

概要

本キットは、マイクロチップ社ナノワット超低電力(XLP)テクノロジー、8ビットマイクロコントローラ、ミッドレンジ、エンハンスシリーズPIC 16F1937を使用した10進変換アダプター(品番HtoD-0297)です。

LCD表示付赤外線リモートコントローラ受信部(品番IR-0294-RX)、表示なし赤外線リモートコントローラ受信部(品番IR-0298-RX)又は、ラジコンリモートコントローラ受信部(品番IR-0299-RX)の出力に本機を繋げば、キースイッチに対応した十進の出力信号が保持されますので、種々の機器の制御が簡単になります。

表示は3ビットデバイスコードを3個のLEDに(二進)、5ビットのキーコードを26個のLEDに十進で表示します。

動作

1. 受信機側のコネクタCN1, CN2と本機のコネクタCN1, CN2を差し込む。
2. 受信機側のリセットスイッチRESET (SW1)を押す。
3. 本機の電源は受信機側から供給される(4V-5V)。本機に必要な電流は40mA以上。
4. 電源が入って数秒後に本機はリセットされデフォルト値(後述)が表示される。
5. 1バイトの入力をボーリングで読み込み、上位3ビットデバイスコードを3個のLEDで表示(負論理)する。下位5ビットキーコードは十進に変換され26個のLEDに表示され、同時にコネクタCN4, 5, 6, 7に負論理で出力される。負論理は0の時LEDが点灯する。
6. 入力データに変化があればブザー(約5KHz)が短時間(約16ms)鳴る。
7. 本機をリセットするには受信側のリセットスイッチを押すか受信側の電源を切る。

準備するもの

1. 10進変換アダプター(品番HtoD-0297)完成品、ICはプログラム済み、動作確認済み
2. LCD表示付赤外線リモートコントローラ受信部(品番IR-0294-RX)、表示なし赤外線リモートコントローラ受信部(品番IR-0298-RX)、又はラジコンリモートコントローラ受信部(品番IR-0299-RX)、何れか1台
3. 受信側の電源(4V-5V, 50mA以上)パワーは必ず受信機側から受ける。
(注)本機の+5Vピンから電源を供給しない。受信側の電源とショートする。
4. ミニスピーカ、圧電スピーカ等。ブザーを鳴らさない時は不要。

出力の使い方

1. CN4からCN7のコネクタに送信機のキーコード(スイッチ番号)に対応した十進のデータが出力される。
2. 二度押しスイッチ出力はトグルスイッチとして使用できる。
例えば、パワースイッチなどON/OFFスイッチ。
3. 四度押しスイッチ出力は4個の設定が必要な制御に使用できる。
例えば、音量調整(切る、小、中、大)、風速調整(切る、弱、中、強)。
4. 単発キー出力は十進のロータリースイッチとして使える。
また、スライドスイッチ(2個以上のスイッチを使う)としても使用できる。
初期値(デフォルト値)は"UNOSW 7"とした。

リセットについて

受信側がリセットされると本機もリセットされデフォルト値に戻る。デフォルト値については、キーコードとLED表示及び出力ピン対照表参照。

送信側と受信側を同時リセットするとキーコードはデフォルトから始まるが、受信側だけリセットした時はデフォルト値から次に押されたキーコードにジャンプする。例えば、リセット前のキーコードがSW5-2の時、受信側だけリセットするとデフォルト値SW5-0になる。再びSW5を押すとSW5-3にジャンプする。電源投入時や停電もリセットと同様な動きをする。このことを十分考慮して設計すること。

注意事項

1. 本機は受信側から電源サプライを受けるので、本機のコネクタCN5の4番ピン、+5Vには他の電源を接続をしない事。
2. 本機の入力コネクタCN1, CN2は高インピーダンスになっており、オープン状態の時は静電気に十分注意する事。本機を使用しないときはアルミホイルなどにくるんでおく事。

