Good Display' co.,

E-ink paper Demo Kit Manual

Edition 1.1.0

Company name: Dalian Good-Display Co.,

www. Good-display.com

No.17, Gonghua St, Shahekou District, Dalian City,Liaoning Province, China

T: 86 411 84619565 **DD:** +86 411 84526151 **E:** info@good-display.com

Summery

Good Display co., had invented 2.1”, 3.5”, 4.3” and 6” inches simplified E-paper demo kits in 2013. Normally this component is very easy for the ‘C code’ programmers to handle it. In additional, the demo kit’s very good compatibility with C code’s underlying system functions.

We strongly recommended you to use the Keil as IDE for compile, it’s simple to modify and observable. The whole procedures rely on to integrate of Debugger and Keil3 software.

From the hardware aspect, there’s a bit different between 2.1” and other types of kits. Even the Arm (MCU) we settled on this kit is the same, but for this specific resolution of use it able to forward images shuffle and contains more lanes of command. Another point, the 2.4” pin connector is not joint the share use with other FPC.

INDEX:

Summery 02

Structure 04

Settings 07

Compiler 12

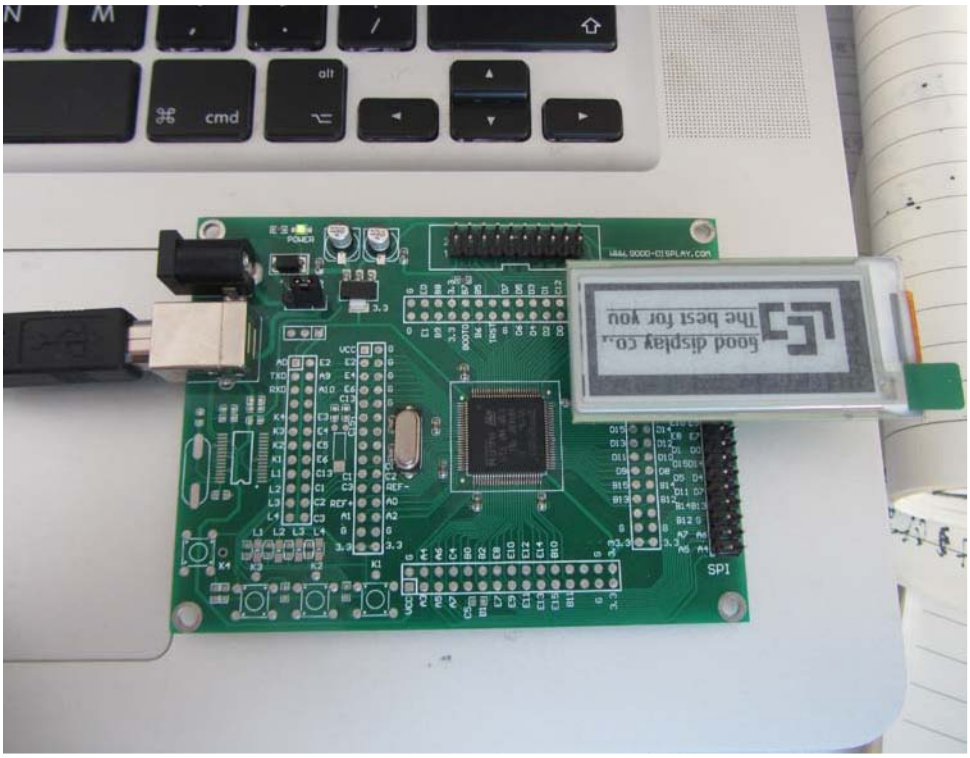
Download 13

Manuel Modify 14

Generate an image..... 14

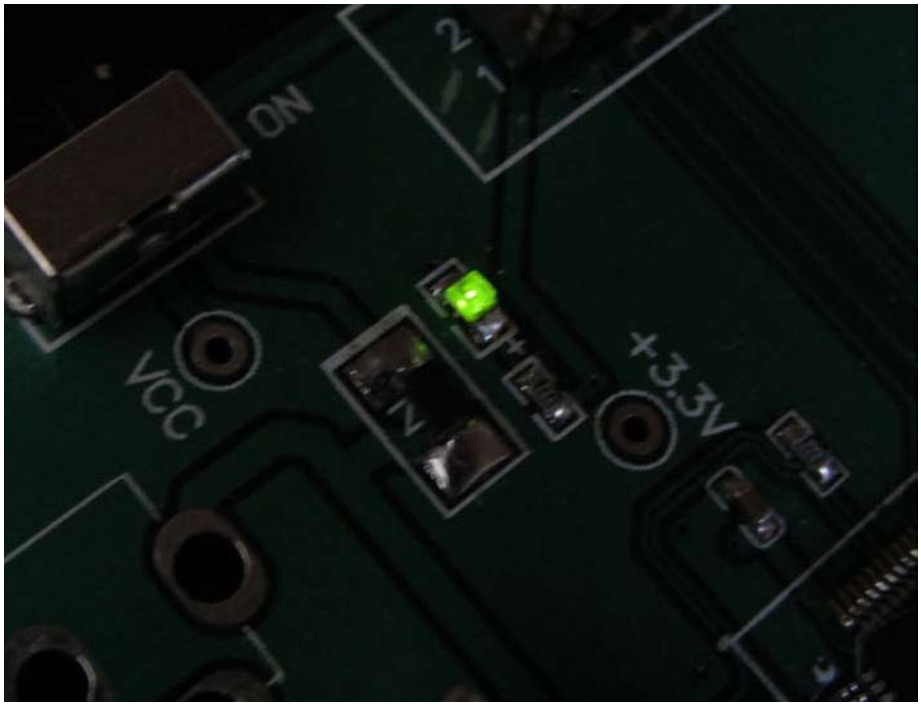
Structure :

For the representing use, you just have to install an e-paper on kit and connected to a PC thru an USB power cable then all will be done.



