

C から体感する 78K0R マイコン

(C)2010 -2011 てきーらサンドム

R1.00 2010/5/18

R1.50 2011/2/27

R2.00 2011/3/29

はじめに

理屈よりも実際の動作でマイコンを覚えたい人向けの解説書です。

- ・プログラム動作例を中心に説明します。
- ・比較的簡単なハードで日本語表示，かな入力ができます
- ・ルネサス エレクトロニクス製 16 ビット・マイコン 78K0R を使用します。

- - - 目次概要 - - -

1章 C 言語のさわり (準備)

- 1.1 最短で動作を見る / 1.2 プログラム動作までの流れ /
- 1.3 開発環境のインストール / 1.4 ボード (実行環境) の製作

2章 C 言語をかじる (見て触って覚える)

- 2.1 行儀を正して / 2.2 固定した文字の表示 / 2.3 数値 (変数) の表示 /
- 2.4 配列と繰返し処理 / 2.5 キー入力に応じた動作の選択 / 2.6 関数電卓 /
- 2.7 文字列の操作と構造体 / 2.8 関数の使い方と変数の通用範囲 / 2.9 補足

3章 ハードな C 言語 (マイコン特有部分)

- 3.1 C から見たマイコンの基礎 / 3.2 ポート操作の基本 / 3.3 キー入力 /
- 3.4 LCD 制御 / 3.5 定期的な処理 (インターバル・タイマと割り込み) /
- 3.6 メロディ制御 (方形波出力) / 3.7 LED 調光 (PWM 制御) /
- 3.8 温度測定 (A/D 変換) / 3.9 時計機能 (RTC) /
- 3.10 パソコン COM ポート接続 (UART 通信) /
- 3.11 赤外線通信 (パルス間隔測定)

4章 リアルな C 言語 (実用的プログラム構造)

- 4.1 プログラムの基本構造 / 4.2 基本構造の役割 / 4.3 基本構造の種類 /
- 4.4 基本構造の設計例 / 4.5 タスクの設計例

- - - 目次詳細 - - -

1章 C言語のさわり(準備)

- 1.1 最短で動作を見る
- 1.2 プログラム動作までの流れ
- 1.3 開発環境のインストール
 - 1.3.1 ソフトウェアのダウンロード
 - 1.3.2 インストール方法
 - 1.3.3 ビルド確認とプログラム作成
 - 1.3.4 デバッガの起動
 - 1.3.5 プログラマの起動
- 1.4 ボード(実行環境)の製作
 - 1.4.1 初期段階の配線
 - 1.4.2 キー入力とLED表示の追加
 - 1.4.3 その他の回路追加

参考文献(1~4章)

(1)ルネサス エレクトロニクス資料(ドキュメント番号からダウンロード可能)

ドキュメント検索ページ <http://japan.renesas.com/search/el/document.html>

- ・U20024J「78K0R/Kx3 L ユーザーズ・マニュアル」
 - ・U17792J「78K0R マイクロコントローラ 命令編」
 - ・U18416J「PM+ Ver.6.30 ユーザーズ・マニュアル」
 - ・U18548J「CC78K0R Ver.2.00 C コンパイラ 言語編」
 - ・U18549J「CC78K0R Ver.2.00 C コンパイラ 操作編」
 - ・U18546J「RA78K0R Ver.1.20 アセンブラ・パッケージ 言語編」
 - ・U18547J「RA78K0R Ver.1.20 アセンブラ・パッケージ 操作編」
 - ・U19613J「ID78K0R QB Ver.3.60 統合デバッガ 操作編」
 - ・U18371J「QB MINI2 ユーザーズ・マニュアル」
 - ・U18527J「QB Programmer 操作編」
 - ・U19532J「78K0R/Kx3 サンプル・プログラム ドット LCD 制御編」
 - ・U19215J「78K0R/Kx3 サンプル・プログラム 簡易 OS 編」
 - ・U20119J「ヒューマン・マシン I/F デモ用 78K0R ボード ユーザーズ・マニュアル」
- (2)その他
- ・JIS X 3010 プログラム言語C (閲覧方法は2.1.4項参照)
 - ・「新ANSI C言語辞典」平林雅英著, 技術評論社
 - ・「プログラミング言語C」カニハン&リッチ - 著, 石田晴久訳, 共立出版株式会社

1 章 C 言語のさわり

この章では、まず簡単なプログラム例を見てから、プログラム開発や実行する環境を準備します。

1.1 最短で動作を見る

まずは次のプログラムを見てください。

```
/* リスト 1-1 最短で動作を見る */
main () {
    printf("今日も元気だ!");
}
```

これを実行すると写真 1-1 のようになります・・・って、ちょっと飛び過ぎですね。準備として、

- ・C コンパイラ（開発環境）をインストールする。
- ・ボード（実行環境）を作る。

という作業が必要です。でも、まずはリスト 1-1 を解説します。

printf 関数

printf は、文字を表示するという関数です。関数というと、数学で $y = f(x)$ とか出てきますが、まさにその関数です。C 言語というのは、数式を寄せ集めたような書き方をします。

