

NS-RX231 のソフトウェアガイド

〈赤外線リモコンの受信編〉

目次

1 プロジェクトのインポート.....	2
2 概要.....	3
3 ソースコード.....	4
3.1 信号フォーマット.....	4
3.2 ソースコード.....	5
3.3 detectIR.c.....	6
4 デバッグ.....	9
5 実行.....	10
6 回路図.....	11

1 プロジェクトのインポート

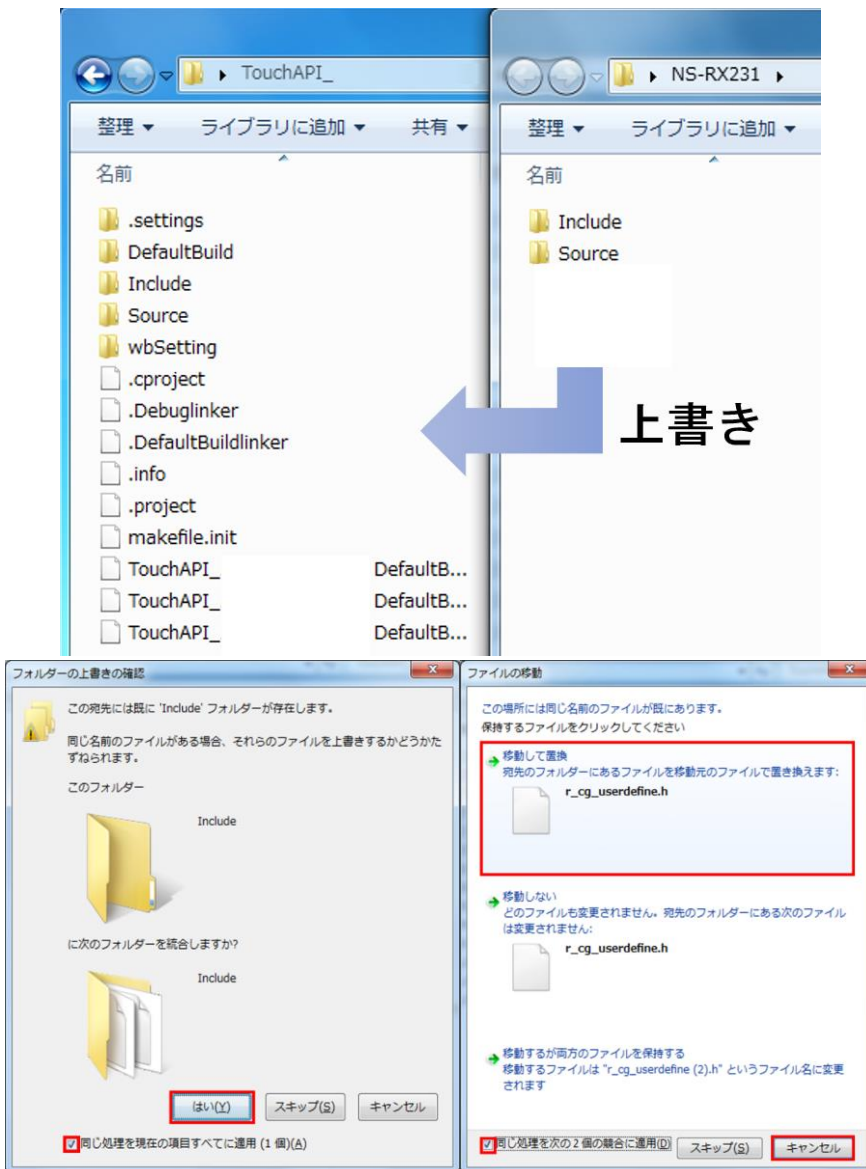


図 1-1 ソースファイルを上書き

添付されたソースファイルをWorkbench6 First step guideに従って、ウィザードで作成したプロジェクトに上書きした後、e2studioを実行します。

