Getting Started with mbed and FRDM-KL25Z

This tutorial guides you through mbed-enabled Freescale Freedom FRDM-KL25Z board, and getting started with the mbed online tools platform. An mbed-enabled FRDM-KL25Z board gives you:

- USB drag and drop programming
- USB Virtual COM port for serial terminal
- CMSIS-DAP interface for programming and debugging from offline tools
- Free access to the mbed online compiler, mbed C/C++ SDK, and developer community

It is assumed from now on that your FRDM-KL25Z is mbed-enabled.

It is really simple to setup and program, and it is free, so let us get started!

1. Connect your mbed-enabled FRDM-KL25Z to a PC

Use the USB lead to connect your KL25Z to a PC, using the USB connector labelled *OpenSDA*.

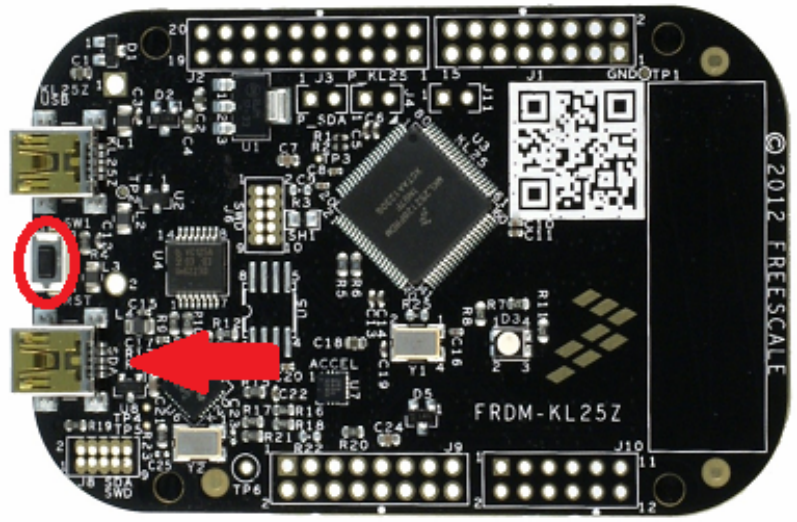

Fig. 1. **FRDM-KL25Z board.**
Read arrow indicates OpenSDA USB connector
and red circle the reset button

The status light will come on, indicating it has power. After a few seconds of activity, the PC will recognise the mbed Microcontroller as a standard USB drive. Also your PC will detect that a driver is needed.

Choose the second option of Advanced installation of driver as shown.

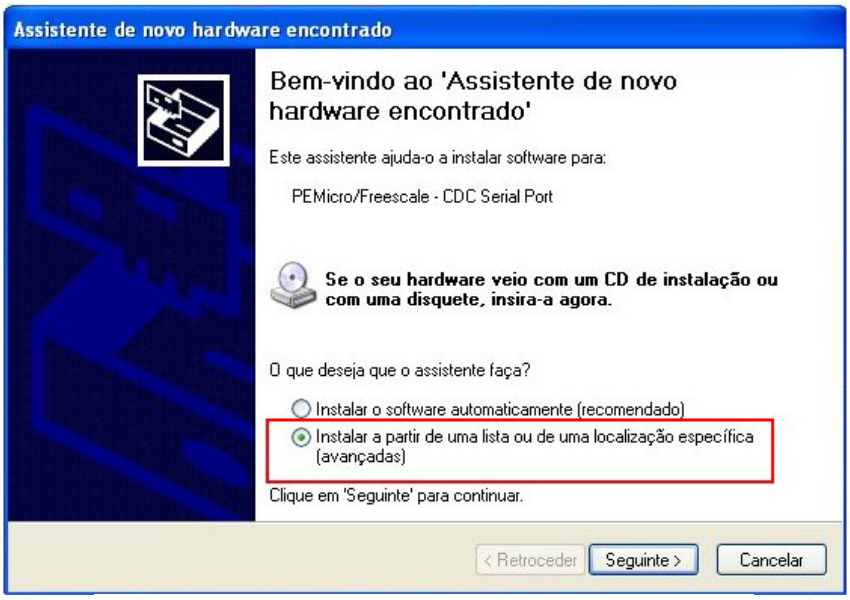

Fig. 2. **Windows' requirement for driver of OpenSDA**

Click *Next* and choose to find the driver (by this time most certainly FRDM-KL25Z is recognised as a drive)