ちょっと違うのは最後にセミコロン (;) を付けて、

```
y = x + 3;
```

```
y = f(x);
```

などのように書きます。printf 関数も

```
y = printf(x);
```

のように書けるのですが、ここでは関数の結果を使わないので省略しています。



写真 1-1 ボード全体図（初期段階）

main 関数

main は、C 言語において必ず最初に実行する関数で、main 関数の実行が終わればプログラムは終了します・・・と覚えましょう。

本当は電源が入ってから main 関数を実行する前に準備があり、main 関数が終わった後はマイコンはどうなるんだという話もありますが、最初の段階では気にしないで下さい。

さて main の行は、main という関数が具体的にどんな動作をするのかを書き始めている部分です。C プログラミングというのは、まさに関数の定義（具体的動作）を書くということなのです。

関数の具体的動作は { } の内側に書きます。リスト 1 は具体的動作が 1 行しかないので、

```
main() { printf("今日も元気だ!"); }
```

とも書けます。普通はたくさんの数式やら文を書くので適当に改行します。

コーディング作法（改行や字下げのマナー）は追々覚えるとして、C 言語自体は名称の途中以外や、' ' や " で囲まれた中以外では、自由に空白や改行を入れることができます。

コメント

/*と*/で囲まれた部分をコメントといいます。ここには何を書いても実行結果に何も影響しません。

1.2 プログラム動作までの流れ

図 1 - 1 にプログラム作成から実行までの作業の流れと、使用するソフトウェア（フリーダウンロード可能）について示します。

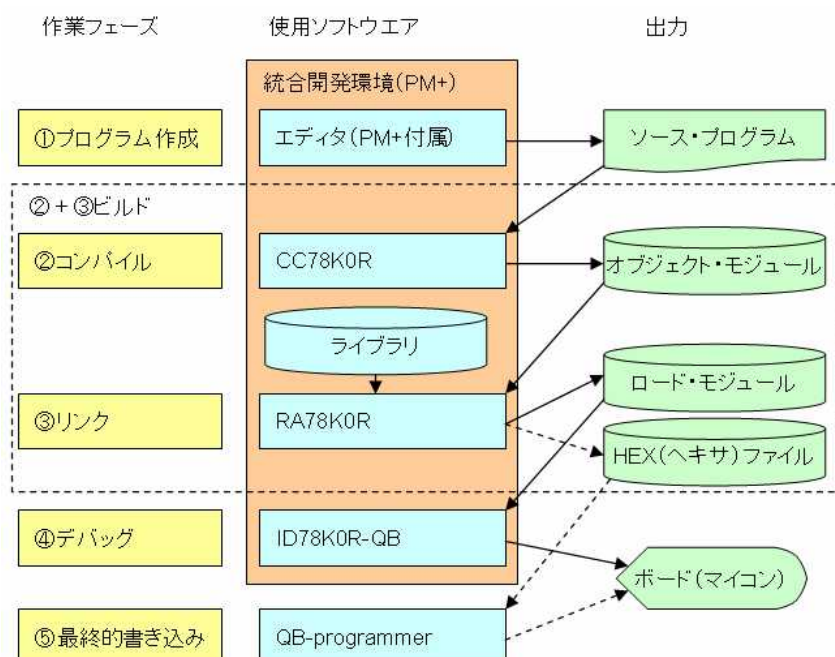


図 1 - 1
プログラム動作
までの流れ

プログラム作成

エディタを使用して C 言語のプログラムを書きます。エディタはメーカ提供の統合開発環境ソフトウェア (PM+) に付属しています。その他の一般的なテキスト・エディタを使うことも出来ます。

作成したプログラムをソース・プログラムと言います。

コンパイル

コンパイラ (CC78K0R) を使って、C 言語のソース・プログラムをマイコンが理解できる機械語に翻訳します。統合開発環境 PM+ の元では 1 クリックでコンパイルできます。

コンパイラは、ソース・プログラムの書き方が悪いとエラー (致命的間違い) やワーニング (致命的かもしれない間違い) を出します。

リンク

リンカ (LK78K0R) を使って、コンパイラが生成したオブジェクト・モジュール・ファイルと、ライブラリ (標準的な関数等) を結合し、マイコンが実行可能なロード・モジュール・ファイルを作ります。

なおリンカ (LK78K0R) は、アセンブラ・パッケージ (RA78K0R) の中に一緒に入っています。

コンパイルとリンクの作業をまとめて、ビルドと言います。統合開発環境 PM+ の元では、ビルドが 1 クリックで出来ます。ソース・プログラムがコンパイル済の場合は再度コンパイルされることなくリンクされます。

再コンパイルから行う作業は、リビルドと言います。

デバッグ

実際にプログラムを動作させて、問題があれば修正します。

まずデバッガ (ID78K0R-QB) を使用して、ロード・モジュール・ファイルをマイコンに書き込んで、マイコンを動作させます。

デバッガを使うと、ソース・プログラムの行単位で動かしたり、途中経過を見ることも出来ます。

最終的書き込み

デバッグが終わったら、最終的なプログラムをマイコンに書き込みします。これによりマイコン単体 (デバッガ接続無し) で動作が可能になります。(注: デバッガで書いた状態で動くこともあるが、デバッグ・コードも書かれていて、それが悪さすることがある)

