

ドットマトリクス LED 基板 C 言語電光掲示板プログラム 解説マニュアル

第 1.02A 版

2015 年 4 月 20 日

株式会社日立ドキュメントソリューションズ

注意事項 (rev.6.0H)

著作権

- ・本マニュアルに関する著作権は株式会社日立ドキュメントソリューションズに帰属します。
- ・本マニュアルは著作権法および、国際著作権条約により保護されています。

禁止事項

ユーザーは以下の内容を行うことはできません。

- ・第三者に対して、本マニュアルを販売、販売を目的とした宣伝、使用、営業、複製などを行うこと
- ・第三者に対して、本マニュアルの使用権を譲渡または再承諾すること
- ・本マニュアルの一部または全部を改変、除去すること
- ・本マニュアルを無許可で翻訳すること
- ・本マニュアルの内容を使用しての、人命や人体に危害を及ぼす恐れのある用途での使用

転載、複製

本マニュアルの転載、複製については、文書による株式会社日立ドキュメントソリューションズの事前の承諾が必要です。

責任の制限

本マニュアルに記載した情報は、正確を期すため、慎重に制作したのですが万一本マニュアルの記述誤りに起因する損害が生じた場合でも、株式会社日立ドキュメントソリューションズはその責任を負いません。

その他

- ・本マニュアルに記載の情報は本マニュアル発行時点のものであり、株式会社日立ドキュメントソリューションズは、予告なしに、本マニュアルに記載した情報または仕様を変更することがあります。製作に当たりますは、最新の内容を確認いただきますようお願いいたします。
- ・すべての商標および登録商標は、それぞれの所有者に帰属します。

連絡先

株式会社 日立ドキュメントソリューションズ

〒135-0016 東京都江東区東陽六丁目3番2号 イースト21タワー

E-mail : himdx.m-carrally.dd@hitachi.com

目次

1. 概要.....	1
2. ワークスペースのインストール.....	2
3. プログラム解説「mini_mcr.c」.....	3
3.1 プログラムリスト.....	3
3.2 メインプログラムを説明する前に.....	15
3.3 R8C/35A の内蔵周辺機能の初期化：init 関数.....	16
3.4 割り込みプログラム：intTRBIC 関数.....	16
3.5 ドットマトリクス LED 表示：dotmatrix 関数.....	17
3.6 メインプログラム：main 関数.....	23
4. 仕様.....	24
4.1 仕様.....	24
4.2 回路図.....	25
4.3 ポート表.....	27
4.4 ピン配置図.....	27

1. 概要

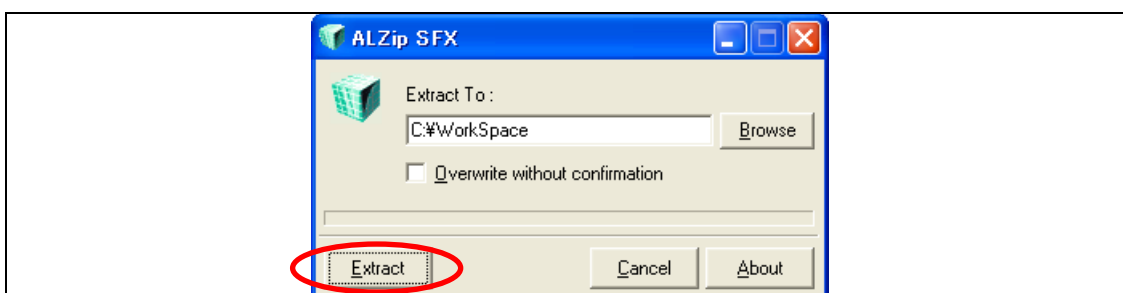
本書では、ドットマトリクス LED 基板を使用した、C 言語電光掲示板プログラムの解説を行います。

開発環境の構築方法やプログラムのビルド、書き込みについては、「ミニマイコンカーVer.2 C 言語走行プログラム解説マニュアル」の「3. インストール」「4. ミニマイコンカーVer.2 の動作確認」を参照してください。

2. ワークスペースのインストール

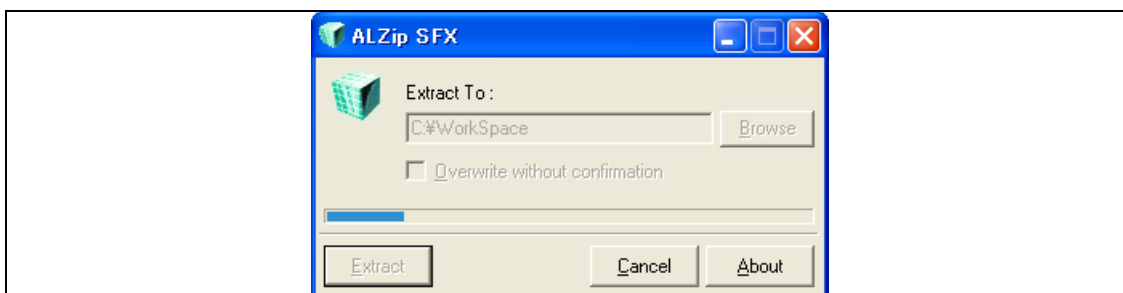
株式会社日立ドキュメントソリューションズのマイコンカーラリー販売ページからワークスペースのインストーラー「mini_mcr2_dotmatrix_vxxx.exe」（xxx はバージョン）をダウンロードします。

ダウンロードした「mini_mcr2_dotmatrix_vxxx.exe」をダブルクリックし、インストーラーを実行します。

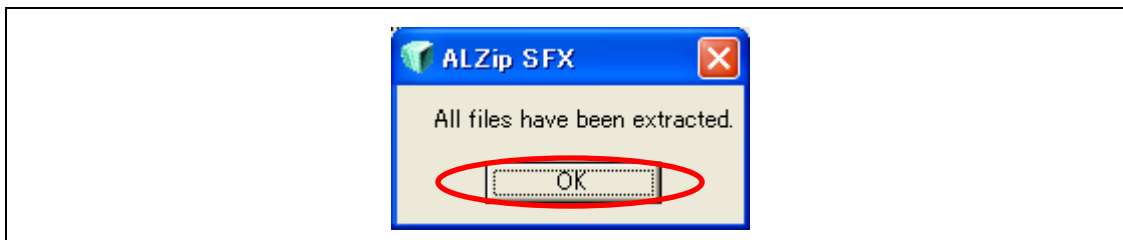


表示されたデフォルトのインストール先のフォルダ「c:\Workspace」を確認して、「Extract」をクリックします。

《補足》別のフォルダを選択する場合は、「Browse」をクリックしてください。



インストールが開始されます。



ワークスペースのインストールが完了しました。「OK」をクリックします。

以上でワークスペースのインストールは完了です。

3. プログラム解説「mini_mcr.c」

Workspace のファイルをそのままコンパイルすると、「ミニマイコンカー製作キット Ver.2」と表示されるだけです。プログラムを変更して、別の文字を表示したり、スクロールの方向を変えたり、工夫してみましょう。

3.1 プログラムリスト

```
1 : //-----  
2 : // 対象マイコン R8C/35A  
3 : // ファイル内容 電光掲示板プログラム  
4 : // バージョン Ver. 1. 00  
5 : // Date 2009. 10. 02  
6 : // Copyright ルネサスマイコンカーラリー事務局  
7 : // 日立インターメディックス株式会社  
8 : //-----  
9 : //-----  
10 : // インクルード  
11 : //-----  
12 : #include "sfr_r835a.h"  
13 :  
14 : //-----  
15 : // シンボル定義  
16 : //-----  
17 : #define TIMER_CYCLE 155 // 1ms:0. 001/(1/(2000000/128))-1  
18 : #define PWM_CYCLE 39999 // 16ms:0. 016/(1/(2000000/8))-1  
19 :  
20 : #define Def_500Hz 4999 // 500Hz:(1/500)/(1/(2000000/8))-1  
21 : #define Def_1000Hz 2499 // 1000Hz:(1/1000)/(1/(2000000/8))-1  
22 :  
23 : #define Def_C3 19083 // ド:(1/131)/(1/(2000000/8))-1  
24 : #define Def_D3 17006 // レ:(1/147)/(1/(2000000/8))-1  
25 : #define Def_E3 15151 // ミ:(1/165)/(1/(2000000/8))-1  
26 : #define Def_F3 14285 // ファ:(1/175)/(1/(2000000/8))-1  
27 : #define Def_G3 12754 // ソ:(1/196)/(1/(2000000/8))-1  
28 : #define Def_A3 11362 // ラ:(1/220)/(1/(2000000/8))-1  
29 : #define Def_B3 10120 // シ:(1/247)/(1/(2000000/8))-1  
30 : #define Def_C4 9541 // ド:(1/262)/(1/(2000000/8))-1  
31 :  
32 : #define DI() asm("FCLR I") // 割り込み禁止  
33 : #define EI() asm("FSET I") // 割り込み許可  
34 :  
35 : //-----  
36 : // 関数プロトタイプの宣言  
37 : //-----  
38 : void init( void );  
39 : unsigned char sensor( void );  
40 : void motor( int data1, int data2 );  
41 : void timer( unsigned long timer_set );  
42 : void beep( int data1 );  
43 : unsigned char dipsw( void );  
44 : unsigned char pushsw( void );  
45 : void dotmatrix( unsigned short *data1 );  
46 :  
47 : //-----
```

