

ND80KL／86取扱説明書

(組立が終わったら、操作の前にお読みください)

目次

| | |
|---|----|
| [1]CPUボードの取り付け | 1 |
| [2]システムROMの取り付け | 1 |
| [3]SP1の設定 | 2 |
| [4]DS1の設定 | 2 |
| [4-1]KL5C80A12の場合 | 2 |
| [4-2]AM188の場合 | 3 |
| [5]電源をセットする | 3 |
| [6]ND80Z3. 5システムROMに入っているマシン語モニタプログラムについて(KL5C80A12CPU対象) | 4 |
| [7]マシン語モニタプログラムの操作説明書について(KL5C80A12CPU対象) | 5 |
| [8]マシン語モニタプログラムのメモリマップ(KL5C80A12CPU対象) | 6 |
| [9]モード2の割込みジャンプテーブル(KL5C80A12CPU対象) | 9 |
| [10]I/Oマップ | 10 |
| [11]プログラム、データの保存 | 11 |
| [12]RS232Cインターフェース | 11 |
| [13]コネクタ端子接続図 | 13 |
| [14]KL5C80A12CPUボードコネクタ端子接続図 | 16 |
| [15]AM188CPUボードコネクタ端子接続図 | 17 |

〒463-0067 名古屋市守山区守山2-8-14
パレス守山305

有限会社中日電工

TEL052-791-6254 Fax052-791-1391

E-mail thisida@alles.or.jp

Homepage <http://www.alles.or.jp/~thisida/>

ND80KL/86本体ボードには2種類のCPUボードが実装できます。

8ビットCPU、KL5C80A12CPUボードと16ビットCPU、AM188CPUボードです。

CPUボードによってシステムROMが異なります。

またSP1、DS1の設定も異なりますから、それぞれCPUボードに合ったシステムROMを実装し、SP1、DS1もCPUボードに合った設定をしてください。

[1]CPUボードの取り付け

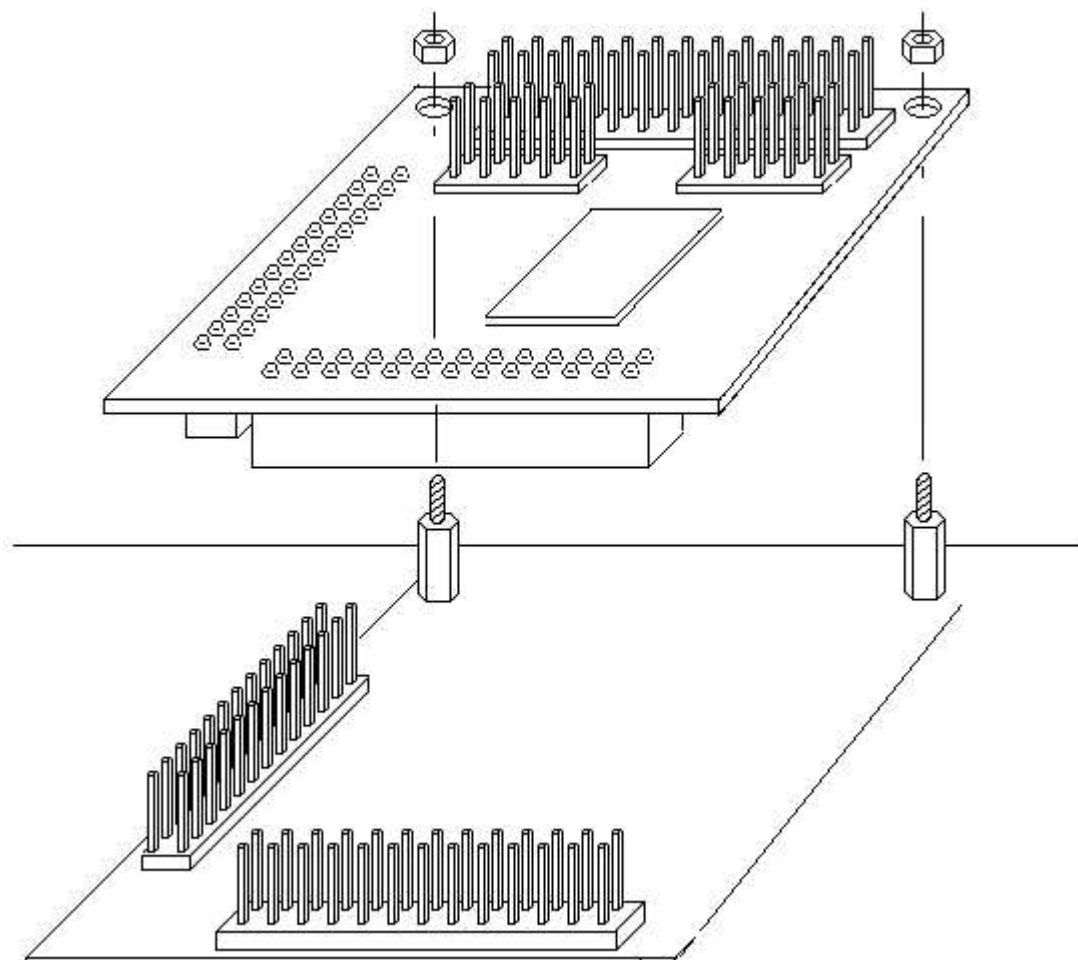
ND80KL/86本体ボード上部、CN2とCN3の26pinヘッダーにCPUボードを取り付けます。

