

```

1 //
2 // system72シリーズ ATAPIコントローラ用リモコン基板
3 //   ・ IRリモコン公開版 V2.0
4 //   ・ Sony/NEC 対応版
5 //
6 //----- 著作権・免責 -----
7 // 著作権は作者であるMI-TAKE が有します。
8 // 本プログラムを使用したことにより生じたいかなる障害、損害(例え本ソフトの
9 // バグによるものであろうとも)において作者は一切責任を負いません。
10 // 各自の責任において使用してください。
11 // 又、作者はバグが発見された場合においても、その修正、バージョンアップの
12 // 義務を負わないものとします。
13 //-----
14 //   *** Copyright (c) 2012 by MI-TAKE *****
15 //
16
17 #include <avr/io.h>
18 #define F_CPU 2000000UL
19 #include <util/delay.h>
20 #include <avr/interrupt.h>
21
22 #define sbi(PORT, BIT) (PORT |= _BV(BIT))
23 #define cbi(PORT, BIT) (PORT &= ~_BV(BIT))
24 #define p_ply 5
25 #define p_stp 6
26 #define p_fwd 7
27 #define p_rwd 4
28
29 int cmd[34];
30 //unsigned char B_cmd[6];
31 int comand,comands1,comands2;
32
33 //-----
34 //waiting loop
35 void waitMsecond(int t)
36 { while (t--) _delay_ms(1); }
37
38 void waitUsecond(int t)
39 { while (t--) _delay_us(1); }
40
41 //-----
42 //all port initialize
43 void port_init(void)
44 {
45     DDRB = 0b00111101; //portB 1:in 1,3-5:out
46     PORTB = 0b00000010; // pullup
47     DDRD = 0xf7; //PD3 input(=INT1)
48     PORTD = 0xff; // pullup
49     cbi(PORTD, 2); //4066 ATAPI側off
50
51     EICRA=0b00001100; //External Interrupt Control Reg.
52     EIMSK=0x02; //INT1
53     sei(); //enabe
54 }
55
56 void remote_code_out(int cmd)
57 {
58     PORTD |= 0b11110000; //ATAPI Hi
59     sbi(PORTD, 2); //4066 ATAPI側on
60     cbi(PORTD, cmd); //cmd out
61     waitMsecond(200); //
62     cbi(PORTD, 2); //4066 ATAPI側off
63     return;
64 }
65 //-----
66 //
67 ISR(INT1_vect)
68 {
69     int p, i, j, k;
70
71     comand = comands1 = comands2 = 0;
72     if(PINB & 0x02) { i = 0; //PORTB2 jamper無=Sony, code#257

```

```

73 do{ p = 0;
74   do{ p++; waitUsecond(80); //80(実測で100uS)でパルス幅カウント
75     } while( PIND & 0x08) ; //IR input == 'H' でカウント継続
76     cmd[i++] = p;
77     waitUsecond(400);
78     for(j=0;j<20;j++) {
79       if((PIND & 0x08) != 0) j = 100;
80       waitUsecond(10);
81     }
82   } while(j > 21) ;
83   if((cmd[0]>20)&&(cmd[0]<50)) { //リード部が20~50のみコマンド
84     j=8;
85     for(i=0;i<4;i++){
86       if(cmd[i+1] > 8) comands1 +=j;
87       if(cmd[i+5] > 8) comands2 +=j;
88       j >>=1;
89     }
90     comand = comands1*16+comands2;
91   }
92 }
93 else{ ////////////////////////////////////////////////// NEC code
94   p = i = 0;
95   do{ p++; waitUsecond(80); // Hedder H read
96     } while( PIND & 0x08) ; //IR input == 'H' でカウント継続
97     cmd[i++] = p;
98     p=0;
99     for(j=0;j<60;j++) { // Hedder L read
100      if((PIND & 0x08) != 0) j = 100;
101      p++; waitUsecond(80);
102    }
103    cmd[i++] = p;
104    if(cmd[0] > 70){
105      for(k=0;k<32;k++){ // cmd read. 8bit +8bit +8bit +8bit
106        do{ waitUsecond(80);
107          } while( PIND & 0x08) ; //IR input == 'H' でcmdread.
108          p=0;
109          for(j=0;j<60;j++) { // Hedder L read (4.4mS).
110            if((PIND & 0x08) != 0) j=100;
111            p++; waitUsecond(80);
112          }
113          cmd[i++] = p;
114        }
115        k = 1;
116        for(j=18;j<26;j++){
117          if(cmd[j] > 12) comands1 +=k;
118          k <<=1;
119        }
120      }
121    switch(comands1){ //共通コード置き換え (NEC => Sony)
122      case (0x84):
123        comand = 0x59; //PLAY
124        break;
125      case (0x83):
126        comand = 0x59; //PLAY
127        break;
128      case (0x87):
129        comand = 0x19; //STOP
130        break;
131      case (0x86):
132        comand = 0x49; //予約
133        break;
134      case (0x8b):
135        comand = 0x19; //STOP
136        break;
137      case (0x8a):
138        comand = 0xc9; //予約
139        break;
140      case (0x90):
141        comand = 0xd9; //F-Rewind
142        break;
143      case (0x8f):
144        comand = 0x39; //F-Foward

```

```

145         break;
146     default:
147         comand = 0;
148     }
149 }
150 switch(comand) {
151     case (0x59): //PLAY
152         remote_code_out(p_ply);
153         break;
154     case (0x19): //STOP
155         remote_code_out(p_stp);
156         break;
157     case (0x39): //F-Foward
158         remote_code_out(p_fwd);
159         break;
160     case (0xd9): //F-Rewind
161         remote_code_out(p_rwd);
162         break;
163     default: //Not match
164         break;
165 }
166 waitMsecond(300);
167 EIFR = 0b00000010;
168 }
169
170 //-----
171 //
172 int main(void)
173 {
174     port_init();
175     waitMsecond(500);
176     while(1) {
177         sbi(PORTD, 1); //dummy
178         waitMsecond(100);
179         cbi(PORTD, 1); //dummy
180         waitMsecond(100);
181     }
182     return(0);
183 }
184
185

```