1.3 開発環境のインストール

ソフトウェアのダウンロード方法，インストール方法，セットアップ方法を説明します。

1.3.1 ソフトウェアのダウンロード

ダウンロードするソフトウェアの一覧を表 1 - 1 に示します。ソフトウェアの掲載場所は次の通りです

(1) ビルド用ソフトウェア

掲載ページは，

「78K0R/Kx3 マイコン用 フリーツール」
で URL は下記です。

<http://www2.renesas.com/micro/ja/freesoft/78k0r/kx3/index.html>

次の 4 つにチェックをつけてダウンロードします (Kx3 L は最初の 2 つでよいです)。

RA78K0R， CC78K0R， 78K0R/Kx3 用デバイス・ファイル，
78K0R/Kx3 用 PG-FP4，MINICUBE2 パラメータ

なお，ダウンロードすると RA78K0R と CC78K0R のライセンスコードが E メールですぐに送られてくるので，記録しておきます。Kx3 L 用のデバイス・ファイルとパラメータ・ファイルは下記から品番を選んでからダウンロードします。

https://www5.renesas.com/micro/tool_reg/OdsListDeviceSeries.do?code=30&lang=ja

(2) デバッガ，プログラマ (MINICUBE2 用)

掲載ページは，「MINICUBE2 シリーズ用ソフトウェア」で URL は下記です。

https://www5.renesas.com/micro/tool_reg/OdsListTool.do?code=460&lang=ja

次の 2 つをダウンロードします。

QB-Programmer

ID78K0R-QB

(3) 78K0R 用各種ドライバ類

本書の例題を動作させるには，筆者提供の各種ドライバ類 (ドライバ・キット) が必要になります。ベクターから次のファイルをダウンロードします。

KLCD_78K0R (<http://www.vector.co.jp/soft/winnt/hardware/se483763.html>)

名称	内容
CC78K0R	Cコンパイラ。C言語のソース・プログラムを翻訳するソフトウェア。
RA78K0R	統合開発環境PM+(エディタ含む)，アセンブラ(リンカ等含む)。
ID78K0R-QB	デバッガ。プログラム実行制御用ソフトウェア。
DF781014, DF781188	デバイス・ファイル。マイコンの型番ごとの情報ファイル。
QB-Programmer	マイコンへのプログラム書込み用ソフトウェア。
PRM78F1014, PRM78F1188	パラメータ・ファイル。書込み用の情報ファイル。
KLCD_78K0R	各種ドライバ類。本書の例題実行に必要。筆者提供。

表 1 - 1 ダウンロード・ソフトウェア一覧

1.3.2 インストール方法

インストール手順の概要を図 1 - 2 に示します。

(1) RA78K0R, CC78K0R

クリックするとインストーラが起動します。
Eメールで通知されたライセンスコードを入力して「インストール」を押します。

インストール開始後、時々インストーラが消えますが、慌てずに辛抱強く「インストール終了」が表示されるまで待ってください。

(2) デバイス・ファイル (DF781014 / 1188)

次の手順でインストールします。

ダウンロードした DF781014 または DF781188 を適当なフォルダに解凍します。

PM+を起動し、ツール・メニューからデバイス・ファイル・インストーラを起動します。デバイス・ファイル製品の「インストール」ボタンを押して、で解凍したデバイス・ファイル・ディレクトリの NECSETUP.INI を開きます。

