HandsOn Technology

17, Jalan Masyhur 2, Taman
Perindustrian Cemerlang,
81800 Ulu Tiram, Johor, Malaysia.
sales@handsontec.com
http://www.handsontec.com

HT-USB5130 USB 8051 Flash µC Development Board

8051-Based System for Rapid Firmware Development

MODEL: HT-USB5130

8051 is one of the most popular 8-bit µController architectures in use today, learn it & use it the practical and HandsOn® way.

Suitable for Industrial Embedded Systems Control & Applications

From Novice to Expert

HT-USB5130 Getting Started Manual

1. INTRODUCTION

Flash microcontrollers are easy to program, which makes them suitable for rapid firmware development environments, professional and educational uses. In the past, program code was usually downloaded via a serial interface, but nowadays many PCs (especially laptops) only have USB ports. Our versatile USB Flash Board provides a solution to this problem. It is built around an AT89C5130/1A, which is an extended 8051-family microcontroller with an 80C52 core and a Full Speed USB port. As a sort of bonus, the IC has a complete update interface for downloading new firmware thru software in the form of its FLIP program, which is available free of charge from Atmel.

As with many other similar boards used for development purposes, the user program code is downloaded from a development PC to the microcontroller via a serial interface. Unfortunately, the good old RS232 interface is becoming increasingly rare. Laptops in particular often have only USB ports and no longer come with printer ports or serial ports.

Handson Technology design team thus developed a version of the Flash µC Board based on a modern microcontroller with a USB interface. For this purpose we selected the Atmel AT89C5130/1, which has an 80C52 core and thus belongs to the 8051 family. The IC incorporates a Full-Speed USB port, and it is specifically designed for use in USB devices such as printers, cameras, and so on. As a sort of bonus, the microcontroller has a complete update interface for downloading new firmware.

HT-uC5130 8051 USB Flash µC Board is suitable for not only learning how to program micro-controllers, but also for mature applications in device controllers, robots, and many other areas. If you want to develop USB software, this gives you everything you could wish for, although you do need a bit of technical expertise. Everyone else can regard this microcontroller board as a normal 8051 device that can be programmed via USB.

Features:

- 80C52x2 Core (6 Clocks per Instruction)
- 32-Kbyte On-chip Flash EEPROM In-System Programming through USB
- 3-KbyteFlash EEPROM for Boot Loader
- 1-Kbyte EEPROM Data
- On-chip Expanded RAM (ERAM): 1024 Bytes
- USB 1.1 and 2.0 Full Speed Compliant Module
- 5 Channels Programmable Counter Array (PCA)
- Programmable Hardware Watchdog Timer
- Keyboard Interrupt Interface on Port P1
- I2C interface 400Kbit/s, SPI Interface
- 32 I/O pins on header connectors located on PCB edge for easy port access
- 0 to 24 MHz On-chip Oscillator
- Switch between Programming& Run modes with the flip of a switch
- Regulated 5VDC/0.5A output for auxiliary circuitboard power supply
- Board powered by USB or 9~12VAC/DC wall adapter

Block Diagram:

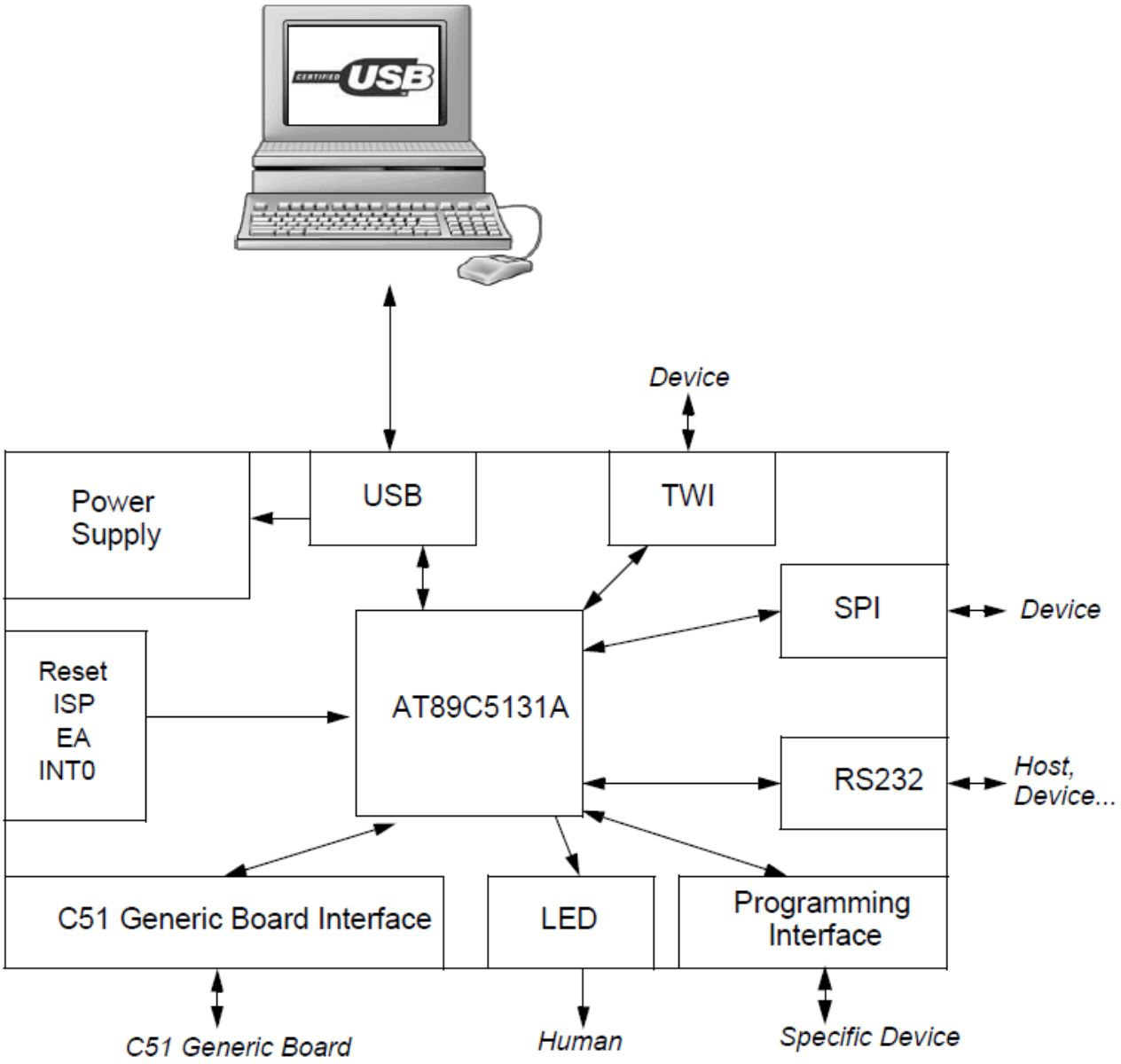
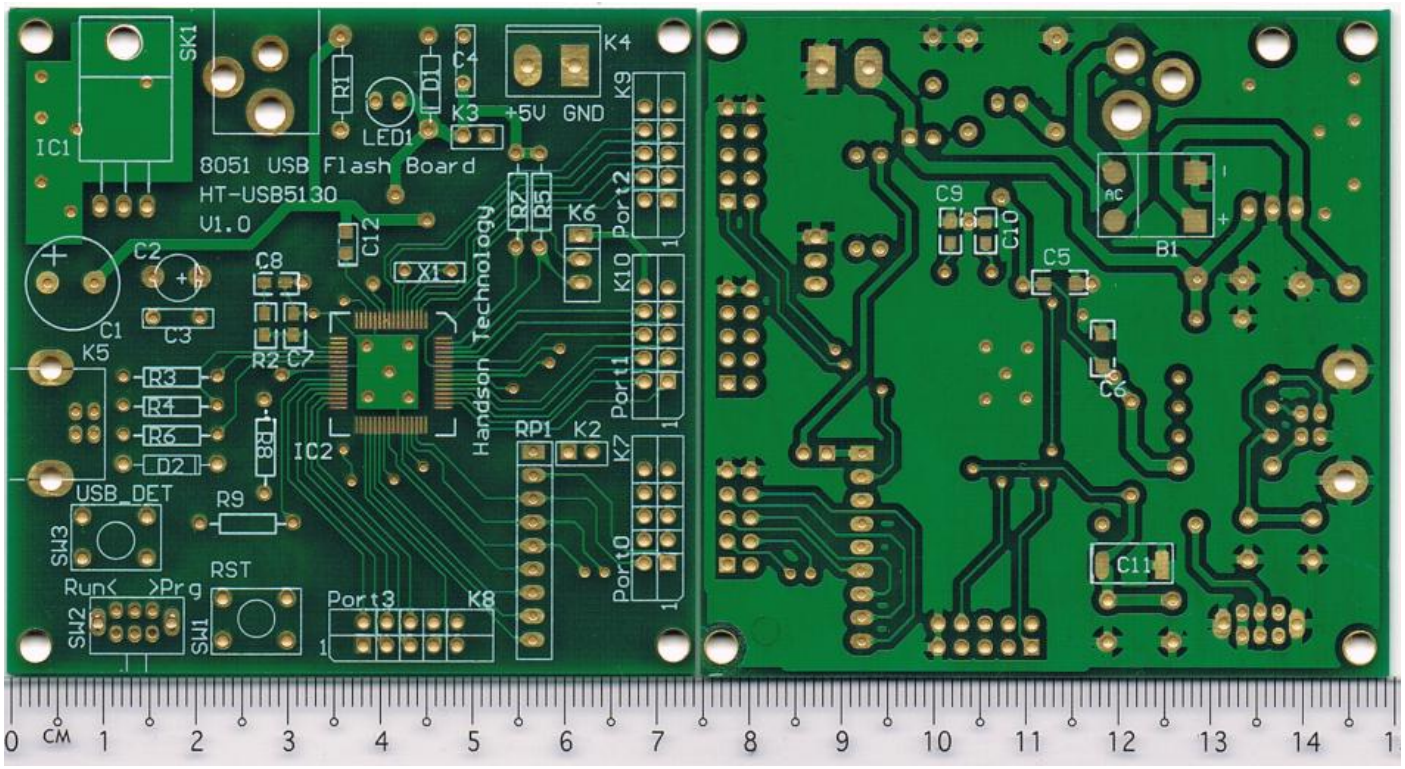