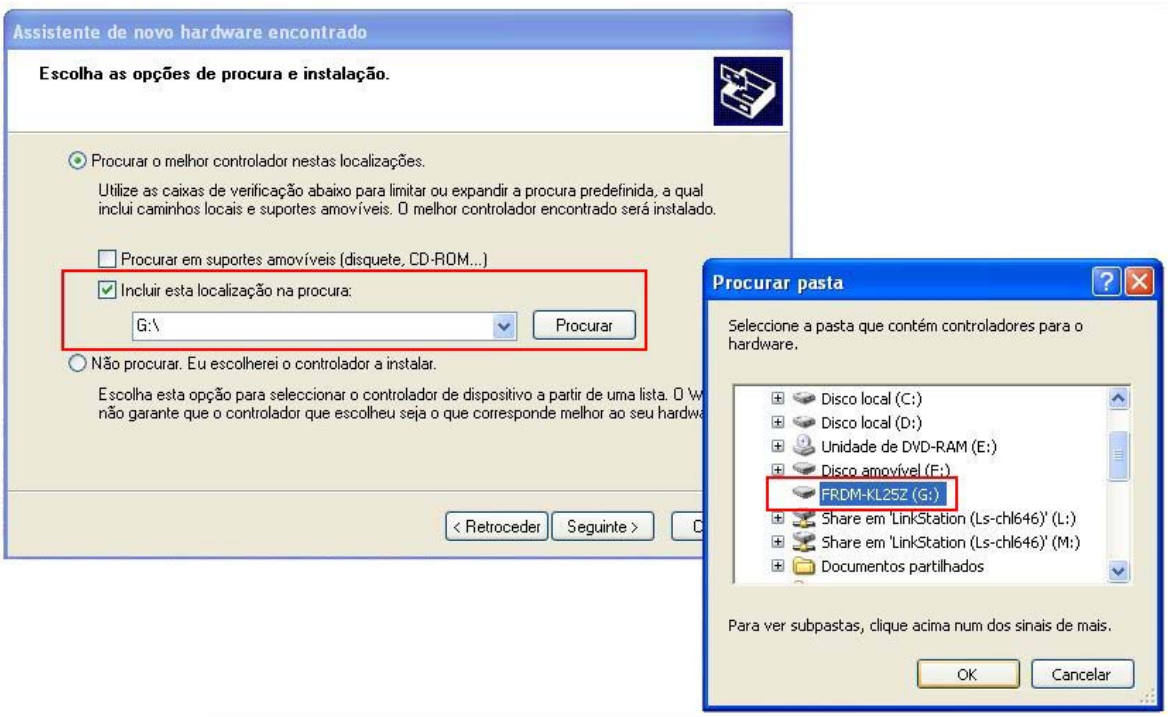

Fig. 3. **FRDM-KL25Z as drive (contains drivers of OpenSDA**

The driver for *OpenSDA* of FRDM-KL25Z is included in its flash memory.

Troubleshooting
If the USB Flash drive does not appear:

- Ensure the USB connector is plugged in fully. The Microcontroller can power up even when only partially connected, but it won't actually work!
- Warning; in newer versions of windows, loading the mass-storage disk driver for the first time can take a long time...
- Windows tends to be lazy, so you'll often have to hit F5 when you're in an explorer window to see new drives.

If any of this doesn't work as described for you, please email support@mbed.co.uk

Install the driver, following all the normal steps of installation of drivers.

NOTE: You may be warned that the driver is not officially recognised to be compatible with Windows. You will need to accept it as trustworthy.

Fig. 4. **Step in installation of driver**

2. Visit the mbed website to get logged in and to compile

The URL for mbed website is:

http://mbed.org

If you do not have an mbed account, choose "Signup", and create your mbed Account. Otherwise, log in with your normal username and password.