2 概要

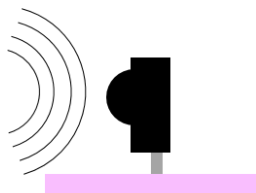


図 2-1 赤外線受信素子

IR受信部は、送信部の赤外線LEDから発光された赤外線を信号として認識する機能を持っています。しかし、光を媒体とする通信の特性上、蛍光灯の光、日光などの影響を受けて信号として誤認識することがあるので、通信システムの設計時に注意しなければいけません。

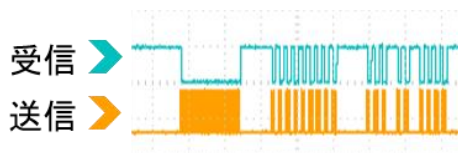


図 2-2 赤外線通信波形例

受信部は、回路の出力は通常はハイレベル状態ですが、一定のキャリア周波数を持つ赤外線信号を受信すると、0（ロウレベル）に低下し、また1（ハイレベル）に戻ります。

キャリア 周波数	30 kHz	TSOP4130
	33 kHz	TSOP4133
	36 kHz	TSOP4136
	38 kHz	TSOP4138
	40 kHz	TSOP4140
	56 kHz	TSOP4156

図 2-3 キャリア周波数と受信デバイスの関係

NS-RX231で使用している赤外線受信素子（TSOP4138）は、38kHzのキャリア周波数に対応した特性を持っているので、この点を留意して設計する必要があります。

詳細は、6回路図を参照してください。

この資料では、簡単なリモコン通信を実装してみましょう。

3 ソースコード

3.1 信号フォーマット

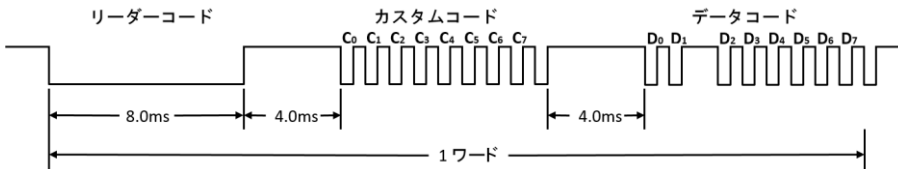


図 3-1 リモコン受信信号フォーマット例

簡単な信号フォーマットを例としてプロジェクトを作成してみました。

信号フォーマットは、通信の開始を知らせるリーダーコード、その後にはリモコン機器の区別のためのカスタムコード、そしてデータを伝達するためのデータコードから構成されています。

