

Internet of Things (IOT) – lab 1

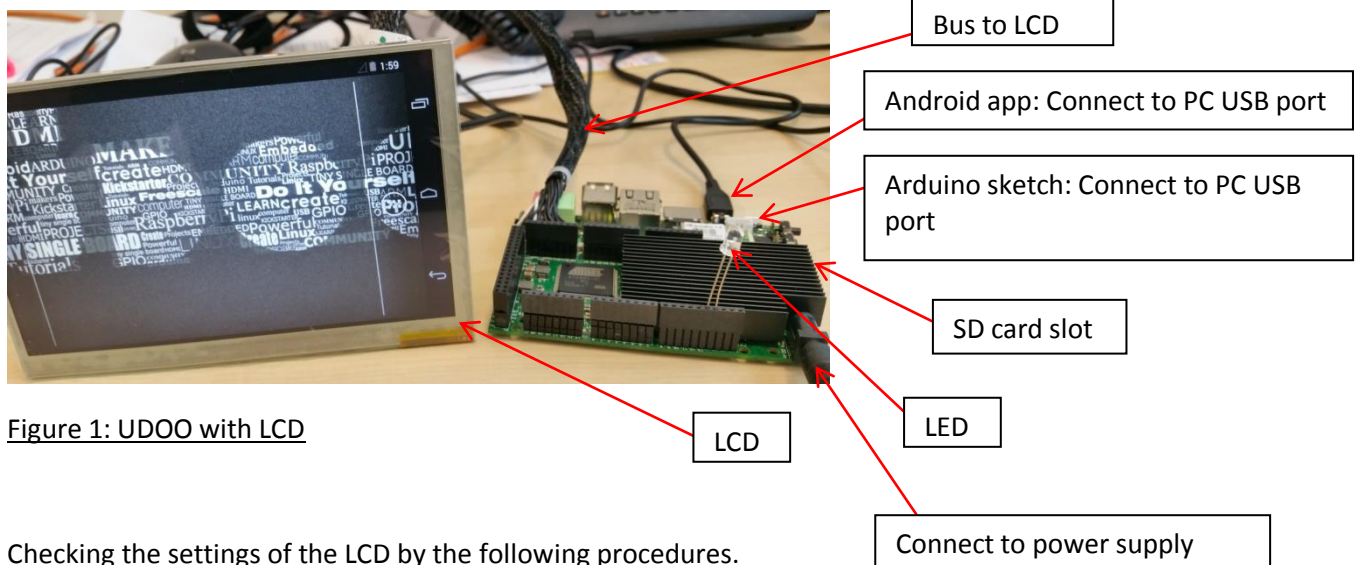
Objective:

- (1) Show the running of an Android application at UDOO
- (2) Demonstrate the LED blinking at UDOO

Android application in UDOO configuration

Insert the SD card into UDOO (Figure 1).

Perform the configuration as shown in Figure 1. Insert a LED (Figure 1) at Grd (shorter lead) and pin 13 (longer lead).



Checking the settings of the LCD by the following procedures.

Settings -> Display -> Sleep -> Select: After 30 minutes of inactivity

Settings -> Developer options -> Select: External OTG port enabled

Settings -> Developer options -> Select: USB debugging

Execute the given Android application, MyApp20, using Android Studio. A screen like Figure 2 will appear. Select the UDOO device. Click OK button. The screen at the LCD (Figure 3) appears.

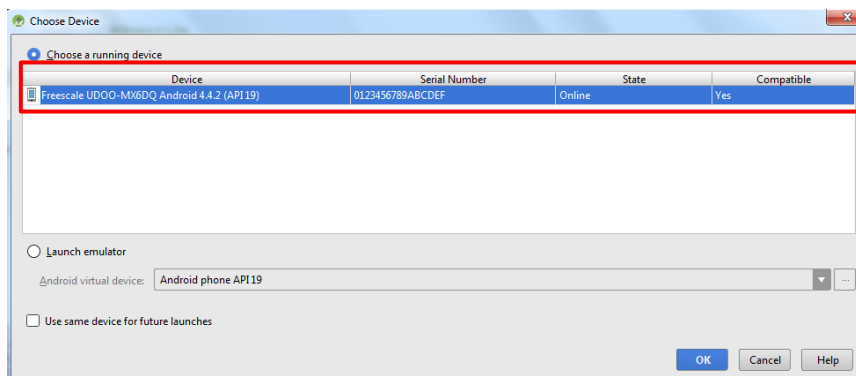


Figure 2: Running device



Figure 3: Android application and LED light

Try the following procedure if Android Studio is not able to detect the UDOO device, at the LCD drag down the Settings icon. Select the connected to a USB accessory option. Then select Camera (PTP).

Test the Android application by typing the statement, "This is a cat" . Then save the text into a file. The display statement, "File saved successfully!", is seen. After that, read the text from a file. The display statement, "This is a cat" is shown. Then another display statement, "File read successfully!" is presented.