Figure 1.1: Board System Block Diagram

Components Layout:



Top Layer Bottom Layer

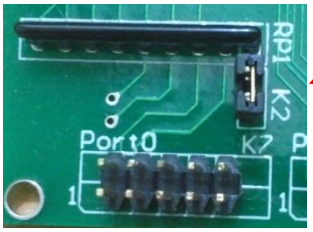
Figure 1.2: Components Layout and board dimension

2. Header I/O Pins Description:

Port-0:

P0.1	P0.3	P0.5	P0.7	GND
P0.0	P0.2	P0.4	P0.6	GND

K7(Port 0) Header Pin Assignment

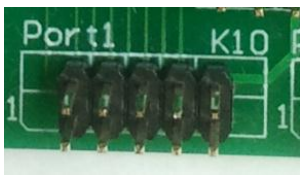


Insert this jumper K2 for P0 pull-up

Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s.By inserting jumper into location K2 will enable the external pull-up through the 10K resistors array RP1 to Port 0. To avoid any parasitic current consumption, Floating P0 inputs must be pulled to VDD or VSS.

Port-1:

P1.1	P1.3	P1.5	P1.7	GND
P1.0	P1.2	P1.4	P1.6	GND



Port 1 is an 8-bit bi-directional I/O port with internal pull-ups on all pins. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups.

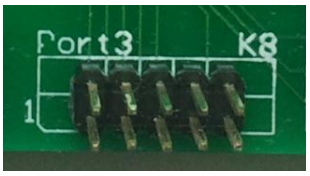
Port-2:

P2.1	P2.3	P2.5	P2.7	GND
P2.0	P2.2	P2.4	P2.6	GND

Port 2 is an 8-bit bi-directional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register.

Port-3:

P3.1	P3.3	P3.5	P3.7	GND
P3.0	P3.2	P3.4	P3.6	GND



Port 3 is an 8-bit bi-directional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups.

3. USB Peripheral

The HT-USB513 development board provides all the required hardware to develop a USB firmware for the 8051 family, this includes:

- USB connector, K5 Figure 3.1
- USB Reset button which allows to disconnect the pull-up on D+ and then to simulate an Attach/Detach of the USB cable, SW3 Figure 3.2
- The USB peripheral can also be used to perform an In-System Programming.

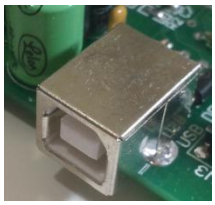


Figure 3.1: USB B-Type Connector K5

Figure 3.2: USB Detach/Attached Switch SW3

Figure 3.4 shown the USB detail schematic connection.