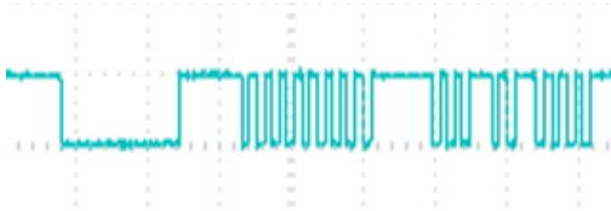


図 3-2 赤外線受信素子の出力波形

そして、ビットの0と1の違いは、次のとおりです。

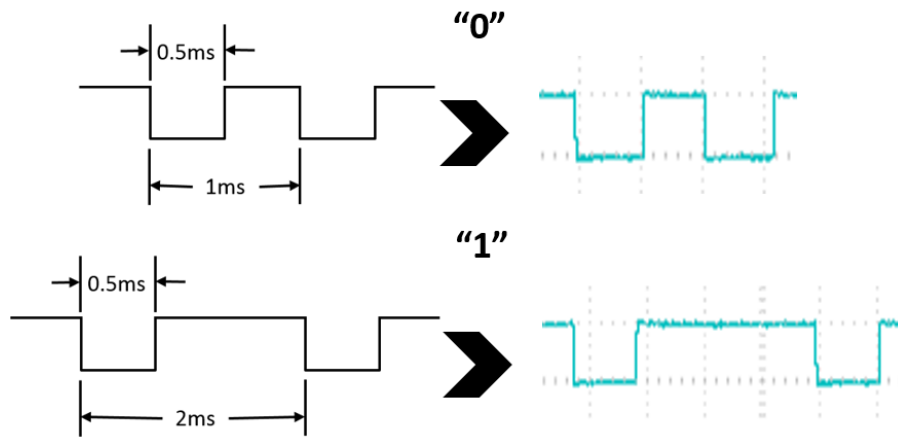


図 3-3 赤外線信号の0と1の違い

3.2 ソースコード

```

if(IR_detectStart()){
  while(1U){
    data = IR_detectData();
    if(data == 1000){
      cnt = 0;
      break;
    }

    if(data != 0){
      buf[cnt] = data;
      cnt++;
      data = 0;
    }else if(cnt == 18){
      cnt = 0;
      data = 0;
      FallingFlg = 0;
      break;
    }
  }

  if(IR_checkStartBit(buf[0],240,260)){
    data = IR_convertData(buf,1);
    if(IR_checkCustomBit(data,0)){
      data = IR_convertData(buf,10);
      switch (data) {
        case 40:
          index--;
          break;
        case 24:
          index = 3;
          break;
        case 168:
          index++;
          break;
      }
      if(index < 0){
        index = 4;
      }else if(index > 4){
        index = 0;
      }
      PORTB.PODR.BYTE = ledNum[index];
    }
  }
}

```

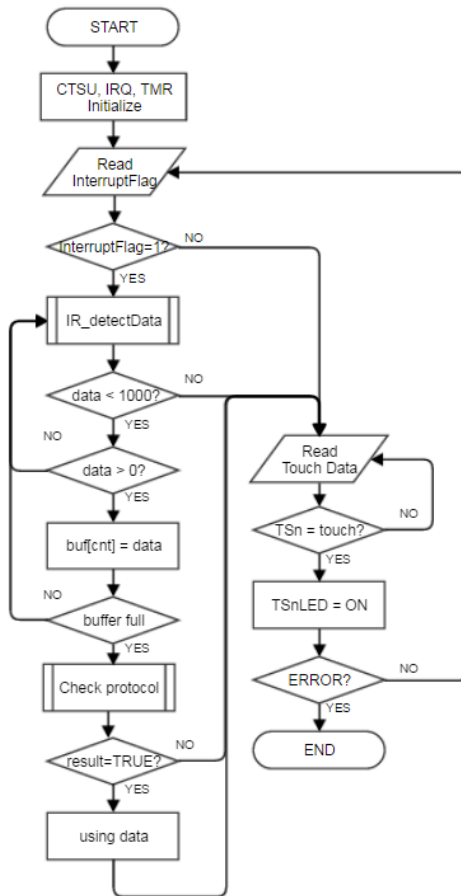


図 3-4 ソースコードとフローチャート



図 3-5 受信信号の立下り割り込みタイミング

受信信号の立ち下がりエッジ検出時に割り込みがかかります。このとき、次の立下りエッジ検出割り込みがかかるまでの時間をはかるためのタイマーを起動します。タイマーは、50usごとにカウントアップするように設定されており、例としてリーダーコードの8ms、4msの12msではカウント値は240になります。タイマーの繰り返し時間は50usなので、タイマーのカウント240は、次のとおりです。

$$240 \times 50\text{us} = 120 \times 100\text{us} = 12000\text{us} = 12\text{ms}$$

メモ 포함[DRm1]: added

3.3 detectIR.c

IR_convertData	
パラメータ	uint16_t * buf バッファ uint8_t startIndex 開始インデックス
戻り値	0 ~ 255
説明	バッファに入っている8個の立下りエッジ間隔データから8ビットのデータを組み立てます。 立下りエッジ間隔データは開始インデックス値から8個を使用します。 ex) buf[startIndex]~buf[startIndex+7] 立下りエッジ間隔データはDISTRVALを閾値にして1と0を判別します。DISTRVALのデフォルト値は40

