Evaluation of RF technologies for Indoor Localization

Xuanjiao Zhu

September 5, 2018

Abstract

Location-based services have become an important market recently, and the market is growing rapidly. These services are generally based on identifying the physical location of individuals or objects. although GPS is typically used outdoors, it is not suitable for indoor localization. Several alternatives have been developed over the years, including vision-based and radio frequency (RF)-based technologies. We report on our experiments with two RF technologies (ANT [1] and Bluetooth Low Energy [2]) and their effectiveness in indoor environments.

1 Introduction

GPS(global positioning system), which uses signals of GPS satellites, is not a suitable solution of indoor localization. Instead, we are implementing a indoor localization system with two RF technologies: ANT and BLE. The system estimate the location of an object by using RSSI value. At the end, we discuss their performance in indoor localization.

2 RF technologies

There are varies of RF technologies. In this project we focus on two RF technologies: ANT and BLE.

2.1 ANT

ANT is a ultra low power wireless protocol. It is suited for multiple topologies: peer to peer, star, tree, mesh. It is a good solution for local area network(LAN). For example, smart home and automation Industries. ANT technology supports the use of any of the available 125 unique RF operating frequencies on 2.4Hz frequency band (2400MHz - 2524MHz)

2.2 BLE

BLE (Bluetooth Low Energy) is designed for low power network. It support peer to peer and broadcast, and mesh topologies. BLE operates on 2.4GHz frequency band. BLE uses frequency-hopping spread spectrum method. During the transmission, the radio signals switch among many frequency channel. The table shows a detailed comparison between ANT and BLE protocol on NRF module

Table 1: ANT and BLE comoarison

Description	BLE	ANT
Frequency range	2.400–2.483 GHz	2.00–2.483
Frequency	FHSS	Fixed frequency
Supported topologies	Peer-to-peer,star	Peer-to-peer star ,mesh
Band	ISM 2.4GHz	ISM 2.4GHz
Security	128-bit AES	64-bit key, 128 - bit AES
Data rate	2 Mbps, 1 Mbps, 500 kbps, 125 kbps	1 Mbps
Number of connections	Up to 20	Very high
Effective throughput	Up to 1.4 Mbps	Up to 60 kbps
Applications	Wearables, automation, sensors, fitness, healthcare, toys, computer peripherals, remote controls, etc.	Wireless sensors, hubs for sports, fitness, healthcare

3 Methodology

To localize an object, first we need to get the distance between object and beacons. The distance can be estimated by RSSI. In this project, I implemented two platform, which is used to measure RSSI based on two RF technologies, ANT and BLE.

3.1 ANT Setup

This section introduces the way to set up a platform for measuring RSSI. Including setup in the host PC, base station and beacons.

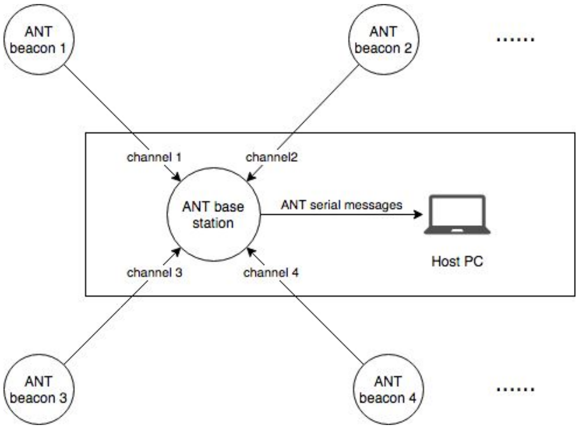


Figure 1: ANT Topology

3.1.1 ANT beacon

The ANT beacons are the masters which initiate communication. They are assigned to different frequency channel. The ANT beacon broadcasts its serial number and its channel ID with designated channel period (Figure 4). In this project, ANT beacons are ANT SoC ”D52QD2M6IA-A” (shown in figure 2) .

3.1.2 ANT base station

The ANT base station is the slave which receives the broadcast messages. It establish multiple channels (in this project maximal 8 channels). Each channel is assigned to one radio frequency and one beacon’s channel ID. This make sure itself listens to the broadcast of only one device. In this project, ANT base station is the ANT MCU (Figure / refANT base station listening)‘D52QD2M6IA‘ connected to host PC through USB interface board. For more detail please refer to section 3.2. Steps to set up ANT communication.

RSSI measuring in ANT base station
If ”Extend Message” is enabled, ANT base station will provide its host application with additional

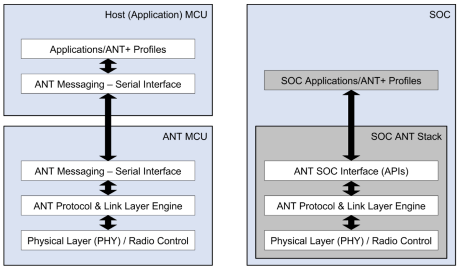

Host (Application) MCU
Applications/ANT+ Profiles
ANT Messaging – Serial Interface
ANT MCU
ANT Messaging – Serial Interface
ANT Protocol & Link Layer Engine
Physical Layer (PHY) / Radio Control
SOC
SOC Applications/ANT+ Profiles
SOC ANT Stack
ANT SOC Interface (APIs)
ANT Protocol & Link Layer Engine
Physical Layer (PHY) / Radio Control

Figure 2: ANT layers

information regarding the received data message. The additional information includes received signal strength indication (RSSI)(shown in the figure 3, step 8 receive messages).