プログラムについて

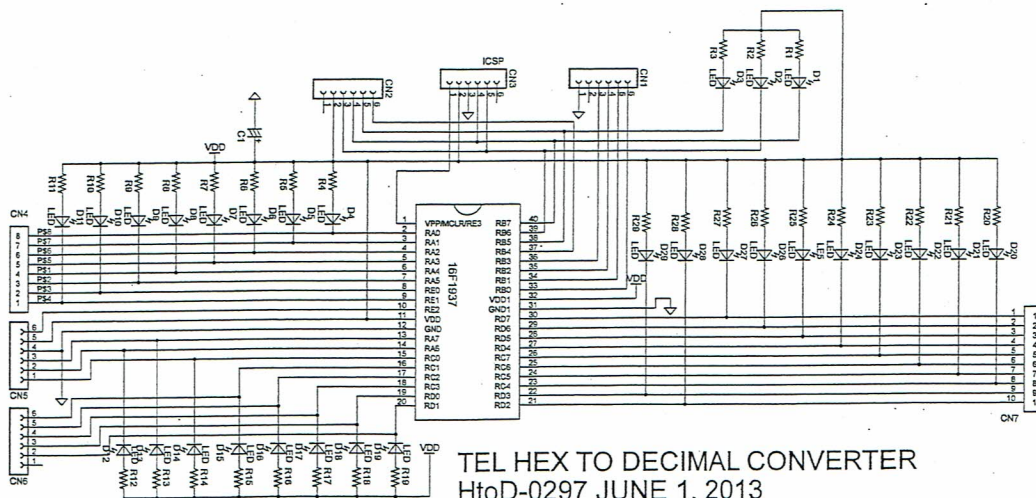
ハイテック C コンパイラ ライトモードV9.83を使用。

ROM(プログラムメモリ) 8192中435ワード使用 約5.3%

RAM(データメモリ) 512中14バイト使用 約1.8%

参考資料としてCソースファイルのリスト(Listing)を添付する。

(注意) CONFIGワードのコードプロテクションビットはOFFになっており、ICプログラムメモリの上書き読み込みが自由にできますが、ICSPを使用しプログラムを変更した場合には、保証できなくなりますから注意してください。また、プログラムに関するご質問はお受けできませんのでご了承ください。



10進変換アダプター(品番HtoD-0297) 回路図

本機は、LCD表示付赤外線リモートコントローラ受信部(品番IR-0294-RX)の他に、表示なし赤外線リモートコントローラ受信部(品番IR-0298-RX)又は、ラジコンリモートコントローラ受信部(品番IR-0299-RX)の出力に接続可能。

TEL エレクトロニクス・キット
(有) 谷岡電子
〒164-0003 東京都中野区
東中野 1-5-1-13
大島ビル第一別館 402
☎ (03)3366-4552

5V電源プラス側
マイナスGNDに接続

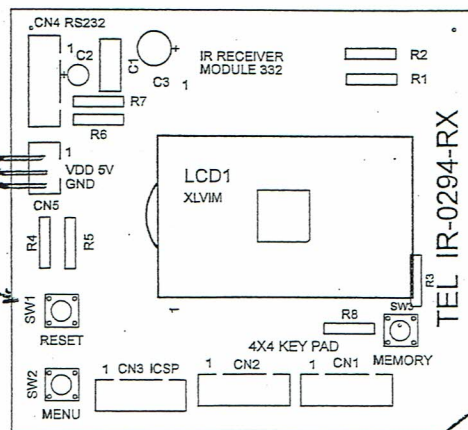
RESET(SW1)
十進アダプターを
初期化する。

キーコード表示用LED(D4-D29)
'0'の時LED点灯(負論理)
デフォルト点灯LEDは
D4、D6、D8、D10、D13、D27、D23
受信側のリセット(SW1)を押すと
デフォルト点灯LEDが点灯

ICSP(In-Circuit Serial Programming)用
コネクタ。使用しないこと。

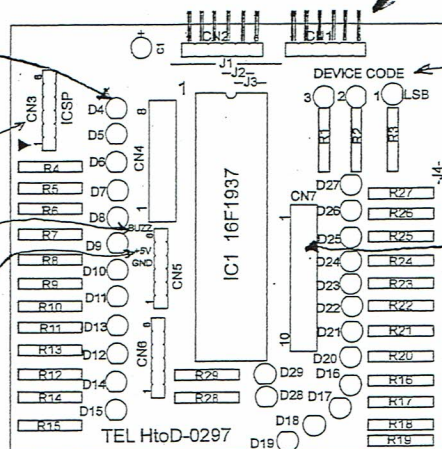
BUZZ(CN5-6ピン)ブザー出力
入力データに変化があればブザー
(約5KHz)が短時間(約16ms)鳴る。