Fig. 5. **Login or signup on top left corner of Home page**

This will give you access to the website, tools, libraries and documentation.

On top left of *Home* page you will see the menu where you will choose your platform by clicking on *Platforms* button.

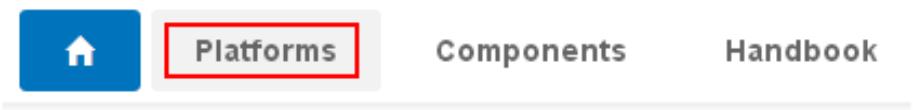

Fig. 6. **Red square indicates Platforms option of menu**

Now choose the right *Platform* that for this case is *FRDM-KL25Z*, by clicking on this option.

Fig. 7. **Blue square indicates Platform option to be chosen for FRDM-KL25Z**

Having chosen the Platform, you will now add your FRDM-KL25Z board to your Compiler.

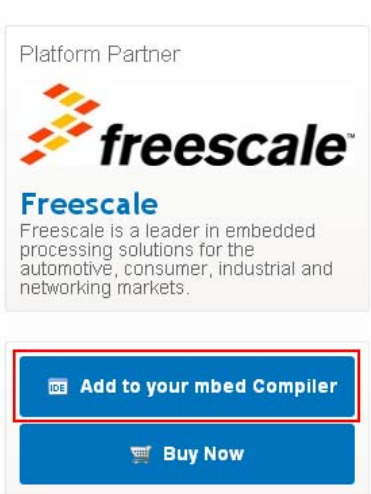

Fig. 8. **Red square shows the option to be chosen for adding the board**

After adding the board to the Compiler you are ready to compile your first code.

3. Open the mbed Compiler

Open the online Compiler clicking the *Compile*r button in the site menu (top-right of the page). This will open the Compiler in a new tab or window.

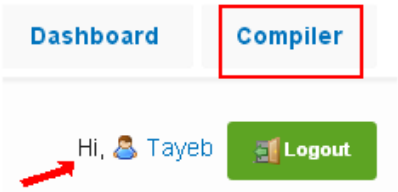

Fig. 9. **Red square shows the option to be chosen, and red arrow shows already logged in**

NOTE: It will be your chosen name that will appear as logged in, obviously!

4. Create a New Program

To create a new program in your personal *Program Workspace*:

- Right-click on *My Programs*, and select *New Program...*

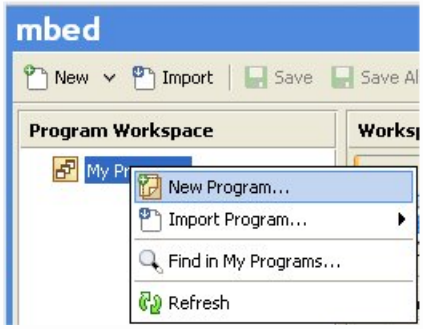

Fig. 10. **Adding "New Program…" in Program Workshpace**

- Enter the name of the new program (e.g. "HelloWorld"), and click *OK*

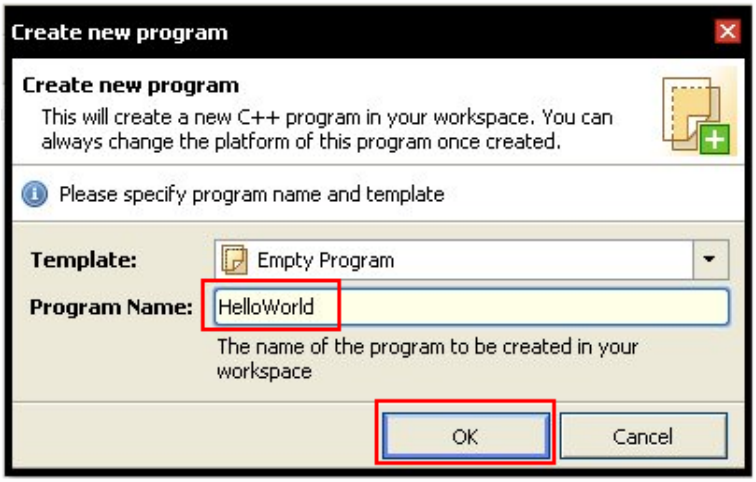

Fig. 11. **Pop up window to create new program name**

Your new program folder will be created under *My Programs*.