コードの例

```
if(IR_checkStartBit(buf[0],240,260)){ //check LeaderCode

    data = IR_convertData(buf,1); //the index 1~8 buffer data convert to decimal data
}
```

IR_detectStart	
パラメータ	Void
戻り値	TRUE 1 FALSE 0
説明	赤外線受信素子 (TSOP4138) からの信号の立下りでFallingFlgが1になります。 FallingFlgの値を確認し、値が1の場合、FallingFlgの値とtmrCntのタイマーのカウント値を初期化した後、TRUEを戻します。

コードの例

```
void main(void)
{
    R_TMR0_Start();
    R_ICU_IRQ6_Start();

    while (1U) /* Main loop */
    {
        if(IR_detectStart()){ //Falling Edge detect

        }
    }
}
```



IR_detectData	
パラメータ	Void
戻り値	0 ~ 65,535
説明	FallingFlgの値が再び1になるまでの時間を計測します。 FallingFlgの値が1になると、FallingFlgの値を0にクリアし、その間にカウントしたタイマーのカウント値（変数tmrCntの値）を戻します。 もし、タイマーのカウント値が1000を超えた場合には、1000を返します。

コードの例

```
while(1){
    data = IR_detectData();
    if(data == 1000){ //if counter over the 1000 then; break
        cnt = 0;
        break;
    }

    if(data != 0){
        buf[cnt] = data; //insert data into a buffer
        cnt++;
        data = 0;
    }else if(cnt == 18){ //buffer maximum then break
        cnt = 0;
        data = 0;
        FallingFlg = 0;
        break;
    }
}
```

IR_checkStartBit	
パラメータ	uint16_t value 比較値 uint16_t compareMin 比較条件 最小値 uint16_t compareMax 比較条件 最大値
戻り値	TRUE 1 FALSE 0
説明	比較値が比較条件の最小値と最大値の間であればTRUEを戻します。

コードの例

```
if(IR_checkStartBit(buf[0],250,260)){ //check LeaderCode
}

for(i=0;i<18;i++){ //initialize buffer
    buf[i] = 0;
}
```



IR_checkCustomBit		
パラメータ	uint16_t value	比較値
	uint8_t compareVal	比較条件値
戻り値	TRUE 1	
	FALSE 0	
説明	比較値と比較条件値が等しい場合にTRUEを戻します。	

コードの例

```
if(IR_checkStartBit(buf[0],250,260)){ //check LeaderCode
    data = binToDec(buf,1); //the index 1~8 buffer binary data return to decimal data
    if(IR_checkCustomBit(data,0)){ //check CustomCode
        data = binToDec(buf,10); //the index 10~17 buffer binary data return to decimal data
        requestData(data); //using data
    }
}

for(i=0;i<18;i++){ //initialize buffer
    buf[i] = 0;
}
```


4 デバッグ



図 4-1 NS-RX231に電源アダプタとE1デバッガを接続した様子

プロジェクトのビルド

右クリック

- ビルド構成
- ビルドから除外...
- 実行(R)
- デバッグ(D)

構成を消去して再ビルド

再ビルドする構成を選択

- TouchAPI_20161213144540
- DefaultBuild [アクティブ]

選択した構成をビルド

OK

デバッグ

右クリック

- 1 EASE Script
- 2 GDB Simulator Debugging (SH, RH850)
- 3 Renesas GDB Hardware Debugging
- 4 Renesas Simulator Debugging (RC, RL78)

Launch Configuration Selection

Choose a launch configuration to debug

- TouchAPI_20161213144540 DefaultBuild
- TouchAPI_20161213144540 DefaultBuild-E1
- TouchAPI_20161213144540 DefaultBuild-E2Lite