+5V(CN5-5ピン)
5V出力ピン
本キットのサプライは
受信機側から受ける。
従って出力ピンには
他の電源をつながない
こと。



LCD表示付赤外線リモート
コントローラ受信部と10進
変換アダプター(品番Hto
D-0297)を組み合わせ、
リモコンとして使用する場合

キーパッド(品番ACC-0295-KEY4x4)と
10進変換アダプター(品番HtoD-0297)
を差し替えたときは必ず受信側RESET(SW1)
スイッチを押してリセットすること。



キーコードとLED表示及び出力ピン対照表

キースイッチ	デフォルト 表示テキスト	点灯LED	出力ピン(負論理) コネクタ	ピン番号	リセット値(LED)
SW1-0	"DUOSW1-0"	D4	CN4	8	デフォルト(点灯)
SW1-1	"DUOSW1-1"	D5	CN4	7	
SW2-0	"DUOSW2-0"	D6	CN4	6	デフォルト(点灯)
SW2-1	"DUOSW2-1"	D7	CN4	5	
SW3-0	"DUOSW3-0"	D8	CN4	4	デフォルト(点灯)
SW3-1	"DUOSW3-1"	D9	CN4	3	
SW4-0	"DUOSW4-0"	D10	CN4	2	デフォルト(点灯)
SW4-1	"DUOSW4-1"	D11	CN4	1	
SW5-0	"QRTSW5-0"	D13	CN5	3	デフォルト(点灯)
SW5-1	"QRTSW5-1"	D12	CN5	2	
SW5-2	"QRTSW5-2"	D14	CN5	1	
SW5-3	"QRTSW5-3"	D15	CN6	6	
SW6-0	"QRTSW6-0"	D27	CN7	1	デフォルト(点灯)
SW6-1	"QRTSW6-1"	D26	CN7	2	
SW6-2	"QRTSW6-2"	D25	CN7	3	
SW6-3	"QRTSW6-3"	D24	CN7	4	
SW7	"UNOSW 7"	D23	CN7	5	デフォルト(点灯)
SW8	"UNOSW 8"	D22	CN7	6	
SW9	"UNOSW 9"	D21	CN7	7	
SW10	"UNOSW 10"	D20	CN7	8	
SW11	"UNOSW 11"	D16	CN6	5	
SW12	"UNOSW 12"	D17	CN6	4	
SW13	"UNOSW 13"	D18	CN6	3	
SW14	"UNOSW 14"	D19	CN6	2	
SW15	"UNOSW 15"	D29	CN7	9	
SW16	"UNOSW 16"	D28	CN7	10	