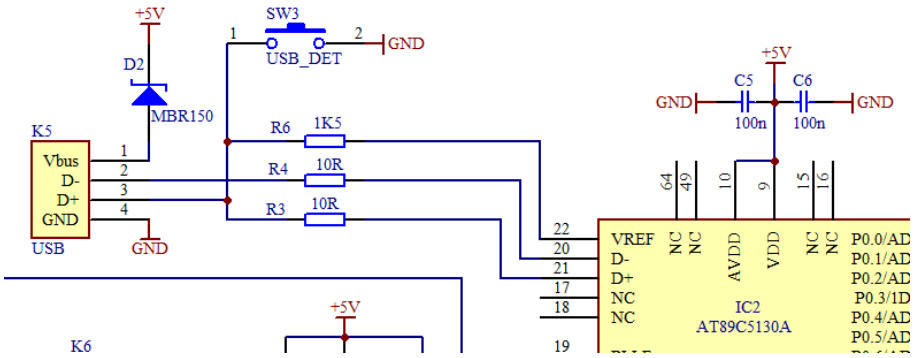
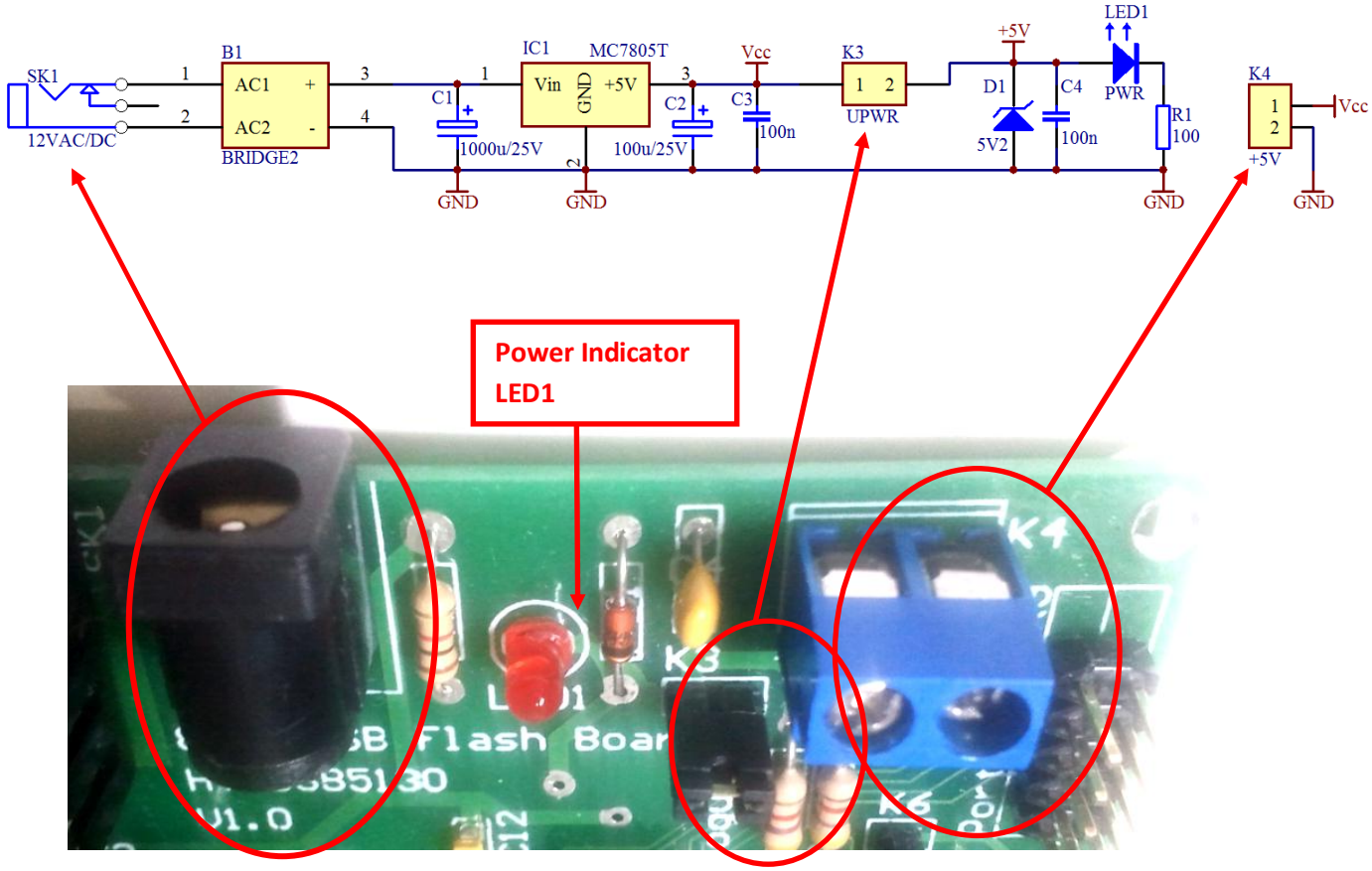


Figure 3.4: USB Connection Diagram

4. Power Supply



The on-board power supply circuitry allows various power supply configurations.

- V_{BUS} from USB (5V)
- External power supply (from 6 to 12V) or 9V battery

The power supply selection is performed using the K3 jumpers and external power socket SK1. The external power supply to SK1 can be any wall adapter which provide 9~12V/1A AC or DC. When K3 jumper is inserted, the board will be supplied by USB bus. Once the power is established, the power LED1 is lit.

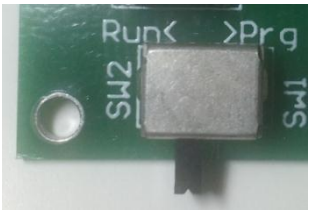
A regulated power supply is provided with K4, limited to output 5VDC/500mA. This provide a convenient way for power supply to other cuircuit board.

5. Function Switch

Reset Switch

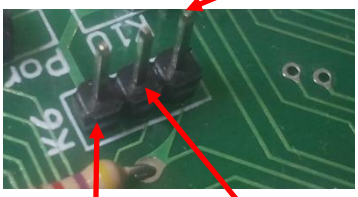
The external Reset push-button (SW1) is provided to easily generate a warm reset. This button is used for ISP process. The reset applied is active low.

Programming/Run Switch

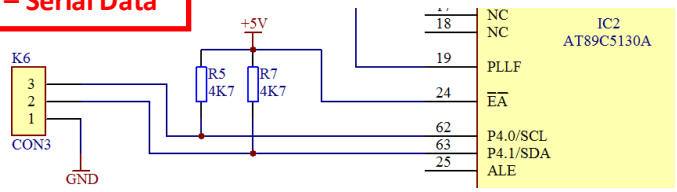


When SW2 is push to <**Run**> position, the board will execute user application firmware in flash memory. For the user code downloading or updating, this switch must push to <**Prg**> position to enter into ISP mode thru USB interface.

I2C Bus Interface



Two Wire Interface (TWI) or I2C Bus interface. K6 provide an easy access to external circuit which required this. SCL & SDA is open drain I/O pins with external pull-up resistors.
Pin 3 – Serial Clock SCL
Pin 2 – Serial Data SDA
Pin 1 – Ground



I2C Bus Connection Diagram

6. Device Programming

In-System Programming

The user flash memory of the HT-USB5131A can be programmed using the ISP mode of the device. To enter in ISP mode, first set SW2 to <PRG> position, follow by press and release the RST button (SW1). The device enters in ISP mode. ISP can then be performed using the USB bus (or with the peripheral corresponding with the boot-loader version). The user may need to re-enumerate the USB bus using the USB DET button (SW3) if the USB cable is already connected.

Flexible In-System Programmer FLIP

FLIP is an ISP programming software free from ATMEL to allows program FLASH C51 µController. It runs under Windows 9x / Me, Windows NT / 2000 / XP and Linux as well. Communicating with the target device may be done through a RS232 link, a CAN or an USB link, depending on the communication media supported by the target device.

FLIP Software Installation and Setup

Windows Installation:

- Copy the "*Flip Installer"* file to an empty temporary directory. The latest copy can be downloaded at http://www.atmel.com/tools/flip.aspx. "Flip installer - 3.4.7.112.exe" is the latest version at the time of this document preparation.
- Run the .exe files and follow the on-screen instruction to perform installation to the local C hard drive.

This below icon will appeared on your computer desktop if the installation is successful.