OK

図 4-2 プロジェクトのビルドとデバッグ

5 実行

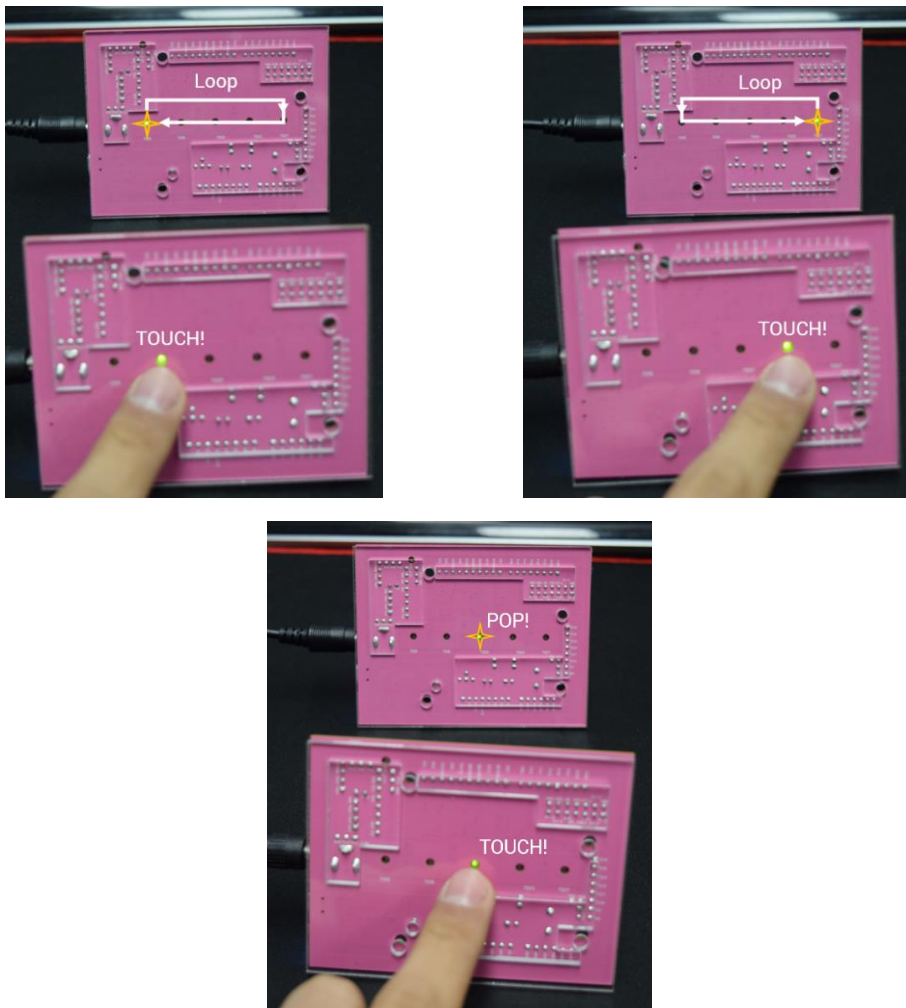


図 5-1 赤外線受信時の動作例

6 回路図

下の図は、赤外線リモコン受信部の回路図です。

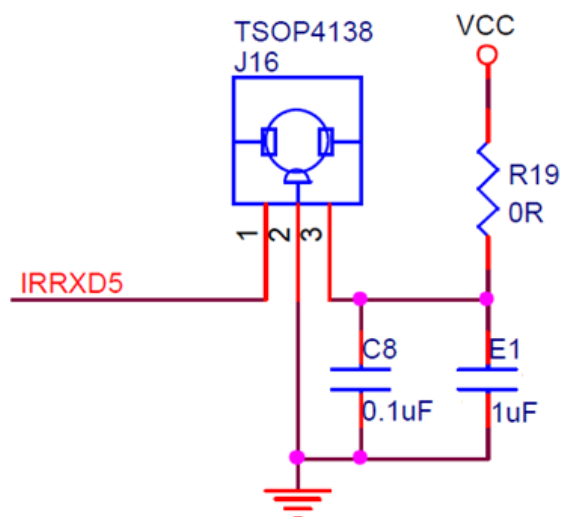


図 6-1 NS-RX231の赤外線受信部回路図