Arduino Due in UDOO configuration

Download from the Internet Arduino IDE 1.5.4, Arduino IDE patch files and serial driver (Figure 4) if you have not done so. 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 .

Disconnect the microUSB cable for Android application and connect the microUSB cable for Arduino sketch (Figure 5).

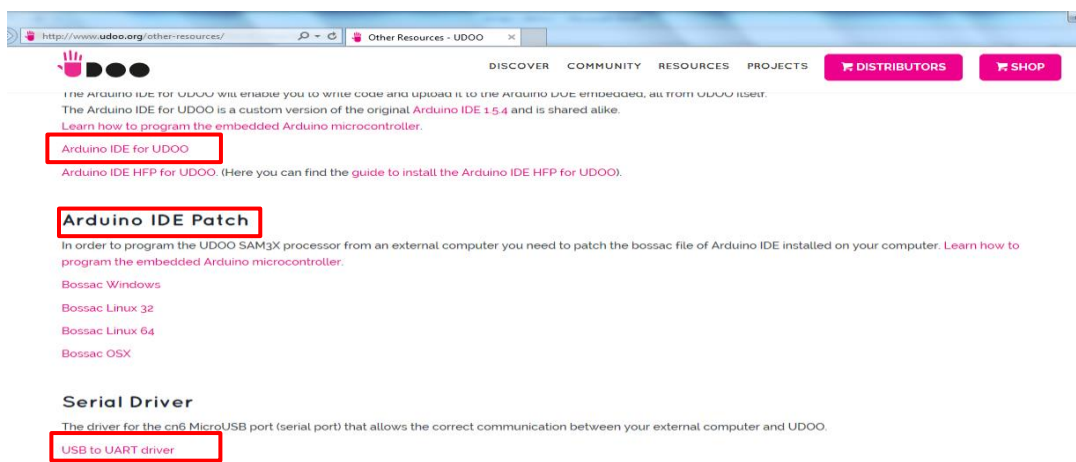


Figure 4: Arduino IDE, patch files and serial driver

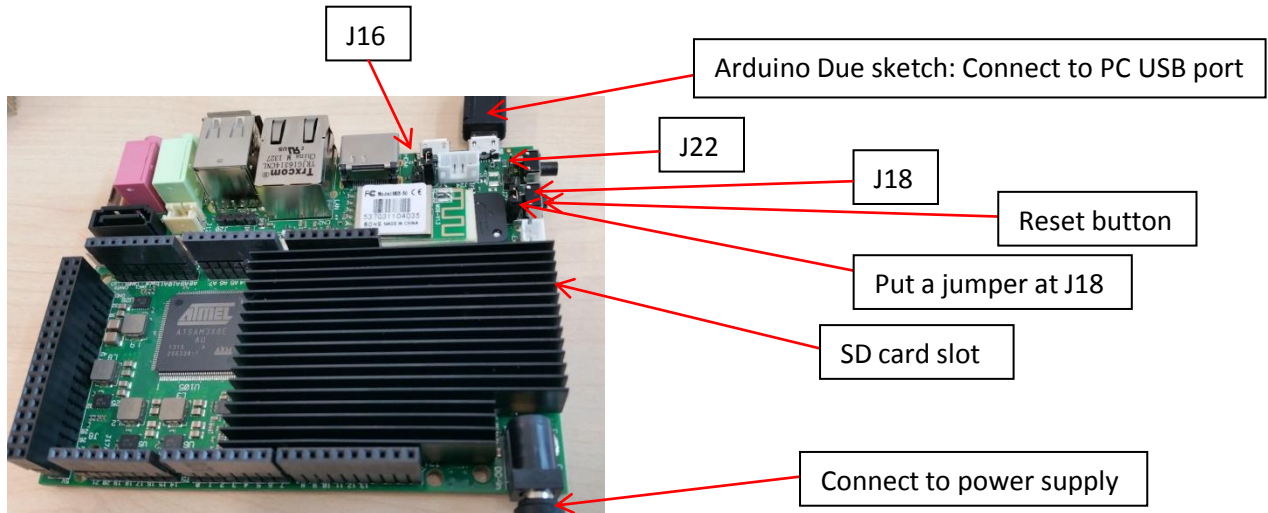


Figure 5: UDOO

If you have not done so, download a serial monitor software (Tera Term for Windows, serial tools for OS X, minicom for LINUX) from the Internet and installed the software. Put a jumper at J18 as shown in Figure 5. 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 6 will appear.

```
COM18:115200baud - Tera Term VT
File Edit Setup Control Window Help
mx6q pll2: 528MHz
mx6q pll3: 480MHz
mx6q pll8: 580MHz
lpg clock : 66000000Hz
lpg per clock : 66000000Hz
vot clock : 80000000Hz
cspi clock : 60000000Hz
shd clock : 132000000Hz
axi clock : 264000000Hz
eml_slow clock : 132000000Hz
drc clock : 52800000Hz
usdhc1 clock : 198000000Hz
usdhc2 clock : 198000000Hz
usdhc3 clock : 198000000Hz
usdhc4 clock : 198000000Hz
nfc clock : 24000000Hz
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
FE00 (PRIME)
Hit any key to stop autoboot: 1
```