CPUボードはKL5C80A12ボードとAM188ボードの2種類がありますが、いずれも取り付ける方法は同じです。

CN2とCN3の26pinヘッダーにCPUボードの26pinメスコネクタがしっかり合わさるように取り付けます。

このとき少しきついかもしれませんが、同時に2本のスペーサにCPUボードの2ヶ所の取付穴を通すようにしながら、コネクタをしっかり差し込みます。最後にナットで固定します。

下の取付参考図はKL5C80A12CPUボードの場合ですがAM188CPUボードの場合でも全く同じです。



[2]システムROMの取り付け

ND80KL/86は2種類のCPUボード、KL5C80A12ボードまたはAM188ボードを実装するように設計されています。

システムROMはCPUボードと組になっています。

ここではCPUボードに合うシステムROMを実装します。

間違ったシステムROMを実装すると、全く動作しませんからよく確認して正しいROMを取り付けてください。

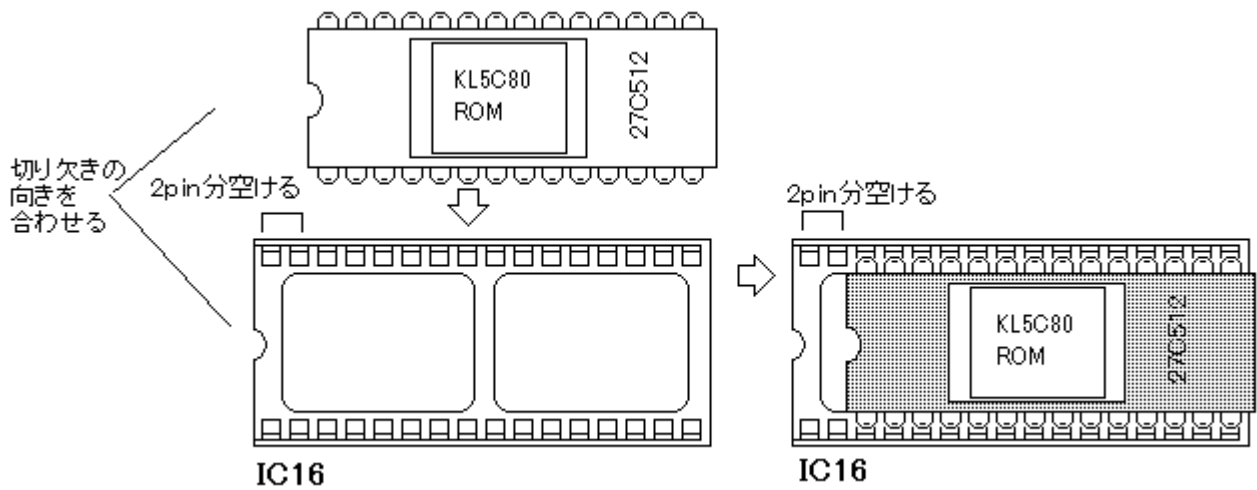
システムROMは、IC16の32pinソケットに取り付けます。

次ページ図をよく見て、向きを間違えないように取り付けてください。

ICをICソケットに取り付けるときの注意については、「ND80KL/86組立説明書」[21]7セグメントLEDの表示テスト①PIC16F883の実装を参照してください。