7. Using the C Compiler:

Now it's time to get down to business! What we want to do is to write the first program, translate it and send it to the uC. For this we need some software. We will use the well-known evaluation software from Keil – uVision2 Integrated Development Environment, which is located on the accompanied CDROM. The Keil Software 8051 development tools are programs you use to compile your C code, assemble your assembly source files, link and locate object modules and libraries, create HEX files, and debug your target program.

The µVision2 IDE is a Windows-based software development platform that combines a robust editor, project manager, and make facility. µVision2 supports all of the Keil tools for the 8051 including the C compiler (a C Compiler translates a source text into pure machine code), macro assembler, linker/locator, and object-HEX converter. µVision2 helps expedite the development process of your embedded applications by providing the following:

- Full-featured source code editor
- Device database for configuring the development tool setting
- Project manager for creating and maintaining your projects,
- Integrated make facility for assembling, compiling, and linking your embedded applications,
- Dialogs for all development tool settings,
- True integrated source-level Debugger with high-speed CPU and peripheral simulator,
- Links to development tools manuals, device datasheets & user's guides.

The µVision2 screen provides you with a menu bar for command entry, a tool bar where you can rapidly select command buttons, and windows for source files, dialog boxes, and information displays. µVision2 lets you simultaneously open and view multiple source files.

Below show the screenshot of the uVision2 IDE, Figure 7.1

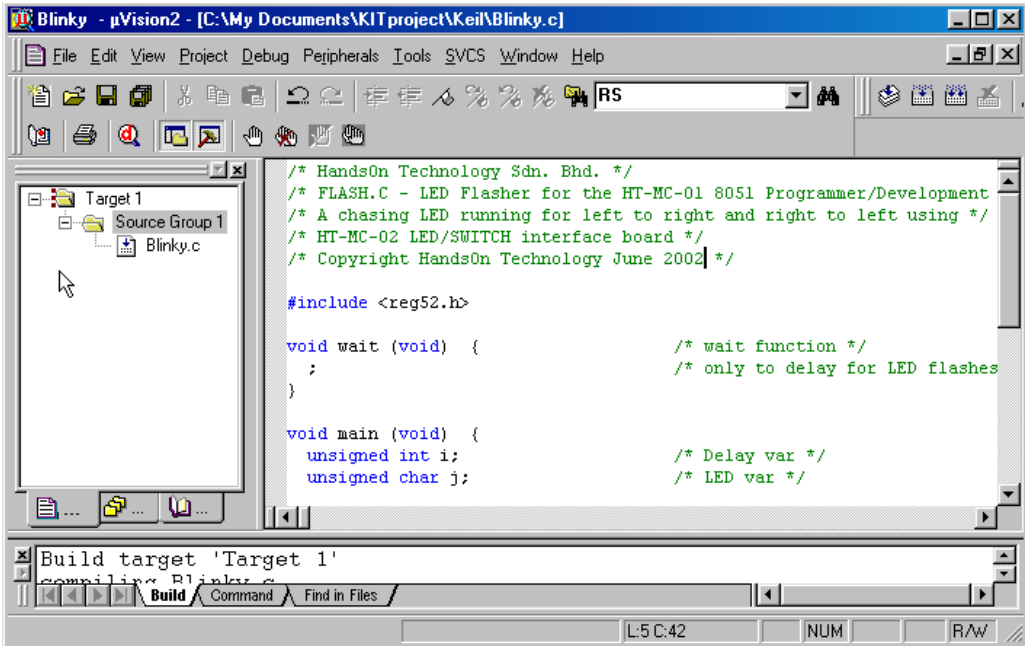


Figure 7.1

8. Creating Projects

This section describes the **Build Mode** of µVision2 and shows you how to use the user interface to create a sample program. A full user manual for µVision2 can be found in the accompanied CDROM. µVision2 includes a project manager that makes it easy to design applications for the 8051 family. You need to perform the following steps to create a new project:

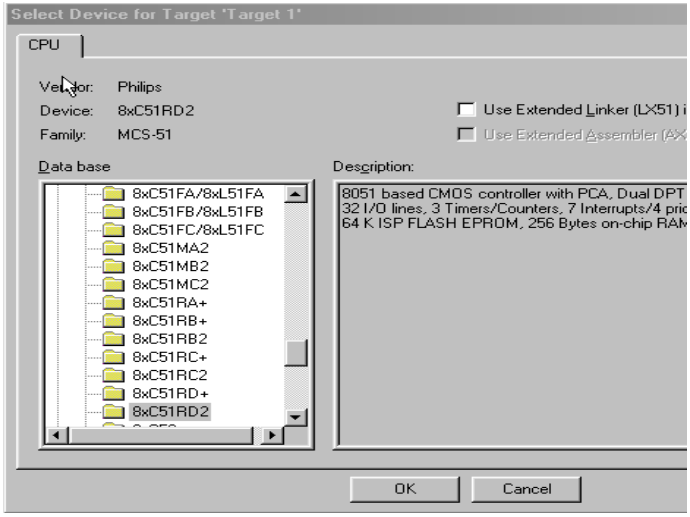
- Start µVision2, create a project file and select a CPU from the device database.
- Create a new source file and add this source file to the project.
- Add and configure the start-up code for the 8051 device
- Set tool options for target hardware.
- Build project and create a HEX file for PROM programming.

The description is a step-by-step tutorial that shows you how to create a simple µVision2 project.

8.1 Starting µVision2 and Creating a Project (Blinking LED)

µVision2 is a standard Windows application and started by clicking on the program icon. To create a new project file select from the µVision2 menu **Project – New Project**…. This opens a standard Windows dialog that asks you for the new project file name. We suggest that you use a separate folder for each project. You can simply use the icon **Create New Folder** in this dialog to get a new empty folder. Then select this folder and enter the file name for the new project, i.e. in this example **Blinky** and click “SAVE” icon.

The “Select Device for Target” dialog box will pop up shows the µVision device database.



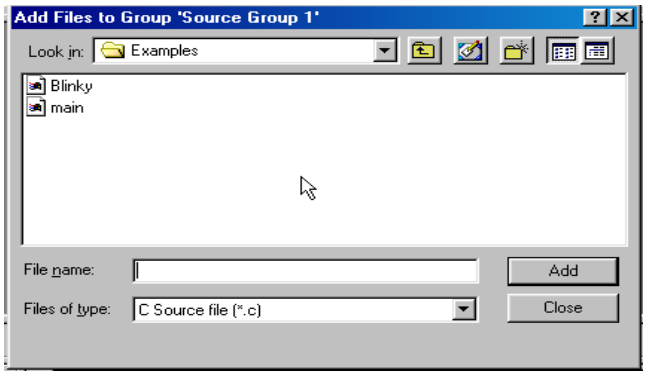
We are using for our examples the Philips P89C51RB2/RC2/RD2 CPU. This selection sets necessary tool options for the P89C51RD2 device and simplifies in this way the tool configuration. µVision2 creates a new project file with the name **Blinky.UV2** which contains a default target and file group name. You can see these names in the **Project Window:**