Channel ID —— a 4 byte value that contains 3 fields: transmission type, device type and device number.

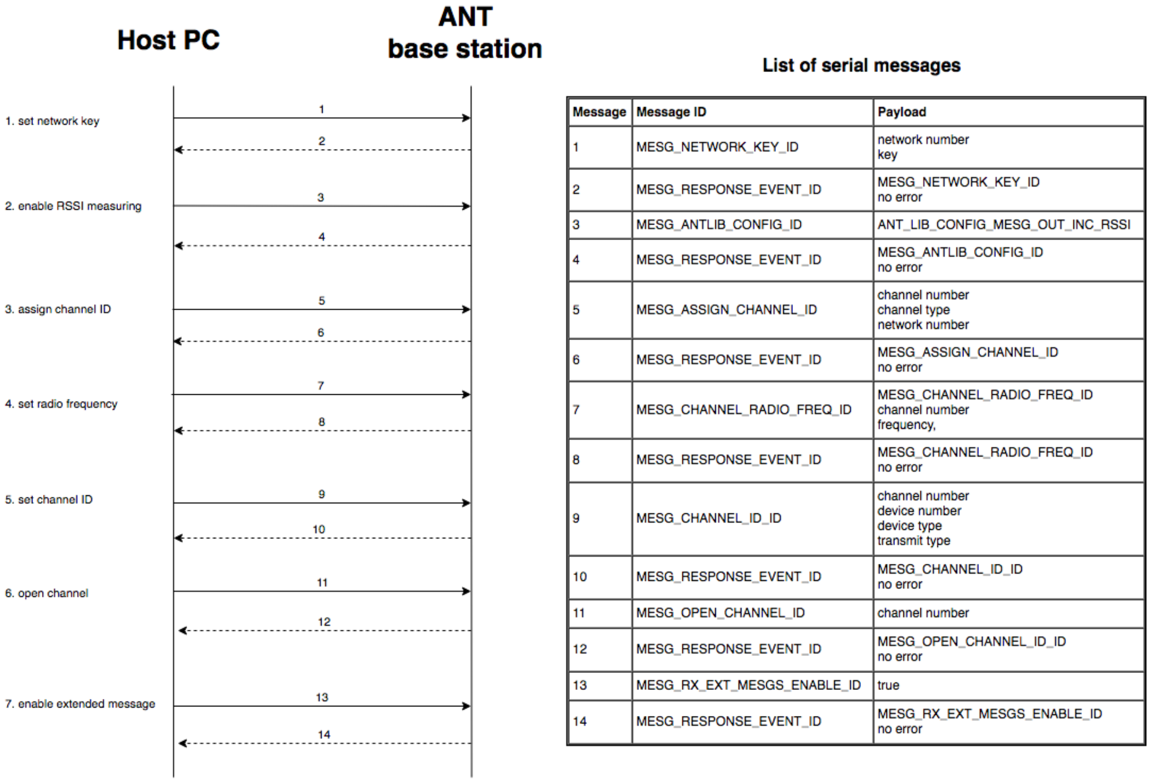
- Device Number
  - A 16-bit unique ID for ANT beacons.
  - The device numbers of ANT beacons are their serial numbers.
- Device Type
  - To define the type (or class) of device.
- Transmission Type
  - To define certain transmission characteristics of a device
- Radio frequency
  - RF frequency value = (desire frequency - 2400MHz)/ 1MHz

3.1.3 ANT channels

In this project we use synchronous, independent, bidirectional channels. In each ANT channel, only one master(transmitter) and one slave(receiver) is allowed. E.g. In the figure 1, channel 1 is establish between beacon 1 and base station.

Table 2: Serial message format: ANT base station to Host PC

Byte 1	Byte 2	Byte3 - N	Byte N + 1	Byte N + 2
Sync	Msg Length	Msg ID	Message Content	Checksum



List of serial messages

Message	Message ID	Payload
1	MESG_NETWORK_KEY_ID	network number key
2	MESG_RESPONSE_EVENT_ID	MESG_NETWORK_KEY_ID no error
3	MESG_ANTLIB_CONFIG_ID	ANT_LIB_CONFIG_MESG_OUT_INC_RSSI
4	MESG_RESPONSE_EVENT_ID	MESG_ANTLIB_CONFIG_ID no error
5	MESG_ASSIGN_CHANNEL_ID	channel number channel type network number
6	MESG_RESPONSE_EVENT_ID	MESG_ASSIGN_CHANNEL_ID no error
7	MESG_CHANNEL_RADIO_FREQ_ID	MESG_CHANNEL_RADIO_FREQ_ID channel number frequency,
8	MESG_RESPONSE_EVENT_ID	MESG_CHANNEL_RADIO_FREQ_ID no error
9	MESG_CHANNEL_ID_ID	channel number device number device type transmit type
10	MESG_RESPONSE_EVENT_ID	MESG_CHANNEL_ID_ID no error
11	MESG_OPEN_CHANNEL_ID	channel number
12	MESG_RESPONSE_EVENT_ID	MESG_OPEN_CHANNEL_ID_ID no error
13	MESG_RX_EXT_MESGS_ENABLE_ID	true
14	MESG_RESPONSE_EVENT_ID	MESG_RX_EXT_MESGS_ENABLE_ID no error