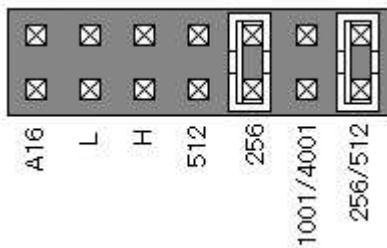
IC16のICソケットは32pinですがシステムROM(27C256または27C512)は28pinです。

ICソケットには図のように1、2ピン側が空きになるように取り付けます。

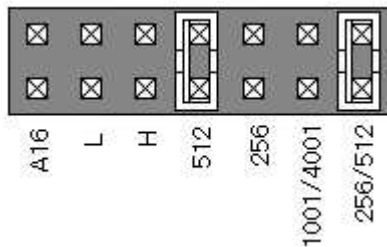


[3]SP1の設定

SP1はIC16に実装するROMのタイプによってそれぞれの設定が必要です。
27C256を実装するときは下図のようにジャンパピンをセットします。



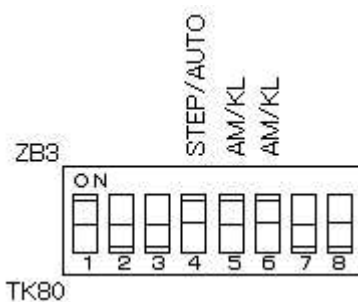
27C512を実装するときは下図のようにジャンパピンをセットします。



このほかの設定は将来のためのものです。
今は使いません。

[4]DS1の設定

[4-1]KL5C80A12の場合



No.5とNo.6は必ずON(上)にします。

No.1はOFFのときTK80モニタプログラムが選択され、ONのときND80Zモニタプログラムが選択されます。

Windows/パソコンとUSB接続をして、リモートプログラム、ZB3BASICやZB3DOS(CP/M 互換DOS)を使うときはNo.1をONにします(TK80モニタモードのとき以外はNo.1はONにしておきます)。