8.2 Creating New Source Files

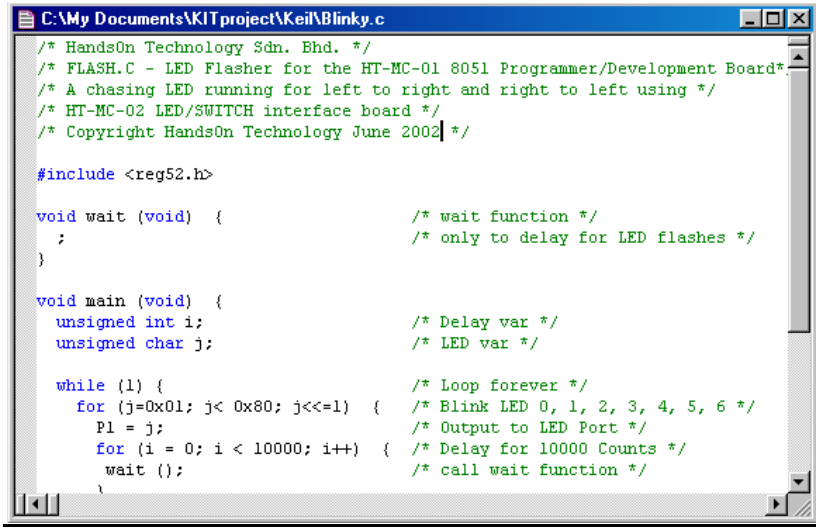
You may create a new source file with the menu option **File – New**. This opens an empty editor window where you can enter your source code. µVision2 enables the C color syntax highlighting when you save your file with the dialog

File – Save As… under a filename with the extension ***.**C. We are saving our example file under the name **Blinky.C**.

Or you can import the blinky.c source file by the following: Move mouse pointer to “Source Group1” icon in the Project Window and right click mouse buttons and select “Add File to Group ‘Source Group 1”. “Add File to Group ‘Source Group 1’” Dialog appeared.

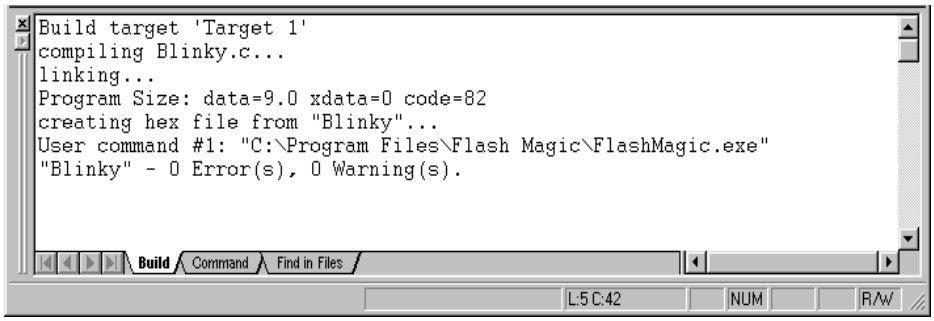


Select “Blinky” by double clicking on or select “Add” button follow by “Close” button. You can view the C source file by double clicking the file name in the Project Window.



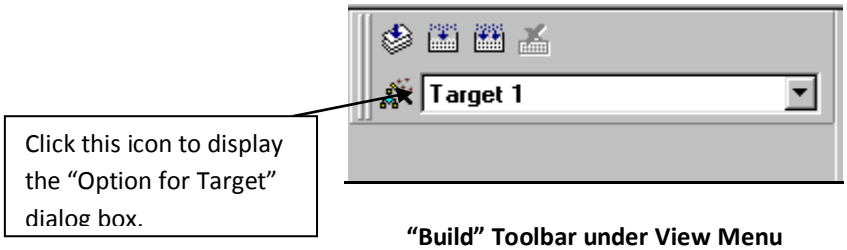
8.3 Building Projects and Creating a HEX Files

Typical, the tool settings under **Options – Target** are all you need to start a new application. You may translate all source files and line the application with a click on the **Build Target** toolbar icon. When you build an application with syntax errors, µVision2 will display errors and warning messages in the **Output Window – Build** page. A double click on a message line opens the source file on the correct location in a µVision2 editor window.

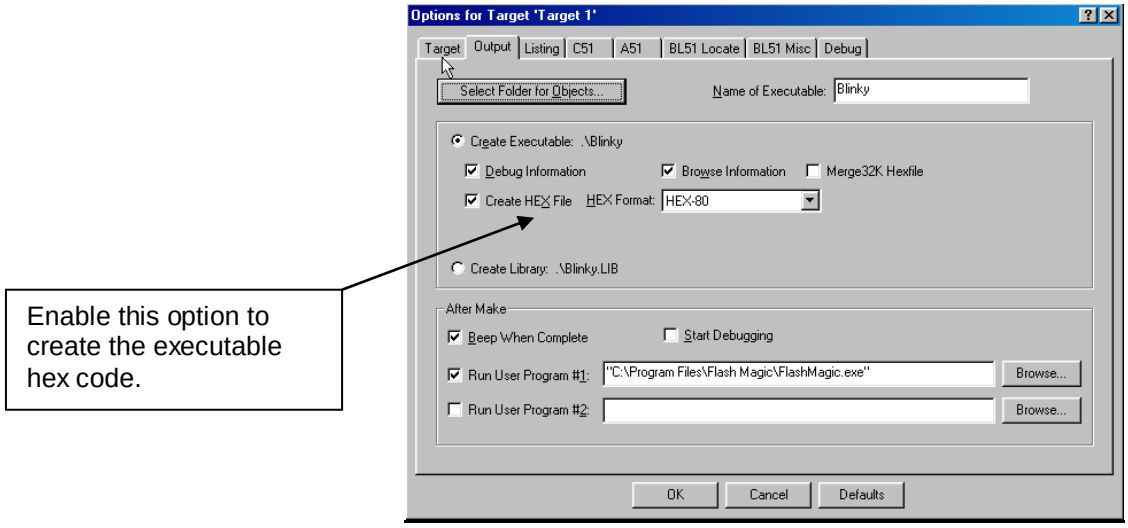


"Output" window under View Menu

After you have tested your application, it is required to create an Intel HEX file to download the software into a Flash programmer. µVision2 creates HEX files with each build process when **Create HEX file** under **Options for Target – Output** is enabled. You may start your Flash programming utility after the make process when you specify the program under the option **Run User Program #1.**



"Build" Toolbar under View Menu



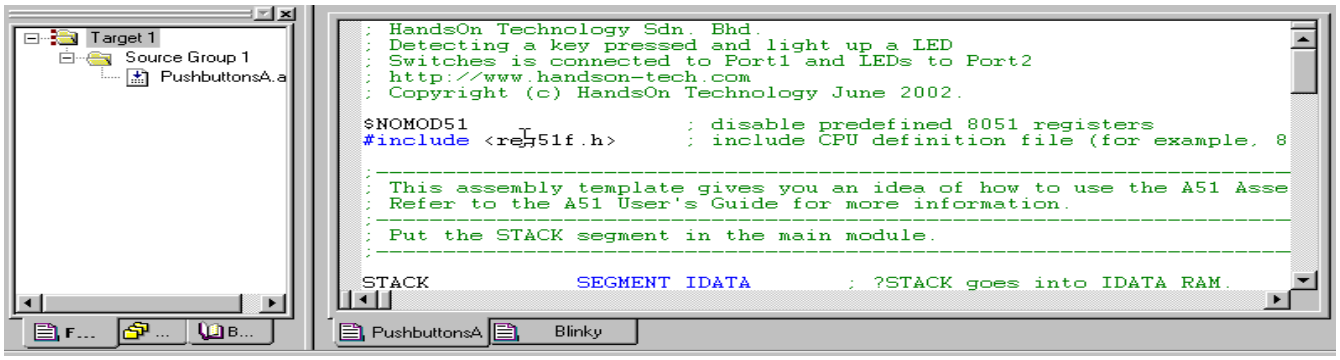
"Option for Target" dialog box

9. Using Assembler

What is an Assembler?