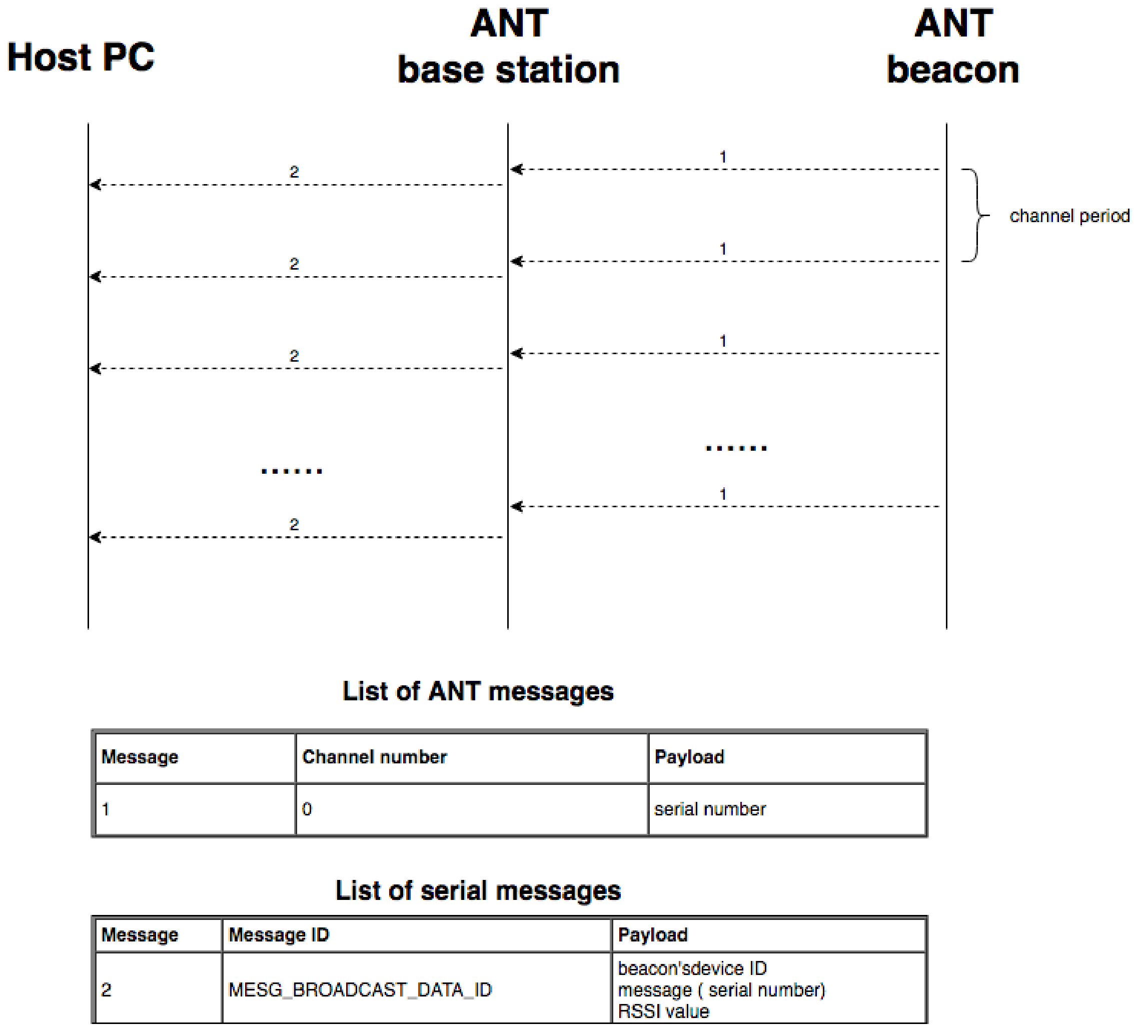
Figure 3: ANT base station configure

Table 3: Message format: ANT beacon to ANT base station

Byte 0	Byte 1	Byte 2	Byte 3	Byte4	Byte 5 - 12	Byte 13
Sync	Msg Length	Msg ID	Channel Number	Payload (serial number)	Flag (0xC0)	Device Number
Byte 14	Byte 15	Byte 16	Byte 17	Byte 18	Byte 19	
Device Type	Transmission Type	Measurement Type	RSSI value	Threshold Configuration Value	Check sum	

3.1.4 Step 1: Implement GUI application for host PC

1. Download ANT PC SDK
2. Open ANT libraries solution in Microsoft Vitual Studio 2017.
3. Build the solution and get library files ‘ANT_LIB.lib‘,‘DSI_SiUSBXp_3_1.dll‘,‘DSI_SiUSBXp7_3_1.dll‘.