5. Adding to program a source code

- Add a new file *main.cpp* by right clicking on *HelloWorld* program name.

Fig. 12. **Adding "New File…" to "HelloWorld" program name**

- Enter the name of the new file *main.cpp* and click *OK:*

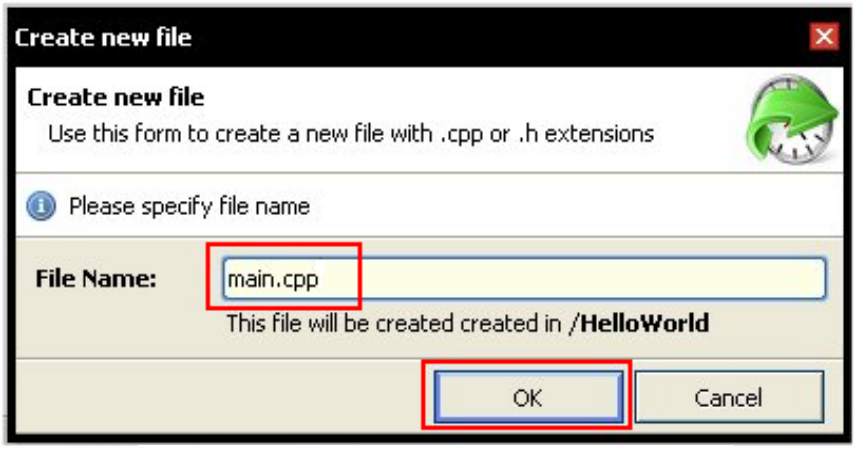

Fig. 13. **Pop up window to create new file name**

Click on the *main.cpp* file in your new program, to open it in the file editor window. This is the main source code file in your program, and by default it is empty. Write the following program:

```
:#include "mbed.h"

DigitalOut myled(LED1);

int main() {
    while(1) {
        myled = 1;
        wait(0.2);
        myled = 0;
        wait(0.2);
    }
}
```

The result will be as shown in following figure:

```
main.cpp
1  #include "mbed.h"
2
3  DigitalOut myled(LED1);
4
5  int main() {
6      while(1) {
7          myled = 1;
8          wait(0.2);
9          myled = 0;
10         wait(0.2);
11     }
12 }
```

Fig. 14. **Code in *main.cpp* file**

- Adding *mbed* library to the program folder

The other item in the program folder is the *mbed* library - this provides all the useful functions to start up and control the mbed Microcontroller, such as the DigitalOut interface used in this example. Right-click on *HelloWorld*, choose *Import Library…* and *From Import Wizard…*

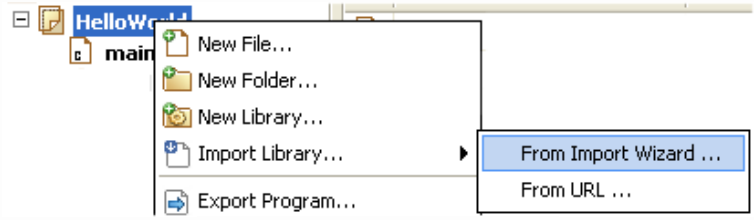

Fig. 15. **Import library wizard**

Now choose *mbed* library and then click on *Import* button on the top right-side.