3. プログラム解説「mini_mcr.c」

```

48 : // グローバル変数の宣言
49 : //-----
50 : unsigned long   cnt0 = 0;           // timer 関数用
51 : unsigned long   cnt1 = 0;           // main 内で使用
52 : int             pattern = 0;        // パターン番号
53 :
54 : int             scroll;
55 :
56 : //22 文字
57 : const unsigned short str0[16*22] = {
58 :
59 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
60 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
61 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
62 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
63 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
64 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
65 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
66 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
67 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
68 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
69 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
70 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
71 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
72 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
73 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
74 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
75 :
76 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
77 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
78 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
79 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
80 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
81 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
82 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
83 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
84 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
85 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
86 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
87 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
88 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
89 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
90 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
91 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
92 :
93 : 0x0844, //○○○○●○○○○●○○○●○○○
94 : 0x0844, //○○○○●○○○○●○○○●○○○
95 : 0x1088, //○○○●○○○○●○○○●○○○
96 : 0x1088, //○○○●○○○○●○○○●○○○
97 : 0x1088, //○○○●○○○○●○○○●○○○
98 : 0x2110, //○○●○○○○●○○○●○○○
99 : 0x2000, //○○●○○○○○○○○○○○○○○○○○○
100 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
101 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
102 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
103 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
104 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
105 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
106 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
107 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
108 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
109 :
110 : 0x1000, //○○○●○○○○○○○○○○○○○○○○○○

```


3. プログラム解説「mini_mcr.c」

```

111 : 0x1008, //○○○●○○○○○○○○●○○○
112 : 0x1008, //○○○●○○○○○○○○●○○○
113 : 0x1008, //○○○●○○○○○○○○●○○○
114 : 0x1008, //○○○●○○○○○○○○●○○○
115 : 0x1008, //○○○●○○○○○○○○●○○○
116 : 0x1008, //○○○●○○○○○○○○●○○○
117 : 0x1008, //○○○●○○○○○○○○●○○○
118 : 0x1000, //○○○●○○○○○○○○○○○○○○
119 : 0x0000, //○○○○○○○○○○○○○○○○○○
120 : 0x0000, //○○○○○○○○○○○○○○○○○○
121 : 0x0000, //○○○○○○○○○○○○○○○○○○
122 : 0x0000, //○○○○○○○○○○○○○○○○○○
123 : 0x0000, //○○○○○○○○○○○○○○○○○○
124 : 0x0000, //○○○○○○○○○○○○○○○○○○
125 : 0x0000, //○○○○○○○○○○○○○○○○○○
126 :
127 : 0x0008, //○○○○○○○○○○○○○○●○○○
128 : 0x0108, //○○○○○○○○●○○○○●○○○
129 : 0x0108, //○○○○○○○○●○○○○●○○○
130 : 0x0208, //○○○○○○●○○○○●○○○
131 : 0x0408, //○○○○●○○○○●○○○
132 : 0x0A08, //○○○○●●○○○○●○○○
133 : 0x3108, //○○●●○○●●○○○○●○○○
134 : 0x00C8, //○○○○○○○○●●○○●○○○
135 : 0x0038, //○○○○○○○○○●●●○○○
136 : 0x0000, //○○○○○○○○○○○○○○○○○○
137 : 0x0000, //○○○○○○○○○○○○○○○○○○
138 : 0x0000, //○○○○○○○○○○○○○○○○○○
139 : 0x0000, //○○○○○○○○○○○○○○○○○○
140 : 0x0000, //○○○○○○○○○○○○○○○○○○
141 : 0x0000, //○○○○○○○○○○○○○○○○○○
142 : 0x0000, //○○○○○○○○○○○○○○○○○○
143 :
144 : 0x0100, //○○○○○○○○●○○○○○○○○
145 : 0x0100, //○○○○○○○○●○○○○○○○○
146 : 0x0080, //○○○○○○○○●○○○○○○○○
147 : 0x0080, //○○○○○○○○●○○○○○○○○
148 : 0x0040, //○○○○○○○○●○○○○○○○○
149 : 0x3FE0, //○○●●●●●●●●○○○○○○
150 : 0x0010, //○○○○○○○○○○○○●○○○
151 : 0x0008, //○○○○○○○○○○○○●○○○
152 : 0x0004, //○○○○○○○○○○○○●○○○
153 : 0x0000, //○○○○○○○○○○○○○○○○○○
154 : 0x0000, //○○○○○○○○○○○○○○○○○○
155 : 0x0000, //○○○○○○○○○○○○○○○○○○
156 : 0x0000, //○○○○○○○○○○○○○○○○○○
157 : 0x0000, //○○○○○○○○○○○○○○○○○○
158 : 0x0000, //○○○○○○○○○○○○○○○○○○
159 : 0x0000, //○○○○○○○○○○○○○○○○○○
160 :
161 : 0x1008, //○○○●○○○○○○○○●○○○
162 : 0x1008, //○○○●○○○○○○○○●○○○
163 : 0x1008, //○○○●○○○○○○○○●○○○
164 : 0x1008, //○○○●○○○○○○○○●○○○
165 : 0x1008, //○○○●○○○○○○○○●○○○
166 : 0x1008, //○○○●○○○○○○○○●○○○
167 : 0x1008, //○○○●○○○○○○○○●○○○
168 : 0x1008, //○○○●○○○○○○○○●○○○
169 : 0x3FF8, //○○●●●●●●●●●●○○○○
170 : 0x0000, //○○○○○○○○○○○○○○○○○○
171 : 0x0000, //○○○○○○○○○○○○○○○○○○
172 : 0x0000, //○○○○○○○○○○○○○○○○○○
173 : 0x0000, //○○○○○○○○○○○○○○○○○○

```

3. プログラム解説「mini_mcr.c」

```

174 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
175 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
176 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
177 :
178 : 0x2008, //○○●○○○○○○○○○○●○○○
179 : 0x2008, //○○●○○○○○○○○○○●○○○
180 : 0x2010, //○○●○○○○○○○○○○●○○○
181 : 0x1020, //○○○●○○○○○○○○●○○○○○
182 : 0x1000, //○○○●○○○○○○○○○○○○○○○
183 : 0x0800, //○○○○●○○○○○○○○○○○○○○○
184 : 0x0400, //○○○○○●○○○○○○○○○○○○○○○
185 : 0x0300, //○○○○○○●●○○○○○○○○○○○○○
186 : 0x00E0, //○○○○○○○○●●●○○○○○○○○○○○
187 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
188 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
189 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
190 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
191 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
192 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
193 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
194 :
195 : 0x2010, //○○●○○○○○○○○○○●○○○○○
196 : 0x1810, //○○○●●○○○○○○○○●○○○○○
197 : 0x0610, //○○○○○●●○○○○○●○○○○○
198 : 0x01FE, //○○○○○○○●●●●●●●●●○○○
199 : 0x0010, //○○○○○○○○○○○○○○●○○○○○
200 : 0x2010, //○○●○○○○○○○○○○●○○○○○
201 : 0x2010, //○○●○○○○○○○○○○●○○○○○
202 : 0x2010, //○○●○○○○○○○○○○●○○○○○
203 : 0x1FF0, //○○○●●●●●●●●●●●○○○○○
204 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
205 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
206 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
207 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
208 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
209 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
210 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
211 :
212 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
213 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
214 : 0x0040, //○○○○○○○○○○○●○○○○○○○○○
215 : 0x0080, //○○○○○○○○○○●○○○○○○○○○
216 : 0x0080, //○○○○○○○○○○●○○○○○○○○○
217 : 0x0080, //○○○○○○○○○○●○○○○○○○○○
218 : 0x0080, //○○○○○○○○○○●○○○○○○○○○
219 : 0x0080, //○○○○○○○○○○●○○○○○○○○○
220 : 0x0080, //○○○○○○○○○○●○○○○○○○○○
221 : 0x0080, //○○○○○○○○○○●○○○○○○○○○
222 : 0x0080, //○○○○○○○○○○●○○○○○○○○○
223 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
224 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
225 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
226 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
227 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
228 :
229 : 0x0000, //○○○○○○○○○○○○○○○○○○○○○
230 : 0x1208, //○○○●○○●○○○○○○●○○○
231 : 0x12F6, //○○○●○○●●●●●●●●●●●○○○
232 : 0x2A14, //○○○●○●○●○○○○○●○○○
233 : 0x3AFE, //○○○●●●●●●●●●●●●●○○○
234 : 0x2614, //○○○●○○●○○○○○○●○○○
235 : 0x1294, //○○○●○○●●●●●●●●●○○○
236 : 0x13F4, //○○○●○○●●●●●●●●●○○○