For every Demo kit, there got be a LED power light exists on board.

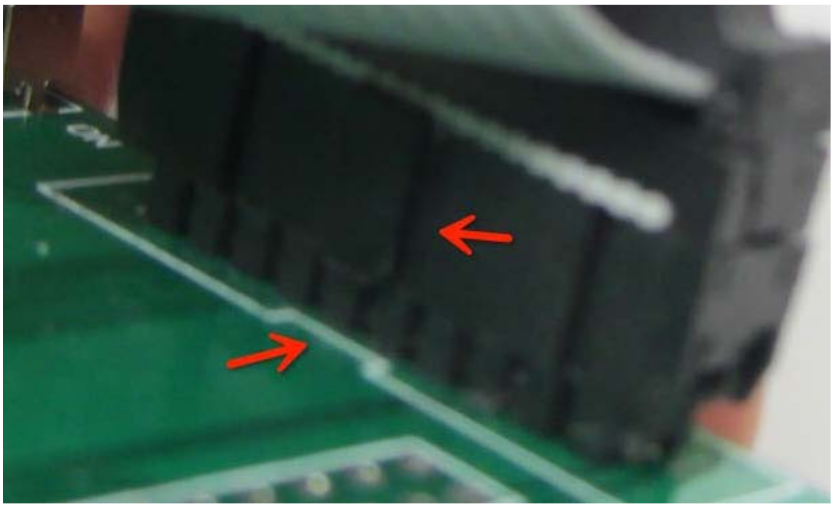
While at working status: Green light is appearing, Otherwise, light distinguished. You could optional choose an USB or AC power supplier to consumes power by switch the jumper wire.



Mentioned that, the USB connector on kit board couldn t use for data stream. Therefor it can t be download by this way. To download your programs from PC to the kit you may use a standard JTAG port debugger to implement.

1. Link the debugger by JTAG cable to the kit. (Image shows the right position)

2. Link the other side of debugger to PC by USB.



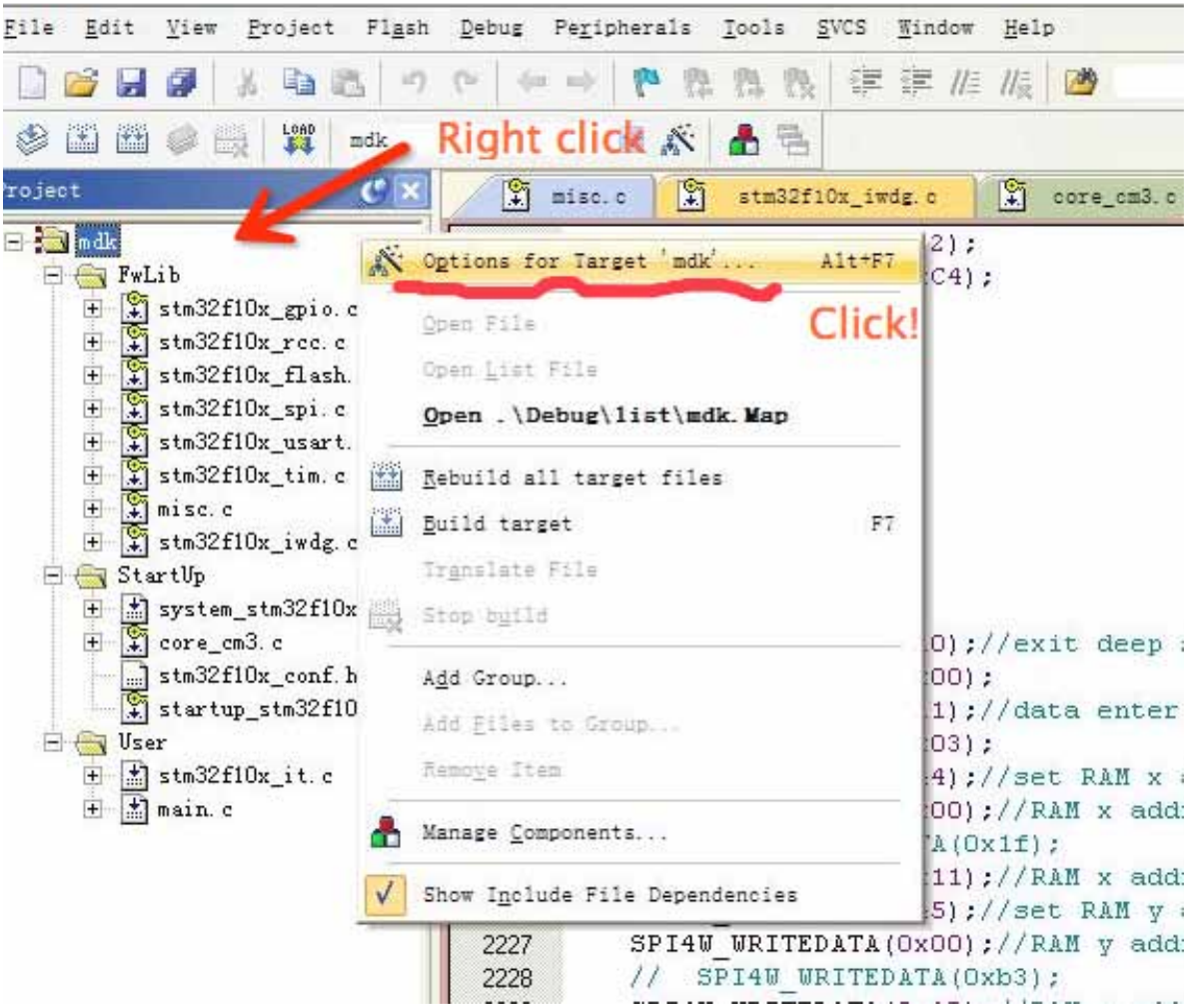
We knew the exactly how to link it up. Now you require a PC that runs a Keils3 (registered) or higher version. In addition you need to make sure all your stuffs are linked in. (5Vpower Demo Kit JTAG Debugger USB)

