# Remote Control Setup

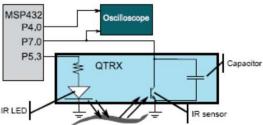
07/2023 N Diep

#### Equipment Needed:

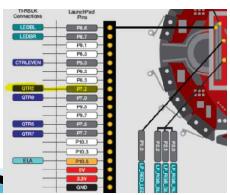
- Texas Instrument MSP432 Launch Pad with RSLK Robotics Kit
- Logic Analyzer
- IR Remote

#### Instructions:

- Use Bottom Side IR TX/RX sensors of RSLK Kit.
- Load code to Launch pad to Enable Output to active.
- Use analyzer to capture waveform.
- Hint: Use Probe pin 7.2 (Easy access)



**Basic IR Sensor Connection Diagram** 



**IR Sensor Port 7.2 Pin Location** 



TI RSLK Robotics Kit



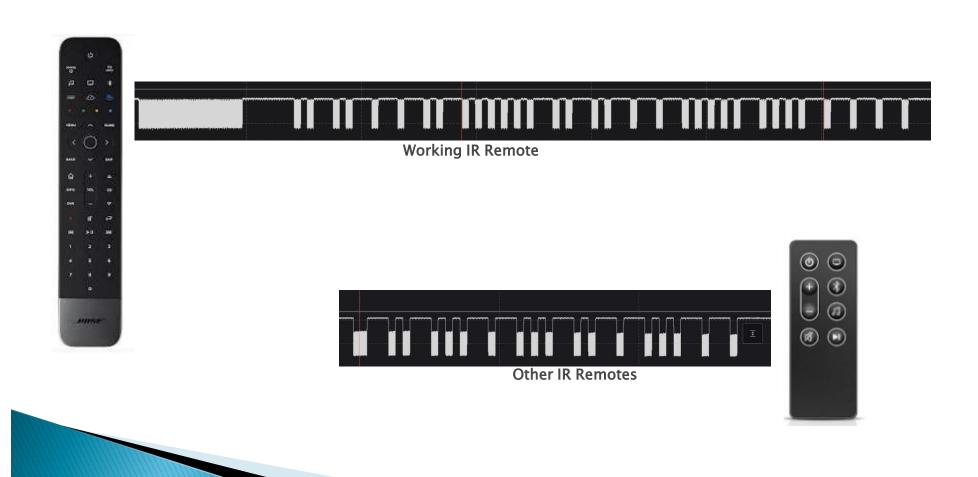
**Bottom Side: IR Sensor** 

```
void BoseIR_Init(void){
    // INI IR INPUT
    // Enable Bottom IR Rx Active. This will allow capture using logic analyzer
    // Use 0x04 = P7.2, //0xFF = P7.0 thru 7.7
    P7->SEL0 &= ~0x04;
    P7->SEL1 &= ~0x04;
                             // 1) Configure P7.x GPIO
    P7->DIR |= 0x04;
                             // 2) Set 7.x to output
             = 0 \times 04;
                             // 3) Enable pull resistors
    P7->REN
    P7->OUT = 0x04;
                                   Set 7.x output active
    // INI SIDE BUTTON INPUT
    // Press button to transmit IR
    P1->SEL0 &= ~0x12;
    P1->SEL1 &= ~0x12;
                              // 1) Configure P1.4, P1.1 as GPIO
    P1->DIR &= ~0x12;
                              // 2) Set P1.4 and P1.1 input
                 0x12; // 3) Enable pull resistors on P1.4 and P1.1
0x12; // P1.4 and P1.1 are pull-up
C Code: IR Sensor Initialization
    P1->REN |= 0x12;
    P1->OUT |= 0x12;
```



IR Remote and Sensor Capture Setup

Logic Analyzer (Salaee Logic 8)



Raw Data



Reviewing the data

- Timing
  - Approx 831.1 Hz (1.2msec)



- Carrier Frequency
  - Approx 38.42kHz



• Data

Measure Modulated Carrier Frequency

Used Visio to calculate spaces to count remote transmitted bits.



