

Internet of Things (IOT) – lab 2

Objective:

- (1) Demonstrate the LED blinking at Arduino UNO
- (2) Establish communication from Arduino Due in UDOO (a coordinator hub) to Arduino UNO

Arduino Due in UDOO

The information provided from page 1 to page 3 is mostly the same as stated in lab 1. This may save you the effort of referring to lab 1.

Insert the SD card into UDOO (Figure 2). Download from the Internet Arduino IDE 1.5.4, Arduino IDE patch files and serial driver (Figure 1). Install the downloaded software. Copy the patch files (bossac and cygwin1.dll) into ...\\hardware\\tools directory of Arduino IDE, for example c:\\Program File(x86)\\Arduino\\hardware\\tools .

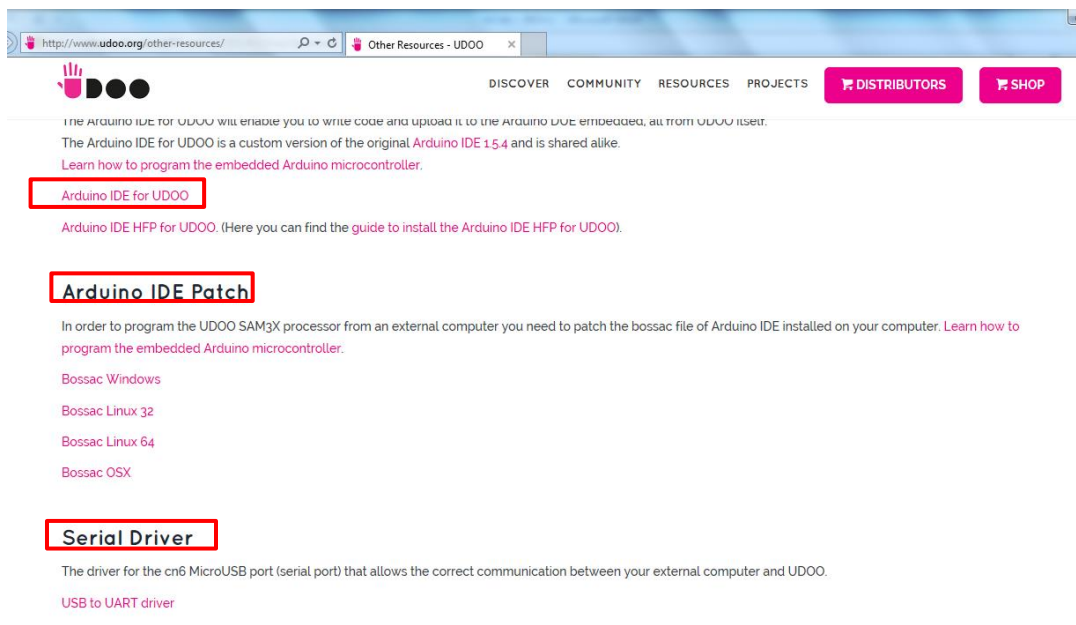


Figure 1: Arduino IDE, patch files and serial driver

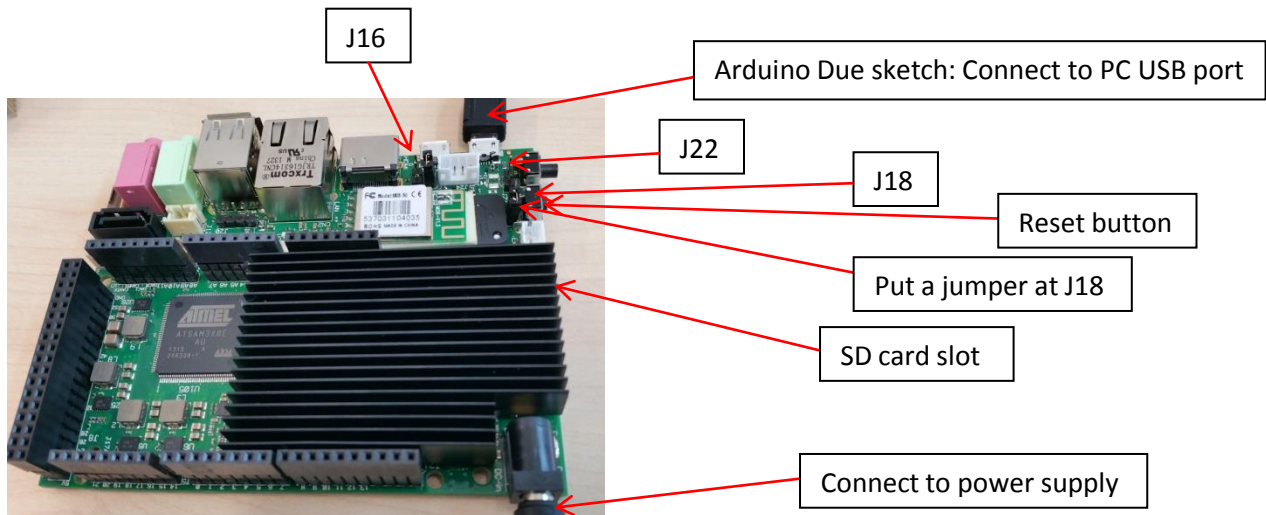


Figure 2: UDOO

Download a serial monitor software (Tera Term for Windows, serial tools for OS X, minicom for LINUX) from the Internet and installed the software. Connect the UDOO board to the PC USB port and power supply (Figure 2). Put a jumper at J18 as shown in Figure 2. As an example, start the Tera Term software. Select Serial and a relevant COM port (for example COM18). At Tera Term, select Setup -> serial port ... -> baud rate 115200. Press the reset button at J18. Messages like Figure 3 will appear.

```
COM18:115200baud - Tera Term VT
File Edit Setup Control Window Help
mx6q pll2: 528MHz
mx6q pll3: 480MHz
mx6q pll8: 50MHz
lpg clock : 660000000Hz
lpg per clock : 660000000Hz
uart clock : 800000000Hz
cspi clock : 600000000Hz
ahb clock : 132000000Hz
axi clock : 264000000Hz
eml_slow clock: 132000000Hz
ddr clock : 528000000Hz
usdhc1 clock : 198000000Hz
usdhc2 clock : 198000000Hz
usdhc3 clock : 198000000Hz
usdhc4 clock : 198000000Hz
nfc clock : 240000000Hz
Board: i.MX6Q-UD00: unknown-board Board: 0x63012 [POR]
Boot Device: NOR
I2C: ready