KL5C80A12用のZBK BASICシステムROMにはTK80モニタプログラムは入っていません。

No.4はTK80モニタプログラムやND80Zモニタプログラムでステップ動作をさせるときにOFFにします。

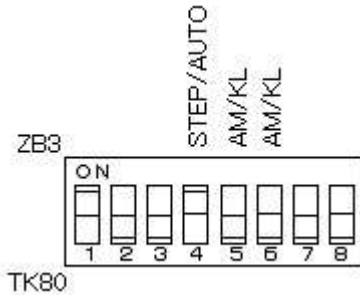
それ以外の場合はONにします。

No.2、No.3はRS232Cのボーレート設定用です。

この説明書の後ろのほうで説明します。

No.7、No.8は将来の機能拡張用です。

[4-2]AM188の場合



No.5とNo.6は必ずOFF(下)にします。

8086システムプログラムにはTK80互換モニタプログラムはありません。

8086システムではNo.1は使用しませんから、ONでもOFFでも構いません。

No.4は7セグメントLED表示を使うTK80風モニタプログラムでステップ動作をさせるときにOFFにします。

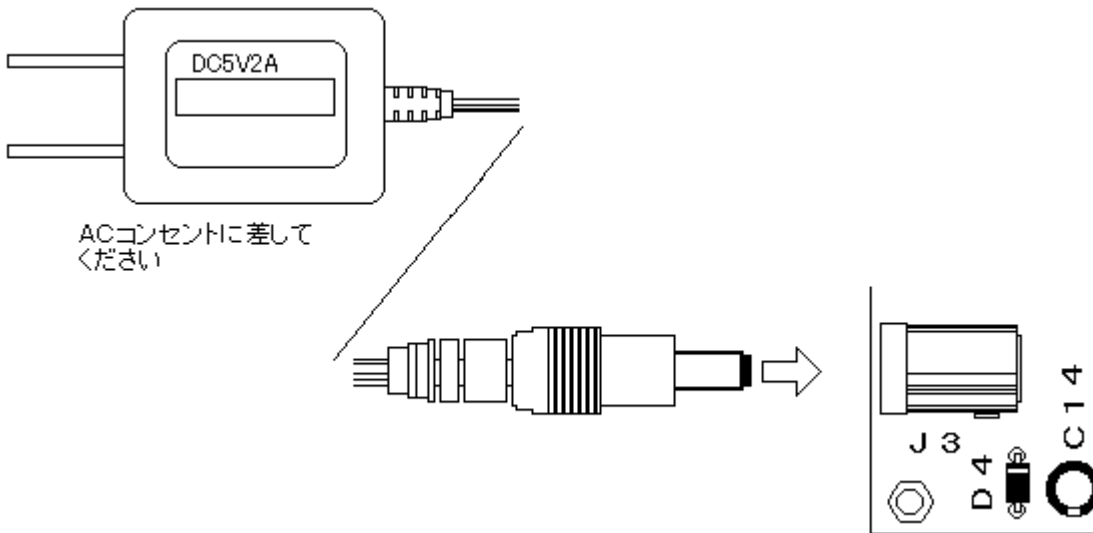
それ以外の場合はONにします。

No.2、No.3はRS232Cのボーレート設定用です。

この説明書の後ろのほうで説明します。

No.7、No.8は将来の機能拡張用です。

[5]電源をセットする



セットに付属している電源のプラグ側をND80KL/86の電源ジャック(J3)にしっかり差し込んで、電源の本体側を家庭用AC100Vのコンセントに差し込むと、ND80KL/86に+5Vが供給されて、7セグメントLEDが点灯します(下図)。

電源の入り切りを簡単にするために、スイッチ付のコンセントを使うこともできます。

電源をONにしてから、7セグメントLEDが点灯するまでに1秒程度かかることがあります。故障ではありません。

電源をONにすると、LEDは00000000を表示します。

システムプログラム起動のタイミングでLEDが点灯しない場合があります。

そのようなときはリセット(MON)キーを押してください。



[6]ND80Z3. 5システムROMに入っているマシン語モニタプログラムについて(KL5C80A12CPU対象)

KL5C80A12CPUボードにはND80Z3. 5システムROMにKL5C80A12の初期設定部分を追加したROMが付属しています。

ND80Z3. 5システムROMには、3つのマシン語モニタプログラムが入っています。

アドレス0000からスタートするTK80モニタプログラムのほかに、アドレス0400からスタートする同じTK80モニタプログラムと、0800からスタートするND80Zモニタプログラムです。

0000スタートのTK80モニタプログラムはNECのTK80モニタプログラムと同じで(一部異なっているところもあります)、RAMワークエリアもTK80と同じ83xx番地台になっています。

0400スタートのTK80モニタプログラムは、0000スタートのモニタプログラムを0400スタートに直ただけで0000スタートのTK80モニタプログラムと同じ動作をしますが、RAMワークアドレスは83xxではなくてFFxxになっています。このモニタプログラムはその基本的な部分をND80Zモニタプログラムが使用するためにあります。

0800スタートのND80Zモニタプログラムは当社オリジナルのND80Zボードに搭載していたモニタプログラムです。TK80モニタプログラムをベースにしていますが、さらに多くの機能が追加されています。RAMワークアドレスはFFxxになっています。

TK80モニタプログラムはCPUが8080としての動作をします。

KL5C80A12はZ80互換CPUですが、TK80モニタプログラムがCPUレジスタをRAMワークエリアにセーブ、ロードする対象は8080のもつレジスタだけです。Z80で追加された、A'、F'、B'、C'、D'、E'、H'、L'、IX、IY、I、R に対しては働きません。

ND80ZモニタプログラムはCPUがZ80であることを前提としていますから、モニタプログラムがCPUレジスタをRAMワークエリアにセーブ、ロードする対象はZ80の全てのレジスタです。