```

3. プログラム解説「mini_mcr.c」

```

237 : 0x0600, //○○○○○●○○○○○○○○○
238 : 0x0A7C, //○○○●○○●●●●○○○
239 : 0x1A00, //○○○●●○○●○○○○○○○
240 : 0x1680, //○○○●●●●○○○○○○○
241 : 0x22FE, //○○●○○○●●●●●●●○
242 : 0x2200, //○○●○○○●○○○○○○○
243 : 0x0000, //○○○○○○○○○○○○○○○
244 : 0x0000, //○○○○○○○○○○○○○○○
245 :
246 : 0x0000, //○○○○○○○○○○○○○○○
247 : 0x0040, //○○○○○○○○○○●○○○○○
248 : 0x0020, //○○○○○○○○○○●○○○○○
249 : 0x3FF8, //○○●●●●●●●●●●○○○
250 : 0x0006, //○○○○○○○○○○○○●●○
251 : 0x0040, //○○○○○○○○○○●○○○○○
252 : 0x0030, //○○○○○○○○○○●●○○○
253 : 0x000E, //○○○○○○○○○○○○●●○
254 : 0x0008, //○○○○○○○○○○○○●○○○
255 : 0x3FF8, //○○●●●●●●●●●●○○○
256 : 0x0248, //○○○○○○○●○○●○○○
257 : 0x0248, //○○○○○○○●○○●○○○
258 : 0x0248, //○○○○○○○●○○●○○○
259 : 0x0248, //○○○○○○○●○○●○○○
260 : 0x0000, //○○○○○○○○○○○○○○○
261 : 0x0000, //○○○○○○○○○○○○○○○
262 :
263 : 0x0220, //○○○○○○○●○○○●○○○
264 : 0x0220, //○○○○○○○●○○○●○○○
265 : 0x0220, //○○○○○○○●○○○●○○○
266 : 0x0116, //○○○○○○○●○○○●●○
267 : 0x0178, //○○○○○○○●●●●○○○
268 : 0x0790, //○○○○○●●●●○○○○○
269 : 0x3910, //○○●●●○○●○○●○○○
270 : 0x0108, //○○○○○○○●○○○○●○○○
271 : 0x0088, //○○○○○○○●○○○●○○○
272 : 0x0080, //○○○○○○○●○○○○○○○
273 : 0x0080, //○○○○○○○●○○○○○○○
274 : 0x0000, //○○○○○○○○○○○○○○○
275 : 0x0000, //○○○○○○○○○○○○○○○
276 : 0x0000, //○○○○○○○○○○○○○○○
277 : 0x0000, //○○○○○○○○○○○○○○○
278 : 0x0000, //○○○○○○○○○○○○○○○
279 :
280 : 0x0040, //○○○○○○○○○○●○○○○○
281 : 0x2180, //○○●○○○○●●○○○○○
282 : 0x2020, //○○●○○○○○●○○○○○
283 : 0x10C0, //○○○●○○○○●○○○○○
284 : 0x0800, //○○○○●○○○○○○○○○
285 : 0x0600, //○○○○○●●○○○○○○○
286 : 0x01E0, //○○○○○○○●●●●○○○
287 : 0x0000, //○○○○○○○○○○○○○○○
288 : 0x0000, //○○○○○○○○○○○○○○○
289 : 0x0000, //○○○○○○○○○○○○○○○
290 : 0x0000, //○○○○○○○○○○○○○○○
291 : 0x0000, //○○○○○○○○○○○○○○○
292 : 0x0000, //○○○○○○○○○○○○○○○
293 : 0x0000, //○○○○○○○○○○○○○○○
294 : 0x0000, //○○○○○○○○○○○○○○○
295 : 0x0000, //○○○○○○○○○○○○○○○
296 :
297 : 0x0000, //○○○○○○○○○○○○○○○
298 : 0x3FFE, //○○●●●●●●●●●●○○○
299 : 0x0020, //○○○○○○○○○○●○○○○○

```

3. プログラム解説「mini_mcr.c」

```
300 : 0x0020, //○○○○○○○○○○●○○○○○
301 : 0x0040, //○○○○○○○○○○●○○○○○
302 : 0x0040, //○○○○○○○○○○●○○○○○
303 : 0x0080, //○○○○○○○○○○●○○○○○
304 : 0x0100, //○○○○○○○○●○○○○○○○
305 : 0x0000, //○○○○○○○○○○○○○○○○○
306 : 0x0000, //○○○○○○○○○○○○○○○○○
307 : 0x0000, //○○○○○○○○○○○○○○○○○
308 : 0x0000, //○○○○○○○○○○○○○○○○○
309 : 0x0000, //○○○○○○○○○○○○○○○○○
310 : 0x0000, //○○○○○○○○○○○○○○○○○
311 : 0x0000, //○○○○○○○○○○○○○○○○○
312 : 0x0000, //○○○○○○○○○○○○○○○○○
313 :
314 : 0x0008, //○○○○○○○○○○○○●○○○
315 : 0x0070, //○○○○○○○○○○●●●○○○
316 : 0x0180, //○○○○○○○○●●●○○○○○
317 : 0x0E00, //○○○○●●●○○○○○○○○○
318 : 0x1000, //○○○●○○○○○○○○○○○○○
319 : 0x0E00, //○○○○●●●○○○○○○○○○
320 : 0x0180, //○○○○○○○○●●●○○○○○
321 : 0x0070, //○○○○○○○○○○●●●○○○
322 : 0x0008, //○○○○○○○○○○○○●○○○
323 : 0x0000, //○○○○○○○○○○○○○○○○○
324 : 0x0000, //○○○○○○○○○○○○○○○○○
325 : 0x0000, //○○○○○○○○○○○○○○○○○
326 : 0x0000, //○○○○○○○○○○○○○○○○○
327 : 0x0000, //○○○○○○○○○○○○○○○○○
328 : 0x0000, //○○○○○○○○○○○○○○○○○
329 : 0x0000, //○○○○○○○○○○○○○○○○○
330 :
331 : 0x0F80, //○○○○●●●●●○○○○○○○
332 : 0x1240, //○○○●○○●○○●○○○○○
333 : 0x1240, //○○○●○○●○○●○○○○○
334 : 0x1240, //○○○●○○●○○●○○○○○
335 : 0x1240, //○○○●○○●○○●○○○○○
336 : 0x0B80, //○○○○●●●●●○○○○○○○
337 : 0x0000, //○○○○○○○○○○○○○○○○○
338 : 0x0000, //○○○○○○○○○○○○○○○○○
339 : 0x0000, //○○○○○○○○○○○○○○○○○
340 : 0x0000, //○○○○○○○○○○○○○○○○○
341 : 0x0000, //○○○○○○○○○○○○○○○○○
342 : 0x0000, //○○○○○○○○○○○○○○○○○
343 : 0x0000, //○○○○○○○○○○○○○○○○○
344 : 0x0000, //○○○○○○○○○○○○○○○○○
345 : 0x0000, //○○○○○○○○○○○○○○○○○
346 : 0x0000, //○○○○○○○○○○○○○○○○○
347 :
348 : 0x0000, //○○○○○○○○○○○○○○○○○
349 : 0x1FC0, //○○○●●●●●●○○○○○○○
350 : 0x0080, //○○○○○○○○○○●○○○○○
351 : 0x0040, //○○○○○○○○○○●○○○○○
352 : 0x0040, //○○○○○○○○○○●○○○○○
353 : 0x0000, //○○○○○○○○○○○○○○○○○
354 : 0x0000, //○○○○○○○○○○○○○○○○○
355 : 0x0000, //○○○○○○○○○○○○○○○○○
356 : 0x0000, //○○○○○○○○○○○○○○○○○
357 : 0x0000, //○○○○○○○○○○○○○○○○○
358 : 0x0000, //○○○○○○○○○○○○○○○○○
359 : 0x0000, //○○○○○○○○○○○○○○○○○
360 : 0x0000, //○○○○○○○○○○○○○○○○○
361 : 0x0000, //○○○○○○○○○○○○○○○○○
362 : 0x0000, //○○○○○○○○○○○○○○○○○
```