以降はデフォルトのまま進めてインストールを完了させます。

いったん PM+を終了してください。

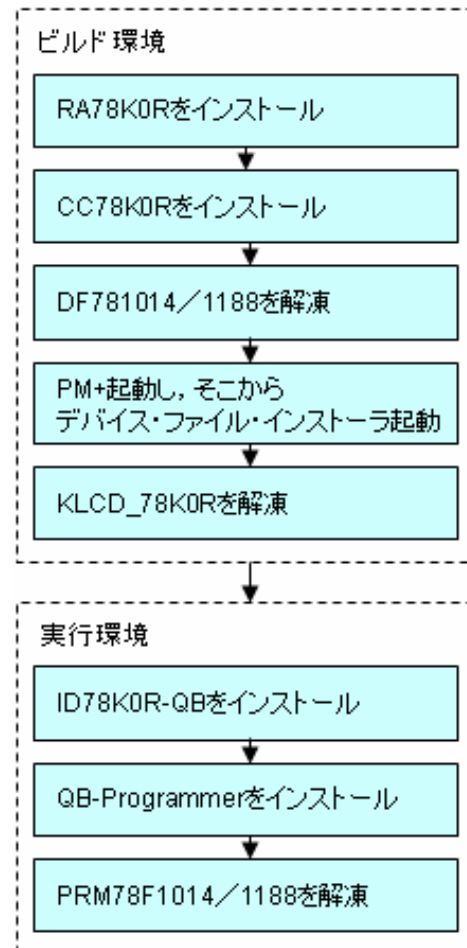


図 1 - 2 インストール手順概要

(3) 78K0R 用各種ドライバ類

ダウンロードした KLCD_78K0R_Rxxx.zip をプログラム開発用の適当なフォルダに解凍します。次のファイルが解凍されます。

- ・ KLCD_78K0R.pdf ドライバ・キットの取扱説明書
- ・ KLCD_78K0R.prw PM+起動用ファイル (64 ピン以下の 78K0R/Kx3 L 用)
- ・ KLCD_DDT8.prw PM+起動用ファイル (デザイン・テクノ No.8 添付基板用)
- ・ KLCD_KORKX3.prw PM+起動用ファイル (78K0R/Kx3 用)
- ・ KLCD_78K0R.zip ドライバのソース・プログラム (ライセンス・コード必要)
- ・ driver フォルダ ドライバ・ライブラリ, ヘッダ, 等
- ・ main フォルダ 本書掲載のサンプル・プログラム
- ・ other フォルダ 回路図, プロジェクト・ファイルの控え, 本書

(4) ID78K0R-QB および QB Programmer

ボードでプログラムを実行する時に必要です。それぞれクリックするとインストールを開始します。

(6) パラメータ・ファイル

QB Programmer を実行するときに必要です。とりあえず適当なフォルダに解凍します。

1.3.3 ビルド確認とプログラム作成

KLCD_78K0R.prw (64ピン以下の Kx3-L 用), KLCD_DDT8.prw (DDT8 添付基板用), KLCD_KORKX3.prw (Kx3 用) のいずれかをクリックすると PM+ が起動します。

(1) 初回の起動の場合だけ、下記手順で設定が必要です。

まず警告「・・・ツールを選択し直してください」が出ます。「OK」を押します。

ツールバージョン設定画面が表示されます。「キャンセル」を押します。

PM+ の「プロジェクト」「プロジェクトの設定」で画面が開きます。

「プロジェクト情報」タブの「デバイス名」が実際と異なる場合は選択し直します。

(デバイス名の初期値は、KLCD_78K0R.prw の場合は UPD78F1006_52 になっています)

「ツールバージョン設定」タブの「詳細設定」を押して、インストールしてあるバージョンを選択してください。

選択後「OK」を押し、さらに「OK」を押すと、「・・・削除しますか？」が出ます。

「はい」を押します。

これで設定完了です。

(2) ビルド確認

まずは正常に環境が整っているか、ビルドしてみます。ビルドのアイコンを押すか、ビルド・メニューの「ビルド」を選ぶとビルドが出来るはずですが。

この段階ではワーニングが 2 つ出ますが、エラーは出ないはずですが。うまくいかない場合はリビルドしてみてください。

(3) プログラムを作成

ProjectWindow (画面左) のソースファイルの下にある main.c を書き換えれば良いです。

main.c 以外の各種ドライバ類については、3 章で解説します。

(4) ポート割り当てやマイコンのシリーズを変更したい場合

ドライバ・キットの取扱説明書を参照してください。

1.3.4 デバッガの起動

以降はボード接続時の手順です。デバッガを起動する前に後述のボードと MINICUBE2 (デバッグ・ツール) を接続してください。

ビルドでエラーが無いことを確認します。

アイコンまたはビルド・メニューで「デバッグ」を選択するとデバッガが起動します。もし「オンチップ・デバッグが禁止されています」というようなメッセージが出たら、いったんプログラマで消去を行います (1.3.5 参照)。

無事起動すると初回起動時は周波数設定が出ます。そのまま「OK」を押しても動くようですが、もし動かない場合は Main Clock 設定で 20MHz を指定してください。

「ダウンロードしますか」が出たら「はい」を選択します。

Open 画面が出た場合は、トレースするソース・プログラムを指定できますが、キャンセルしても問題ありません。

以降は、Go や Stop などのアイコンでプログラムを走らせたり、止めたりできます。

なお、**最初は LCD のコントラスト調整が外れている可能性があります**。何か表示するプログラム (デフォルトの main.c など) を動かしている状態で、コントラストを調整してみてください。

1.3.5 プログラムの起動

デバッガが起動していない状態でプログラム (QB Programmer) を起動してください。

最初は、

「Device」 「Setup」 「PRM File Read」

を押し、パラメータ・ファイルを解凍したフォルダを開き、対象となるデバイスを選択します。

(1) 消去の場合

「Device」 「Erase」を選択してください。

(2) プログラム書き込みの場合

「File」 「Load」を押し、

ビルドした HEX ファイル (a.hex) を選択して開きます。

「Device」 「Autoprocedure (EPV)」

を選択すると、消去、書き込み、照合を自動的に行います。

MINICUBE2 のケーブルをボードから外すと、ボードが動作します。

ただし、初期段階では電源を MINICUBE2 から取っているので別途電源を接続します。

1.4 ボード (実行環境) の製作

ボードは少しずつ作れるようになっていきます。

- ・初期段階：LCD と MINICUBE2 (デバッグ・ツール) を接続します。
2章の途中までは、これだけで使用できます。
- ・第2段階：キーとLEDを追加します。
2章の途中から使用します。
- ・第3段階：好みに応じて圧電ブザー、調光用LED、温度センサ、時計、COMポート、赤外線送受信の追加を行います。3章以降にプログラム例があります。
- ・第4段階：一部の型番では、SDカード、音声コーデックの追加が可能です。ただし本書およびドライバ・キットのサポート対象外で、筆者ページでの参考公開です。