10進変換アダプター(品番HtoD-0297)
Cソースファイルリスト(Listing)

```
C:\Documents and Settings\User\My Documents\TEL_kits\FIC_IR\HtoD_0297.c
1  /******
2  * Project Name: Hex to decimal converter
3  * Project number: HtoD-0297
4  * Micro controller: Enhanced mid-range PIC 16f1937
5  * Features:
6  * *Conversion from Hex 0x00 to 0x19 to Decimal 0 to 25.
7  * *26 LED indicator for corresponding ir keys
8  * *3 LED indicator for device codes
9  * *Buzzer output
10 * Compiler: Hi-Tech C lite mode v9.83
11 * JUNE 3, 2013. Coded by Yuji Tanaka(TEL company)
12 *****/
13
14 #include <htc.h>
15
16 #CONFIG(MCLR_ON & CP_OFF & BOREN_OFF & WDTF_OFF &
17 FOSC_INTOSC & FPRTE_ON & CPD_OFF & CLKOUTEN_OFF &
18 FCMEN_OFF);
19
20 #CONFIG(WRT_OFF & STVRN_OFF & BORV_HI & LVP_OFF & FLEN_OFF);
21
22 #define XTAL_FREQ 4000000 //4Mhz
23
24 typedef volatile bit Bool; //flags
25
26 //Function prototype declaration
27 void reg_init(void); //F1
28 void interrupt_isr(void); //F2
29 void buzzer(void); //F3
30 void get_data(unsigned char hex_data); //F4
31
32 #define BUZZER_TONE 100 //100us
33 #define TRUE 1
34 #define FALSE 0
35
36 //Global variable
37 Bool buzzer_flag = FALSE;
38
39 //Main function
40 void main(void)
41 {
42     unsigned char sPORTB1, sPORTB2;
43
44     reg_init();
45
46     //Main loop
47     for(;;)
48     {
49         sPORTB1 = PORTB;
50         sPORTB1 &= 0xE0;
51         if(!sPORTB1) //device code(000)
52         {
53             LATA = 0x6A; //RESET
54             LATB = 0x00; //0b01101010
55             LATC = 0x7F; //0b01111111
56             LATD = 0x7F; //0b01111111
57             LATE = 0x02; //0b-----10
58         }
59
60         sPORTB1 = PORTB;
61         get_data(sPORTB1);
62         delay_us(1);
63         sPORTB2 = PORTB;
64         if(sPORTB1 != sPORTB2)
65         {
66             buzzer_flag = TRUE;
67             buzzer(); //F3
68         }
69     }
70 }
71
72 //Function definition F1, register initialization
73 void reg_init(void)
74 {
75     OSCCON = 0x68; //IRCF<3>=0=1101 4Mhz
76     OSC1DIV = 0x00;
77
78     TRISA = 0x00; //OUTPUT
79     TRISB = 0xFF; //INPUT
80     TRISC = 0x00;
81     TRISD = 0x00;
82     TRISE = 0x00;
83     ANSELA = 0x00; //DIGITAL I/O
84     ANSELB = 0x00;
85     ANSELC = 0x00;
86     WPUB = 0x1F; //PORTB<4>=0 WEAK PULL-UP
87     LATA = 0x6A; //0b01101010
88     LATB = 0x00;
89     LATC = 0x7F; //0b01111111
90     LATD = 0x7F; //0b01111111
91     LATE = 0x02; //0b-----10
92     PORTB = 0x00; //CLEAR PORTB REGISTER
93
94     T2CON = 0b00000011; //0x0B
95     //-----11 POSTSCALER TOUTPS<3>=0
96     //-----0- TIMER2 OFF
97     //-----11 PRESCALER IS 64 T2CKPS<1>=0
98     TMR2 = 0x00; //TIMER 2 CLEAR
99     PR2 = 0xFF; //TOTAL BUZZER SOUND DURATION:
100 //1us*64*256=16.384ms
101 TMR2IE = 1; //TIMER 2 ENABLED
102
103 PEIE = 1;
104 ei();
105 }
106
107 //Function definition, interrupt isr
108 void interrupt_isr(void) //F2
109 {
110     if(TMR2IE && TMR2IF)
111     {
112         TMR2IF = 0;
113         TMR2ON = 0;
114         buzzer_flag = FALSE;
115     }
116 }
117
118 void get_data(unsigned char hex_data) //F4
119 {
120     hex_data &= 0x1f;
121     switch(hex_data)
122     {
123         // PORT CONNECTER PIN
124         case 0: //RA0 CN4-5
125             LATA0 = 0;
126             LATA1 = 1;
127             break;
128         case 1: //RA1 CN4-7
129             LATA1 = 0;
130             LATA0 = 1;
131             break;
132         case 2: //RA2 CN4-6
133             LATA2 = 0;
134             LATA3 = 1;
135             break;
136         case 3: //RA3 CN4-5
137             LATA3 = 0;
