## Hardware

### Overview

The system consists of three main devices: tags, detectors and proxy. Tag is attached to the tracking asset and broadcast its data to detectors. Detectors are on fixed locations, they whether receive data from tag or from another detector. All data will be relayed toward proxy by detectors. For system mobility, proxy is a standalone module that can connects to local wireless LAN and relays data to back-end system. Since the system requires large range of tracking, tags, detectors and proxy are active broadcasting devices and need to be powered from power source.

Front-end system devices use RF technologies for communicating and locating. Data on the front-end system are devices info which are used for device identifications and Received Signal Strength Indication (RSSI) (measured by detector as receiving a broadcast from a tag) used for locating method. The hardware are designed to give reliable signal strength reading as well as data transferring.

### Component Selections

These are selected carefully to make sure they meet these main requirements: small size, low power consumption and low price. Also, it is important that they are available and well-supported.

|  |  |  |  |
| --- | --- | --- | --- |
| D:\Study\Capstone Proj\Docs\Photos\09582-08.jpg | D:\Study\Capstone Proj\Docs\Photos\download.jpg | D:\Study\Capstone Proj\Docs\Photos\ATMegaDIP.jpg | D:\Study\Capstone Proj\Docs\Photos\328PQFP.jpg |
| **RFM12B-S2 Wireless Transceiver Module** | **WiFly GSX Breakout** | **ATMega328P – 28 DIP Pinout** | **ATMega32P – 32 TQFP Pinout** |

Refer to datasheets provided by manufacturers for more information (See **Useful Links** section in **Appendices**).

### Tag

An asset tag consists of two major parts:

* Microcontroller: ATMega328P – 32 TQFP
* RF transceiver: RFM12B-S2 Wireless Transceiver Module

Table 1: Tag Specifications

Table 2: Electrical Characteristics

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Specification** | **Min** | **Typ** | **Max** | **Unit** |
| Input voltage | 3.3 | 5.7 | 6.0 | V |
| Operating voltage | 3.2 | 3.3 |  | V |
| Transmit Current |  | 30 |  | mA |
| Receive Current |  | 30 |  | mA |
| Idle Current |  | 40 |  | uA |
| Power-down Current |  | 2 |  | uA |
| Power Dissipation(transmit/receive) |  | 99 |  | mW |
| Operating Temperature | -25 |  | +85 | 0C |

The microcontroller controls the RF transceiver. The RF transceiver is responsible for communicating information to the system.

Figure 6: Board layout of the controller module on a tag

Figure 7: Board layout of the RF module on a tag

### Detector

A detector consists of two major parts:

* Microcontroller: ATMega328P – 28 DIP
* RF transceiver: RFM12B-S2 Wireless TransceiverTable 3: Detector Specifications

|  |  |
| --- | --- |
| **Specification** | **Description** |
| Microcontroller | ATMega328P – 28 DIP |
| RF Transceiver | RFM12B-S2 Wireless Transceiver |
| Voltage regulator | MIC5205 150mA Low-Noise LDO Regulator |
| Power supply | 9V Battery |
| Program Interface | ISP 6-pin header |
| Operating Frequency Band | ISM 433 MHz |
| Power Indicator | Yes (regular red LED) |
| RF Indicator | Yes (green LED) |
| PCB size | 25.4mm x 88.9mm |
| Outline Dimension |  |

Table 4: Electrical Characteristics

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Specification** | **Min** | **Typ** | **Max** | **Unit** |
| Input voltage | 3.3 | 9.0 | 16.0 | V |
| Operating voltage | 3.2 | 3.3 |  | V |
| Transmit Current |  | 30 |  | mA |
| Receive Current |  | 30 |  | mA |
| Idle Current |  | 30 |  | mA |
| Power-down Current |  | 2 |  | uA |
| Power Dissipation |  | 99 |  | mW |
| Operating Temperature | -25 |  | +85 | 0C |

Figure 8: Board layout of a detector

**Programming firmware:**

1. What you need:

AVRISP MK II (recommended)



AVR studio

<http://www.atmel.com/dyn/products/tools_card.asp?tool_id=2725&category_id=163&family_id=607&subfamily_id=760>

Arduino environment

<http://www.arduino.cc/en/Main/Software>

rf12 library (modified version, support RSSI analog reading), can be found in Firmware/arduinolibs