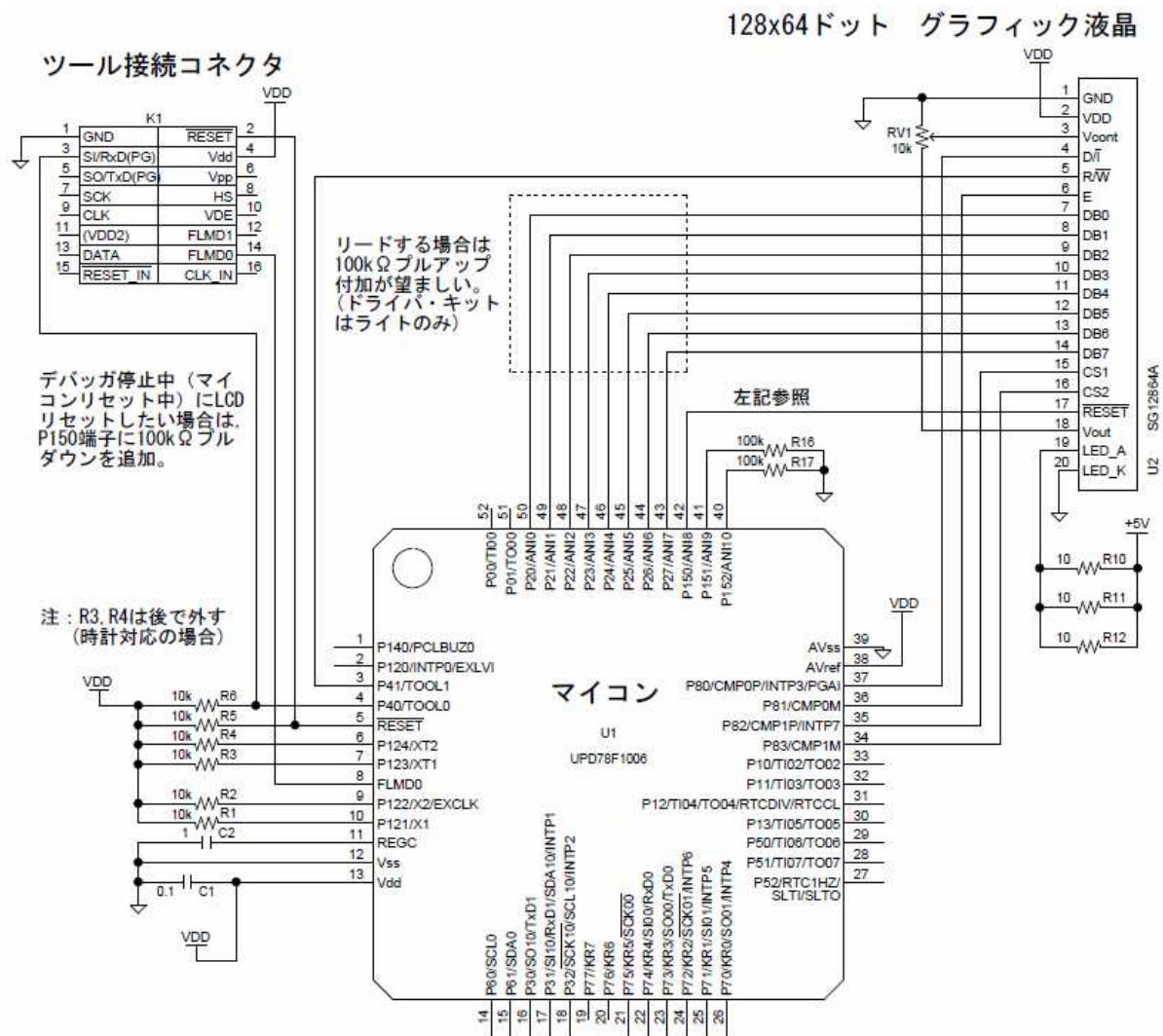


図 1 - 3 初期段階の回路図 (LCD 表示のみ)

1.4.1 初期段階の配線

代表として 78KOR/KD3-L (UPD78F1006) の回路図を図 1-3 に示します。

部品表 (第 2 段階含む) を表 1-2 に示します。

PDF 図面 (KD3-L 以外も含む), CAD データ等は, ドライバ・キットの other フォルダ中にあります。CAD データは D2CAD 用です。

ボードの電源は MINICUBE2 から取っています。ただし, MINICUBE2 の出力は 100mA までとなっているので, LCD バックライトは別電源から取ります。第 3 段階の回路図では USB シリアル変換から配線しています。