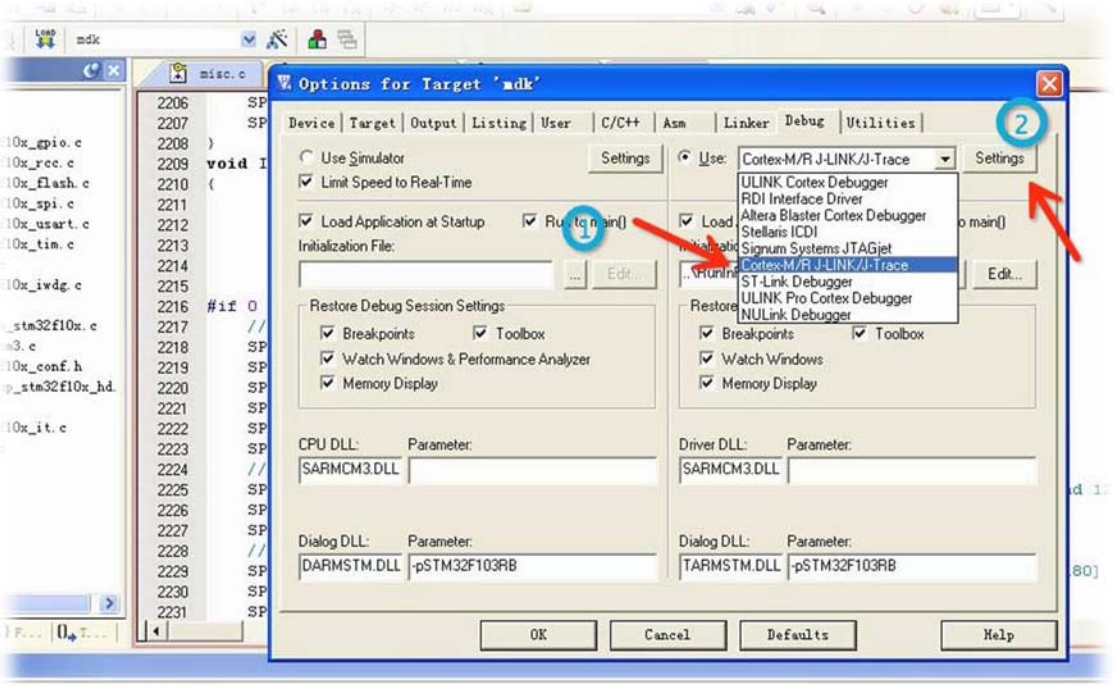
Settings :

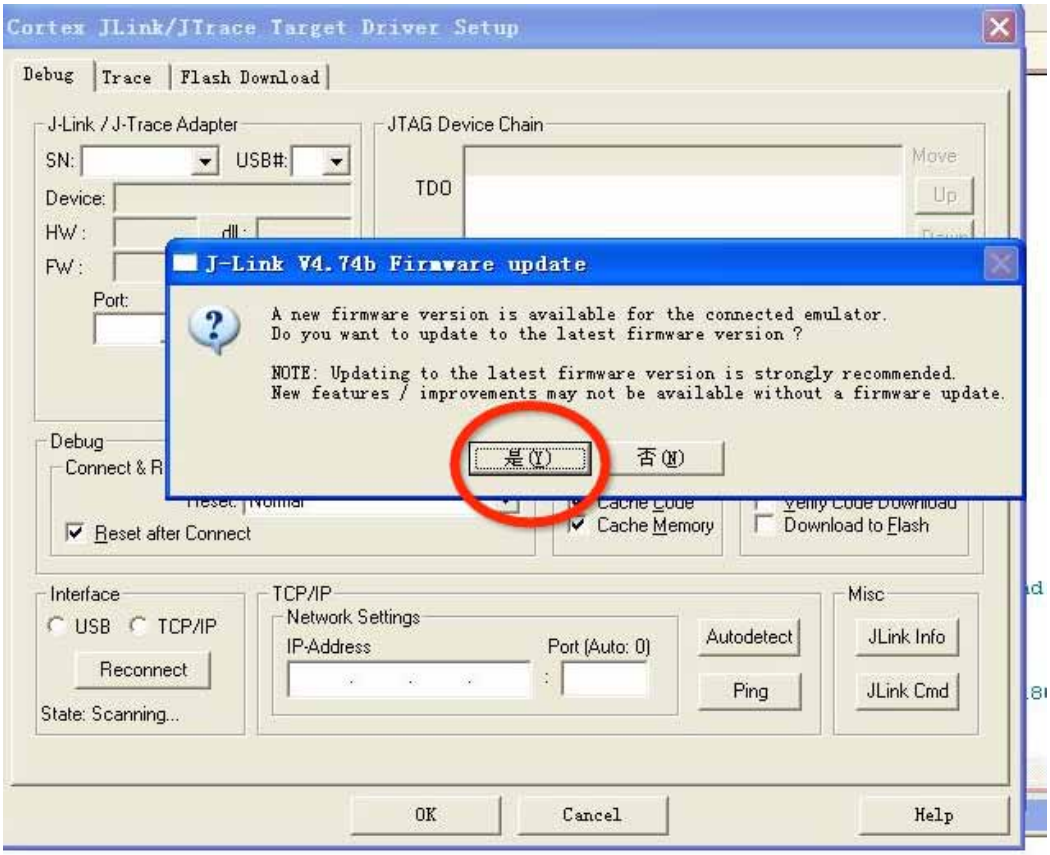
Double clicks your Keil icon on desktop or windows start to run it:

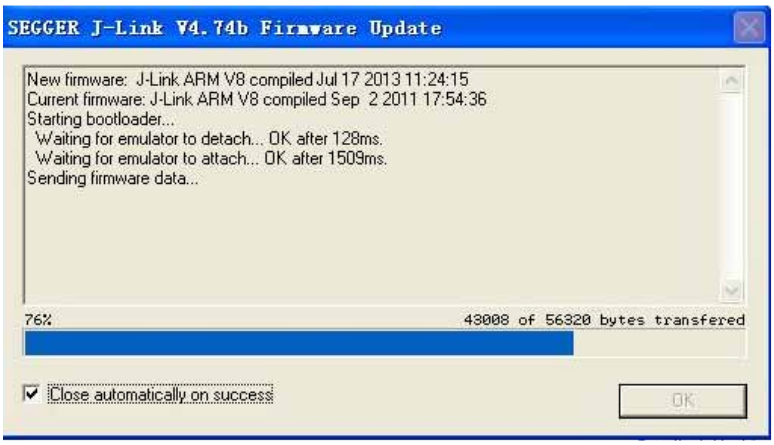
1. Open the project > Right click the project root DIR > Select Options for Target > Click Debug Tab, Find the J-link JTAG debugger and press the setting.

   (Procedures listed on pictures)




Options for Target 'mdk'
Device
Target
Output
Listing
User
C/C++
Asm
Linker
Debug
Utilities
Use Simulator
Limit Speed to Real-Time
Settings
Use:
Cortex-M/R J-LINK/J-Trace
ULINK Cortex Debugger
RDI Interface Driver
Altera Blaster Cortex Debugger
Stellaris ICDI
Signum Systems JTAGjet
Cortex-M/R J-LINK/J-Trace
ST-Link Debugger
ULINK Pro Cortex Debugger
NULink Debugger
Load Application at Startup
Initialization File:
Edit...
Restore Debug Session Settings
Breakpoints
Toolbox
Watch Windows & Performance Analyzer
Watch Windows
Memory Display
CPU DLL:
Parameter:
SARMCM3.DLL
Driver DLL:
DARMSTM.DLL
-pSTM32F103RB
TARMSTM.DLL
Dialog DLL:
OK
Cancel
Defaults
Help
1
2


Cortex JLink/JTrace Target Driver Setup
Debug
Trace
Flash Download
J-Link / J-Trace Adapter
SN:
USB#:
Device:
HW:
FW:
Port:
JTAG Device Chain
TDO
Move
Up
J-Link V4.74b Firmware update
A new firmware version is available for the connected emulator.
Do you want to update to the latest firmware version ?
NOTE: Updating to the latest firmware version is strongly recommended.
New features / improvements may not be available without a firmware update.
是(Y)
否(N)
Debug
Reset after Connect
Cache Code
Cache Memory
Verify Code Download
Download to Flash
Interface
USB
TCP/IP
Reconnect
State: Scanning...
TCP/IP
Network Settings
IP-Address
Port (Auto: 0)
Autodetect
Ping
Misc
JLink Info
JLink Cmd
OK
Cancel
Help