またTK80モニタプログラムにはない、ND80Z独自のモニタ機能が多く含まれています。

3つのモニタプログラムのうち、0400スタートのTK80モニタプログラムは、それだけを単独で使うことはできません。ND80Zモニタの制御のもとで、基本的なサブルーチンをコールして使うことはできます。

0000スタートのTK80モニタと0800スタートのND80Zモニタのどちらを使うかは、ディップスイッチ(DS1)のNo.1で選択できます。

(1)0000スタートのTK80モニタプログラムは、No.1を下側(OFF)にしたときに選択されます。

TK80のために書かれたプログラムの多くは、TK80モニタプログラムのサブルーチンをCALLしたり、TK80のワークエリアアドレスをアクセスしていますから、そのようなプログラムをキーから入力して実行させる場合には、0000スタートのTK80モニタプログラムを選択しておくことが必要になります。

(2)0800スタートのND80Zモニタプログラムは、No.1を上側(ON)にしたときに選択されます。

ND80Zモニタプログラムを選択していても、0000番地からにあるTK80モニタプログラムのサブルーチンがCALLできないようになっているわけではありません。ただTK80モニタサブルーチンをCALLすると、83xx番地台のメモリがアクセスされて書き換わるほか、7セグメントLEDに表示することができません。

ND80Zモニタを選択した状態で、TK80モニタのサブルーチンを使用する場合は、0000スタートのTK80モニタプログラムではなくて0400スタートのTK80モニタプログラムのサブルーチンをコールするようにしてください。

ディップスイッチによるモニタプログラムの選択は、電源をONしたときか、RESET(MON)スイッチを押したときに行われます。それ以外のときにディップスイッチの設定を変更しても、そのままではモニタプログラムは切り換わりませんが、LED表示アドレスだけが強制的に切り換わりますから、LEDの表示がデタラメになります。

モニタプログラムを切り換えるには、ディップスイッチの設定を変更したあとでRESET(MON)スイッチを押してください。

または電源OFFの状態でも切り換えてから、電源をONにしてください(こちらの方法を推奨します)。

[7]マシン語モニタプログラムの操作説明書について(KL5C80A12CPU対象)

(1)「TK80モニタプログラム操作説明書」は、TK80モニタプログラムの操作を説明しています。

TK80モニタプログラムはCPUが8080であることを前提にしていますから、説明の中で例示するプログラムリストはZ80ニーモニックではなくて、8080ニーモニックの表記になっています。

Z80は8080のマシン語命令コード(16進コード)をそのまま実行できますから、ニーモニックが8080で書かれていても、そのマシン語命令コードをキー入力して実行することには支障はありません。

Z80に比べて8080の方がレジスタ数が少なく、命令数も少ないので、いきなりZ80を対象にするよりも、まず8080の命令に慣れる方が楽です。

モニタプログラムもND80ZモニタプログラムよりもTK80モニタプログラムの方が機能が少ないので、最初はまずTK80モニタプログラムの操作に慣れることからスタートしたほうがわかりやすいかもしれません。

ND80KL/86のキートップに貼られているキーシールはND80Zモニタプログラムの機能に合わせてあります。

TK80モニタプログラムでは、一部のキーの機能がND80Zモニタプログラムとは異なっています。

「TK80モニタプログラム操作説明書」はTK80モニタの機能を説明するために、そのようなキーについては、TK80のキー名で説明をしています。

もしわかりづらいようでしたら、アクリルの透明キャップを外して、キーシールにエンピツなどで、TK80でのキー名を書き込んでください(大きく相違しているのは下の4個のキーです)。

| [TK80キー] | [ND80Zキー] |
|----------|-----------|
| RET | CONT |
| STORE | *(I/O) |
| LOAD | REG |
| RESET | MON |

RDINC(READING)、RDDEC(READDEC)、WRINC(WRITEINC)はわずかな相違ですから気になるほどではないでしょう。

(2)「ND80Zモニタプログラム操作説明書」は、ND80Zモニタプログラムの操作を説明しています。

説明の中で例示するプログラムリストはZ80ニーモニックの表記になっています。

ND80ZモニタプログラムにはTK80モニタプログラムにはない、いろいろな機能が追加されていますが、基本的な機能やキー入力の仕方などはTK80モニタプログラムと同じです。

