

main.c

```
1  /*****
2  /*          赤外線リモコンの信号をキャンセルする装置のメイン関数          */
3  /*
4  /*  2024. mm. dd - 2024. mm. dd                      by S. Suzuki          */
5  /*****
6  /* 注意 : 1. このプログラムはATtiny44A-PU用です。          */
7  /*      : 2. クロック8MHz用です。ヒューズビットCKDIV8=1としてください。  */
8
9  #define F_CPU 8000000UL
10 #include <avr/io.h>
11 // #include <avr/wdt.h>
12 #include <avr/interrupt.h>
13
14 #include "hardwareInfo.h"
15 #include "IrRcCanSend.h"
16 #include "IrRcReceive.h"
17
18 /*****
19
20 #if 0
21 /*****
22 /*          定周期割り込み発生          */
23 /*****
24 /* 備考 : 使えるタイマがないため、ウォッチドッグタイマWDTの割り込み発生機能を */
25 /*      : 使って実現する。発生する定周期割り込みは約16mS周期。          */
26 /* 注意 : WDTCSRのWDIEビットは、割り込み発生で自動的に'0'になるので、そのまま */
27 /*      : だと次回はリセット動作になってしまうので、割り込み発生でWDIEビット */
28 /*      : を再設定する必要がある。          */
29
30 #define wdt_reset() __asm__ __volatile__ ("wdr")
31 static volatile WORD s_wTmr16m_main;
32
33 static void Start_IntTimer16m(void)
34 {
35     /* 割り込みコードでWDT許可 (WDP3~WDP0=0で最速の16mS周期のはず) */
36     WDTCSR = _BV(WDCE) | _BV(WDE);
37     WDTCSR = _BV(WDIE) | _BV(WDE);
38     /* ウォッチドッグタイマリセット */
39     wdt_reset();
40 }
41
42
43 ISR(WATCHDOG_vect)
44 {
45     /* WDTCSR.WDIEビットは自動的に'0'になるので再設定する */
46     WDTCSR = _BV(WDIE) | _BV(WDE);
47     /* タイマ用カウンタをデクリメント */
48     if(s_wTmr16m_main != 0){
49         --s_wTmr16m_main;
50     }
51     /* 以降は16mS周期で実行するもの */
52
53 }
54 #endif
55
56 /*****
57 /*          LEDインジケータ用I/Oポートの初期化          */
58 /*****
59
60 static void Init_LedPort(void)
61 {
62     LED_PORT |= LED_BIT;          /* LED用I/Oポートにあらかじめ OFFを出力 */
63     LED_DDR |= LED_BIT;          /* LED用I/Oポートを出力ポートにする */
64 }
65
66 void Indicate_LED(BYTE byLighting)
67 {
68     if(byLighting == 0){
69         LED_OFF();          /* LEDインジケータ消灯 */
70     }
71     else{
72         LED_ON();          /* LEDインジケータ点灯 */
73     }
74 }
75
76 /*****
77 /*****
78 /*          */
```

main.c

```
79 /*                      メイン関数                      */
80 /*                      */
81 /******  
82 /******  
83  
84 int main(void)  
85 {  
86     BYTE    byRcType;  
87  
88 // /* 定周期割り込みの初期化 */  
89 // Start_IntTimer16m();  
90 /* LEDインジケータ用I/Oポートの初期化 */  
91 Init_LedPort();  
92 /* タイマ0の初期化とキャンセル信号送信準備 */  
93 Init_TIMER0();  
94 /* タイマ1の初期化と赤外線リモコンデータの受信準備 */  
95 Init_TIMER1();  
96  
97 /* 割り込み許可 */  
98 sei();  
99  
100 /* 赤外線リモコンの受信を開始 */  
101 Ctrl_IrRcReceive(1);  
102 /* メインループです */  
103 while(1){  
104     /* 赤外線リモコンの信号受信をチェック (戻り値≠TYPE_UNDETなら受信あり) */  
105     byRcType = Get_IrRcDetect(); /* 戻値 : TYPE_UNDET, TYPE_NEC, TYPE_AEHA, TYPE_SONY */  
106     if(byRcType != TYPE_UNDET){  
107         /* 赤外線リモコンの受信を停止 */  
108         Ctrl_IrRcReceive(0);  
109         /* 38KHzキャンセル信号送信開始 */  
110         Send_IrRcCancel(100); /* 約10mS送出 */  
111         /* 送信完了を待つ */  
112         while(Check_IrRcCancel() != 0);  
113         /* 赤外線リモコンの受信を開始 */  
114         Ctrl_IrRcReceive(1);  
115     }  
116 }  
117  
118 return 0;  
119 }  
120
```