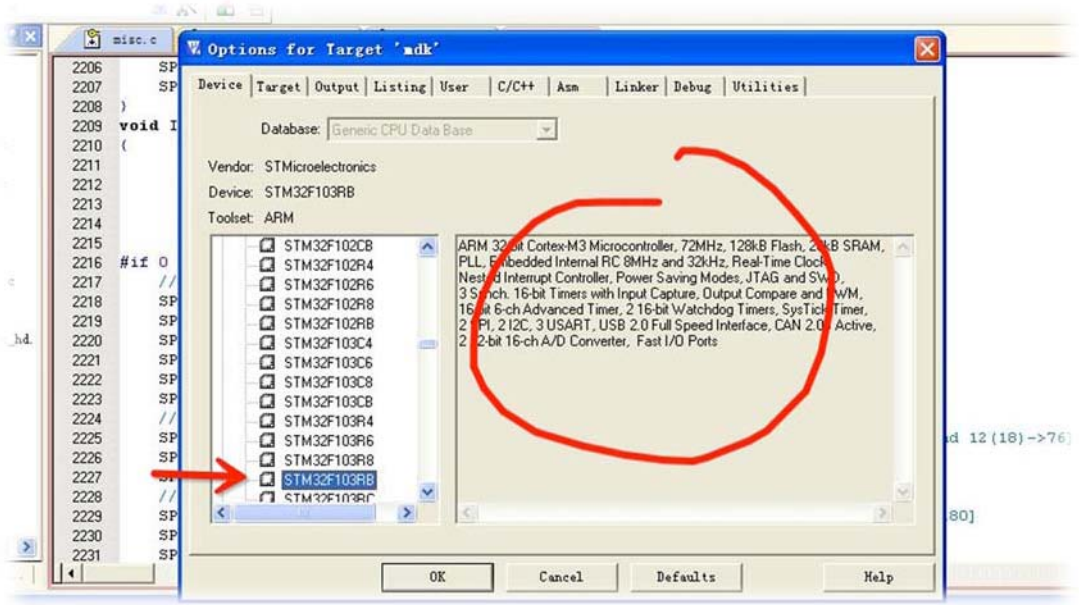


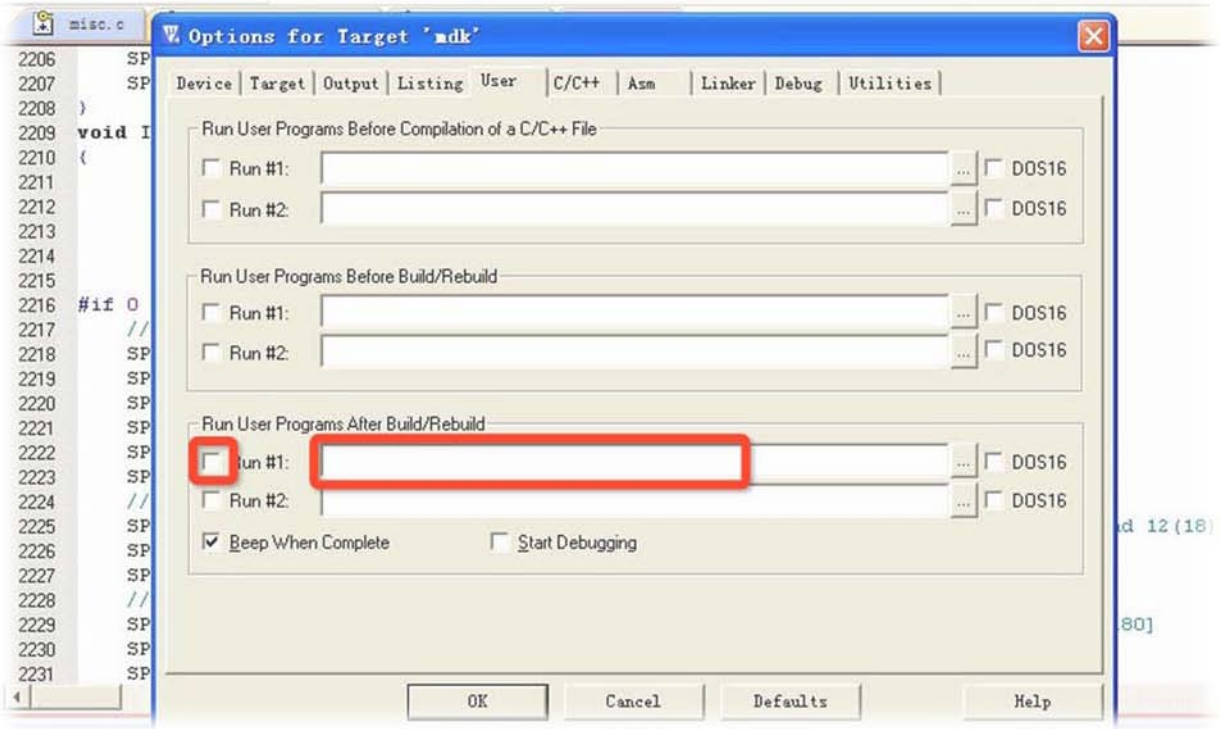
After UPLOAD your Debugger to Keil press OK to save it.

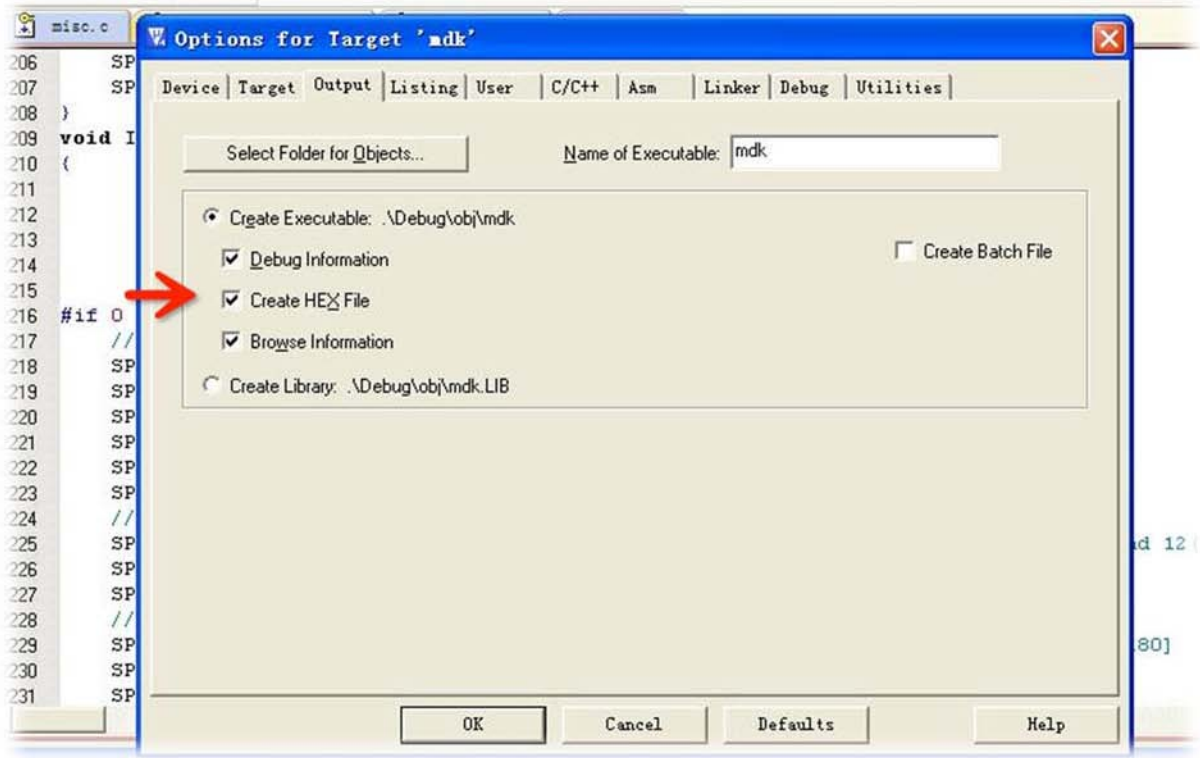
2. Click the Utilities tab select J-Link JTAG and do the similar steps copy previous one, also settings.

WARNING : When you done your Debug and Utilities Settings, remember to check the FlashDownload tab in those settings. If it does come up with STM32F10x 128K, please manually added it.