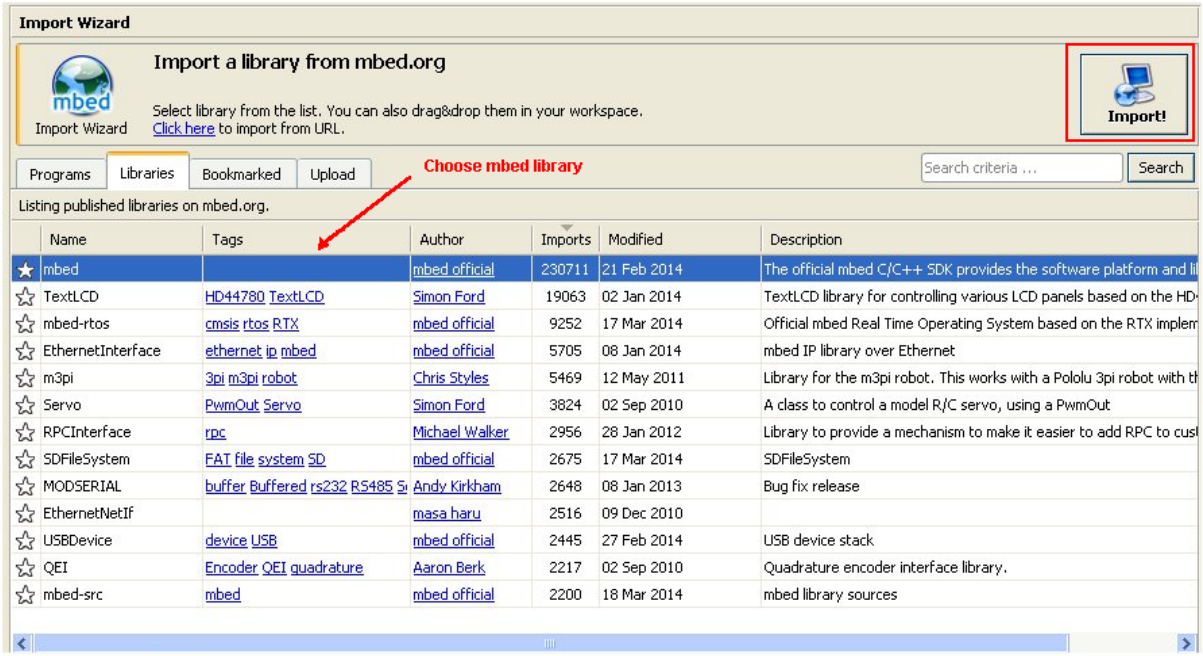

Fig. 16. **Import mbed library**

Program Workspace will show that mbed library has been added to *HelloWorld* program:

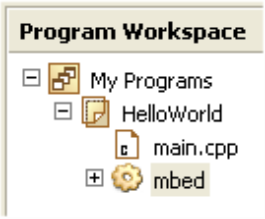

Fig. 17. **Program Workspace with HelloWorld and main.cpp file and mbed library**

In our *main.cpp* file LED1 stands for red colour LED on the FRDM-KL25Z. Before proceeding any further, make changes to second line at the end of *DigitalOut* command, substituting LED1 with LED3. On the right-side of editor you will see the Enumerator telling you that LED3 corresponds to blue LED on FRDM-KL25 RGB LED.

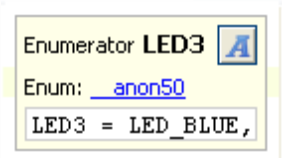

Fig. 18. **LED3 enumerator indicating what it corresponds to in Compiler**

Our *main.cpp* file's content will be now:

```
:#include "mbed.h"

DigitalOut myled(LED3);

int main() {
    while(1) {
        myled = 1;
        wait(0.2);
        myled = 0;
        wait(0.2);
    }
}
```

Before proceeding save *main.cpp* file by right-clicking on it, and choosing the option *Save…*

4. Compile and Download the Program

To compile the program, click the *Compile* button in the top toolbar.

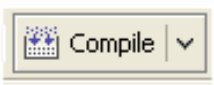

Fig. 19. **Compile button**

This will compile all the program source code files within the program folder to create a binary program.