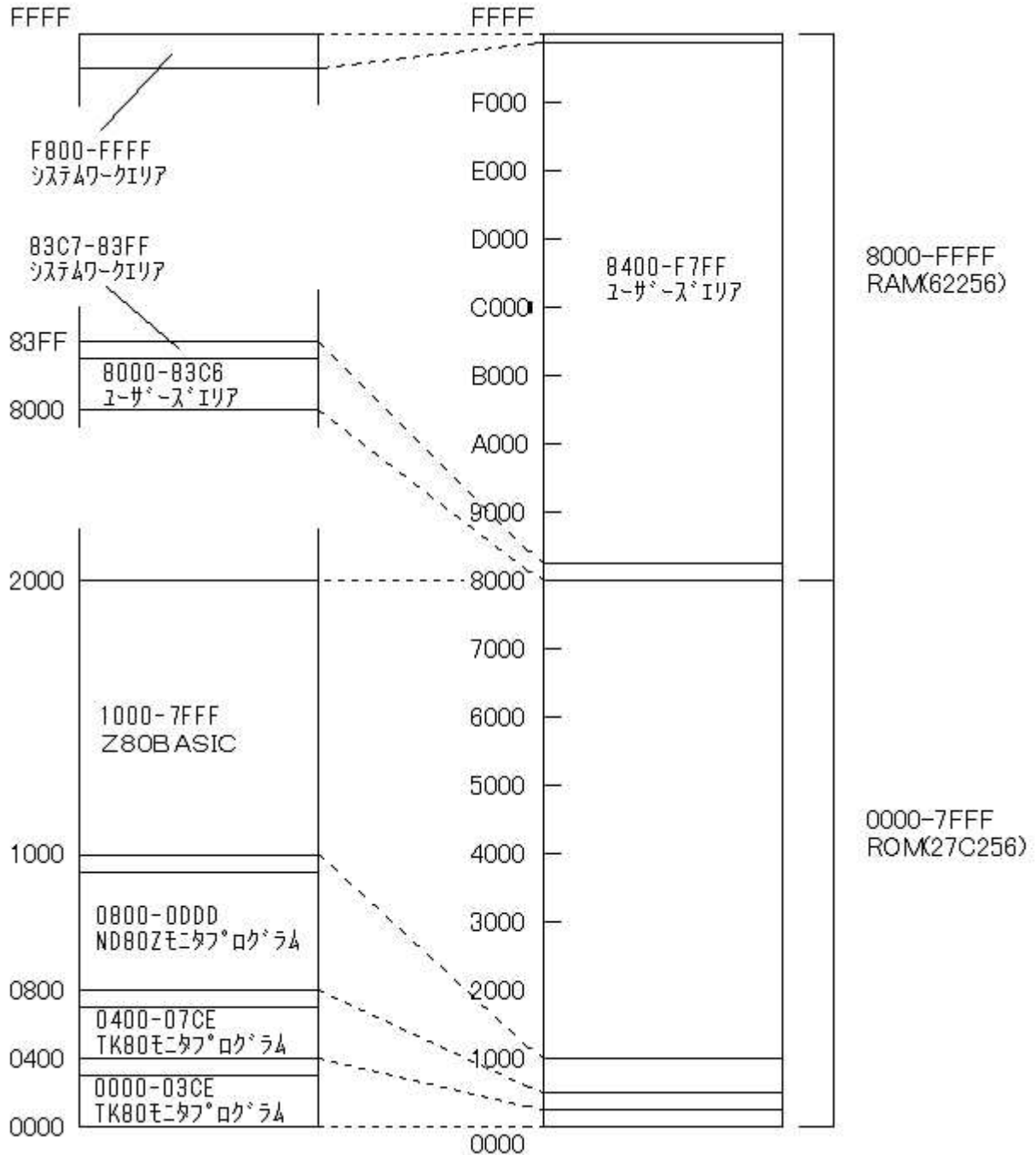
説明が重複しますから、そのような基本的な機能や、キー入力の仕方の説明は「ND80Zモニタプログラム操作説明書」では省略してあります。

ですからキー操作やモニタプログラムの基本的な操作に慣れるために、まず「TK80モニタプログラム操作説明書」から読むことをおすすめします。

[8]マシン語モニタプログラムのメモリマップ(KL5C80A12CPU対象)