3. Find our MCU module in Device Tab and press [OK] to save, After that click User Tab and deselect Run user pro keep the tick box and text field empty, finally go to Output Tab and tick the create a HEX file after compile .

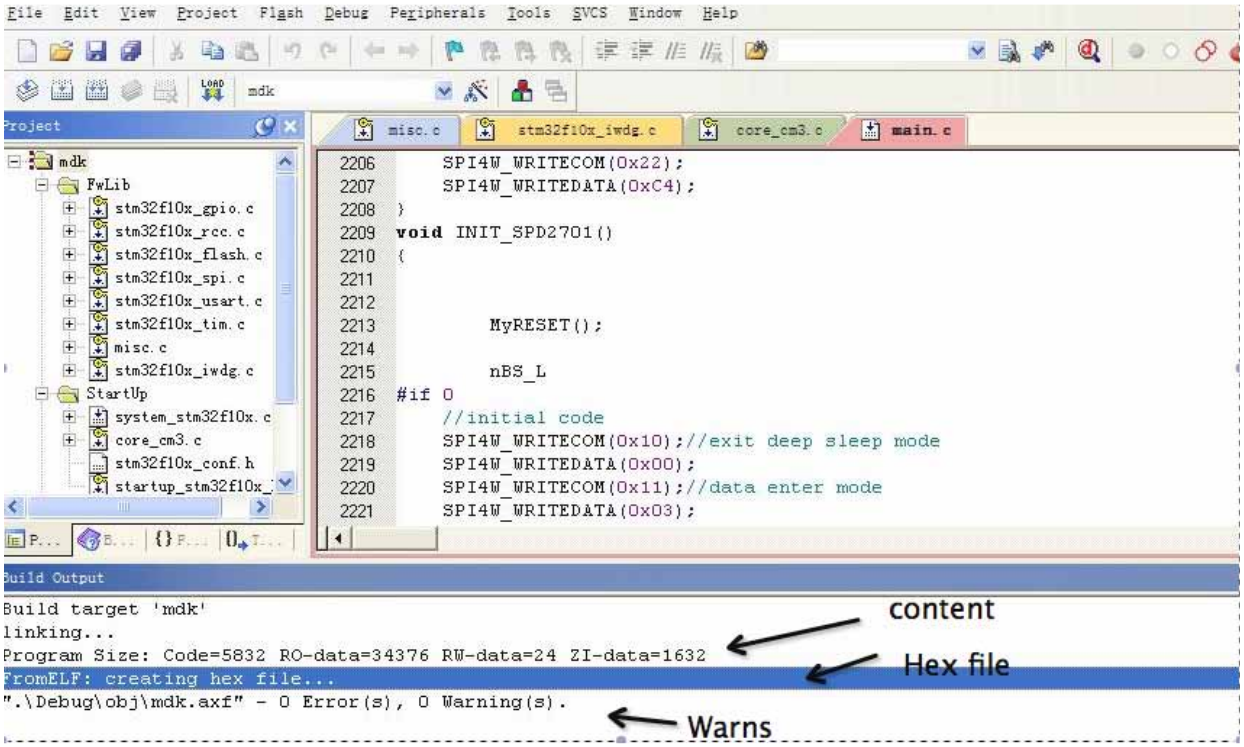



misc.c
Options for Target 'mdk'
Device
Target
Output
Listing
User
C/C++
Asm
Linker
Debug
Utilities
Run User Programs Before Compilation of a C/C++ File
Run #1:
Run #2:
DOS16
Run User Programs Before Build/Rebuild
Run User Programs After Build/Rebuild
Beep When Complete
Start Debugging
OK
Cancel
Defaults
Help


misc.c
Options for Target 'mdk'
Device
Target
Output
Listing
User
C/C++
Asm
Linker
Debug
Utilities
Select Folder for Objects...
Name of Executable:
mdk
Create Executable: .\Debug\obj\mdk
Debug Information
Create HEX File
Browse Information
Create Batch File
Create Library: .\Debug\obj\mdk.LIB
OK
Cancel
Defaults
Help