List of ANT messages

Message	Channel number	Payload
1	0	serial number

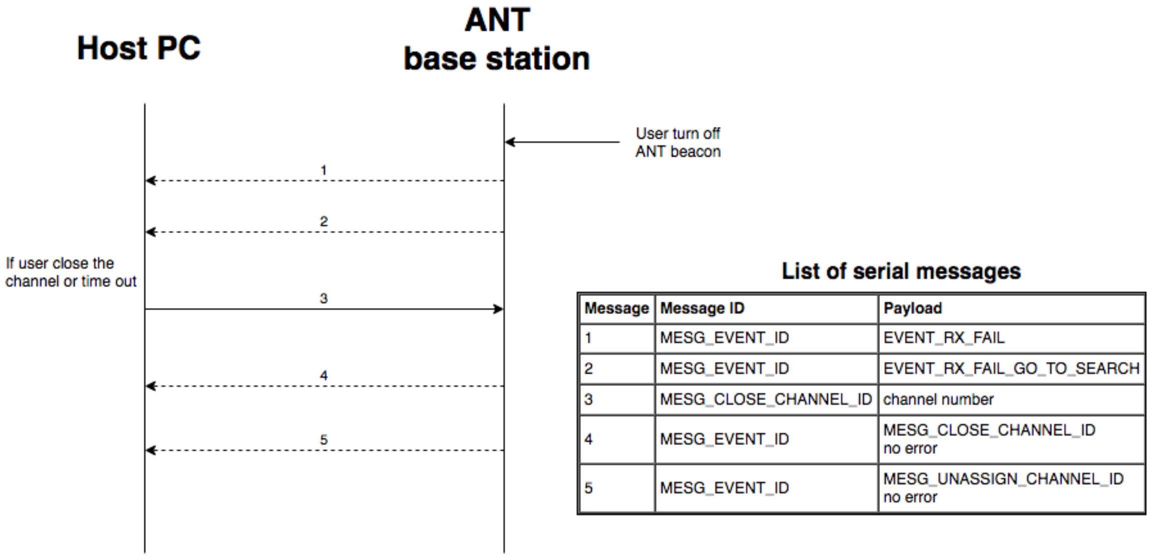
List of serial messages

Message	Message ID	Payload
2	MESG_BROADCAST_DATA_ID	beacon'sdevice ID message (serial number) RSSI value

Figure 4: ANT base station listening

4. Other library file: `C:\ProgramFiles(x86)\WindowsKits\10\Lib\10.0.17134.0\um\x86\User32.Lib`
5. Implement an application on QT Using these library files. The structure of the application is shown in figure 7.
6. Deploy the applicaion to a .exe file.

To see how to use ANT GUI application, please refer to 3.2.5. Step 4: Measure the RSSI value.



List of serial messages

Message	Message ID	Payload
1	MESG_EVENT_ID	EVENT_RX_FAIL
2	MESG_EVENT_ID	EVENT_RX_FAIL_GO_TO_SEARCH
3	MESG_CLOSE_CHANNEL_ID	channel number
4	MESG_EVENT_ID	MESG_CLOSE_CHANNEL_ID no error
5	MESG_EVENT_ID	MESG_UNASSIGN_CHANNEL_ID no error

Figure 5: ANT base station close channel

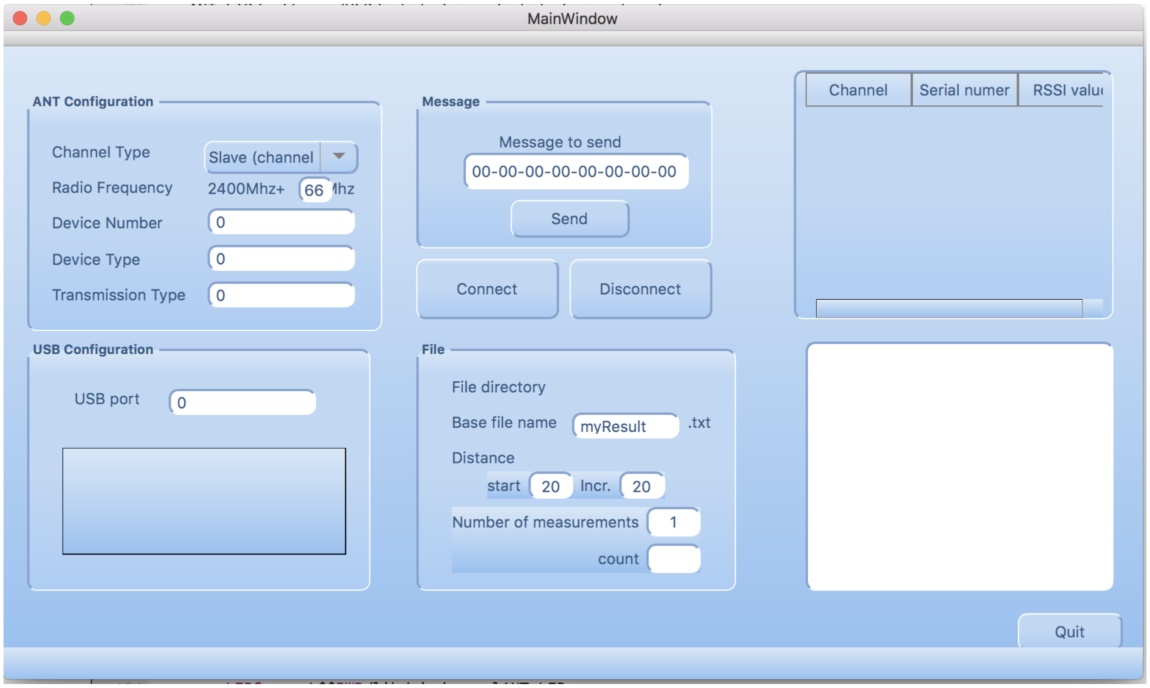
3.1.5 Step 2: Choose proper soft device for ANT beacons

Please refer to IC revisions, SDK, and SoftDevice compatibility matrix for nRF52832. link: `http://infocenter.nordicsemi.com/index.jsp?topic=%2Fcom.nordic.infocenter.nrf52%2Fdita%2Fnrf52%2Fnrf52_comp_matrix.html`
To make sure the ANT and BLE application based on same soft device, we use SoftDevice S332 v5.0.0 and nRF5 SDK 14.2.0.