ND80Z3. 5システムROMの3つのマシン語モニタプログラムはRAMワークエリアのアドレスなどが異なります。参考までに全体のメモリマップと、各モニタプログラムのRAMワークエリアのメモリマップを示します。

(1)全体のメモリマップ



(2)TK80モニタプログラム(0000スタート)ワークエリア

| | | |
|------|--------------------------------------|------------------|
| FFFF | ND80Zモニタ作業エリア (TK80モニタでは使用されない) | |
| FFCF | | |
| FFCE | | RST7ジャンプ先アドレス(H) |
| FFCD | | RST7ジャンプ先アドレス(L) |
| FFCC | | RST7ジャンプコード(C3) |
| FFCB | | RST6ジャンプ先アドレス(H) |
| FFCA | | RST6ジャンプ先アドレス(L) |
| FFC9 | | RST6ジャンプコード(C3) |
| FFC8 | | RST5ジャンプ先アドレス(H) |
| FFC7 | | RST5ジャンプ先アドレス(L) |
| FFC6 | RST5ジャンプコード(C3) | |
| FFC5 | RST4ジャンプ先アドレス(H) | |
| FFC4 | RST4ジャンプ先アドレス(L) | |
| FFC3 | RST4ジャンプコード(C3) | |
| FFC2 | RST3ジャンプ先アドレス(H) | |
| FFC1 | RST3ジャンプ先アドレス(L) | |
| FFC0 | RST3ジャンプコード(C3) | |
| FFBF | RST2ジャンプ先アドレス(H) | |
| FFBE | RST2ジャンプ先アドレス(L) | |
| FFBD | RST2ジャンプコード(C3) | |
| FFBC | RST1ジャンプ先アドレス(H) | |
| FFBB | RST1ジャンプ先アドレス(L) | |
| FFBA | RST1ジャンプコード(C3) | |
| FFB9 | ND80Zモニタ作業エリア | |
| FFB8 | Z32Cエラーコード | |
| FFB7 | ND80Zモニタ作業エリア (TK80モニタでは使用されない) | |
| FF68 | | |
| FF67 | ND80Zモニタリザーブエリア (TK80モニタでは使用されない) | |
| F800 | | |
| F7FF | ユーザー用プログラムエリアとして使用可 | |
| 83F8 | | |

| | |
|------|-------------------------|
| 83FF | LED表示用セグメントデータバッファ No.8 |
| 83FE | LED表示用セグメントデータバッファ No.7 |
| 83FD | LED表示用セグメントデータバッファ No.6 |
| 83FC | LED表示用セグメントデータバッファ No.5 |
| 83FB | LED表示用セグメントデータバッファ No.4 |
| 83FA | LED表示用セグメントデータバッファ No.3 |
| 83F9 | LED表示用セグメントデータバッファ No.2 |
| 83F8 | LED表示用セグメントデータバッファ No.1 |
| 83F7 | LED表示用データレジスタ No.4 |
| 83F6 | LED表示用データレジスタ No.3 |
| 83F5 | LED表示用データレジスタ No.2 |
| 83F4 | LED表示用データレジスタ No.1 |
| 83F3 | キー入力フラグ |
| 83F2 | ブレイクカウンタ |
| 83F1 | ブレイクアドレス(H) |
| 83F0 | ブレイクアドレス(L) |
| 83EF | アドレスレジスタ(H) |
| 83EE | アドレスレジスタ(L) |
| 83ED | データレジスタ(H) |
| 83EC | データレジスタ(L) |
| 83EB | CPUレジスタセーブエリア A |
| 83EA | CPUレジスタセーブエリア F |
| 83E9 | CPUレジスタセーブエリア B |
| 83E8 | CPUレジスタセーブエリア C |
| 83E7 | CPUレジスタセーブエリア D |
| 83E6 | CPUレジスタセーブエリア E |
| 83E5 | CPUレジスタセーブエリア H |
| 83E4 | CPUレジスタセーブエリア L |
| 83E3 | CPUレジスタセーブエリア SP(H) |
| 83E2 | CPUレジスタセーブエリア SP(L) |
| 83E1 | CPUレジスタセーブエリア PC(H) |
| 83E0 | CPUレジスタセーブエリア PC(L) |
| 83DF | RST6ジャンプテーブル |
| 83DD | |
| 83DC | RST5ジャンプテーブル |
| 83DA | |
| 83D9 | RST4ジャンプテーブル |
| 83D7 | |
| 83D6 | RST3ジャンプテーブル |
| 83D4 | |
| 83D3 | RST2ジャンプテーブル |
| 83D1 | |
| 83D0 | モニタ用スタックエリア |
| 83C7 | |
| 83C6 | |
| | ユーザー用スタック・プログラムエリア |
| 8000 | |