表 1-2 初期段階～第 2 段階の部品表

段階	番号	型番 / 仕様	購入先例	
1	U1	UPD78F1003GB -GAF -AX (44 ピン, 0.8 ピッチ)	ルネサス エレクトロニクスの特約店各社。	
		UPD78F1003GA -HAA -AX (48 ピン, 0.5 ピッチ)		
		UPD78F1006GB -GAG -AX (52 ピン, 0.65 ピッチ)		
	U1	UPD78F1009GK -GAJ -AX (64 ピン, 0.65 ピッチ)	RS コンポーネンツ	
		UPD78F1009GB -GAH -AX (64 ピン, 0.5 ピッチ)		
			78KOR/Kx3 シリーズ (UPD78F1146 ~ 1168)	
			デジタル・デザイン・テクノロジー No.8 添付基板 (UPD78F1003GB 搭載)	CQ 出版
		U2	128x64 ドット LCD。SG12864A, BG12864A など。	秋月電子通商など
		C1	0.1 μ F 積層セラコン	どこでも
		C2	1 μ F。OS-CON やタンタルコンなど特性の良い物。積層セラは B 特性, R 特性などの特性の良い物。(F 特性の安い積層セラなどは非推奨)	サトー電気など
		R1 ~ R6	10k 。R3, R4 は時計対応時は必ず外して下さい。	どこでも
		R10 ~ R12	10 。1/4W 以上。	どこでも
		R16 ~ R18	100k 。入力保護用なので, 温度センサなどが接続されていれば不要です。	どこでも
	RV1	10k 半固定。LCD コントラスト調整用。	どこでも	
	CN1	16 ピン・ヘッダー	どこでも	
	-	MINICUBE2 (デバッグ・ツール)。8 ~ 32bit の All Flash MCU で使用可能です。	マルツ電波, デジキー (英語版) など	
2	SW1 ~ SW25	タクトスイッチ	どこでも	
	D10 ~ D13	LED。KC3-L44pin 版では D12, D13 は常時点灯。	どこでも	
	R20 ~ R23	1k 。	どこでも	

1.4.2 キー入力とLED表示の追加

図 1 - 4 に第 2 段階で追加する回路図を示します。2 章後半のキー入力プログラム例で使用します。LED はキーのシフト状態表示に使います。

・キーの割り当て (表 1 - 4 参照)

BS : バックスペース。Enter キーを押すまでは修正可能です。

NUM : 数字シフト

ALPA : 英字 A シフト。英字 A シフト状態で押すと小文字と大文字の切り替えになります。

ALPB : 英字 B シフト。英字 B シフト状態で押すと小文字と大文字切り替えになります。

SYMB : 記号シフト

ENT : Enter キー。入力を確定するのに使います。入力例では 記号で示します。

その他のキー: シフト状態により 4 種類の文字に対応します。表 1 - 4 に、NUM, ALPA,

ALPB, SYMB の順に対応関係が示してあります。ただし , は次の意味です。

: スペースです。

: かなシフトです。1 回目でひらがな変換。2 回目でカタカナ変換, 3 回目で無変

換に戻ります。ローマ字かな変換は, 表 1 - 3 の組合せのみ受け付けます。

表 1 - 3 ローマ字かな変換表

a: あア	i: いイ	u: うウ	e: えエ	o: おオ	vv: ヴヴ
xa: アア	xi: イイ	xu: ウウ	xe: エエ	xo: オオ	
ka: かカ	ki: きキ	ku: くク	ke: けケ	ko: こコ	
ga: がガ	gi: ぎギ	gu: ぐグ	ge: げゲ	go: ごゴ	
sa: さサ	si: しシ	su: すス	se: せセ	so: そソ	
za: ざザ	zi: じジ	zu: ずズ	ze: ぜゼ	zo: ぞゾ	
ta: たタ	ti: ちチ	tu: つツ	te: てテ	to: とト	xt: っツ
da: だダ	di: ぢヂ	du: づヅ	de: でデ	do: どド	
na: なナ	ni: にニ	nu: ぬヌ	ne: ねネ	no: のノ	
ha: はハ	hi: ひヒ	hu: ふフ	he: へヘ	ho: ほホ	
ba: ばバ	bi: びビ	bu: ぶブ	be: べベ	bo: ぼボ	
pa: ぱパ	pi: ぴピ	pu: ぷプ	pe: ぺペ	po: ぽポ	
ma: まマ	mi: みミ	mu: むム	me: めメ	mo: もモ	
ya: やヤ	va: やヤ	yu: ゆユ	vu: ゆユ	yo: よヨ	vo: よヨ
ra: らラ	ri: りリ	ru: るル	re: れレ	ro: ろロ	
wa: わワ				wo: をヲ	nn: んン

