

Practice work #2

At the first lesson we installed yocto-linux at our sd cards and tested how it works. At this lesson we are going to use library mraa for creating “hello world” and blink LED via linux, instead of Arduino. All links and source codes you can find on github:

- 1) **If you have not UART**, you will plug in Ethernet-cable and use this sketch: Use Putty at COM of Galileo. Write ifconfig and find ipv4 address.
- 2) Connect via Putty using ipv4 address or COM (if connected via UART).

We have connected to yocto and let's turn on WiFi.

- 3) Check you WiFi connection (use: ifconfig).
- 4) Connect to WiFi

```
root@quark0196c6:~# connmanctl
connmanctl> scan wifi
Scan completed for wifi
connmanctl> services
      kartoshka          wifi_0cd292960041_6b6172746f73686b61_managed_psk
                        wifi_0cd292960041_hidden_managed_psk
      Sea                wifi_0cd292960041_536561_managed_psk
      DIR-300NRUB6       wifi_0cd292960041_4449522d3330304e52554236_managed_psk
      mik-mik            wifi_0cd292960041_6d696b2d6d696b_managed_psk
      Don't Worry Papa  wifi_0cd292960041_446f6e277420576f7272792050617061_managed_psk
      MJ-0721            wifi_0cd292960041_4d4a2d30373231_managed_psk
      TP-LINK_AC470C     wifi_0cd292960041_54502d4c494e4b5f414334373043_managed_psk
      lolow              wifi_0cd292960041_6f6c6f6c6f77_managed_psk
      G est              wifi_0cd292960041_4720657374_managed_psk
      MacBook - Marina  wifi_0cd292960041_4d6163426f6f6b202d204d6172696e61_managed_wep
      DIRECT-Pn pass-PHILIPS TV wifi_0cd292960041_4449524543542d506e20706173732d5048494c4950532054
      1115                wifi_0cd292960041_31313135_managed_psk
      NETGEAR            wifi_0cd292960041_4e455447454152_managed_psk
      D&D                wifi_0cd292960041_442644_managed_psk
      MADBOYZ            wifi_0cd292960041_4d4144424f595a_managed_psk
      chikipidia-free    wifi_0cd292960041_6368696b6970696469612d66726565_managed_psk
      pokkydo            wifi_0cd292960041_706f6b6b79646f_managed_psk
      Sobaka@            wifi_0cd292960041_536f62616b6140_managed_psk
connmanctl> agent on
Agent registered
connmanctl> connect wifi_0cd292960041_6b6172746f73686b61_managed_psk
Agent RequestInput wifi_0cd292960041_6b6172746f73686b61_managed_psk
  Passphrase = [ Type=psk, Requirement=mandatory, Alternates=[ WPS ] ]
  WPS = [ Type=wpspin, Requirement=alternate ]
Connected wifi_0cd292960041_6b6172746f73686b61_managed_psk
Passphrase? nenuchoona?
connmanctl> [ 181.097379] wlp1s0: authenticate with 64:70:02:9c:71:86
[ 181.179607] wlp1s0: send auth to 64:70:02:9c:71:86 (try 1/3)
[ 181.188096] wlp1s0: authenticated
[ 181.230309] wlp1s0: associate with 64:70:02:9c:71:86 (try 1/3)
[ 181.260481] wlp1s0: RX AssocResp from 64:70:02:9c:71:86 (capab=0x431 status=0 aid=5)
[ 181.295012] wlp1s0: associated
[ 181.299796] IPv6: ADDRCONF(NETDEV_CHANGE): wlp1s0: link becomes ready
exit
root@quark0196c6:~#
```

Let us consider how to work with mraa library:

5) Firstly, update mraa:

```
echo "src maa-upm http://iotdk.intel.com/repos/1.1/intelgalactic" > /etc/opkg/intel-iotdk.conf
```

```
opkg update
```

```
opkg upgrade
```

6) Create txt file on your PC and give a name with extension .cpp . Copy the next code for hello world:

```
#include "mraa.h"
int main(int argc, char** argv)
{
    char* board_name = mraa_get_platform_name();
    fprintf(stdout, "hello mraa\n Version: %s\n Running on %s\n", mraa_get_version(),
    board_name);
    mraa_deinit();
    return MRAA_SUCCESS;
}
```

7) Upload your new file via pscp. How to use(windows):

```
C:\Program Files (x86)\Putty>pscp main.cpp root@192.168.0.105:/home/root/
The server's host key is not cached in the registry. You
have no guarantee that the server is the computer you
think it is.
The server's rsa2 key fingerprint is:
ssh-rsa 2048 9f:8c:35:0f:9e:70:4f:98:4f:11:6e:55:01:6c:70:b6
If you trust this host, enter 'y' to add the key to
PuTTY's cache and carry on connecting.
If you want to carry on connecting just once, without
adding the key to the cache, enter 'n'.
If you do not trust this host, press Return to abandon the
connection.
Store key in cache? (y/n) y
main.cpp          1 0 kB |   0.2 kB/s | ETA: 00:00:00 | 100%
```

8) For compile and execute you can use the following commands:

```
root@quark0196c6:~# make main.cpp
make: Nothing to be done for 'main.cpp'.
root@quark0196c6:~# g++ main.cpp -lmraa -o output.out
root@quark0196c6:~# ./output.out
hello mraa
Version: v0.6.1
Running on Intel Galileo Gen 2
```

9) Compile and execute code for Led blink (led.cpp).

10) Connect analog sensor, and using code sensor.cpp for get data from device.