DRAM: 1 GB
MMC: FSL_USDHC: 0_FSL_USDHC: 1_FSL_USDHC: 2_FSL_USDHC: 3
*** Warning - bad CRC or MMC, using default environment
In: serial
Out: serial
Err: serial
Net: got MAC address from IIM: 00:c0:08:88:9c:8d
FEC0 (PRIME)
Hit any key to stop autoboot: 1
```

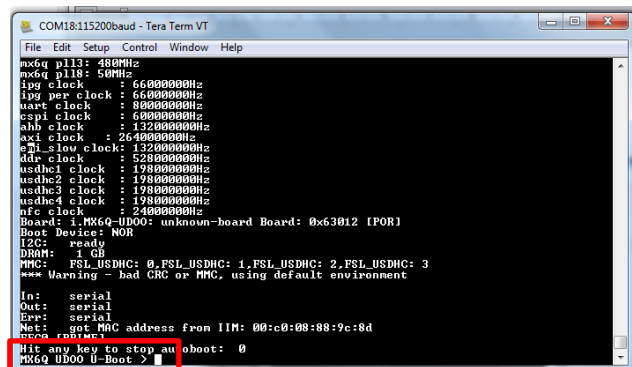
Figure 3: To stop autoboot

When the message, "Hit any key to stop autoboot:", quickly hit any key from keyboard like b key or z key to stop the kernel boot.

```
COM18:115200baud - Tera Term VT
File Edit Setup Control Window Help
Net: got MAC address from IIM: 00:c0:08:88:9c:8d
FEC0 (PRIME)
Hit any key to stop autoboot: 0
Kernel 2.0.0-00000 <4329640>
ramdisk 0 11800000 <497845>
kernel cmdline:
use uboot command line:
console=ttymxc1,115200 init=/init video=mx6fb0:dev=hdmi,1920x1080M@60,if=RGB
24,bpp=32 video=mx6fb1:off video=mx6fb2:off fbmem=28M vmlinux=400M androidboot.conso
le=ttymxc1 androidboot.hardware=freescale nwn=1024M
Starting kernel ...
Initializing cgroup subsys cpuset
Initializing cgroup subsys cpu
Linux version 3.0.35-sun73 (udoo@ubuntu) (gcc version 4.6.x-google 20120106 (prerele
ase) <GCC>) #2 SMP PREEMPT Thu Jan 22 02:43:19 PST 2015
CPU: ARMv7 Processor [412fc09a] revision 10 (ARMv7), cr=10c53c7d
CPU: UPT nonaliasing data cache, UPT aliasing instruction cache
Machine: i.MX6 UDOO Board
Ignoring unrecognized tag 0x41000901
Memory policy: ECC disabled, data cache writealloc
CPU identified as i.MX6Q, silicon rev 1.2
PERCPU: Embedded 7 pages/cpu @c14d5000 s5536 r8192 d14944 u32768
Built 1 zonelists in Zone order, mobility grouping on. Total pages: 223232
Kernel command line: console=ttymxc1,115200 init=/init video=mx6fb0:dev=hdmi,1920x10
80M@60,if=RGB24,bpp=32 video=mx6fb1:off video=mx6fb2:off fbmem=28M vmlinux=400M andr
oidboot.console=ttymxc1 androidboot.hardware=freescale nwn=1024M
PID hash table entries: 2048 (order: 1, 8192 bytes)
```