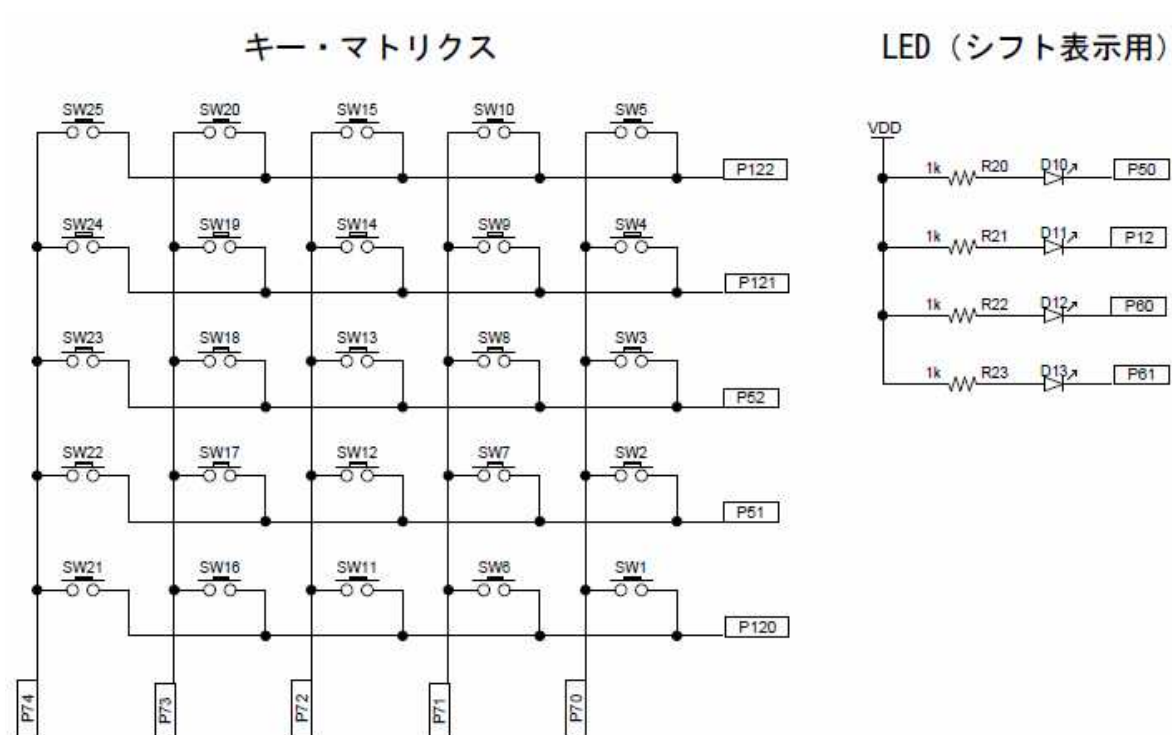


図 1 - 4 第 2 段階の追加回路図 (キー入力と LED 表示追加)

表 1 - 4 キー配置

NUM	/ B P =	* K G ~	- A A {	B S
7 N Q &	8 M C	9 S Z ^	+ I I }	A L P A
4 " ' (5 Y F)	6 T D ¥	% U U [A L P B
1 ? ! <	2 V X >	3 H W \$	E E E]	S Y M B
0 R L ;	. , J :	—	O O #	E N T ()

表 1 - 5 キーのシフト状態表示 (色は一例)

LED	数字シフト	英字 A シフト	英字 B シフト	記号シフト
D10				
D11				
D12				

D13 は、かなシフトの場合に点灯 () します。

78K0R/KC3 L 44pin 版は端子不足により D12, D13 は常時点灯です。

1.4.3 その他の回路追加

図1-5の各ブロックは、3章で説明するマイコン内蔵周辺機能に対応しています。使ってみたい部分だけ実装すれば良いです。

- ・圧電ブザー：内蔵タイマ機能により方形波を生成してメロディを出力します。
- ・調光用LED：内蔵タイマでPWM（パルス幅変調）波形を生成してLED輝度を変えます。
- ・温度センサ：内蔵A/Dコンバータで温度を読み取ります。
- ・時計：内蔵RTC（リアルタイム・カウンタ）により時計を動作させます。
- ・COMポート：内蔵シリアル・インタフェース機能によりパソコンと通信を行います。
- ・赤外線送受信：内蔵タイマ機能により送信波形を生成したり，入力パルス幅測定を行って受信を行います。Ir-DAではありません。

表1-6 第3段階で追加する部品の一覧

番号	型番 / 仕様	購入先例
U3	USB シリアル変換 AE-UM232R	秋月電子通商
U4	赤外受光モジュール 38kHz (PL-IRM2161-C438)	秋月電子通商
U5	温度センサ LM61BIZ など	秋月電子通商など
Q1, Q2(形式1)	汎用トランジスタ 2SC945, 2SC1815 など。 電流を上げる場合は 2SC2120, 2SC2001 など。	どこでも
Q3(形式2)	Nch MOS FET 2SK2961 など。	サトー電気など
D3	赤外 LED (940nm)。OSI5FU5111C-40 など。	秋月電子通商など
D4(形式1)	各種 LED (V_F はおおむね 3.2V 程度まで)。	どこでも
D5(形式2)	各種 LED (V_F はおおむね 4V 程度まで)。	どこでも
BZ1	圧電ブザー (他励式), 圧電スピーカ。	どこでも
X1	32.768kHz 発振子。	サトー電気, 秋月電子通商など
C3, C4	33pF (サトー電気扱い 2 品など), 15~22pF (秋月扱い VT200 など), 温度補償型セラコン。発振子により調整。	どこでも
C10	10 μ F 以上, 10V 以上	どこでも
R30~R36	100 Ω , 1/4W 以上。	どこでも
R14~R15	100k	どこでも

