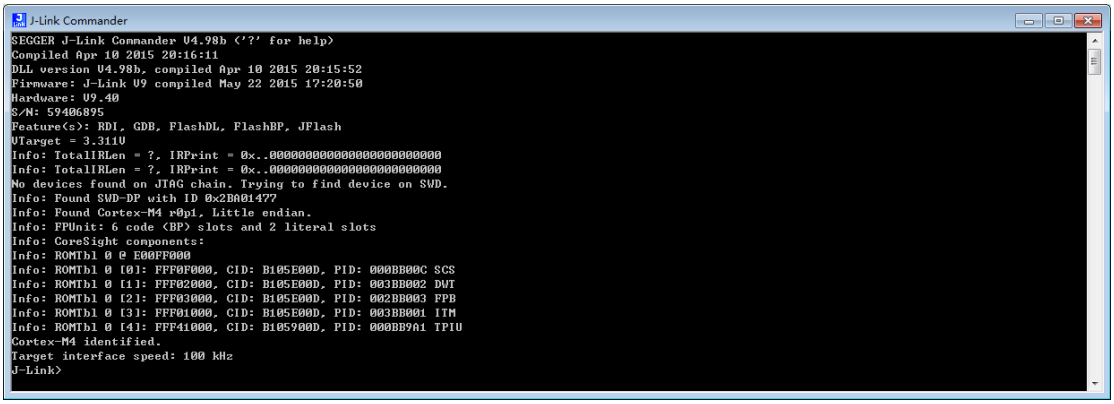


Figure 31 Proper connection

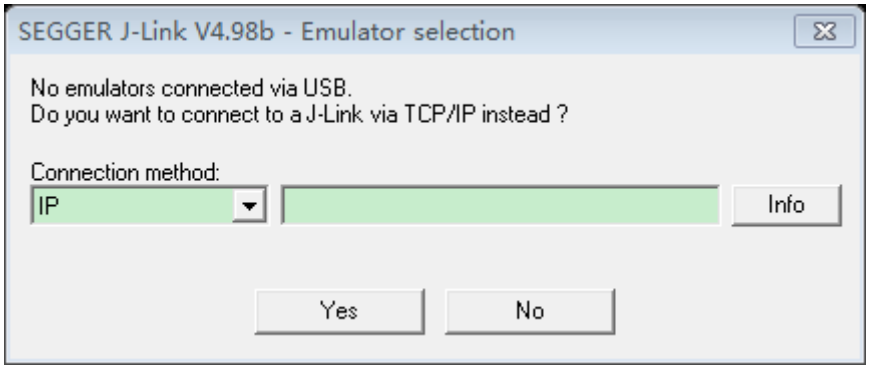


Figure 32 Error connection 1

Figure 31 shows that the connection with PC is error.

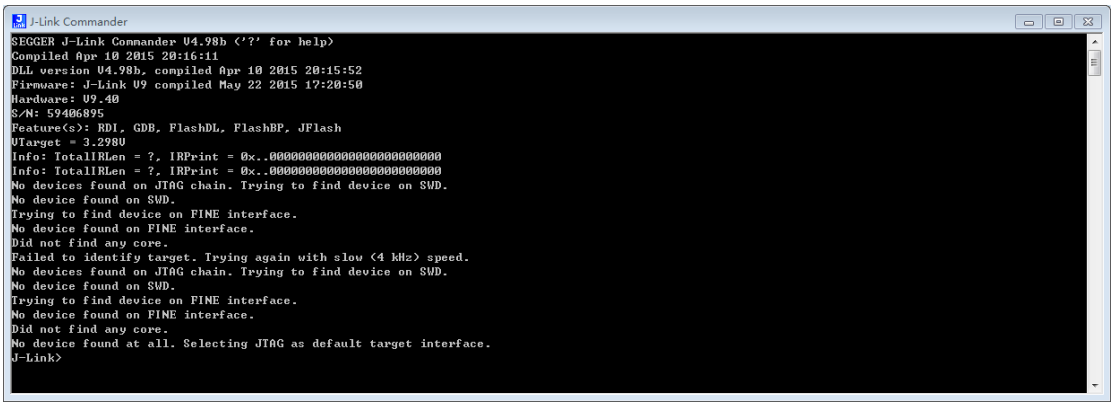


Figure 33 Error connection 2

Figure 32 shows that the connection with development board is not good

C ：Modify the path of the SDK. The path should be local, not Chinese. Make sure it is English.

Q2：Build or Download Error

Answer：Confirm whether the environment of the project is configured correctly.