EE2016 - Microprocessor Theory + Lab
Experiment-8: Introduction to the LPC2378 Microcontroller

Few experiments in this lab will use the LPC2378 microcontroller. The LPC2378 microcontroller is based on a 16-bit/32-bit ARM7TDMI-S CPU. This document gives an introduction and some aspects of programming for the first experiment on the LPC2378.

1 The LPC2378 Microcontroller

The LPC2378 microcontroller to be used in this experiment is from NXP semiconductors. LPC stands for *Low Power Consumption*. The LPC2378 is based on the ARM7TDMI-S CPU so we will discuss aspects of ARM below.

1.1 The ARM and ARM7TDMI-S

In this experiment, we will use an ARM7-based processor, namely LPC2378. The ARM core here is called the ARM7TDMI-S (where T stands for Thumb Instructions, D for on-chip debugging support, M for multiplier, I for embedded In circuit emulation hardware and S for the synthesizable option based on RTL provided).

ARM7TDMI was the first of a range of processors introduced in 1995 by ARM. ARM7TDMI is the first core to include the Thumb instruction set. The Thumb instruction set contains 16-bit instructions. These serve as a 'compact shorthand' for a subset of the 32-bit instructions of the ARM. For 32-bit systems, ARM instructions are recommended for higher performance. Thumb instructions are appropriate for memory constrained systems.

1.2 Block Diagram of LPC2378

The LPC2378 block diagram is shown in Fig. **??**. The instructions for the board to be used in the experiments are available as a separate pdf file.

2 Brief Description of the Software Environment

In this laboratory experiment, programming will be done in C. The instructions for writing C language programs in Keil μVision are as follows.

```
1. Open Keil uVision and  Click project  > New uVision Project

2. Select the 'Legacy device Database' under device dropdown. Search for  LPC2378 and select it.

3. Choose Yes when prompted to copy LPC2300.s Startup  file.

4. Right Click on Target 1, choose 'Options for Target 1'.

5. Under the Target tab tick 'Use Microlib'. Under Output tab tick 'Create Hex File'

6. Right click Source Group1 under Target in the left side of the keil window. Select Add
new Item to group ..

7.  Select C file in the list, give a filename and save.

8. Write your program.
```

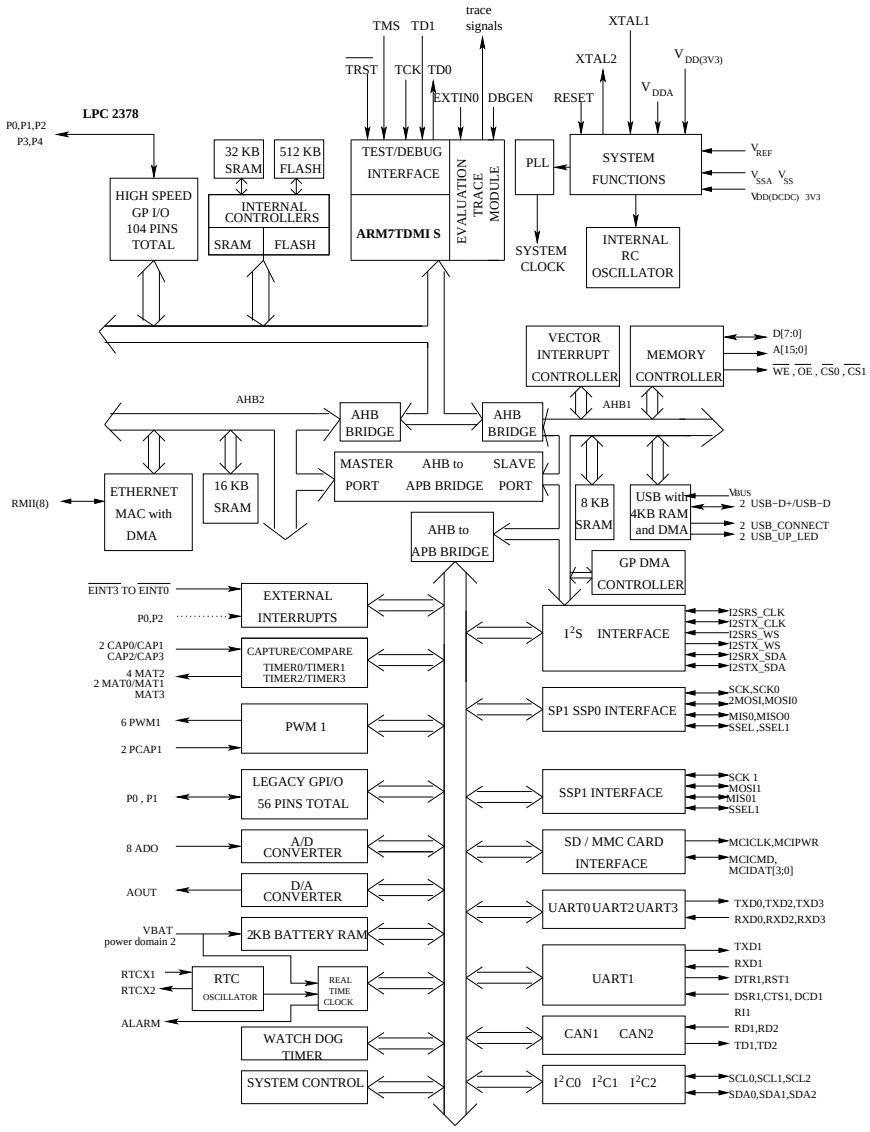

Figure 1: LPC2378

9. Go to Project tab and click on Build Target (or press F7).

3 How to Upload the Program to the Flash on LPC2378 in the Board

The instructions for uploading the program on to the flash memory on the LPC2378 are as follows.

1. Run FlashMagic.

2. Select LPC2378.

3. Choose the correct COM port to which the USB to Serial converter is connected.
 If you do not know the COM Port, Use Device Manager to find out.

4. Set baud rate to 19200 and Oscillator to 12 MHz.

5. Choose the Hex file to be transferred.

6. On the board, put the "Program/Execute" slide Switch to "Program" mode (slide it down) and press reset.

7. On FlashMagic click 'Start' to start   programming.

8. Once programming is done, put the "Program/Execute" slide Switch to "Execute" mode
 (slide it up) and press reset to execute the program.

4 Tasks for the Experiment

In this experiment, we will write C programs for various tasks.

Task 1: Complete the following program to cause the LEDs on the ARM-board to blink.

```
#include "LPC23xx.h"

int main ()

{



while(1)
{
FIO3DIR=0x--; // LEDs are connected to lower 8 bits of
  // Fast IO port 3 (FIO3PIN). Hence set the lower 8 bits
              //   as output in the corresponding direction
              //   register (FIO3DIR)
FIO3PIN=0x--; Port to which actual data is to be written
delay_code ; to be written
FIO3PIN=0x--
delay_code; to be written
```

```
}
return 0;
}
```

Task 2: In the previous task, we focussed on output on the LEDs. In this task, we will take input. In particular, read the settings of an 8-way DIP (Dual Inline Package) switch and display it on the LEDs. Use FIO4DIR AND FIO4PIN for data input.

Task 3: Write a C program to read a DIP switch, split into two nibbles (4 bits), multiply them and display the product on the LEDs.