Figure 4: Starting kernel

If user is too slow, the kernel will start to boot (red rectangular box in Figure 4). Then the user needs to redo again. A successful stop will see output like Figure 5 below. Notice the prompt in the red rectangular box.



```

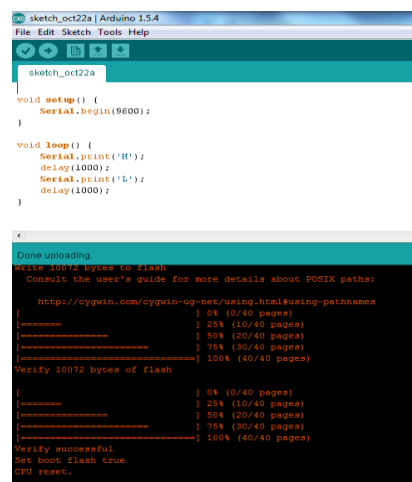
COM18:115200baud - Tera Term VT
File Edit Setup Control Window Help
mx6q pll1: 480MHz
mx6q pll18: 50MHz
lpg clock : 66000000Hz
lpg_per clock : 66000000Hz
uart clock : 8000000Hz
cspl clock : 60000000Hz
ahb clock : 132000000Hz
axi clock : 264000000Hz
apb_slow clock : 132000000Hz
drc clock : 528000000Hz
usdhc1 clock : 198000000Hz
usdhc2 clock : 198000000Hz
usdhc3 clock : 198000000Hz
usdhc4 clock : 198000000Hz
nrc clock : 240000000Hz
Board: 1.MX6Q-UD00: unknown-board Board: 0x63012 [POR]
Boot Device: NOR
I2C: ready
DRAM: 1 GB
MMC: FSL_USDHC: 0.FSL_USDHC: 1.FSL_USDHC: 2.FSL_USDHC: 3
*** Warning - bad CRC or MMC, using default environment
In: serial
Out: serial
Err: serial
Net: got MAC address from IIM: 00:c0:08:88:9c:8d
Hit any key to stop autoboot: 0
MX6Q UD00 U-Boot >

```

Figure 5: Kernel boot stopped

Close the serial monitor, for example Tera Term. Remove J18 jumper. Put the jumper J22 for one second and remove it. This is to erase the previous sketch. Do the same for jumper J16. This is to reset the SAM3X8E processor. This processor is used for Arduino Due sketch programming.

At Arduino IDE, select Tools -> Board -> Arduino Due (Programming Port). At Arduino IDE, select Tools -> Port -> select relevant port (for example COM18). At Arduino IDE, compile and upload the lab1_Due file which contains the arduino sketch. If successful, messages like in Figure 6 will be displayed.



```

sketch_oct22a | Arduino 1.5.4
File Edit Sketch Tools Help
sketch_oct22a
void setup() {
  Serial.begin(9600);
}

void loop() {
  Serial.print('H');
  delay(1000);
  Serial.print('L');
  delay(1000);
}

Done uploading
Write 10072 bytes to flash
Consult the user's guide for more details about POSIX paths:
http://cygwin.com/cygwin-ug-net/using.html#using-pathnames
[=====] 100% (40/40 pages)
Verify 10072 bytes of flash
[=====] 100% (40/40 pages)
Verify successful
Set boot flash true
CPU reset.

```

Figure 6: Successfully compile and upload sketch

Connect the XBee shield on top of the UDOO (Figure 7). Notice that there is a XBee module. A red light is seen when there is a communication with another XBee module.