3. プログラム解説「mini_mcr.c」

```
426 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
427 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
428 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
429 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
430 : 0x0000, //○○○○○○○○○○○○○○○○○○○○
431 : 0x0000 //○○○○○○○○○○○○○○○○○○○○
432 :
433 : };
434 :
435 :
436 : //-----
437 : // メインプログラム
438 : //-----
439 : void main(void)
440 : {
441 :     // 初期化
442 :     init();
443 :
444 :     cnt1 = 0;
445 :     scroll = 0;
446 :
447 :     while(1){
448 :         // 25ms を超えたら
449 :         if( cnt1 >= 25 ){
450 :
451 :             cnt1 = 0;                // タイマーカウンタをクリアする
452 :             scroll++;                // スクロールさせるための変数を+1 する
453 :
454 :         }
455 :
456 :         // スクロールの終了位置まで来たら
457 :         if( scroll >= 16 * ( 22 - 2 ) ){
458 :             scroll = 0;                // 変数をクリアする
459 :         }
460 :     }
461 : }
462 :
463 :
464 : //-----
465 : // R8C/35A の内蔵周辺機能の初期化
466 : //-----
467 : void init( void )
468 : {
469 :     unsigned char i = 0;
470 :
471 :     // 割り込み禁止
472 :     DI();
473 :
474 :     // クロック発生回路の XIN クロック設定
475 :     prc0 = 1;
476 :
477 :     cm13 = 1;
478 :     cm05 = 0;
479 :     while(i <= 50) i++;
480 :     ocd2 = 0;
481 :
482 :     prc0 = 0;
483 :
484 :     // I/O ポートの入出力設定
485 :     prc2 = 1;                // pd0 レジスタへの書き込み許可
486 :     pd0 = 0xe0;            // P0_0~P0_3:センサー
487 :                            // P0_4:マイクロスイッチ
488 :                            // P0_5~P0_7:LED
```

3. プログラム解説 「mini_mcr.c」

```

489 :         prc2 = 0;                // pd0 レジスタへの書き込み禁止
490 :         pd1 = 0xdf;              // P1_0~P1_3:LED
491 :                                     // P1_4:TXD0
492 :                                     // P1_5:RXD0
493 :         pd2 = 0xfe;              // P2_0:スイッチ
494 :                                     // P2_1:AIN1
495 :                                     // P2_2:PWMA
496 :                                     // P2_3:BIN1
497 :                                     // P2_4:PWMB
498 :                                     // P2_5:SERVO
499 :                                     // P2_6:AIN2
500 :                                     // P2_7:BIN2
501 : //         pd3 = 0xfb;            // P3_2:赤外線受信
502 : //                                     // P3_4:ブザー
503 :         pd3 = 0x00;              // P3_0~P3_7:タクトスイッチ
504 :         pd4 = 0x80;              // P4_2:VREF
505 :                                     // P4_3~P4_5:DIPSW
506 :                                     // P4_6:XIN
507 :                                     // P4_7:XOUT
508 :         pd5 = 0x40;              // P5_7:DIPSW
509 :         pd6 = 0xff;              // P6_7:¥ENABLE¥
510 :                                     // P6_6:LATCH
511 :                                     // P6_5:¥SCLR¥
512 :                                     // P6_4:SCLK
513 :                                     // P6_3:ROWA_SIN        行 A
514 :                                     // P6_2:ROWB_SIN        行 B
515 :                                     // P6_1:COL_SIN        列
516 :                                     // P6_0:
517 :
518 :
519 :
520 :         mstcr = 0x00;            // モジュールストップ解除
521 :
522 :
523 :
524 : // タイマ RB の 1ms 割り込み設定
525 :         trbmr = 0x00;            // カウントソースは f1
526 :         trbpre = 128 - 1;        // プリスケーラ
527 :         trbpr = TIMER_CYCLE;     // プライマリカウンタ
528 :         trbic = 0x01;            // タイマ RB の割り込みレベル設定
529 :         trbcr = 0x01;            // カウントを開始
530 :
531 : // タイマ RC の PWM モード
532 : //         trcrr1 = 0xb0;          // カウントソースは f8
533 : //         trcgra = 0;            // 圧電サウンダの周期
534 : //         trcgrc = 0;           // 圧電サウンダのデューティ比
535 : //         trcrr2 = 0x02;        // TRCIOC 端子はアクティブレベル H
536 : //         trcoer = 0x0b;        // TRCIOC 端子の出力許可
537 : //         trcpsr1 = 0x02;       // TRCIOC 端子を P3_4 に割り当て
538 : //         trcmr = 0x8a;         // カウントを開始
539 :
540 : // タイマ RD のリセット同期 PWM モード
541 :         trdpsr0 = 0x08;          // TRDIOB0 端子を P2_2 に割り当て
542 :         trdpsr1 = 0x05;          // TRDIOB1 端子を P2_5 に割り当て
543 :                                     // TRDIOA1 端子を P2_4 に割り当て
544 :         trdmr = 0xf0;            // レジスタをバッファ動作にする
545 :         trdfer = 0x01;          // リセット同期 PWM モードに設定
546 :         trdoer1 = 0xcd;         // TRDIOB1 の出力許可
547 :                                     // TRDIOA1 の出力許可
548 :                                     // TRDIOB0 端子の出力許可
549 :         trdcr0 = 0x23;          // カウントソースは f8
550 :         trdgra0 = trdgrc0 = PWM_CYCLE; // 周期
551 :         trdgrb0 = trdgrd0 = 0;  // TRDIOB0 端子 (左モータ)

```

3. プログラム解説 「mini_mcr.c」

```
552 :         trdgral = trdgrcl = 0;           // TRDIOA1 端子 (右モータ)
553 :         trdgrbl = trdgrdl = 0;           // TRDIOB1 端子 (サーボ)
554 :         trdstr = 0x0d;                    // カウントを開始
555 :
556 :         // 割り込み許可
557 :         EI();
558 :     }
559 :
560 : //-----
561 : // 割り込み
562 : //-----
563 : #pragma interrupt intTRBIC (vect=24)
564 : void intTRBIC( void )
565 : {
566 :     tundf_tracr = 0;
567 :
568 :     p0_7 = ~p0_7;
569 :
570 :     if( p0_7 == 0 ){
571 :         //p0_1、p0_3 のモニタが可能
572 :         p0_5 = ~p0_1;
573 :         p0_6 = ~p0_3;
574 :     }else{
575 :         //p0_0、p0_2 のモニタが可能
576 :         p0_5 = p0_0;
577 :         p0_6 = p0_2;
578 :     }
579 :
580 :     cnt0++;
581 :     cnt1++;
582 :
583 :     // ドットマトリクス LED 表示処理
584 :     dotmatrix( &str0[scroll] );
585 :
586 : }
587 :
588 : //-----
589 : // センサー状態検出
590 : // 引数      なし
591 : // 戻り値    センサ値
592 : //-----
593 : unsigned char sensor( void )
594 : {
595 :     volatile unsigned char  data1;
596 :
597 :     data1 = ~p0;                // ラインの色は白
598 :     data1 = data1 & 0x0f;
599 :
600 :     return( data1 );
601 : }
602 :
603 : //-----
604 : // モーター速度制御
605 : // 引数      左モータ:-100~100、右モータ:-100~100
606 : //           0 で停止、100 で正転 100%、-100 で逆転 100%
607 : // 戻り値    なし
608 : //-----
609 : void motor( int data1, int data2 )
610 : {
611 :     volatile int    motor_r;
612 :     volatile int    motor_l;
613 :     volatile int    sw_data;
614 :
```