An assembler is a software tool designed to simplify the task of writing computer programs. It translates symbolic code into executable object code. This object code may then be programmed into a µcontroller and executed. Assembly language programs translate directly into CPU instructions which instruct the processor what operations to perform. Therefore, to effectively write assembly programs, you should be familiar with both the µcomputer architecture and the assembly language.

This section shows you how to create **8051** program, developed in assembly language. The program will light up one of the eight LEDs when one of the eight push buttons is pressed using 8LEDs+Switches interface board. Follow the procedure as in Section 4 above to open up a project "Pushbutton.uv2" project file. Assemble the program to create the HEX file to download it to the ***HT-USB513 DEVELOPMENT BOARD.***



10. Programming the 8051:

10.1 Introduction

To load a program into the µC on the ***HT-USB5130 Development Kit***, you will need the Windows program "***FLIP"*** programming software from ATMEL which can be found on the accompanied CDROM or free download it from the ATMEL Website: http://www.atmel.com and install this to your C drive.

FLIP is Windows software from the ATMEL that allows easy access to all the ISP features provided by the P89C51Rx2 devices. These features include:

- Intel MCS-86 Hexadecimal Object, Code 88 file format supported for data fileloading and saving Optional address offset for loading
- Buffer control:
  - data byte modification, address range fill-in, goto a specified address, data byte sequence search
- Target device control:
  - Blank check, Program, Read, Verify, Erase, Special bytes edition capability
- Permanently displayed and updated information about the buffer options and the target device status
- Customizable flow for quick repeating programming
- Command window allowing control through a command file (Tcl language)
- Commands recording for automatic reconfiguration and commands replay.
- Progress bar, tooltips and on-line help

In this section the user will program the AT89C5131A microcontroller via USB using FLIP software. The following procedure will guide you through the programming of the demonstration program.

1. Run FLIP software, screen at start up as see in Figure 10.1.

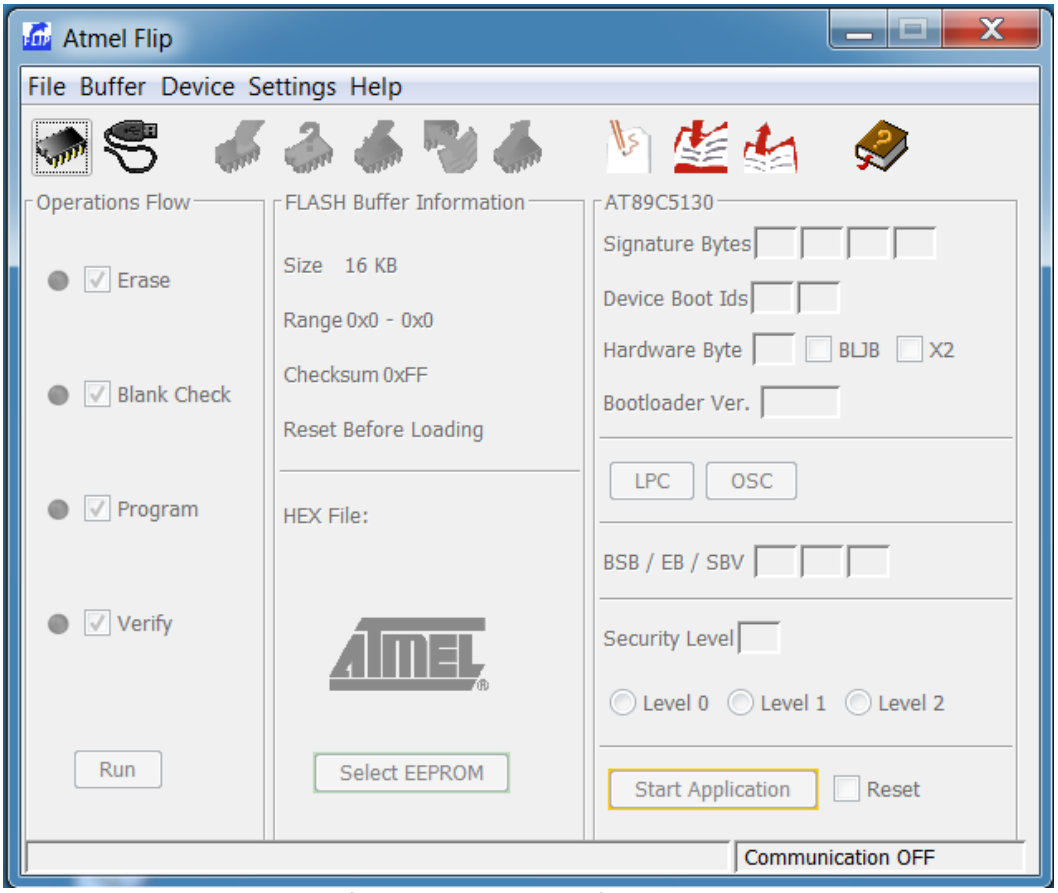


Figure 10.1: FLIP starting screen

2. From the Device Menu, choose “Select” and select the device (AT89C5130) that is connected to the evaluation board, Figure 10.2.

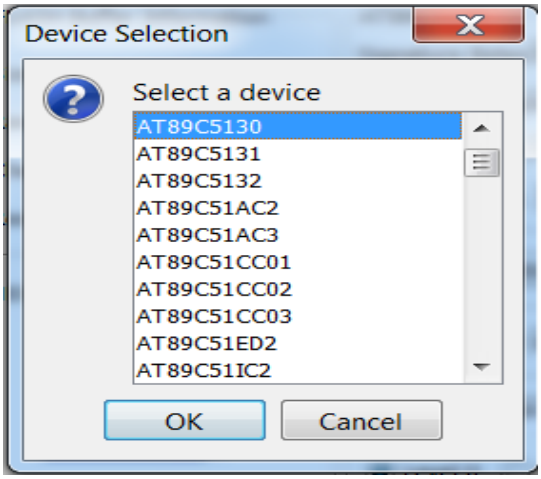


Figure 10.2: Device Selection

3. Click the “Setting” > “Communication” > USB. Initialize the communication by clicking the “Open” button in the USB Port Connection pop-up window, Figure 10.3.