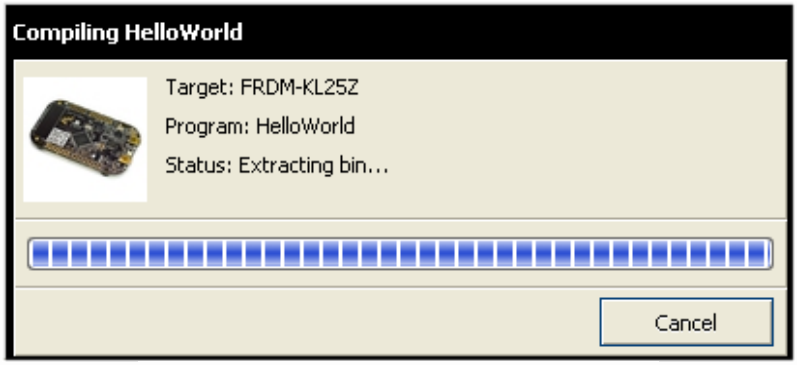

Fig. 20. **Compiling "HelloWorld" program**

- After a successful compile, you will get a *Success!* message in the compiler output and the download dialog will pop up to Save the binary program, or it will be automatically downloaded to your default *Downloads* folder.

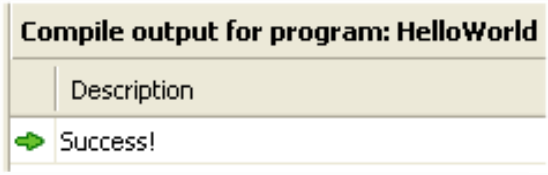

Fig. 21. **Compiler output for "HelloWorld"**

- Save it to the location of your PC in case of download dialog pop up.

NOTE: In Goggle's *Chrome* browser, the complied file will be downloaded to *Downloads* folder

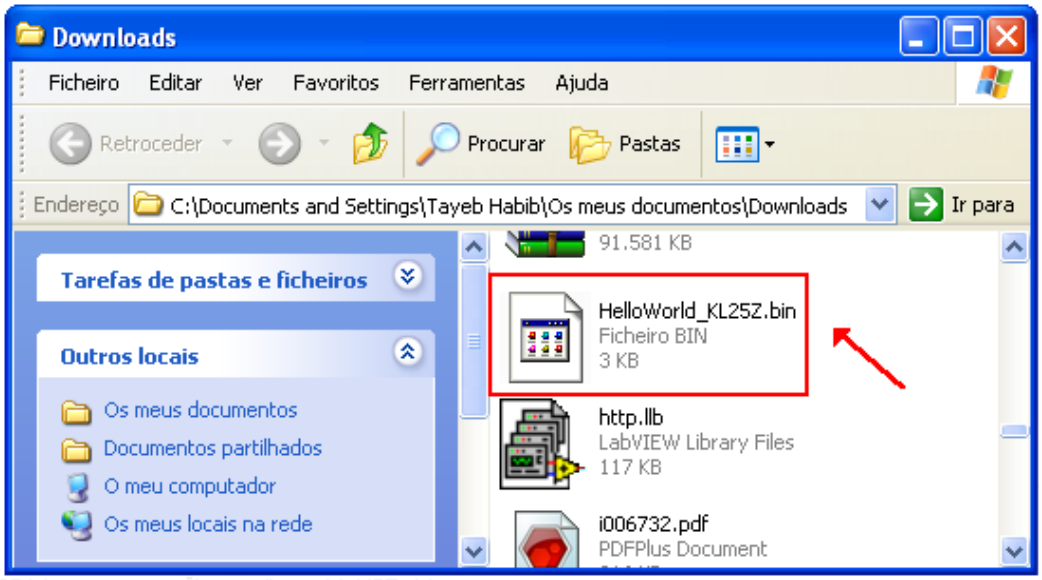

Fig. 22. **Compiler output for "HelloWorld" in Downloads folder of PC**

NOTE: If there are errors, they will show up in the "Compiler Output" window, and will need to be fixed!

- FRDM-KL25Z behaves a drive, so you find it in *My Computer* of your PC.