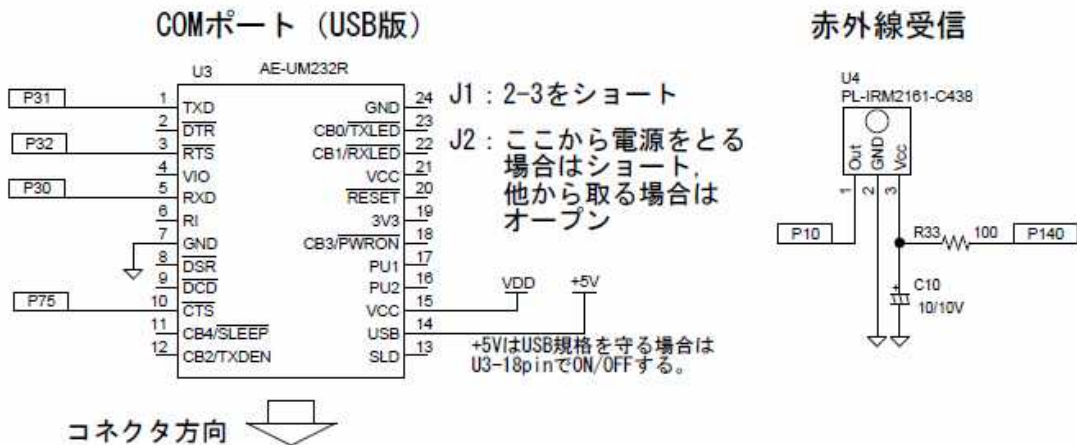
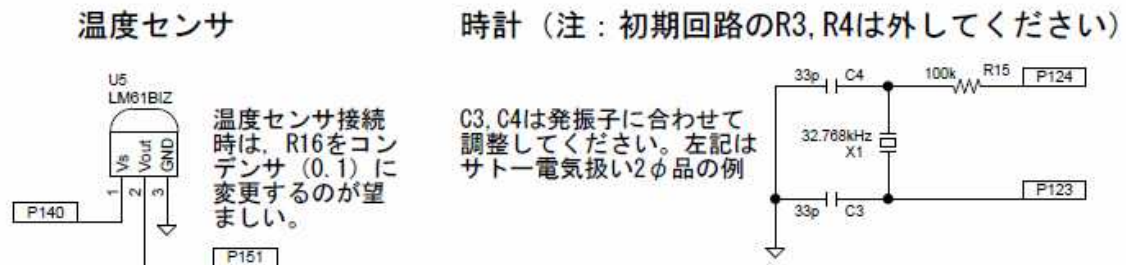
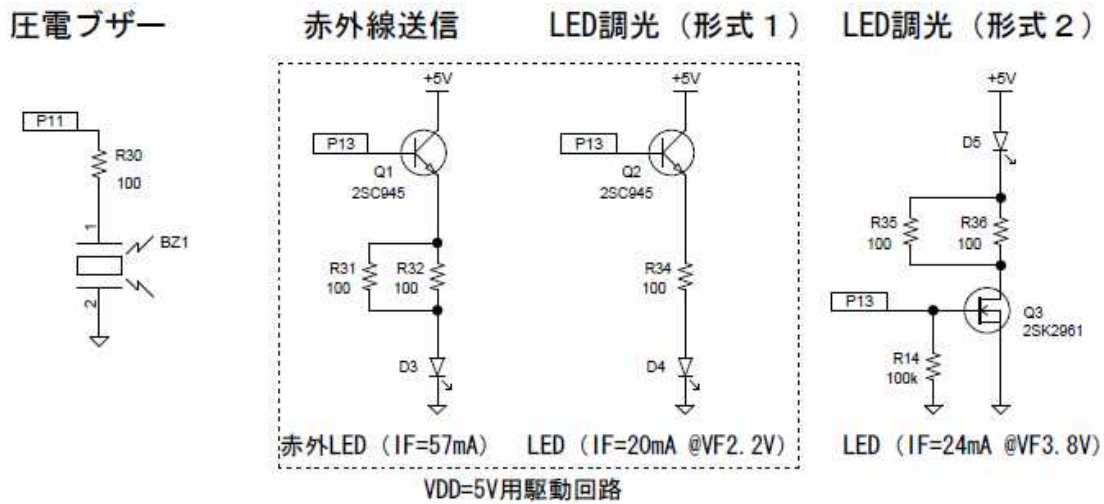


図1-5 第3段階の追加回路図（第3章で使用）

（参考）LED 電流 I_F の計算方法（VDD = 5V 時，typical）

形式1： $I_F = \{ (\text{マイコン P11 の } V_{OH}) - (Q1 V_{BE}) - (LED V_F) \} / \text{制限抵抗値}$
 $\{ 4.2 - (LED V_F) \} / \text{制限抵抗値}$

形式2： $I_F = \{ VDD - (LED V_F) \} / \{ \text{制限抵抗値} + (Q2 R_{on}) \}$
 $\{ 5 - (LED V_F) \} / \text{制限抵抗値}$