

```

1  /*-----*/
2  /*  ものづくりコンテスト山形県大会08 電子回路組立部門 */
3  /*-----*/
4  /* 全工長協会HP掲示 制御プログラム課題(例) */
5  /*-----*/
6  /* 課題3 物体の位置に相当する数値を7セグメントLEDで点灯させる。 */
7  /*-----*/
8  /* センサーの前に物体を置き、その位置の十の位の数値を7セグメントLEDで */
9  /* 点灯させる。次に物体を移動させた時に7セグメントLEDの数値も変化させる。 */
10 /* この時LED0~7は消灯しておく。範囲、測定位置は課題2と同じとする。 */
11 /*-----*/
12 /*      範囲(目安)      測定位置      7セグメントLED */
13 /*      10      15cm      10cm      0 */
14 /*      15      25cm      20cm      1 */
15 /*      25      35cm      30cm      2 */
16 /*      35      45cm      40cm      3 */
17 /*      45      55cm      50cm      4 */
18 /*      55      65cm      60cm      5 */
19 /*      65      75cm      70cm      6 */
20 /*      75      80cm      80cm      7 */
21 /*                               80cm以上      消灯 */
22 /*-----*/
23 /* 初期状態 */
24 /* (1) センサは1m以上前に何もおいていない。 */
25 /* (2) トグルスイッチはLの位置にある。 */
26 /* (3) タクトスイッチは押されていない。 */
27 /* (4) 光センサは光を受光している。 */
28 /* (5) タクトスイッチ動作による状態変化は、特に指定がない限り押した瞬間 */
29 /*     に行うものとする。 */
30 /*-----*/
31 /* 今後確認 */
32 /* 距離のデータはあらかじめ測定したものを使用した。 */
33 /*-----*/
34 /*                               山形電波工業高等学校 情報技術科 & 電子工学科 */
35 /*                               Ver. 1.0 平成20年5月 6日(火) */
36 /*                               Ver. 2.0 平成20年5月10日(土) */
37 /*-----*/
38 /*-----*/
39 /*#include "comment08.h" */
40 /*-----*/
41 /*-----*/
42 /* 組み込みファイル定義 */
43 /*-----*/
44 #include <3048.h>
45 #include "macro08.h"
46
47
48 /*-----*/
49 /* グローバル変数宣言(どの関数からも参照、変更できる) */
50 /*-----*/
51
52
53 /*-----*/
54 /* このプログラム中で使用する関数の記述 */
55 /*-----*/
56 /*#include "dempalib08.h" */
57
58
59 /*-----*/
60 /* モジュール名 Init_Port */
61 /* 処理概要 H8 ポート初期化 */
62 /* 引数 なし */
63 /* 戻り値 なし */
64 /*-----*/
65 void Init_Port( void )
66 {
67     /* I/Oポート設定 0:入力 1:出力 */
68     P6.DDR = 0x00; /* on bord デイップスイッチ */
69                 /* 0x00 = 0000 0000 全bit入力設定 */
70
71     P4.DDR = 0xff; /* 事前配布基板(1) LED & 7segLED */
72                 /* 0xff = 1111 1111 全bit出力設定 */
73     P4.DR.BYTE = 0x00;
74
75     PA.DDR = 0xff; /*
76                 /* 0x00 = 0000 0000 全bit入力設定 */
77     PA.DR.BYTE = 0x00;
78
79     PB.DDR = 0xff; /*
80                 /* 0xff = 1111 1111 全bit出力設定 */
81     PB.DR.BYTE = 0x00;
82
83     P5.DDR = 0xff; /* 事前配布基板(1) LED or 7segLED指示端子 */
84                 /* 0xff = 1111 1111 全bit出力設定 */
85     P5.DR.BYTE = 0x00;
86

```

```

87                                     /* 当日作成基板はP7ポート(P7は入力専用なので設定なし) */
88 }
89
90
91 /*-----*/
92 /* モジュール名 wait */
93 /* 処理概要 ソフトウェアタイマー */
94 /* 引数 タイマー値 1: 10[ms] */
95 /* 10: 100[ms] = 0.1[s] */
96 /* 50: 500[ms] = 0.5[s] */
97 /* 100:1000[ms] = 1.0[s] */
98 /* 戻り値 なし */
99 /*-----*/
100 void wait( int iTimer )
101 {
102     int i;
103     while( iTimer ) {
104         for( i = 0; i < 5000; i++ ); /* H8/3048f */
105         /*for( i = 0; i < 8333; i++ );*/ /* H8/3048f-one */
106         iTimer--;
107     }
108 }
109
110
111 /*-----*/
112 /* モジュール名 seq_cls */
113 /* 処理概要 7セグメントLED消去 */
114 /* 引数 なし */
115 /* 戻り値 なし */
116 /*-----*/
117 void seq_cls( void )
118 {
119     KIBAN081= 0xff; /* KIBAN081 = ~0x00; */
120 }
121
122
123
124 /*-----*/
125 /* モジュール名 opening */
126 /* 処理概要 オープニング(山形電波工高固有) */
127 /* 引数 なし */
128 /* 戻り値 なし */
129 /*-----*/
130 void opening08( void )
131 {
132     signed char i; /* -128 ~ +127 */
133
134     SEG_SELECT = T7SEG; /* 表示は7セグメントLED */
135     for( i = 3; i >= 0; i-- ) {
136         KIBAN081= tendo[ i ]; /* 表示 */
137         wait( 50 );
138         seq_cls(); /* 消去 */
139         wait( 50 );
140     }
141 }
142
143
144 /*-----*/
145 /* モジュール名 Init_ad */
146 /* 処理概要 A D 初期化 */
147 /* 引数 なし */
148 /* 戻り値 なし */
149 /*-----*/
150 void Init_ad( void )
151 {
152     /* ADコントロールレジスタ設定 */
153     AD.CSR.BYTE = 0x00; /* ch0(AN0 (P70) ) 単一モード */
154     /*AD.CSR.BYTE = 0x01;*/ /* ch1(AN1 (P71) ) 単一モード */
155     /*AD.CSR.BYTE = 0x02;*/ /* ch2(AN2 (P72) ) 単一モード */
156     /*AD.CSR.BYTE = 0x03;*/ /* ch3(AN3 (P73) ) 単一モード */
157     /*AD.CSR.BYTE = 0x04;*/ /* ch4(AN4 (P74) ) 単一モード */
158     /*AD.CSR.BYTE = 0x05;*/ /* ch5(AN5 (P75) ) 単一モード */
159     /*AD.CSR.BYTE = 0x06;*/ /* ch6(AN6 (P76) ) 単一モード */
160     /*AD.CSR.BYTE = 0x07;*/ /* ch7(AN7 (P77) ) 単一モード */
161 }
162 }
163
164
165 /*-----*/
166 /* モジュール名 start_ad */
167 /* 処理概要 A D 変換スタート */
168 /* 引数 なし */
169 /* 戻り値 なし */
170 /*-----*/
171 void start_ad( void )
172 {