3. プログラム解説 「mini_mcr.c」

```
615 :         sw_data = dipsw() + 5;
616 :         motor_l = (long)data1 * sw_data / 20;
617 :         motor_r = (long)data2 * sw_data / 20;
618 :
619 :         if( motor_l >= 0 ) {
620 :             p2_1 = 0;
621 :             p2_6 = 1;
622 :             trdgrd0 = (long)( PWM_CYCLE - 1 ) * motor_l / 100;
623 :         } else {
624 :             p2_1 = 1;
625 :             p2_6 = 0;
626 :             trdgrd0 = (long)( PWM_CYCLE - 1 ) * ( -motor_l ) / 100;
627 :         }
628 :
629 :         if( motor_r >= 0 ) {
630 :             p2_3 = 0;
631 :             p2_7 = 1;
632 :             trdgrc1 = (long)( PWM_CYCLE - 1 ) * motor_r / 100;
633 :         } else {
634 :             p2_3 = 1;
635 :             p2_7 = 0;
636 :             trdgrc1 = (long)( PWM_CYCLE - 1 ) * ( -motor_r ) / 100;
637 :         }
638 :     }
639 :
640 : //-----
641 : // 時間稼ぎ
642 : // 引数      タイマ値 1=1ms
643 : // 戻り値    なし
644 : //-----
645 : void timer( unsigned long data1 )
646 : {
647 :     cnt0 = 0;
648 :     while( cnt0 < data1 );
649 : }
650 :
651 : //-----
652 : // 音を鳴らす
653 : // 引数      (1/音の周波数)/(1/(クロック周波数/8))-1
654 : // 戻り値    なし
655 : //-----
656 : void beep( int data1 )
657 : {
658 :     trcgra = data1;           // 周期の設定
659 :     trcgrc = data1 / 2;     // デューティ 50%のため周期の半分の値
660 : }
661 :
662 : //-----
663 : // DIP スイッチ状態検出
664 : // 引数      なし
665 : // 戻り値    0~15、DIP スイッチが ON の場合、対応するビットが 0 になります。
666 : //-----
667 : unsigned char dipsw( void )
668 : {
669 :     volatile unsigned char  data1;
670 :
671 :     data1 = ( ( p5 >> 4 ) & 0x08 ) | ( ( p4 >> 3 ) & 0x07 );
672 :
673 :     return( data1 );
674 : }
675 :
676 : //-----
677 : // プッシュスイッチ状態検出
```

3. プログラム解説 「mini_mcr.c」

```

678 : // 引数      なし
679 : // 戻り値    スイッチが押されていない場合:0、押された場合:1
680 : //-----
681 : unsigned char pushsw( void )
682 : {
683 :     unsigned char data1;
684 :
685 :     data1 = ~p2;
686 :     data1 &= 0x01;
687 :
688 :     return( data1 );
689 : }
690 :
691 : //-----
692 : // ドットマトリクス LED 表示
693 : // 引数      表示配列のポインタ
694 : // 戻り値    なし
695 : //-----
696 : void dotmatrix( unsigned short *data1 )
697 : {
698 :
699 :     // P6_7:¥ENABLE¥
700 :     // P6_6:LATCH
701 :     // P6_5:¥SCLR¥
702 :     // P6_4:SCLK
703 :     // P6_3:ROWA_SIN      行 A
704 :     // P6_2:ROWB_SIN     行 B
705 :     // P6_1:COL_SIN      列
706 :     // P6_0:
707 :
708 :     unsigned short rowa_sin;
709 :     unsigned short rowb_sin;
710 :     unsigned short col_sin;
711 :     static int x=0;
712 :     int y;
713 :
714 :     // シフトレジスタをクリアしない
715 :     p6_5 = 1;                               // ¥SCLR¥
716 :
717 :     // ドットマトリクス LED へ出力許可
718 :     p6_7 = 0;                               // ¥ENABLE¥
719 :
720 :     rowa_sin = data1[x];                    // 行 A データ
721 :     rowb_sin = data1[x+16];                // 行 B データ
722 :     col_sin = 1 << ( 15 - x );            // 列データ
723 :
724 :     for( y = 0; y < 16; y++ ){
725 :
726 :         p6_3 = (rowa_sin >> 15) & 0x0001; // ROWA_SIN
727 :         rowa_sin = rowa_sin << 1;
728 :
729 :         p6_2 = (rowb_sin >> 15) & 0x0001; // ROWB_SIN
730 :         rowb_sin = rowb_sin << 1;
731 :
732 :         p6_1 = col_sin & 0x0001;          // COL_SIN
733 :         col_sin = col_sin >> 1;
734 :
735 :         // クロックの立ち上りでシフトレジスタにロード
736 :         p6_4 = 0;                          // SCLK
737 :         p6_4 = 1;                          // SCLK
738 :
739 :     }
740 :

```

3. プログラム解説「mini_mcr.c」

```
741 : // シフトレジスタのデータをラッチ
742 :     p6_6 = 0; // LATCH
743 :     p6_6 = 1; // LATCH
744 :
745 :     x++;
746 :     if( x >= 16 ){
747 :         x = 0;
748 :     }
749 :
750 : }
```

3.2 メインプログラムを説明する前に

main 関数は、main 関数の後に記載されている関数を組み合わせてプログラムしていますので、先に main 関数以外の関数の解説を行います。

dotmatrix 以外の関数については「ミニマイコンカーVer.2 C 言語走行プログラム解説マニュアル」の「5. プログラム解説「mini_mcr.c」」を参照してください。

3.3 R8C/35A の内蔵周辺機能の初期化 : init 関数

init 関数は「ミニマイコンカーVer.2 C 言語走行プログラム解説マニュアル」で解説されています。ここでは、init 関数の変更部分のみを解説します。

```
501 : //      pd3 = 0xfb;           // P3_2:赤外線受信
502 : //      // P3_4:ブザー
503 :      pd3 = 0x00;           // P3_0~P3_7:タクトスイッチ
```

ドットマトリクス LED 基板のタクトスイッチを使用するため、ポート 3 を入力ポートにする必要がありますので、pd3 レジスタを“0x00”にします。

```
532 : //      trcrr1 = 0xb0;       // カウントソースは f8
533 : //      trcgra = 0;          // 圧電サウンダの周期
534 : //      trcgrc = 0;          // 圧電サウンダのデューティ比
535 : //      trcrr2 = 0x02;       // TRCIOC 端子はアクティブレベル H
536 : //      trcoer = 0x0b;       // TRCIOC 端子の出力許可
537 : //      trcpsr1 = 0x02;      // TRCIOC 端子を P3_4 に割り当て
538 : //      trcmr = 0x8a;        // カウントを開始
```

ドットマトリクス LED 基板のタクトスイッチを使用するため、ポート 3 を入力ポートにする必要があります。タイマー RC の初期化部分を削除して、TRCIOC 端子の出力が p3_4 端子に割り当てられないようにします。

3.4 割り込みプログラム : intTRBIC 関数

intTRBIC 関数は「ミニマイコンカーVer.2 C 言語走行プログラム解説マニュアル」で解説されています。ここでは、intTRBIC 関数の変更部分のみを解説します。

```
584 :      dotmatrix( &str0[scroll] );
```

dotmatrix 関数を呼び出す部分を追加しています。割り込みに追加しているのは、各ドットマトリクス LED で 1 列ごとに表示を行い、残像で全体が表示されているように見せるためです。割り込みは 1 [ms] ごとに行われるので、全体を表示するには 16 [ms] の時間が必要です。

3.5 ドットマトリクス LED 表示 : dotmatrix 関数

dotmatrix 関数は、引数の 16×32 のデータを各ドットマトリクス LED で 1 列ごとに表示します。表示する列は、左端の列からで、呼び出すごとに表示列はインクリメントされていきます。