138             LATA2 = 1;
139             break;
140         case 4: //RA4 CN4-4
141             LATA4 = 0;
142             LATA5 = 1;
143             break;
144         case 5: //RA5 CN4-3
145             LATA5 = 0;
146             LATA4 = 1;
147             break;
148     }
149 }
```

```
C:\Documents and Settings\User\My Documents\TEL_kits\FIC_IR\HtoD_0297.c
153 break;
154
155 case 6: //RE0 CN4-2
156     LATE0 = 0;
157     LATE1 = 1;
158     break;
159
160 case 7: //RE1 CN4-1
161     LATE1 = 0;
162     LATE0 = 1;
163     break;
164
165 case 8: //RA7 CN5-3
166     LATA7 = 0;
167     LATA6 = 1;
168     LATC0 = 1;
169     LATC1 = 1;
170     break;
171
172 case 9: //RA5 CN5-2
173     LATA7 = 1;
174     LATA6 = 0;
175     LATC0 = 1;
176     LATC1 = 1;
177     break;
178
179 case 10: //RC0 CN5-1
180     LATA7 = 1;
181     LATA6 = 1;
182     LATC0 = 0;
183     LATC1 = 1;
184     break;
185
186 case 11: //RC1 CN6-6
187     LATA7 = 1;
188     LATA6 = 1;
189     LATC0 = 1;
190     LATC1 = 0;
191     break;
192
193 case 12: //RD7 CN7-1
194     LATD7 = 0;
195     LATD6 = 1;
196     LATD5 = 1;
197     LATD4 = 1;
198     break;
199
200 case 13: //RD6 CN7-2
201     LATD7 = 1;
202     LATD6 = 0;
203     LATD5 = 1;
204     LATD4 = 1;
205     break;
206
207 case 14: //RD5 CN7-3
208     LATD7 = 1;
209     LATD6 = 1;
210     LATD5 = 0;
211     LATD4 = 1;
212     break;
213
214 case 15: //RD4 CN7-4
215     LATD7 = 1;
216     LATD6 = 1;
217     LATD5 = 1;
218     LATD4 = 0;
219     break;
220
221 case 16: //RC7 CN7-5
222     LATC7 = 0; LATC6 = 1; LATC5 = 1; LATC4 = 1; LATC2 = 1;
223     LATC3 = 1; LATD0 = 1; LATD1 = 1; LATD3 = 1; LATD2 = 1;
224     break;
225
226 case 17: //RC6 CN7-6
227     LATC7 = 1; LATC6 = 0; LATC5 = 1; LATC4 = 1; LATC2 = 1;
228     LATC3 = 1; LATD0 = 1; LATD1 = 1; LATD3 = 1; LATD2 = 1;
229     break;
230
231 case 18: //RC5 CN7-7
232     LATC7 = 1; LATC6 = 1; LATC5 = 0; LATC4 = 1; LATC2 = 1;
233     LATC3 = 1; LATD0 = 1; LATD1 = 1; LATD3 = 1; LATD2 = 1;
234     break;
235
236 case 19: //RC4 CN7-8
237     LATC7 = 1; LATC6 = 1; LATC5 = 1; LATC4 = 0; LATC2 = 1;
238     LATC3 = 1; LATD0 = 1; LATD1 = 1; LATD3 = 1; LATD2 = 1;
239     break;
240
241 case 20: //RC2 CN6-5
242     LATC7 = 1; LATC6 = 1; LATC5 = 1; LATC4 = 1; LATC2 = 0;
243     LATC3 = 1; LATD0 = 1; LATD1 = 1; LATD3 = 1; LATD2 = 1;
244     break;
245
246 case 21: //RC3 CN6-4
247     LATC7 = 1; LATC6 = 1; LATC5 = 1; LATC4 = 1; LATC2 = 1;
248     LATC3 = 0; LATD0 = 1; LATD1 = 1; LATD3 = 1; LATD2 = 1;
249     break;
250
251 case 22: //RD0 CN6-3
252     LATC7 = 1; LATC6 = 1; LATC5 = 1; LATC4 = 1; LATC2 = 1;
253     LATC3 = 1; LATD0 = 0; LATD1 = 1; LATD3 = 1; LATD2 = 1;
254     break;
255
256 case 23: //RD1 CN6-2
257     LATC7 = 1; LATC6 = 1; LATC5 = 1; LATC4 = 1; LATC2 = 1;
258     LATC3 = 1; LATD0 = 1; LATD1 = 0; LATD3 = 1; LATD2 = 1;
259     break;
260
261 case 24: //RD3 CN7-9
262     LATC7 = 1; LATC6 = 1; LATC5 = 1; LATC4 = 1; LATC2 = 1;
263     LATC3 = 1; LATD0 = 1; LATD1 = 1; LATD3 = 0; LATD2 = 1;
264     break;
265
266 case 25: //RD2 CN7-10
267     LATC7 = 1; LATC6 = 1; LATC5 = 1; LATC4 = 1; LATC2 = 0;
268     LATC3 = 1; LATD0 = 1; LATD1 = 1; LATD3 = 1; LATD2 = 0;
269     break;
270
271 default: break;
272 }
273
274 //Function definition F3, buzzer signal output
275 void buzzer(void) //F3
276 {
277     PR2 = 0xFF;
278     TMR2IE = 1;
279     TMR2ON = 1;
280     while(buzzer_flag)
281     {
282         LATE2 = 1;
283         delay_us(BUZZER_TONE);
284         LATE2 = 0;
285         delay_us(BUZZER_TONE);
286     }
287 }
```