3.1.6 Step 3: Implement application with SDK for ANT beacons

The following step shows how to implement a ANT application using Keil.

1. Download NordicSemiconductor.nRF_DeviceFamilyPack.8.17.0 and install it.
2. Connect a ANT SoC(D52QD2M6IA-A) to nRF 52 DK. And connect nRF52 DK to the host PC.
3. If any SDK is missing, SDK install window will automatically pop out. User can install all SDK packets for nRF52 DK.
4. Create a new project called ANT beacon project.
5. Go to Project menu, choose Option for Target... and configure you target as figure 8 and figure 9.
6. Write program for ANT beacon project according to the work flow (figure 10) and compile it



ANT Configuration

Channel Type: Slave (channel

Radio Frequency: 2400Mhz+ 66 Mhz

Device Number: 0

Device Type: 0

Transmission Type: 0

USB Configuration

USB port: 0

Message

Message to send: 00-00-00-00-00-00-00-00

File

File directory

Base file name: myResult .txt

Distance

start 20 Incr. 20

Number of measurements: 1

count

Channel	Serial numer	RSSI valu

Figure 6: ANT GUI window

3.1.7 Step 4: Measure the RSSI value

The following step shows how to measure RSSI value for one ANT beacon in different distance.

1. Preparation on ANT beacons
   (a) Connect a ANT SoC(D52QD2M6IA-A) to nRF 52 DK. And connect nRF52 DK to the host PC.
   (b) Open application ”nRF studio” in host pc and flash ANT_s332_nrf52_5.0.0.hex(see download link in software list) on the beacon(SoC).
   (c) Go to application tab and flash application .hex file(found in ”_build” folder of Keil project) on the beacons.
      - Another option: open Keil project and click ”Load” icon to flash the ANT beacon project
   (d) Place an ANT beacon in desired start distance.
2. Preparation on host PC The following step shows how to Compiling and flashing the project onto the DK

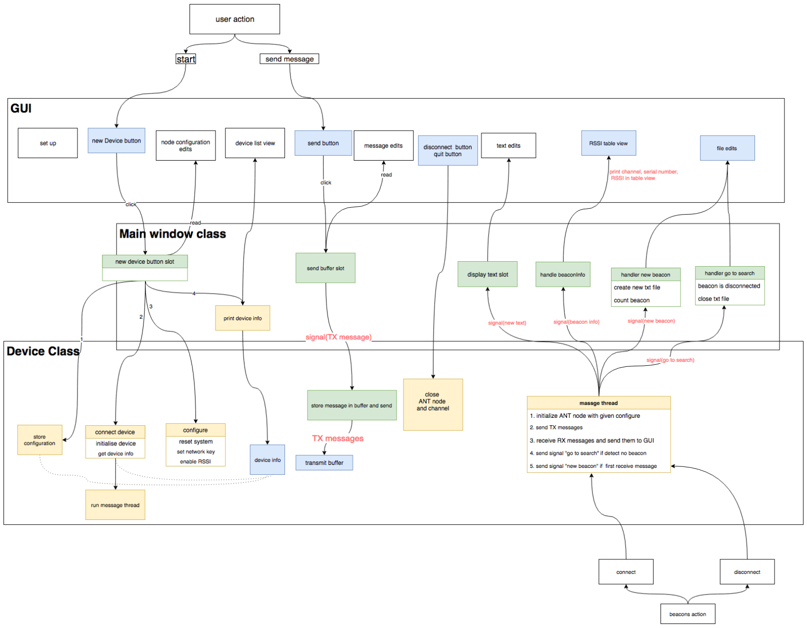
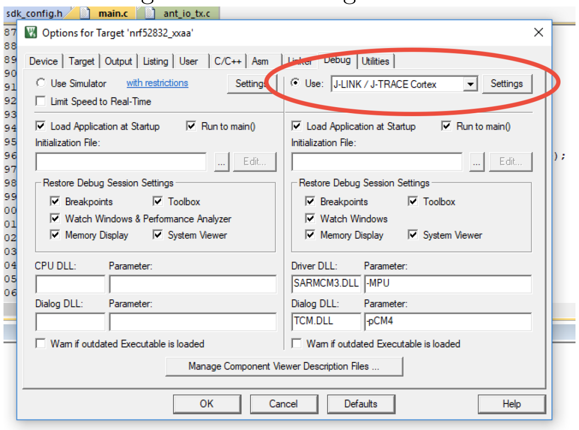


Figure 7: ANT GUI application structure

Figure 8: Keil configuration 1



○ Use Simulator with restrictions

□ Limit Speed to Real-Time

☑ Load Application at Startup ☑ Run to main()

Initialization File:

Restore Debug Session Settings

☑ Breakpoints ☑ Toolbox

☑ Watch Windows & Performance Analyzer

☑ Memory Display ☑ System Viewer

CPU DLL: Parameter:

Dialog DLL: Parameter:

□ Warn if outdated Executable is loaded

● Use: J-LINK / J-TRACE Cortex

☑ Load Application at Startup ☑ Run to main()