プログラム

```

696 : void dotmatrix( unsigned short *datal )
697 : {
698 :
699 :     // P6_7:¥ENABLE¥
700 :     // P6_6:LATCH
701 :     // P6_5:¥SCLR¥
702 :     // P6_4:SCLK
703 :     // P6_3:ROWA_SIN      行 A
704 :     // P6_2:ROWB_SIN     行 B
705 :     // P6_1:COL_SIN      列
706 :     // P6_0:
707 :
708 :     unsigned short rowa_sin;
709 :     unsigned short rowb_sin;
710 :     unsigned short col_sin;
711 :     static int x=0;
712 :     int y;
713 :
714 :     // シフトレジスタをクリアしない
715 :     p6_5 = 1;                               // ¥SCLR¥
716 :
717 :     // ドットマトリクス LED へ出力許可
718 :     p6_7 = 0;                               // ¥ENABLE¥
719 :
720 :     rowa_sin = datal[x];                    // 行 A データ
721 :     rowb_sin = datal[x+16];                 // 行 B データ
722 :     col_sin = 1 << ( 15 - x );              // 列データ
723 :
724 :     for( y = 0; y < 16; y++ ){
725 :
726 :         p6_3 = (rowa_sin >> 15) & 0x0001;   // ROWA_SIN
727 :         rowa_sin = rowa_sin << 1;
728 :
729 :         p6_2 = (rowb_sin >> 15) & 0x0001;   // ROWB_SIN
730 :         rowb_sin = rowb_sin << 1;
731 :
732 :         p6_1 = col_sin & 0x0001;           // COL_SIN
733 :         col_sin = col_sin >> 1;
734 :
735 :         // クロックの立ち上りでシフトレジスタにロード
736 :         p6_4 = 0;                           // SCLK
737 :         p6_4 = 1;                           // SCLK
738 :
739 :     }
740 :
741 :     // シフトレジスタのデータをラッチ
742 :     p6_6 = 0;                               // LATCH
743 :     p6_6 = 1;                               // LATCH
744 :
745 :     x++;
746 :     if( x >= 16 ){
747 :         x = 0;

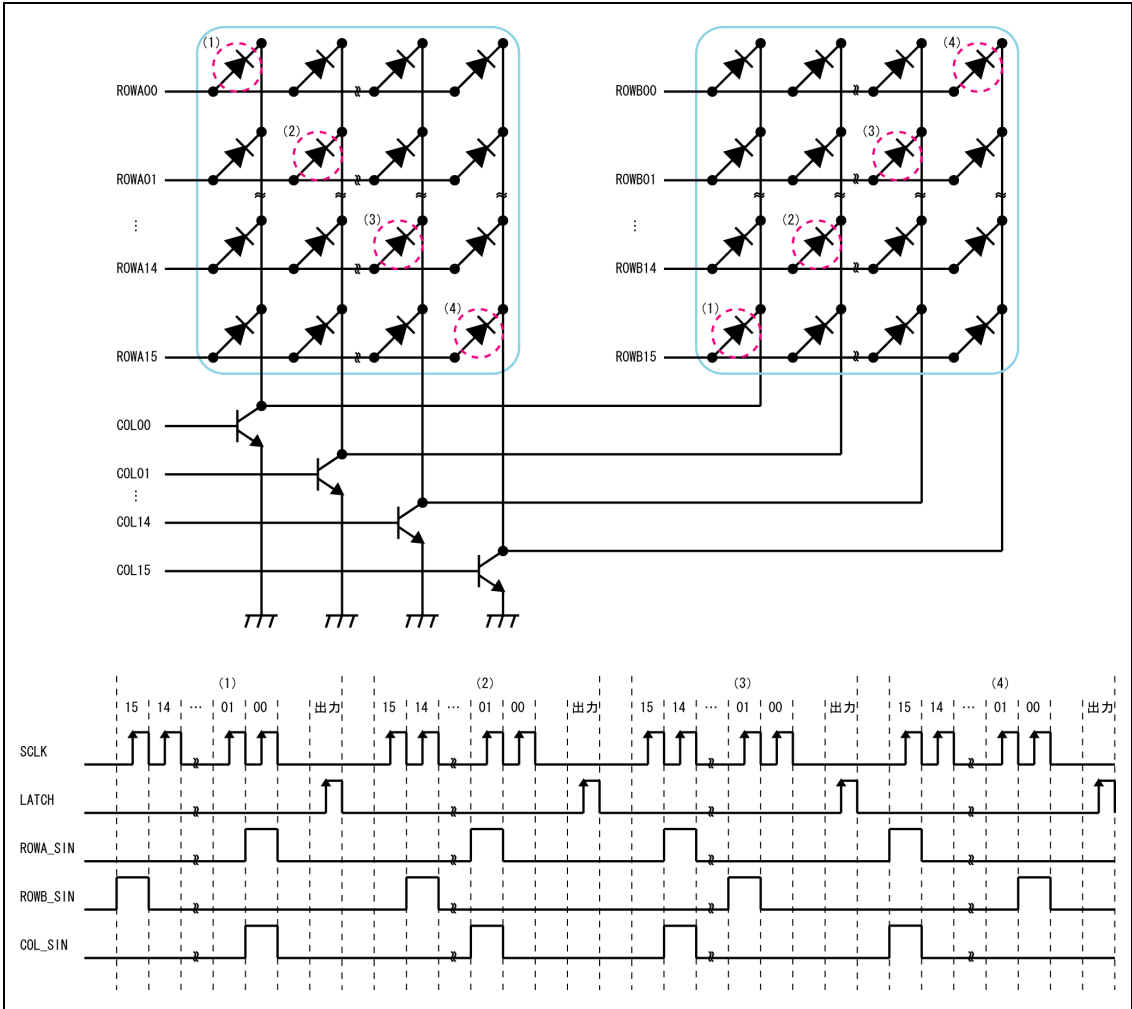
```

3. プログラム解説 「mini_mcr.c」

```

748 :     }
749 :
750 : }
    
```

回路図 (簡略図)



3. プログラム解説 「mini_mcr.c」

```
715 :      p6_5 = 1;                                // ¥SCLR¥
```

p6_5 端子を “1” にすることにより、シフトレジスタをクリアしないようにします。クリアするときは、p6_5 端子を “0” にします。

```
718 :      p6_7 = 0;                                // ¥ENABLE¥
```

p6_7 端子を “0” にすることにより、パラレルレジスタのドットマトリクス LED への出力を許可します。禁止するときは、p6_7 端子を “1” にします。

```
720 :      rowa_sin = data1[x];                      // 行 A データ
721 :      rowb_sin = data1[x+16];                  // 行 B データ
722 :      col_sin = 1 << ( 15 - x );              // 列データ
```

x=0 のとき rowa_sin 変数、rowb_sin 変数に入るデータ

0x0844, //○○○○●○○○○●○○○●○○○	←	x=0 のとき、rowa_sin 変数 に、この部分の行データが入 る。
0x0844, //○○○○●○○○○●○○○●○○○		
0x1088, //○○○●○○○○●○○○●○○○	←	x=0 のとき、rowb_sin 変数 に、この部分の行データが入 る。
0x1008, //○○○●○○○○●○○○●○○○		

0x1088, //○○○●○○○○●○○○●○○○

0x1088, //○○○●○○○○●○○○●○○○

0x1088, //○○○●○○○○●○○○●○○○

0x2110, //○○●○○○○●○○○●○○○

0x2000, //○○●○○○○●○○○●○○○

0x0000, //○○○○○○○○○○○○○○○○○○

0x0000, //○○○○○○○○○○○○○○○○○○

0x0000, //○○○○○○○○○○○○○○○○○○

0x0000, //○○○○○○○○○○○○○○○○○○

0x0000, //○○○○○○○○○○○○○○○○○○

0x0000, //○○○○○○○○○○○○○○○○○○

0x0000, //○○○○○○○○○○○○○○○○○○

0x0000, //○○○○○○○○○○○○○○○○○○

0x0000, //○○○○○○○○○○○○○○○○○○

0x0000, //○○○○○○○○○○○○○○○○○○

0x0000, //○○○○○○○○○○○○○○○○○○

0x1000, //○○○●○○○○○○○○○○○○○○

0x1008, //○○○●○○○○○○○○○○●○○○

0x1008, //○○○●○○○○○○○○○○●○○○

0x1008, //○○○●○○○○○○○○○○●○○○

0x1008, //○○○●○○○○○○○○○○●○○○

0x1008, //○○○●○○○○○○○○○○●○○○

0x1008, //○○○●○○○○○○○○○○●○○○

0x1008, //○○○●○○○○○○○○○○●○○○

0x1000, //○○○●○○○○○○○○○○○○○○

0x0000, //○○○○○○○○○○○○○○○○○○

0x0000, //○○○○○○○○○○○○○○○○○○

0x0000, //○○○○○○○○○○○○○○○○○○

0x0000, //○○○○○○○○○○○○○○○○○○

0x0000, //○○○○○○○○○○○○○○○○○○

0x0000, //○○○○○○○○○○○○○○○○○○

0x0000, //○○○○○○○○○○○○○○○○○○

0x0000, //○○○○○○○○○○○○○○○○○○

x=0 のとき col_sin 変数に入るデータ

```
0x8000 //1000 0000 0000 0000
```

3. プログラム解説 「mini_mcr.c」

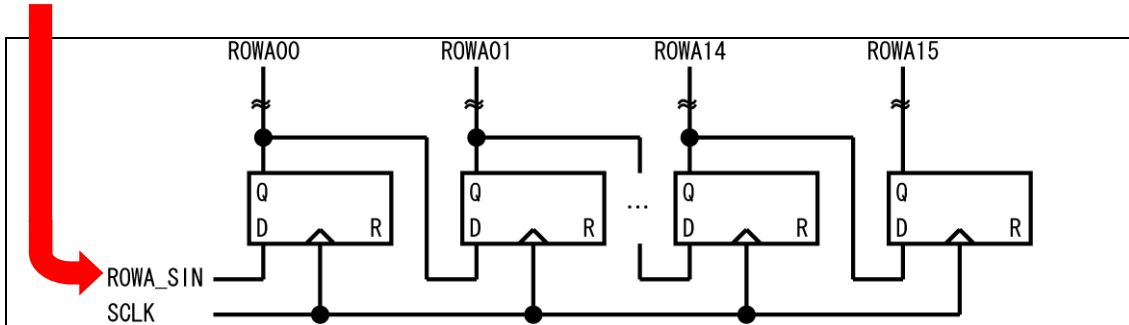
```
724 :         for( y = 0; y < 16; y++ ){
725 :
726 :             p6_3 = (rowa_sin >> 15) & 0x0001;        // ROWA_SIN
727 :             rowa_sin = rowa_sin << 1;
728 :
729 :             p6_2 = (rowb_sin >> 15) & 0x0001;        // ROWB_SIN
730 :             rowb_sin = rowb_sin << 1;
731 :
732 :             p6_1 = col_sin & 0x0001;                 // COL_SIN
733 :             col_sin = col_sin >> 1;
734 :
735 :             // クロックの立ち上りでシフトレジスタにロード
736 :             p6_4 = 0;                                // SCLK
737 :             p6_4 = 1;                                // SCLK
738 :
739 :         }
```

p6_4 端子(SCLK)の立ち上り時の p6_3 端子(ROWA_SIN)、p6_2 端子(ROWB_SIN)、p6_1 端子(COL_SIN)の状態が、シフトレジスタにラッチされます。