<http://code.google.com/p/tiu-tracking/>

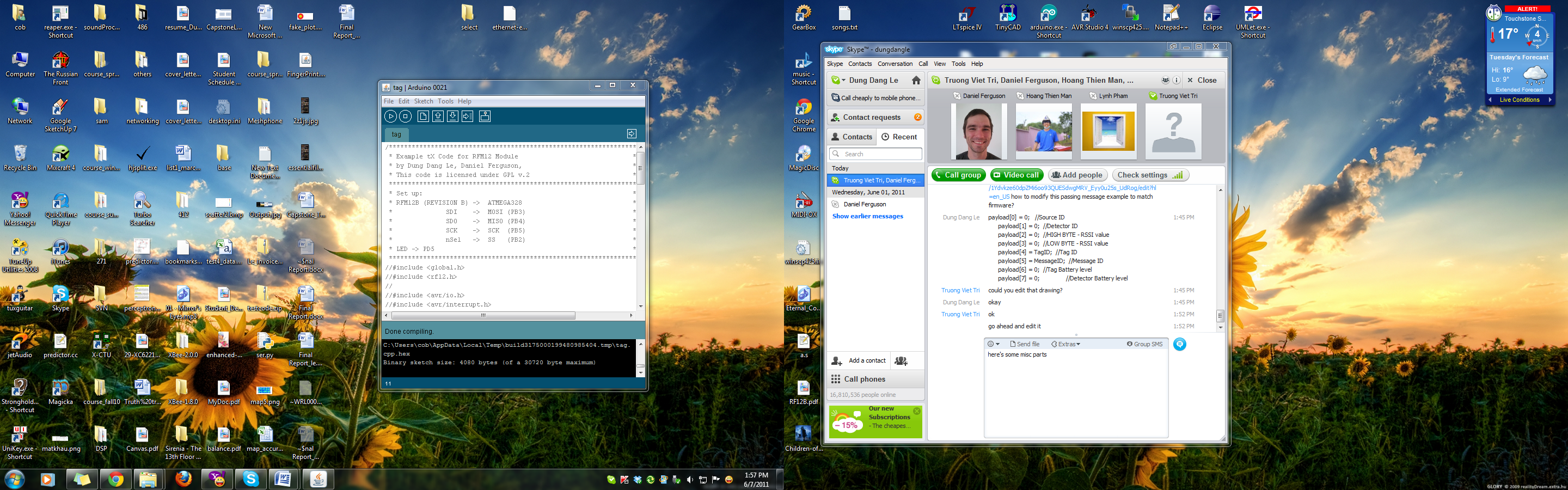
copy RF12 folder into ...arduino-0021\libraries\ (Windows machine,

yeah, under Linux you 'll find it somewhere else...)

device firmware, can be found in Firmware

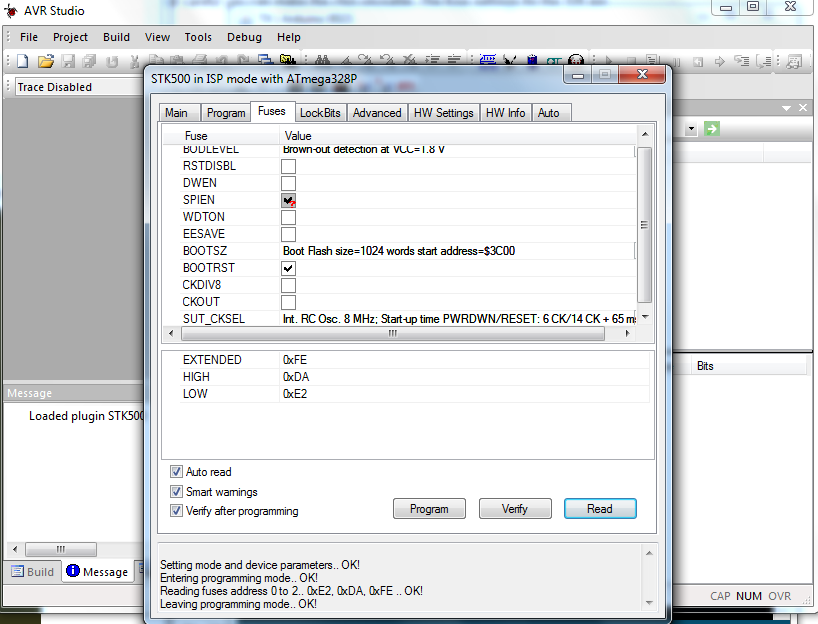
<http://code.google.com/p/tiu-tracking/>

1. compile source
   * open firmware source in arduino IDE
   * hold Shift key and hit compile button
   * go to the end of printout window, copy the “C:\Users\...\Temp\build3902875270031311054.tmp\....cpp.hex” to clipboard
   * this is the compiled hex file that can be used for burning device flash



1. programming

* open AVR studio
* make sure the MKII programmer is connected to PC and the target device
* make sure device is powered
* Burning fuses:
  + fuse setting for chips that are off-arduino and 3.3V rail:



* + program the fuse bits
* programming flash
  + select program tab
  + paste the path copied above
  + program the chip