Initialization File:

Restore Debug Session Settings

☑ Breakpoints ☑ Toolbox

☑ Watch Windows

☑ Memory Display ☑ System Viewer

Driver DLL: SARMCM3.DLL Parameter: -MPU

Dialog DLL: TCM.DLL Parameter: -pCM4

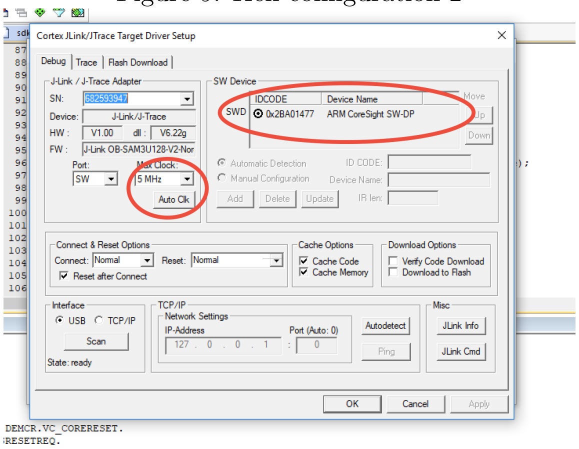
□ Warn if outdated Executable is loaded

(a) Connect a ANT SoC ”D52QD2M6IA” to nRF 52 DK. And connect nRF52 DK to the host PC.

(b) go to soft device tab and flash ANT soft device ANT_s332_nrf52_5.0.0.hex on the ANT base station(SoC).

(c) Download Network Processor Source Code. Build this project and get ”ant_network_processor_s332.hex”.

Figure 9: Keil configuration 2



Cortex JLink/JTrace Target Driver Setup

J-Link / J-Trace Adapter
SN: 682593947
Device: J-Link/J-Trace
HW : V1.00 dll : V6.22g
FW : J-Link OB-SAM3U128-V2-Nor
Port: SW
Max Clock: 5 MHz

SW Device

	IDCODE	Device Name
SWD	⦿ 0x2BA01477	ARM CoreSight SW-DP

⦿ Automatic Detection
○ Manual Configuration
ID CODE:
Device Name:
IR len:

Connect & Reset Options
Connect: Normal Reset: Normal
☑ Reset after Connect

Cache Options
☑ Cache Code
☑ Cache Memory

Download Options
☐ Verify Code Download
☐ Download to Flash

Interface
⦿ USB ○ TCP/IP
State: ready

TCP/IP
Network Settings
IP-Address: 127 . 0 . 0 . 1
Port (Auto: 0): 0

```
DEMCR.VC_CORERESET.
RESETREQ.
```

(d) Open application "nRF studio" in host PC

(e) Go to application tab and flash application .hex file "ANT network proccessor S332" The following step shows how set up GUI applicaion.

(f) Open ANT GUI application on the host PC

(g) Enter the configuration of the ANT beacons (Table 4). The channel type should be "Slave (channel 0 only)"

(h) Enter the base "file name" (e.g.myResult), "start distance" and "number of measurements".

3. Start measuring

   (a) Turn on the ANT beacon.

   (b) Click "Connect" bottom in GUI application.

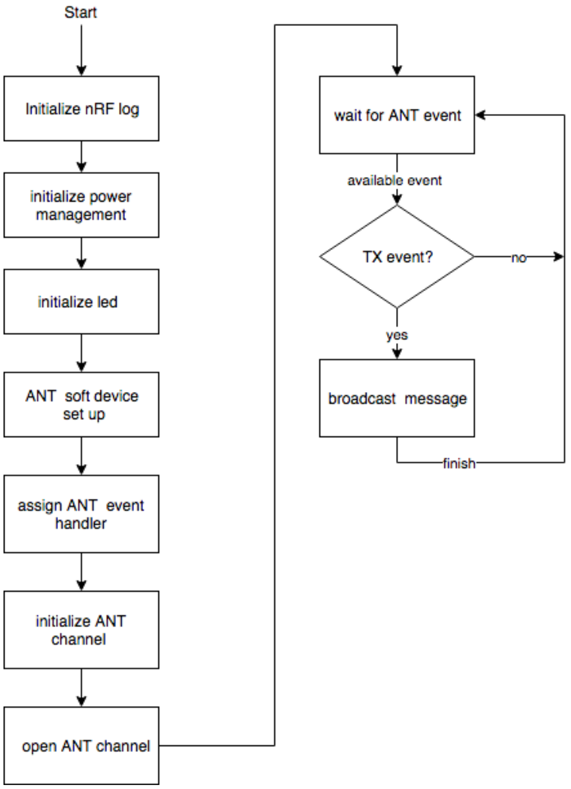
   (c) If an ANT base station is recognized, the VID and PID of the ANT base station will be shown in the text view. In addition, the received messages will be shown in Table view and text view. The "count" value should be 1. The measuring result will be stored in a txt file.

      - Txt file: application path/file/base name_start distance.txt
      - E.g. application folder/file/myResult_20.txt

   (d) Turn off the ANT ANT beacon. The base station will recognize its absence and the text view will print "RX Fail" and "Go to Search". if reach maximal number of measurement, the application will quit.

   (e) Move ANT beacon to next destance (start distance + Turn on the ANT beacon after "Go to search" is printed. The count value will puls 1. Now the measuring result will be store in a new txt tile.