**Count Individual Captured Bits** 

- Convert each bit to HEX value.
  - Used windows calculator
- To transmited code, replay with IR\_Play
- Note some limitations
  - Raw data capture is not a common format.
  - Does not contain carrier frequency information

```
Calculator
                                               ≡ Programmer
18 IR Recorded
19 0000000001000100
                            0044
                                                                     0001 0101 0101 0100
20 0100010001010101
                            4455
21 0001010101010001
                            1551
22 0001000101000101
                            1145
23 0001010101010100
                                               OCT 12 524
                            1554
                                              BIN 0001 0101 0101 0100
24 0101000100010001
                            5111
                                                                   OWORD
                                                                                        MS
25 0100010100000000
                            4500
26 1111111111111111
```

Convert Each Bits to Hex format

```
const uint16_t BoseCtrl_ON[8]={
     0xFFFF, 0x4500, 0x5111, 0x1554, 0x1145, 0x1551, 0x4455, 0x0044 }; // IR Recorded
```

C Code: Output Encoded Carrier Frequency and Data

#### IR Transmitter

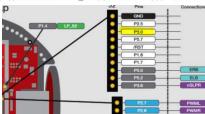
- Equipment Needed:
  - Texas Instrument MSP432 Launch Pad
  - IR Emitter (Generic Bose Replacement Remote Used Below)
- Removed IR Transmitter from non working remote





**Attached IR Remote to PCB** 

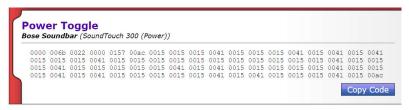
11#define IROUT3\_0 (\*((volatile uint8\_t \*)(0x42098440))) // P3.0 IR ODD Port 3 Output P3OUT 022h



Used Pin 3.0 for IRTransmit

#### IR Transmit Pronto Codes

- Accepts standard HEX Code found on internet
- https://files.remotecentral.com/library/3-1/bose/soundbar/soundtouch\_300\_(power)/index.html



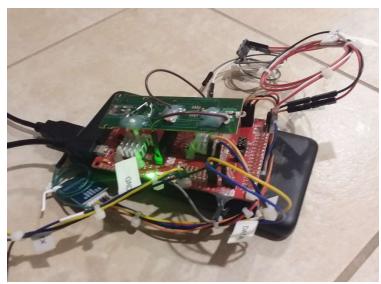
- Return to the Bose Discrete Infrared Hex Codes index...
- Understanding Hex codes
  - ProntoBoseCtrl\_ON[1] = 006b, (107)
  - Formula for index[1], Carrier frequency (KHZ) = 1000000/(N \* .241246)
  - $\circ$  Using 0x006b (107), Cf = 1000000/(107\*0.241246) = 38739.7 KHz
  - For values larger than 3
  - ProntoBoseCtrl\_ON[EVEN] = Count of ON burst pairs
    - ProntoBoseCtrl\_ON[4] = 0x0157, (343) qty ON burst pairs
  - ProntoBoseCtrl\_ON[ODD] = Count of OFF burst pairs
    - ProntoBoseCtrl\_ON[5] = 0x00AC, (172) qty OFF burst pairs



Green = ON burst, Red = OFF burst

#### IR Transmit Pronto Codes

- Use IR Pronto Play
  - Copy from Website, replace " " with ", 0x"
  - Function takes in data and array element index



**Remote Powered by Battery Pack** 

C Code: Raw Data Array

C Code: Accepts Data and Array Element Index

# **Application**

- Use emulator in combination with Online HEX library to create custom remote.
- Generic IR Learn Remote from Amazon (ASIN: B07DCM9B1N)



Using RSLK to teach Learning Remote

### Appendix

Resources used for this project:

To learn more about HDMI CEC:

https://www.nakamichi-usa.com/what-is-hdmi-cec

Bose Soundtouch 300 User Manual:

 $\frac{https://assets.bose.com/content/dam/Bose\_DAM/Web/consumer\_electronics/global/products/speakers/st\_30}{0\_product\_page/pdf/773965\_og\_soundtouch-300-soundbar\_en.pdf}$ 

Texas Instrument RSLK Robotics Kit:

https://university.ti.com/en/faculty/ti-robotics-system-learning-kit/ti-robotics-system-learning-kit-archive

Remotecentral.com IR Code List:

https://files.remotecentral.com/library/3-1/bose/soundbar/soundtouch\_300\_(power)/index.html