Embedded System Design

Chapter 4: Development Tools

1. Code Composer Studio
2. Keil Tools by ARM
1. Code Composer Studio - Overview

- Code Composer Studio™ (CCStudio) is an integrated development environment (IDE) for Texas Instruments (TI) **embedded processor families**.
- CCStudio comprises a suite of tools used to develop and debug **embedded applications**.
1. Code Composer Studio – Create a project

- Create new project
  - Family: ARM
  - Variant: LM4F120H5QR
  - Connection: ICDI
  - Project templates: Empty project
1. Code Composer Studio – Project view

• C/C++ Projects view
1. Code Composer Studio – Properties

- Setup properties...
1. Code Composer Studio – Properties

- Include options:
  - C:\StellarisWare
1. Code Composer Studio – Properties

- File Search Path
  - Driverlib-cm4f.lib
1. Code Composer Studio – Build a project

- Build your project
## 1. Code Composer Studio – Debug

<table>
<thead>
<tr>
<th>Image</th>
<th>Name</th>
<th>Description</th>
<th>Availability</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image" /></td>
<td>New Target Configuration</td>
<td>Creates a new target configuration file.</td>
<td>File New Menu Target Menu</td>
</tr>
<tr>
<td><img src="image2.png" alt="Image" /></td>
<td>Debug</td>
<td>Opens a dialog to modify existing debug configurations. Its drop down can be used to access other launching options.</td>
<td>Debug Toolbar Target Menu</td>
</tr>
<tr>
<td><img src="image3.png" alt="Image" /></td>
<td>Connect Target</td>
<td>Connect to hardware targets.</td>
<td>TI Debug Toolbar Target Menu Debug View Context Menu</td>
</tr>
<tr>
<td><img src="image4.png" alt="Image" /></td>
<td>Terminate All</td>
<td>Terminates all active debug sessions.</td>
<td>Target Menu Debug View Toolbar</td>
</tr>
</tbody>
</table>
2. Keil Tools by ARM

• The Keil products from ARM support over 700 of the most popular ARM microcontrollers.
  – Includes RealView® Compilation Tools including C/C++ Compiler, Macro Assembler, and Linker
  – Includes Debuggers, Real-time Kernels, Single-board Computers, and Emulators
  – All tools are integrated into µVision which provides interfaces to ULINK and other third-party debug adapters.
2. Keil Tools by ARM – Create a project
2. Keil Tools by ARM – Select device

- Vendor: Texas Instruments
- Device: LM4F120H5QR
2. Keil Tools by ARM – Add files

• Add `main.c`

```c
#include "inc/hw_types.h"
#include "inc/hw_memmap.h"
#include "driverlib/sysctl.h"
#include "driverlib/gpio.h"

int main(void)
{
    int LED = 2;
    SysCtlClockSet(SYSCTL_SYSDIV_4|SYSCTL_USE_PLL);
    SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOF);
    GPIOPinTypeGPIOOutput(GPIO_PORTF_BASE, GPIO_)
    while(1)
```
2. Keil Tools by ARM – Add files

- Add C:\StellarisWare\driverlib\rvmdk-cm4\driverlib-cm4f.lib
2. Keil Tools by ARM – Options

- Include Paths: C:\StellarisWare
2. Keil Tools by ARM – Build target
2. Keil Tools by ARM – Run on Kit
2. Keil Tools by ARM – Debug

- Use Simulator or Use Stellaris ICDI
2. Keil Tools by ARM – Debug
2. Keil Tools by ARM – Simulation
Sample Codes

```c
#include "inc/hw_memmap.h"
#include "inc/hw_types.h"
#include "driverlib/gpio.c"
#include "driverlib/sysctl.h"

unsigned int i;

int main()
{
    //Enable GPIO Port F
    SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOF);
    // Set pins PF3 as output.
    GPIOPinTypeGPIOoutput(GPIO_PORTF_BASE,GPIO_PIN_3);
    while(1)
    {
        GPIOPinWrite(GPIO_PORTF_BASE,GPIO_PIN_3,GPIO_PIN_3); //SET PF3
        for(i=0;i<1000000;i++) {} //Delay
        GPIOPinWrite(GPIO_PORTF_BASE,GPIO_PIN_3,0); //Clear PF3
        for(i=0;i<1000000;i++) {} //Delay
    }
}
```