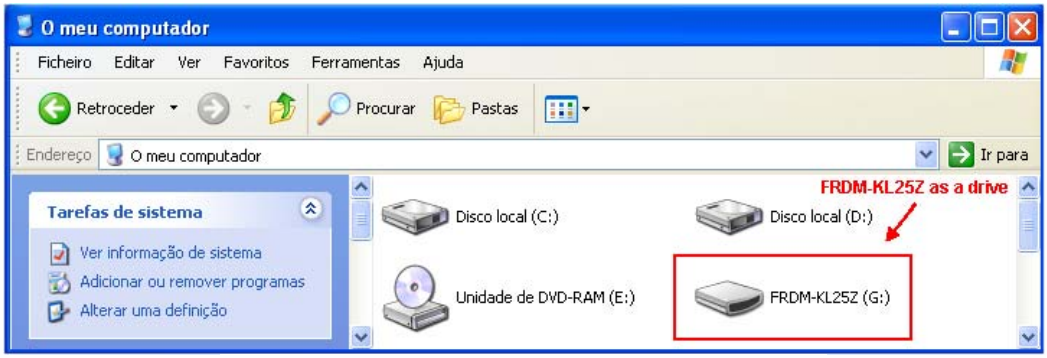

Fig. 23. **FRDM-KL25Z as a drive in *My Computer***

- Click twice on the icon FRDM-KL25Z to open the contents of the drive that will be something similar to:

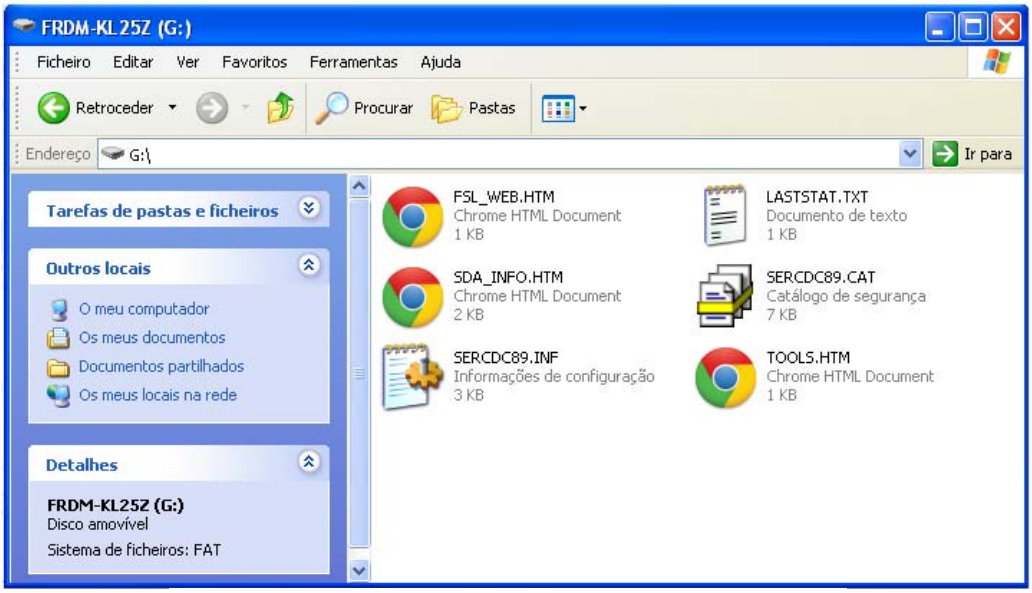

Fig. 24. **Contents of FRDM-KL25Z behaving as a drive**

- Cut, or drag, the HelloWorld_KL25Z.bin file in your PC's Downloads folder and paste, or drop, in the mbed Microcontroller folder, wait a few seconds for the drive folder to close automatically, and then hit reset on the microcontroller to start it running!

If everything goes well the RGB LED on FRDM-KL25Z will blink blue LED!

Main source of this tutorial is mbed.org website, specially the Handbook section at: https://mbed.org/handbook/mbed-FRDM-KL25Z-Getting-Started

See also:

- Examples programs for FRDM-KL25Z
  http://mbed.org/handbook/mbed-FRDM-KL25Z-Examples

- Explore the mbed platform
  https://mbed.org/explore/

And an enormous wealth of information on mbed's website!