3. プログラム解説「mini_mcr.c」

rowa_sin 変数

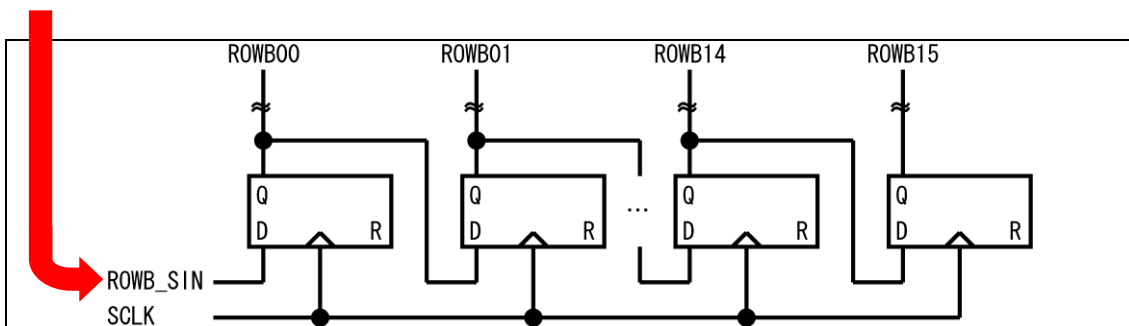
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0



rowa_sin 変数を右に 15 回シフトしたデータを p6_3 端子から出力しています。その後 rowa_sin 変数を左に 1 回シフトしていますので、p6_3 端子からは、rowa_sin 変数の 15、14…1、0 ビットの順番でデータが出力されます。

rowb_sin 変数

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

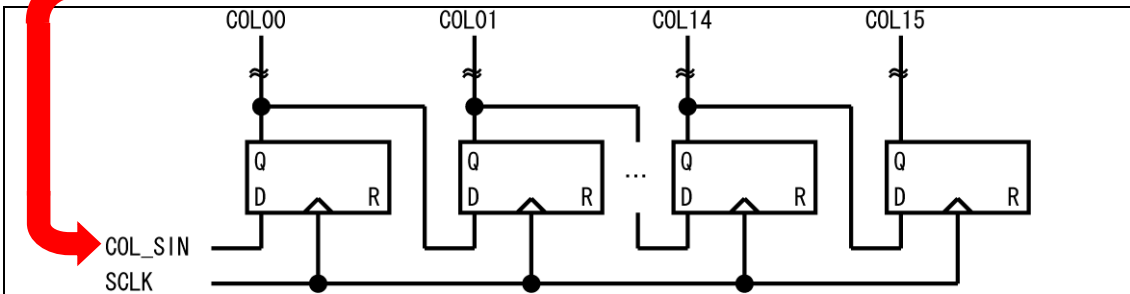


rowb_sin 変数を右に 15 回シフトしたデータを p6_2 端子から出力しています。その後 rowb_sin 変数を左に 1 回シフトしていますので、p6_2 端子からは、rowb_sin 変数の 15、14…1、0 ビットの順番でデータが出力されます。

3. プログラム解説 「mini_mcr.c」

col_sin 変数

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

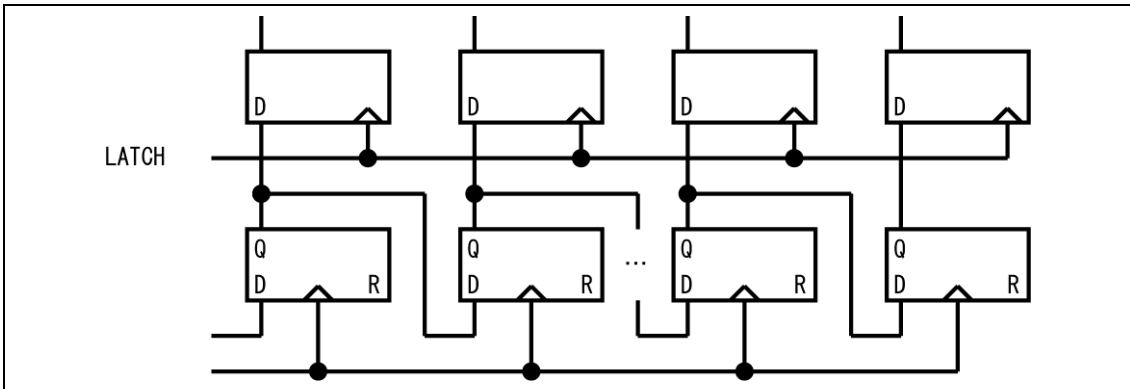


col_sin 変数のデータを p6_1 端子から出力しています。その後 col_sin 変数を右に 1 回シフトしていますので、p6_1 端子からは、col_sin 変数の 0、1…14、15 ビットの順番でデータが出力されます。

```

742 :      p6_6 = 0;           // LATCH
743 :      p6_6 = 1;           // LATCH
    
```

p6_6 端子 (LATCH) の立ち上り時のシフトレジスタの状態が、パラレルレジスタにラッチされます。



3.6 メインプログラム : main 関数

main 関数は、スタートアップルーチンから呼び出され、最初に行われる C 言語のプログラムです。

プログラム

```

439 : void main(void)
440 : {
441 :     // 初期化
442 :     init();
443 :
444 :     cnt1 = 0;
445 :     scroll = 0;
446 :
447 :     while(1){
448 :         // 25ms を超えたら
449 :         if( cnt1 >= 25 ){
450 :
451 :             cnt1 = 0;           // タイマーカウンタをクリアする
452 :             scroll++;          // スクロールさせるための変数を+1 する
453 :
454 :         }
455 :
456 :         // スクロールの終了位置まで来たら
457 :         if( scroll >= 16 * ( 22 - 2 ) ){
458 :             scroll = 0;        // 変数をクリアする
459 :         }
460 :     }
461 :
462 : }
```

main 関数では、25 [ms] ごとに表示する文字を左スクロールさせるための処理をしています。

```

444 :     cnt1 = 0;
445 :     scroll = 0;
```

1[ms]ごとに時間をカウントする cnt1 変数と、表示する文字を左スクロールさせるための scroll 変数をクリアします。

```

449 :         if( cnt1 >= 25 ){
450 :
451 :             cnt1 = 0;           // タイマーカウンタをクリアする
452 :             scroll++;          // スクロールさせるための変数を+1 する
453 :
454 :         }
```

25 [ms] 経過したら、次の時間をカウントするため cnt1 変数をクリアし、表示する文字を左スクロールさせるため、scroll 変数をインクリメントします。

4. 仕様

```

457 :           if( scroll >= 16 * ( 22 - 2 ) ){
458 :               scroll = 0;           // 変数をクリアする
459 :           }

```

表示する文字は、「ミニマイコンカーVer.2」で18文字、前後にスペースが2文字ずつ入って22文字となっています。最後の2文字を表示したときに先頭に戻す必要があり、1文字あたりの列数は16列ですので、

16×(文字数-2)