Figure 10: ANT beacon workflow



- New txt file: application path/file/base name_start distance + increment.txt
- E.g. application folder/file/myResult_40.txt

(f) Repeat step 5-6

Table 4: Default ANT GUI Application configuration for single channel base station

Parameter	Value	Description
Channel type	Slave (multi channel)	Each channel listens to one beacon
	Slave (single channel)	to listen to one beacon
Device number	0	wildcard
Device Type	2	same as ANT beacon's device type
Transmission Type	5	same as ANT beacon's transmission type
Radio frequency	50	desire frequency = 2450MHz
USB port	0	First USB device connected to PC

Figure 11: BLE topology

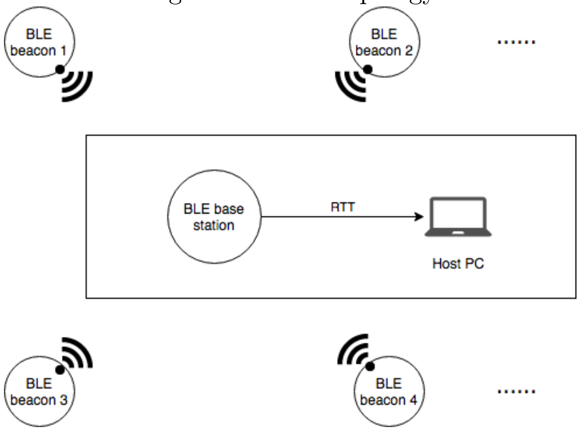
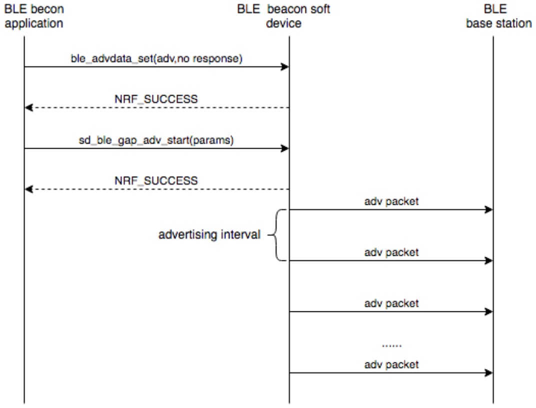


Figure 12: BLE beacon diagram



3.2 BLE Setup

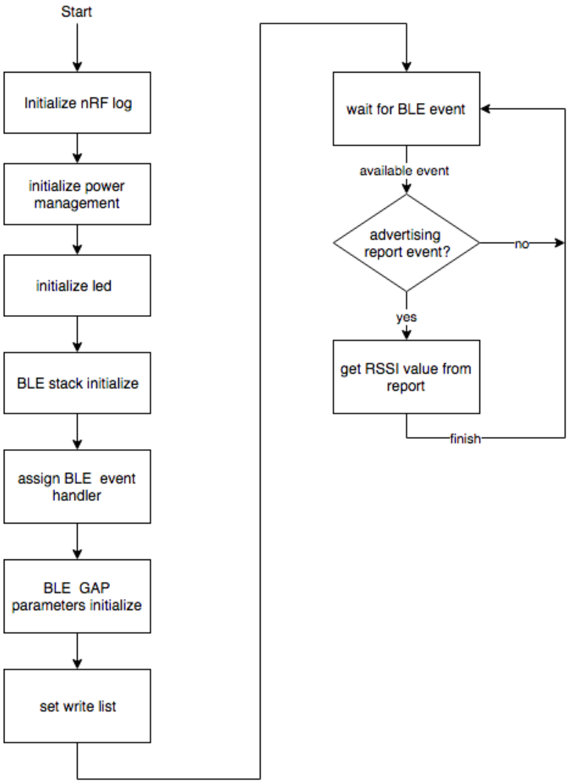
The BLE topology is shown in figure 11. the BLE beacons are broadcasters and the BLE base station is an observer. The observer listens to what's happening on the air. The broadcasters advertise but don't receive advertisement. They only use advertising but never establish a connection.

3.2.1 BLE beacons

The BLE beacons are broadcasters. they broadcasts its GAP name every time interval, known as the advertising interval. In this project, the BLE beacons are ANT SoC "D52QD2M6IA-A".

GAP (Generic Access Profile): GAP is the lowest layer of the Bluetooth stack that an application interfaces with. It includes parameter that govern advertising and connection. e.g.

Figure 13: BLE base station workflow



name, address.

3.2.2 Step 1: Implement application for BLE base staion

The BLE base station is nrf52 DK. The application is implemented on Keil. The Keil project configuration is the same as ANT project (figure 8 and figure 9). The workflow of application on BLE base station is shown in figure 13.