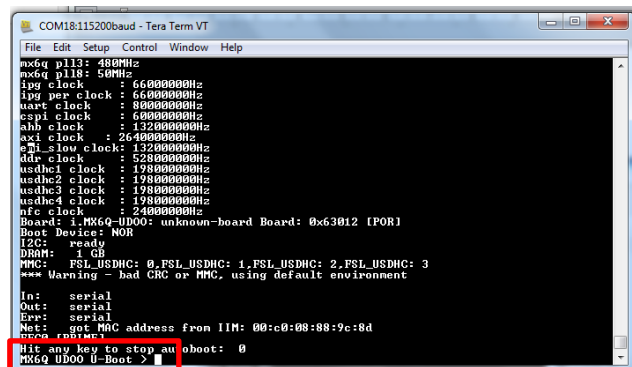
Figure 6: To stop autoboot

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
FE00 (PRIME)
Hit any key to stop autoboot: 0
Kernel: [0] 108000000 <4379648>
readlink 0 118000000 <4379648>
Kernel cmdline:
use uboot command line:
console=ttymxc1,115200 init=/init video=mx6fb0:dev=hdmi,1920x1080M660.if-RGB
24.hpp32 video=mx6fb1:off video=mx6fb2:off fmem=28M ualloc=400M androidboot.conso
le=ttymxc1 androidboot.hardware=freescale nwn-1024M
Starting kernel ...
[0] 108000000 <4379648>
Initializing cgroup subsys cpuset
Linux version 3.0.35-00000 (kudo@ubuntu) (gcc version 4.6.x-google 20120105 (prerelease)
) <GCC> > #2 SMP PREEMPT Thu Jan 22 02:43:19 PST 2015
CPU: ARMv7 Processor [412fc09a] revision 10 (ARMv7), cr=10c53c7d
CPU: DIT not implemented, data cache, DIT not implemented, instruction cache
Machine: i.MX6 UDOO Board
Ignoring unrecognised key 0x41000000
Memory policy: ECC disabled, Data cache writealloc
CPU identified as i.MX6Q, silicon rev 1.2
PERCPU: Embedded 7 pages/cpu @c1dc0000 s536 r8192 d14944 u32768
Built 1 zonelists in zone order, mobility grouping on. Total pages: 232332
Kernel command line: console=ttymxc1,115200 init=/init video=mx6fb0:dev=hdmi,1920x10
80M660.if-RGB24.hpp32 video=mx6fb1:off video=mx6fb2:off fmem=28M ualloc=400M andr
oidboot.console=ttymxc1 androidboot.hardware=freescale nwn-1024M
PID hash table entries: 2648 (order: 1, 8192 bytes)
```

Figure 7: Starting kernel

If user is too slow, the kernel will start to boot (red rectangular box in Figure 7). Then the user needs to redo again. A successful stop will see output like Figure 8 below. Notice the prompt in the red rectangular box.



```

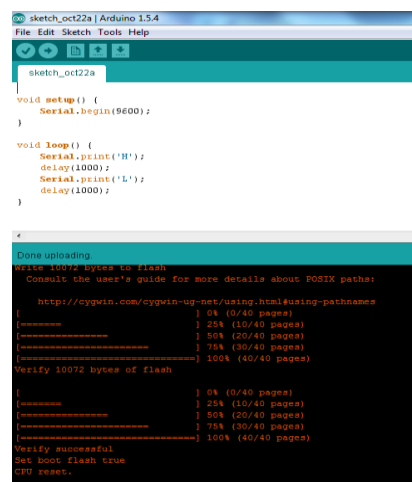
COM18:115200baud - Tera Term VT
File Edit Setup Control Window Help
ax6q pll13: 480MHz
ax6q pll18: 50MHz
ipg clock : 66000000Hz
ipg per clock : 66000000Hz
uart clock : 8000000Hz
cspl clock : 6000000Hz
ahb clock : 132000000Hz
axi clock : 264000000Hz
apl_slow clock : 132000000Hz
dly clock : 528000000Hz
usdhc1 clock : 198000000Hz
usdhc2 clock : 198000000Hz
usdhc3 clock : 198000000Hz
usdhc4 clock : 198000000Hz
nrc clock : 240000000Hz
Board: 1, MX5Q-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
MX5Q UD00 U-Boot >

```

Figure 8: 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 lab0_Due file which contains the arduino sketch. If successful, messages like in Figure 9 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
[Progress bar] 100% (40/40 pages)
Verify 10072 bytes of flash
[Progress bar] 100% (40/40 pages)
Verify successful
Set boot flash true
CPU reset.

```

Figure 9: Successfully compile and upload sketch

The LED light at UD00 will blink every second.

After you have finished, get a check off from one of the instructors or TA's by system demonstration.