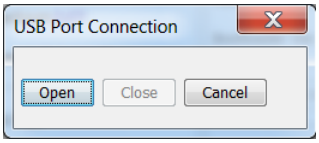


Figure 10.3

4. If the connection is successful, the FLIP window should look like Figure 10.4.

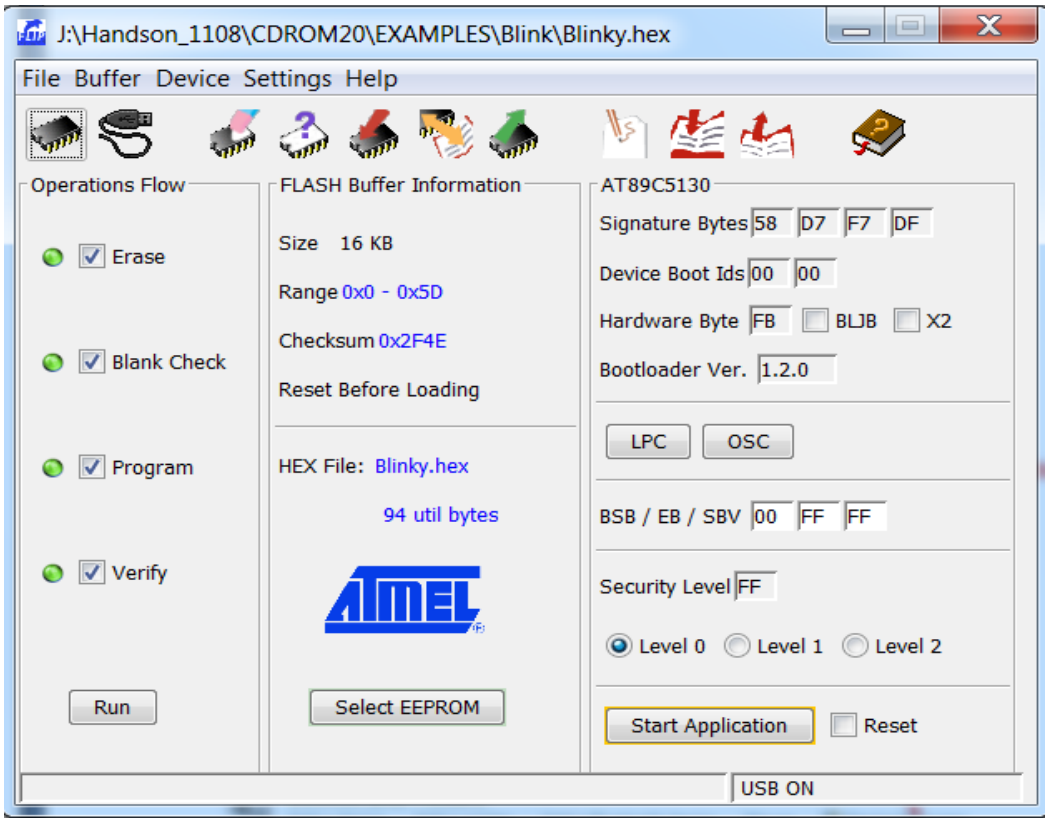


Figure 10.4

5. In the File menu, select “Load HEX” and choose the demonstration program “blinky.hex”.
6. The message “HEX file parsed” is displayed at the bottom of the FLIP window, Figure 10.5.

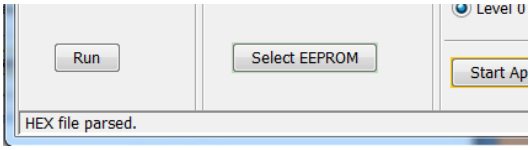


Figure 10.5

7. Ensure the following check boxes are selected in the “Operations Flow” section of FLIP, as in Figure 10.4:
    a. Erase
    b. Blank Check
    c. Program
    d. Verify

    These are the operations that will be performed on the microcontroller.

8. Make sure the “BLJB” bit is uncheck, Figure 10.6.

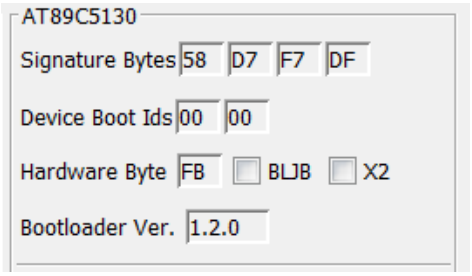


Figure 10.6

8. Press the “Run” button. Programming is executed. The “Memory Verify Pass” message confirms programming is successful and that the microcontroller has been programmed.
9. Push “SW2” to “Run” position and press “SW1” “RST” reset button once, Figure 10.7. The user application should start executing.

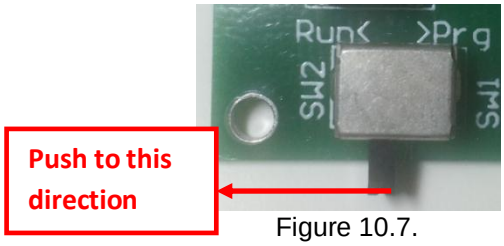
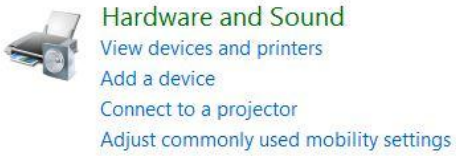


Figure 10.7.

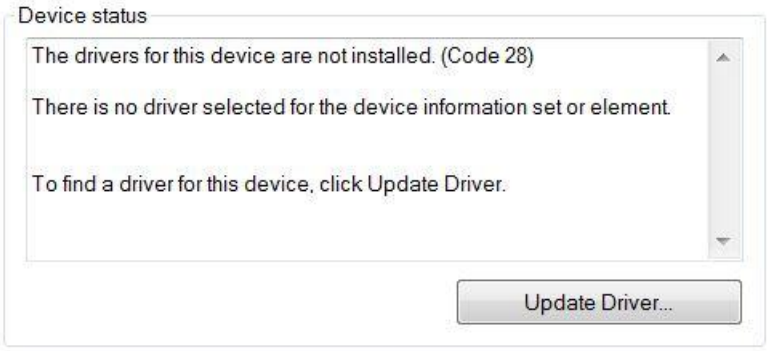
Please follow the following steps if HT-USB5130 Development Board is not able to recognise by your PC Host after installed FLIPS software.

1. Go to "Control Panel" by clicking Start icon. 
2. Click the " Hardware and Sound"

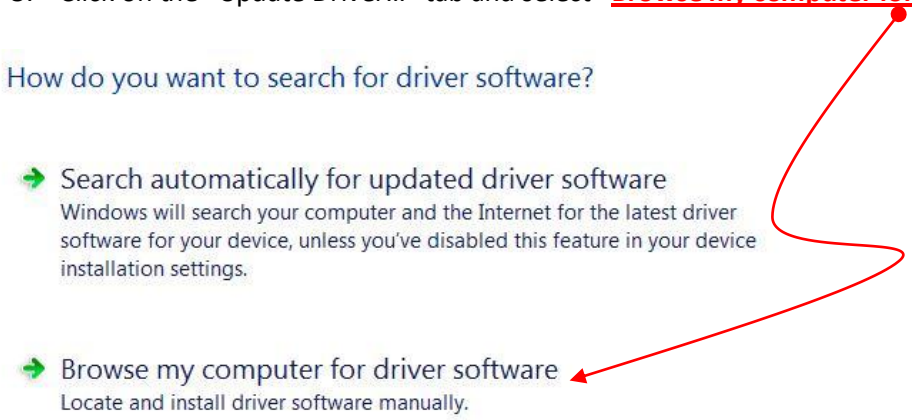


3. From " Devices and Printers" icon, click " **Device Manager**".

4. Under the "Device Manager" panel, locate " Unknown device" and double click on it. This will bring up "Update Driver…" tab under Device status panel.