```

```

173         AD.CSR.BIT.ADST = 1; /* AD変換開始 */
174
175     }
176
177
178     /*-----*/
179     /* モジュール名 end_ad */
180     /* 処理概要     A D 変換終了 */
181     /* 引数         なし */
182     /* 戻り値       なし */
183     /*-----*/
184     void end_ad( void )
185     {
186         while( AD.CSR.BIT.ADF != 1 ){ /* AD変換終了チェック */
187
188         }
189
190
191     /*-----*/
192     /* モジュール名 get_ad8 */
193     /* 処理概要     データ読み込み */
194     /*               上位10bitにデータが格納されるので、下位8bitにシフトし(右に8bit) */
195     /*               変換データを確定する。(下位2bitゴミとしてカット) */
196     /* 引数         なし */
197     /* 戻り値       8bit data */
198     /*               1111 1111 11-- ---- : 変換後データ */
199     /*               ---- ---- 1111 1111 : 8bitシフト後データ */
200     /*               ---- ---- 0000 0000 : 0x00      0 */
201     /*               :                          :      */
202     /*               ---- ---- 1111 1111 : 0xff      255 0~255 の 256通りのデータ */
203     /*-----*/
204     unsigned char get_ad8( void )
205     {
206         unsigned char indata; /* unsigned intで 0 ~ 255 まで扱える */
207
208         indata = AD.DRA >>8; /* 上位10bitを右に8bitシフトする。下位2bit無視 */
209                             /* 1111 1111 1100 0000 */
210                             /* ---- ---- 1111 1111 */
211
212         /* indata = AD.DRB >>8; */ /* 上位10bitを右に8bitシフトする。 */
213         /* indata = AD.DRC >>8; */ /* 上位10bitを右に8bitシフトする。 */
214         /* indata = AD.DRD >>8; */ /* 上位10bitを右に8bitシフトする。 */
215         //indata = indata & 0xff;
216
217         return indata;
218     }
219 }
220
221
222 /*-----*/
223 /* モジュール名 main */
224 /* 処理概要     メイン処理 */
225 /* 引数         なし */
226 /* 戻り値       なし */
227 /*-----*/
228 int main( void )
229 {
230     unsigned char psd_data;
231
232     /*-----*/
233     /* H8のポート初期化 */
234     /*-----*/
235     Init_Port();
236
237     /*-----*/
238     /* オープニング(山形電波固有) */
239     /*-----*/
240     opening08(); /* プログラムスタートの知らせ */
241
242
243     /*-----*/
244     /* 課題処理プログラム */
245     /*-----*/
246
247     /*-----*/
248     /* AD初期化 */
249     /*-----*/
250     Init_ad();
251
252     /*-----*/
253     /* 表示先設定 */
254     /*-----*/
255     SEG_SELECT = T7SEG; /* 表示先は7segLED */
256
257     while(1) {

```

```
259
260 start_ad();          /* AD変換スタート */
261 end_ad();            /* AD変換終了チェック */
262 psd_data = get_ad8(); /* psd読み込み */
263
264
265     if( psd_data >= 117 ) {
266         KIBAN081 = tendo[ 1 ];
267
268     } else if( psd_data >= 80 ) {
269         KIBAN081 = tendo[ 2 ];
270
271     } else if( psd_data >= 54 ) {
272         KIBAN081 = tendo[ 3 ];
273
274     } else if( psd_data >= 41 ) {
275         KIBAN081 = tendo[ 4 ];
276
277     } else if( psd_data >= 30 ) {
278         KIBAN081 = tendo[ 5 ];
279
280     } else if( psd_data >= 26 ) {
281         KIBAN081 = tendo[ 6 ];
282
283     } else if( psd_data >= 23 ) {
284         KIBAN081 = tendo[ 7 ];
285
286     } else if( psd_data >= 20 ) {
287         KIBAN081 = tendo[ 8 ];
288
289     } else {
290         KIBAN081 = ~0x00;
291     }
292
293     /* end of while(1) */
294
295 }
```