

Multi-Robot Localization Update

Caleb Lu

Outline

- ▶ Goals from Last Meeting
- ▶ Goals Achieved
- ▶ Issues
- ▶ Next Step
- ▶ Questions

Goals from Last Meeting

- ▶ Get Communication between BBB and the USB interface board
- ▶ Write C code to send and receive messages
 - Properly parse messages to interpret message
 - Send and receive specific message to a beacon

Goals Achieved/Status

- ▶ Still currently having issues getting connection from the Beagle Bone Blue to the Zigbee board.
- ▶ Changed port to the USB serial port and when running code there is a stall. This indicates multiple potential causes:
 - Wrong port
 - Code crash/wrong
 - Wrong setup elsewhere

Issues / Pictures

[illegible]

```
debian@beaglebone:~/localization$ ./a.out
^C
debian@beaglebone:~/localization$ jobs
debian@beaglebone:~/localization$
```

Issues/Pictures

```
#include <stdio.h>
#include <unistd.h>
#include <fcntl.h>
int main() {
    int tx_byte[8] = {0x7E, 0x00, 0x04, 0x08, 0x01, 0x64, 0x62, 0x30};
    int rx_byte[8]={0};
    int *buff,*rx;
    int fd = open("/dev/ttyUSB0", O_RDWR);
    int i;
    rx = rx_byte;
    buff = tx_byte;
    write(fd, tx_byte, 8);

    ssize_t size = read(fd, &rx, 8);

    for(i = 0; i<8;i++)
        printf("Read byte %X\n", rx[i]);
    return 0;
}
```

Issues/Pictures

```
debian@beaglebone:~/localization$ ls /dev
apm_bios      log          spidev1.1    tty29        tty5         ttyS5
autofs        loop-control stderr        tty3         tty5         ttyUSB0
block         mapper       stdin        tty30        tty5         ubi_ctrl
btrfs-control mem          stdout       tty31        tty5
bus           memory_bandwidth tty          tty32        tty57        uinput
char          mmcblk1     tty0         tty33        tty58        urandom
console       mmcblk1boot0 tty1         tty34        tty59        vcs
cpu_dma_latency mmcblk1boot1 tty10        tty35        tty6         vcs1
cuse          mmcblk1p1   tty11        tty36        tty60        vcs2
disk          mmcblk1rpmb tty12        tty37        tty61        vcs3
dri           mqqueue     tty13        tty38        tty62        vcs4
fd            net         tty14        tty39        tty63        vcs5
full          network_latency tty15        tty4         tty7         vcs6
fuse          network_throughput tty16        tty40        tty8         vcsa
gpiochip0     null        tty17        tty41        tty9         vcsa1
gpiochip1     ppp         tty18        tty42        ttyGS0       vcsa2
gpiochip2     ptmx        tty19        tty43        tty00        vcsa3
gpiochip3     pts         tty2         tty44        tty01        vcsa4
hwrng         random      tty20        tty45        tty02        vcsa5
i2c-0         rfkill      tty21        tty46        tty03        vcsa6
i2c-1         rtc         tty22        tty47        tty04        vhci
i2c-2         rtc0        tty23        tty48        tty05        watchdog
iio:device0   serial      tty24        tty49        ttyS0        watchdog0
initctl       shm         tty25        tty5         ttyS1        watchdog1
input         snapshot    tty26        tty50        ttyS2        zero
kmem          snd         tty27        tty51        ttyS3
kmsg          spidev1.0   tty28        tty52        ttyS4
```

Next Step

- ▶ Confirm communication between BBB and the USB interface board
- ▶ Write C code to :
 - Properly parse messages to interpret message
 - Send and receive specific message to a beacon
 - Output incoming data into a table

Questions?