のときに scroll 変数をクリアします。

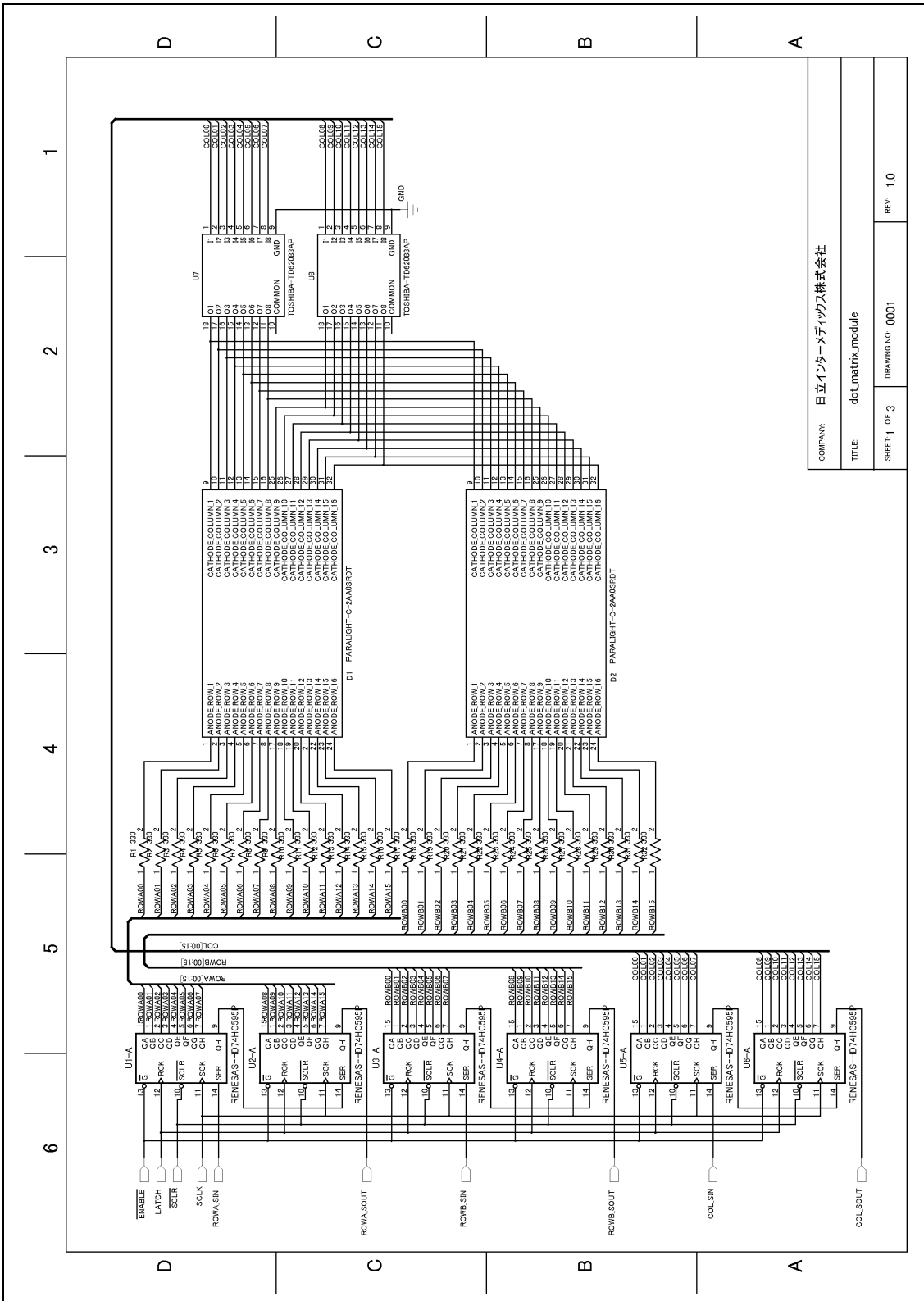
4. 仕様

4.1 仕様

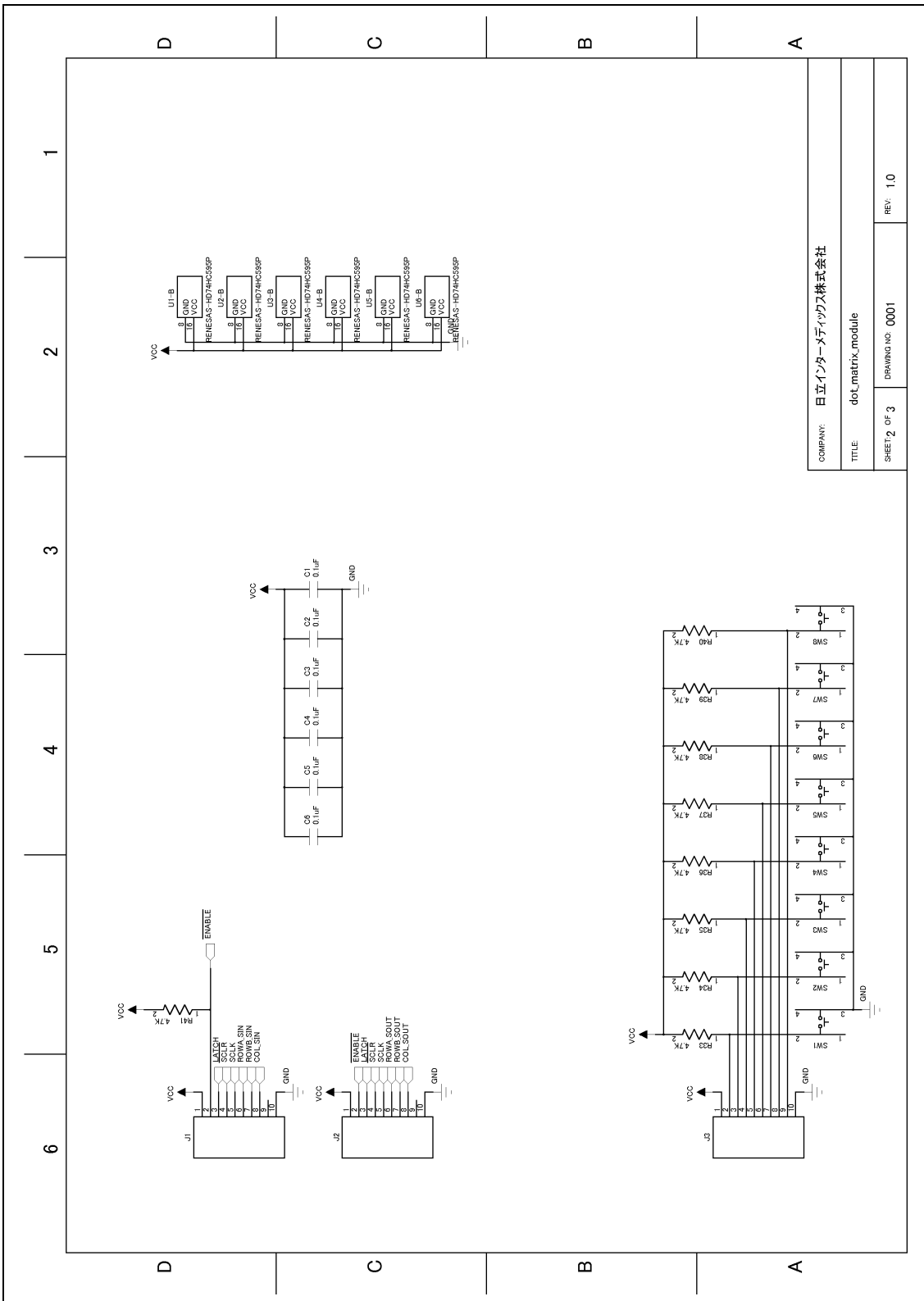
内容	詳細
電源	DC+5V ※ドットマトリクス LED 制御信号入力用コネクタに供給
表示エリア	16×32 ドット ※基板を最大4枚まで連結し、16×128まで拡張可能
I/O	<ul style="list-style-type: none"> ・ドットマトリクス LED 制御信号入力用コネクタ×1個 ・ドットマトリクス LED 制御信号出力用コネクタ×1個 ・タクトスイッチ信号出力用コネクタ×1個 ・タクトスイッチ×8個

4. 仕様

4.2 回路図



COMPANY: 日立インターメディアシステム株式会社
 TITLE: dot_matrix_module
 SHEET 1 of 3 DRAWING NO. 0001 REV. 1.0



4. 仕様

4.3 ポート表

コネクタ	番号	端子名	機能
J1	1	VCC	
	2	¥ENABLE	L: パラレルレジスタを LED に出力
	3	LATCH	立ち上がり時にシフトレジスタをパラレルレジスタへラッチ
	4	¥SCLR	L: シフトレジスタをクリア
	5	SCLK	立ち上がり時に ROWA_SIN、ROWB_SIN、COL_SIN の入力をシフトレジスタへラッチ
	6	ROWA_SIN	左側 LED の行データ
	7	ROWB_SIN	右側 LED の行データ
	8	COL_SIN	左右 LED の列データ
	9		未接続
	10	GND	

コネクタ	番号	端子名	機能
J2	1	VCC	
	2	¥ENABLE	J1 と同じ
	3	LATCH	J1 と同じ
	4	¥SCLR	J1 と同じ
	5	SCLK	J1 と同じ
	6	ROWA_SOUT	左側 LED の行データ (拡張用)
	7	ROWB_SOUT	右側 LED の行データ (拡張用)
	8	COL_SOUT	左右 LED の列データ (拡張用)
	9		未接続
	10	GND	

4.4 ピン配置図

コネクタ