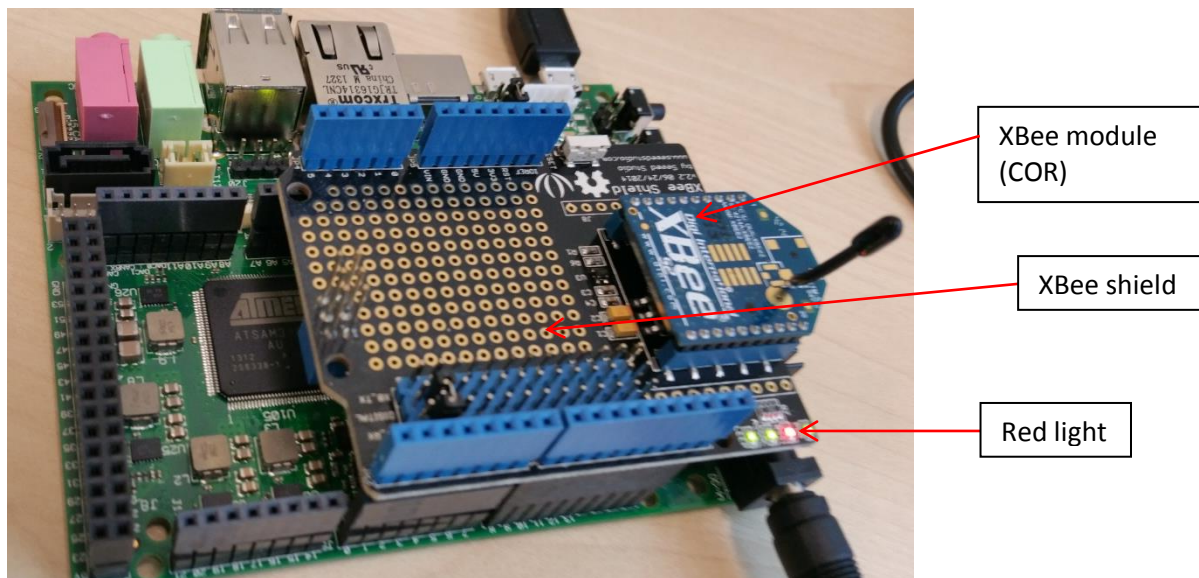


Figure 7: XBee shield on top of UDOO

Arduino UNO configuration

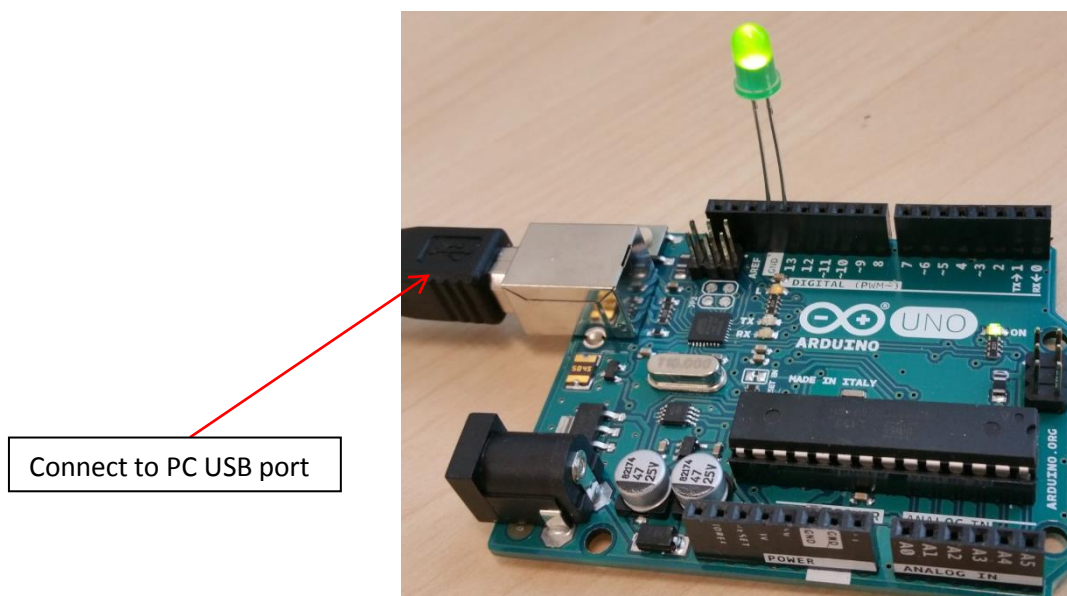


Figure 8: Arduino UNO

Connect the Arduino UNO circuit board to the USB port of a PC (Figure 8). This is to provide power supply and communication with the PC. Connect to power bank only provides power supply to Aduino UNO. Insert a LED (Figure 8) at Grd (shorter lead) and pin 13 (longer lead).