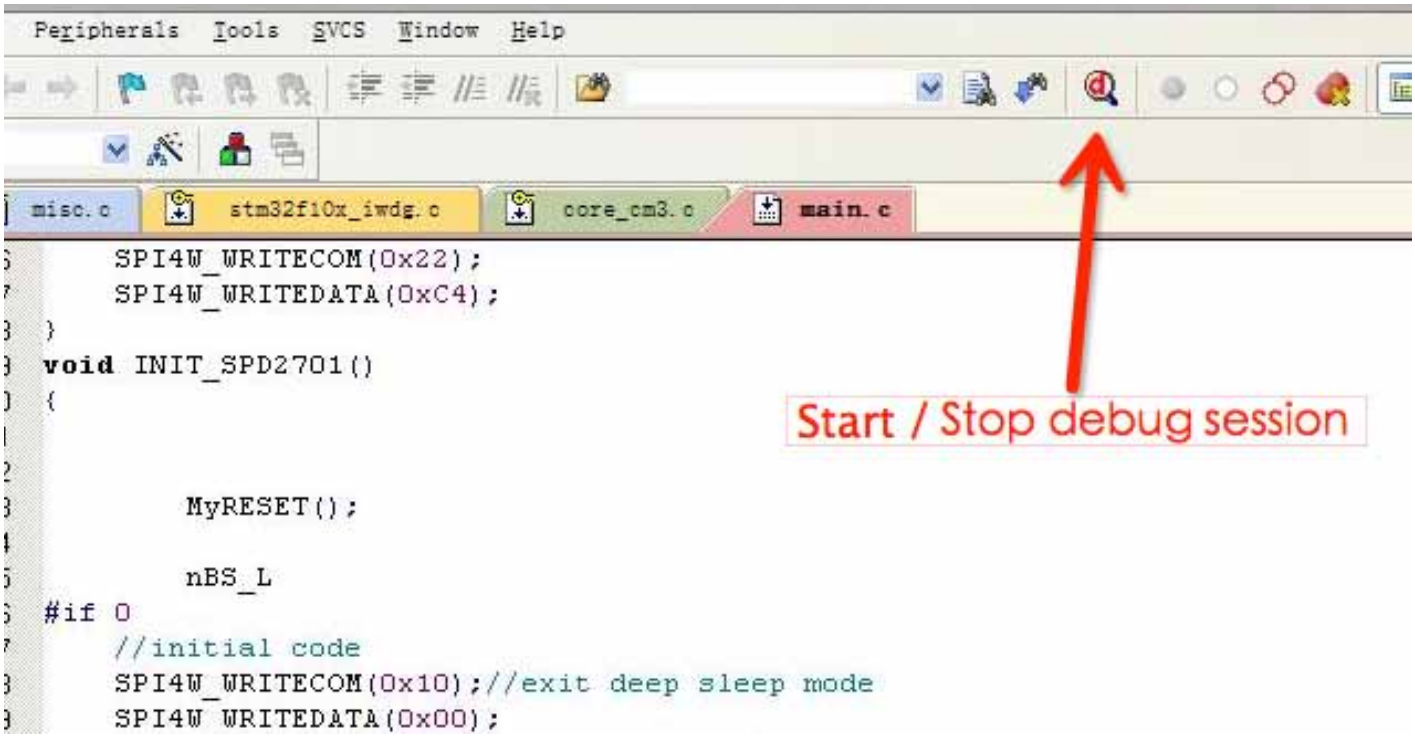
Compile：

To compile your code just click on the build button on left top of window. You must compile before debug. The Build Output window shows all detail you may interest to know. You also can observe if there s any error or interrupt existed and mark the lane it is. Careful check that the HEX file is successfully created for your next step.



Download :

Moans that again please check your Debugger and Demo kit correctly connected and powered up. The debugger and your kit s LED bulbs should be lightning or flashing. Click the Start/Stop Debugger session button to download it to your MCU and enter to debug mod.



The button is on your top right.

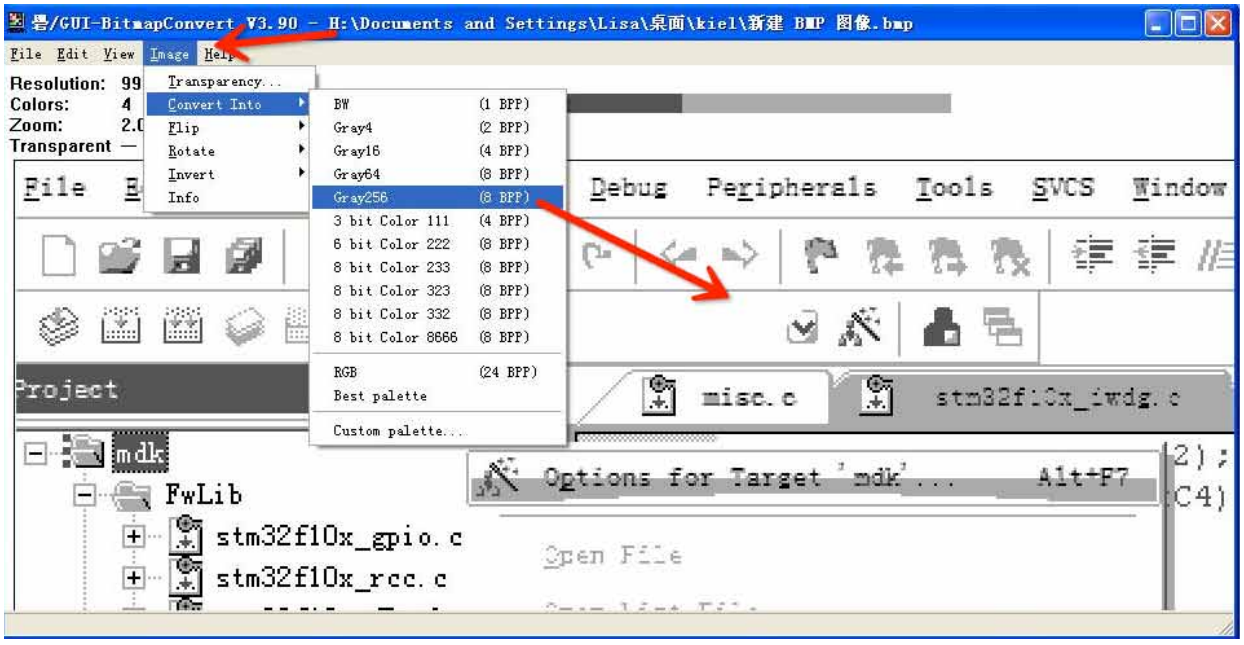
Modify :

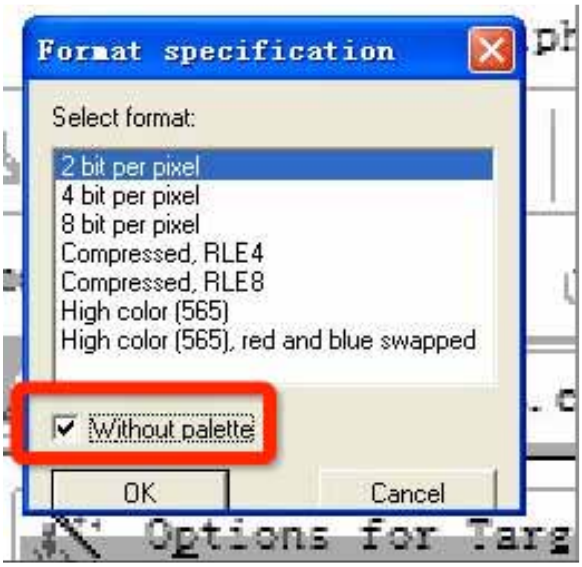
You may set the temporary pause by double clicks a particular lane of source and observe it. Use step button can move to your the next pause. When you use this to perform your work to check the reactions please always download it to ARM after modify. That's because of it's more reliable to keep the ARM had your latest update of the program.