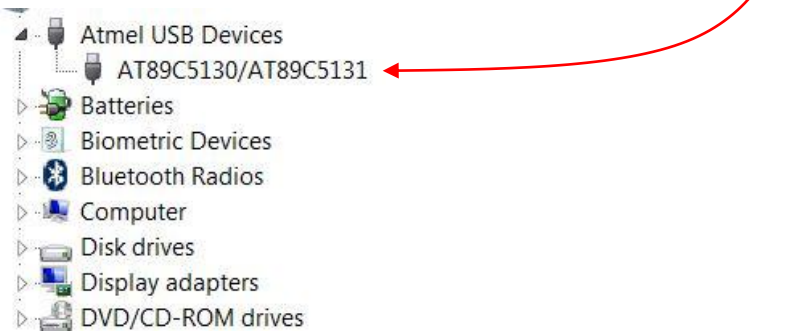
5. Click on the "Update Driver…" tab and select "**Browse my computer for driver software**".



6. Locate FLIP software in your computer.

Note: The default installation path for FLIPS is C:\Program Files\Atmel\Flip3.4.5.

7. The HT-USB5130 should now recognised by your host PC as "**Atmel USB Devices**">> "AT89C5130/AT89C5131"



11. LEDs & SWITCH BOARD

8 switches + 8 LEDs (HT-MC-02) test and experiment board .

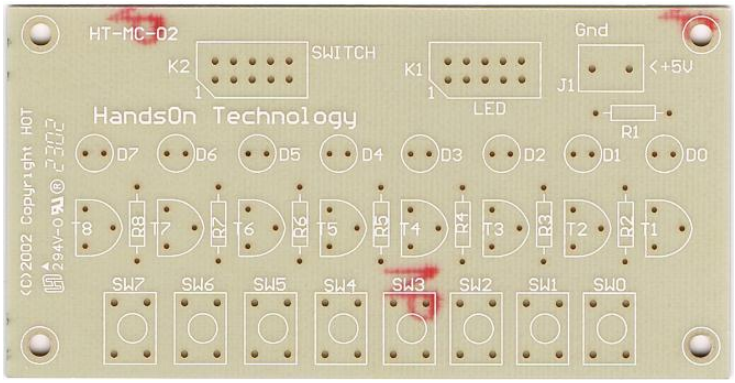


Figure 11.1: LEDs + Switches Board Layout

Note: This test board is use with HT-USB5130 Development board (to be purchased separately)

Test Setup for LEDs + Switches experiment:

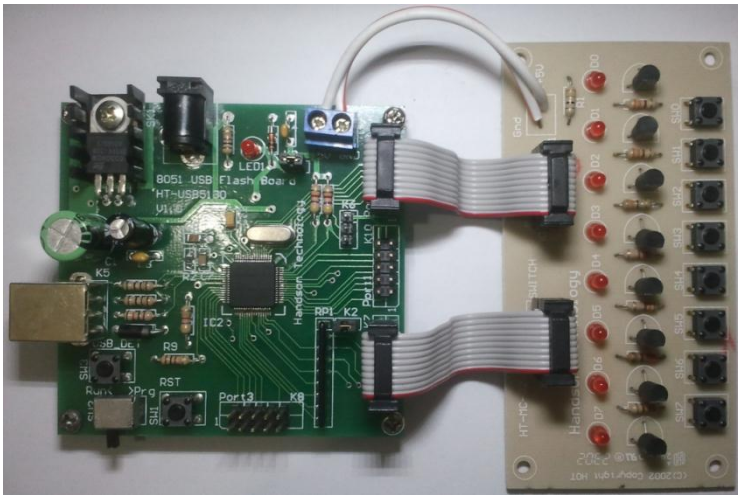


Figure 11.2: Test Setup.

12. System Requirements

- PC running with Windows®
- 16 MB RAM minimum
- 20 MB free disk space
- Mouse
- USB Port

13. CDROM Contents:

Useful files in the accompanied CDROM:

1. **ek51v701** application file. This is the Keil C-Compiler and Assembler windows program. Locates this file in the CDROM and run this program to install the Keil Compiler to the PC.
2. **FLIP** application file. This is Flash Memory µC programming software from ATMEL. Locates this file in the CDROM and install it into the hard disc.
3. **AcrobatReader910** application file. Acrobat Reader program to read and print all the documentations contain in this CDROM. Run this application program to install the Acrobat Reader 9 into the hard disc.
4. **8051µC** tutorial, application note and IC data sheets related to this development board.
5. **Examples Files.** A hands-on examples code for you to get familiar with the compiler, assembler and programming the ***HT-USB5130*** *development board*.
6. **Project Folder:** Collection of constructional projects based on 8051 uC.
7. Schematic diagram and technical detail for ***HT-USB5130 Development Board***.

14. Resources:

The following are development resources for the 8051 with samples code and other helpful utilities.

1. Atmel Corporation :
    a. http://www.atmel.com
    b. FLIP flash programming software http://www.atmel.com/tools/FLIP.aspx
    c. Documentations: http://www.atmel.com/devices/AT89C5130A-M.aspx?tab=documents
2. Vault Information Services has one of the best online tutorials for the 8051 that we have seen to date. http://www.8052.com.
3. MCU Tools supplier, emulators, compilers & assembler: http://www.mcu123.com
4. Professional Industrial compiler & emulator. http://www.keil.com , evaluation available for free download.
5. Open Source IDE: http://www.codeblocks.org

14. CONTACT:

For Sales : sales@handsontec.com

For technical support: techsupport@handsontec.com

For general inquiry: inquiry@handsontec.com

15. On-Line Purchase:

Log-in to http://handsontec.com/index.php/product/8051-development-board/

HandsOn Technology provides a multimedia and interactive platform for everyone interested in electronics. From beginner to diehard, from student to lecturer. Information, education, inspiration and entertainment. Analog and digital, practical and theoretical; software and hardware.

Hands*On* Technology support Open Source Hardware (OSHW) Development Platform.

Learn : Design : Share

www.handsontec.com

The Face behind our product quality…
In a world of constant change and continuous technological development, a new or replacement product is never far away – and they all need to be tested.
Many vendors simply import and sell wihtout checks and this cannot be the ultimate interests of anyone, particularly the customer. Every part sell on Handsotec is fully tested. So when buying from Handsontec products range, you can be confident you're getting outstanding quality and value.

We keep adding the new parts so that you can get rolling on your next project.

Breakout Boards & Modules

Connectors

Electro-Mechanical Parts

Engineering Material

Mechanical Hardware

P Electronics Components

Power Supply

Arduino Board & Shield

Tools & Accessory