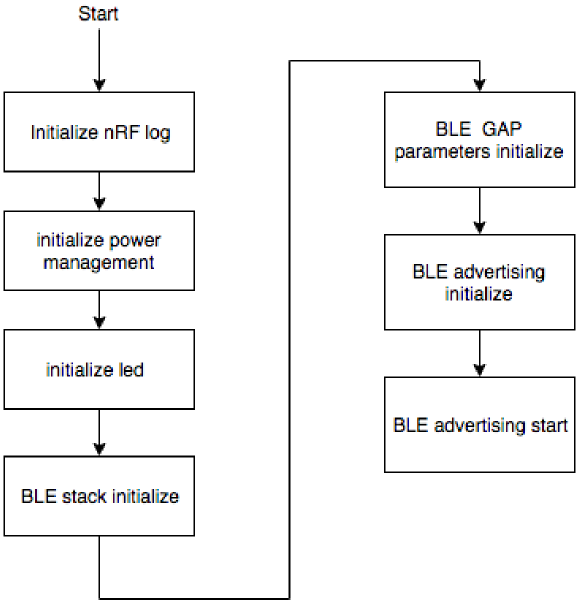
3.2.3 Step 2: Implement application with SDK for BLE beacons

The BLEbeacon is SoC (D52QD2M6IA-A). The application is implemented on Keil. The Keil project configuration is the same as ANT project (figure 9 and figure 10). The workflow of applicaton on BLE beacon is shown in figure 14.

3.2.4 Step 3: Inplement application for host PC

this part is uncertain now because I may modify the application on host PC

Figure 14: BLE beacon workflow


Start
Initialize nRF log
initialize power management
initialize led
BLE stack initialize
BLE GAP parameters initialize
BLE advertising initialize
BLE advertising start

3.2.5 Step 4: Measure the RSSI value

The following step shows how to measure RSSI value for one BLE beacon in different distance.

1. Preparation on BLE beacon
   (a) Connect a ANT SoC(D52QD2M6IA-A) to nRF 52 DK. And connect nRF52 DK to the host PC.
   (b) Open application "nRF studio" in host pc and flash "ANT_s33_nrf52_5.0.0.hex" on the beacon(SoC).
   (c) Go to application tab and flash application .hex file(found in "_build" folder of Keil project) on the beacons.
       - another option: open Keil project and click "Load" icon to flash the BLE beacon

   project
   (d) Place an BLE beacon in desired start distance.
2. Preparation on host PC

   The following step shows how to Compiling and flashing the project onto the DK.

   (a) Connect nRF52 DK to the host PC.
   (b) go to soft device tab and flash soft device "ANT_s332_nrf52_5.0.0.hex" on the nRF52 DK.

    (c) go to application tab and flash application .hex file(found in ”_build” folder of Keil project) on the BLE beacons.
    (d) another option: open Keil project and click ”Load” icon to flash the BLE beacon project

3. Start measuring *this part is uncertain now because I may modify the application on host PC*

4 Evaluation

Present the measurements in graphs and elaborate

5 Conclusions and Future Work

Summarize your findings and elaborate on what else can be done

References

[1] ANT Network. `https://en.wikipedia.org/wiki/ANT_(network)`. Accessed: 2018-08-10.

[2] Bluetooth (BR) and Bluetooth Low Energy (BLE). `https://www.bluetooth.com`. Accessed: 2018-08-10.

6 Appendix

The following software and hardware are used in the project:

A Software

- Mac OS 10.13
- Virtualbox Windows 10 virtual machine
- nRFgo Studio: a GUI applicaton for programming nRF5x SoftDevices, applications, and bootloaders.
  Download here: `https://www.nordicsemi.com/eng/Products/Bluetooth-low-energy`
- QT creater4.7 and QT : a complete cross-platform development framework and IDE
  Dowload here: `https://www.qt.io/download`
- ANTware: an ANT Desktop applications used for testing, debugging and configuring
  Download here: `https://www.thisisant.com/developer/resources/downloads/#software_tab`
- ARM Keil MDK 5: software develop solution for ARM based microcontrollers.
  Download here: `http://www2.keil.com/mdk5`

- ANT USB Interface Board Driver - Windows 1.2.40.201: driver for the USB interface board in developer kit.
  Download here: https://www.thisisant.com/developer/resources/downloads/#software_tab

- Microsoft Visual Studio 2017 Full-featured IDE for Windows.
  Download here: https://visualstudio.microsoft.com/downloads/

- Network Processor Source Code source code for ANT network processor firmware. it can build network processor firmwares for SoftDevices S212/S332, which is the application firmware for ANT base station.
  Download here: https://www.thisisant.com/developer/resources/downloads/

- soft device: ANT_s332_nrf52_5.0.0.hex BLE and ANT stack solution
  Download here: https://www.thisisant.com/developer/components/nrf52832#tab_protocol_stacks_tab

- nrf SDK 14: Source code of examples and libraries for application developing on SoC.
  Download here: https://developer.nordicsemi.com/nRF5_SDK/nRF5_SDK_v14.x.x/

- ANT PC SDK 3.5: Source code of examples and libraries for Windows application developing.

- ANT Mac SDK 3.5: Source code of examples and libraries for Mac application developing.
  Download ANT SDKs here: https://www.thisisant.com/developer/resources/downloads/

- nRF5x MDK for Keil4: nRF5x MDK for Keil4 and Keil5 compatibility version with 3-clause BSD license https://www.nordicsemi.com/eng/Products/Bluetooth-low-energy/nRF52832

B Hardware

- nRF52 DK: BLE base station

- ANT/BLE SoC Module: D52QD2M6IA-A: BLE and ANTbeacons

- ANT SoC Module: D52QD2M6IA: ANT base station

- ANTUIF1: USB interface board with a Molex socket: used to connect ANT base station with host PC.

- ANTBAT3: Combined battery and I/O board with a Molex socket, a reset button, I/O. it is used to provide power for ANT beacons.

- ANTUSB-M: ANT USB dongle