Generate an image static number table [] :

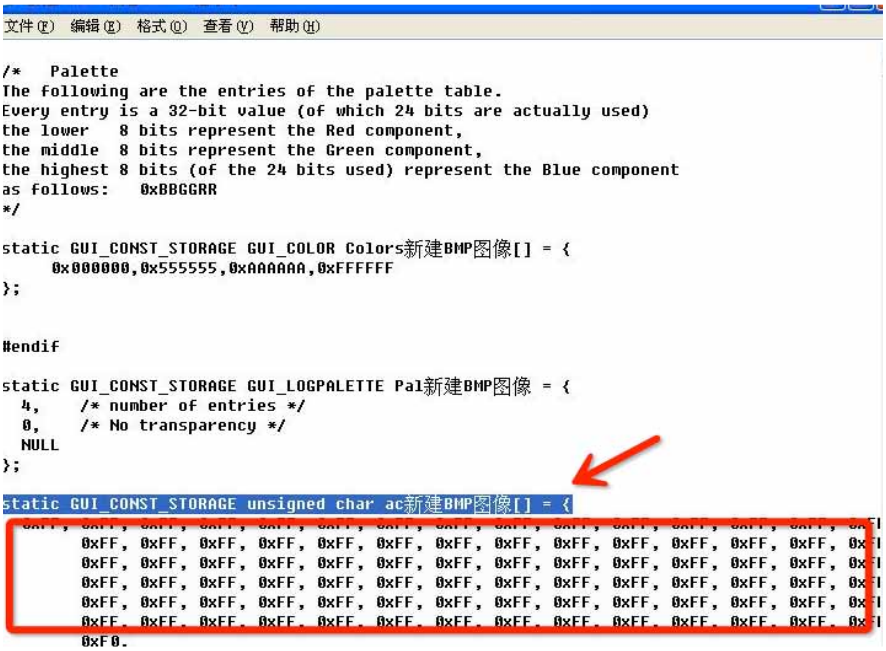
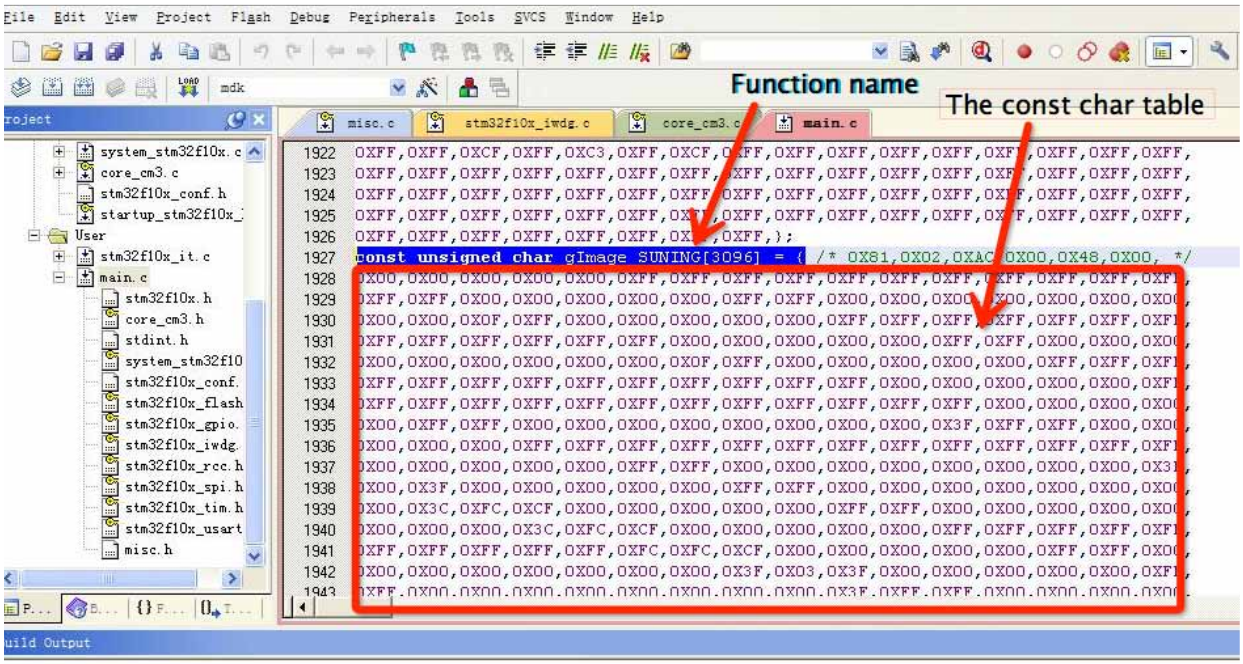
- Technically it's perfectly fine for any software you could download from the Internet that make the same purpose. However if you don't have any idea of what it is I would recommend you to use the BitmapCovert to implement. You need to insert a [*.bmp] format image to the window and drill the unattainable color off. So make sure the resolution is right, the gray scale is attainable by check the datasheet. Check the frame is portrait or landscape

then flip turn the image in window to correct. When everything is right, just click the file in toolbar, and then click save as [.c] format txt file. Remember to fill the [without template] tick box, that makes sure you won t see too much useless code in that file. Replace the const number [] of our program with the new one we just generated. Now an image is replaced.

- 



During you save it from BitmapCovert to local disk, the format you need to select in this widow is the numbers of bit will applied per pixel. So you better to check it with a proper datasheet.



The red square marked the hex numbers in text that you looking for to copy. And these numbers below need to be replaces with.