At Arduino IDE, select Tools -> Board -> Arduino Uno. At Arduino IDE, select Tools -> Port -> select relevant port (for example COM13). At Arduino IDE, compile and upload your file. The LED in Figure 8 should blink every 100 milliseconds.

Now, at Arduino IDE, compile and upload another file, lab1__1UNO file.

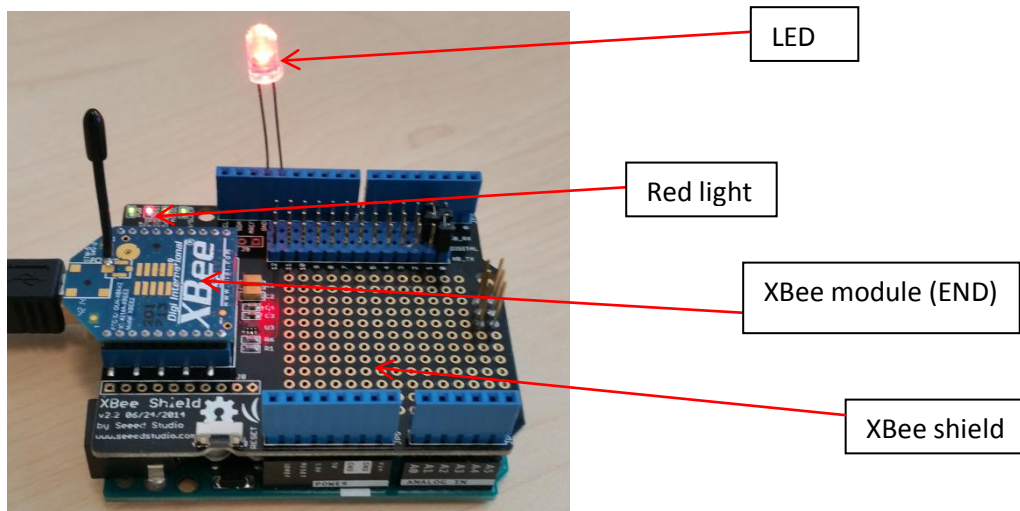


Figure 9: XBee shield above Arduino UNO

Connect the XBee shield on top of the Arduino UNO as shown in Figure 9. Notice that there is a XBee module. Insert a LED (Figure 9) at Grd (shorter lead) and pin 13 (longer lead).

A red light is seen when there is a communication with another XBee module.

The UDOO and Arduino UNO communication set-up should look like Figure 10.

A successful connection with the Arduino Due in UDOO will show messages like HLHLHLH ... (Figure 11) displayed in the serial monitor screen and the LED light will blink every 10 milliseconds.

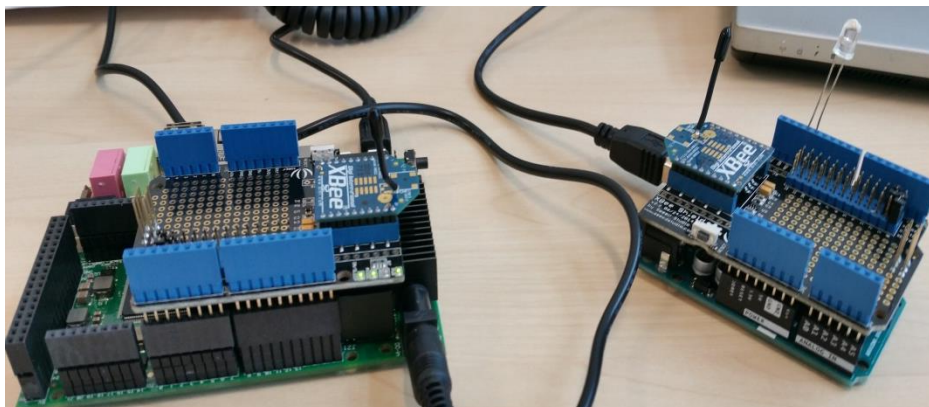


Figure 10: UDOO and Arduino UNO communication setup