太線で囲んだ範囲はTK80モニタプログラムとND80Zモニタプログラムがともにアクセスする領域です。
RSTジャンプテーブルについての説明は次ページの(4)にあります。

(3)ND80Zモニタプログラム(0800スタート)ワークエリア

| | | | |
|------|-------------------------|------|------------------------------------|
| FFFF | LED表示用セグメントデータバッファ No.8 | FFCE | RST7ジャンプ先アドレス(H) |
| FFFE | LED表示用セグメントデータバッファ No.7 | FFCD | RST7ジャンプ先アドレス(L) |
| FFFD | LED表示用セグメントデータバッファ No.6 | FFCC | RST7ジャンプコード(C3) |
| FFFC | LED表示用セグメントデータバッファ No.5 | FFCB | RST6ジャンプ先アドレス(H) |
| FFFB | LED表示用セグメントデータバッファ No.4 | FFCA | RST6ジャンプ先アドレス(L) |
| FFFA | LED表示用セグメントデータバッファ No.3 | FFC9 | RST6ジャンプコード(C3) |
| FFF9 | LED表示用セグメントデータバッファ No.2 | FFC8 | RST5ジャンプ先アドレス(H) |
| FFF8 | LED表示用セグメントデータバッファ No.1 | FFC7 | RST5ジャンプ先アドレス(L) |
| FFF7 | LED表示用データレジスタ No.4 | FFC6 | RST5ジャンプコード(C3) |
| FFF6 | LED表示用データレジスタ No.3 | FFC5 | RST4ジャンプ先アドレス(H) |
| FFF5 | LED表示用データレジスタ No.2 | FFC4 | RST4ジャンプ先アドレス(L) |
| FFF4 | LED表示用データレジスタ No.1 | FFC3 | RST4ジャンプコード(C3) |
| FFF3 | キー入力フラグ | FFC2 | RST3ジャンプ先アドレス(H) |
| FFF2 | ブレイクカウンタ | FFC1 | RST3ジャンプ先アドレス(L) |
| FFF1 | ブレイクアドレス(H) | FFC0 | RST3ジャンプコード(C3) |
| FFF0 | ブレイクアドレス(L) | FFBF | RST2ジャンプ先アドレス(H) |
| FFEF | アドレスレジスタ(H) | FFBE | RST2ジャンプ先アドレス(L) |
| FFEE | アドレスレジスタ(L) | FFBD | RST2ジャンプコード(C3) |
| FFED | データレジスタ(H) | FFBC | RST1ジャンプ先アドレス(H) |
| FFEC | データレジスタ(L) | FFBB | RST1ジャンプ先アドレス(L) |
| FFEB | CPUレジスタセーブエリア A | FFBA | RST1ジャンプコード(C3) |
| FFEA | CPUレジスタセーブエリア F | FFB9 | リモートプログラム選択マーク |
| FFE9 | CPUレジスタセーブエリア B | FFB8 | 232Cエラーコード |
| FFE8 | CPUレジスタセーブエリア C | FFB7 | |
| FFE7 | CPUレジスタセーブエリア D | | モニタ用スタックエリア |
| FFE6 | CPUレジスタセーブエリア E | FFB6 | |
| FFE5 | CPUレジスタセーブエリア H | | システムリザーブエリア |
| FFE4 | CPUレジスタセーブエリア L | FFB5 | |
| FFE3 | CPUレジスタセーブエリア SP(H) | | システムリザーブエリア |
| FFE2 | CPUレジスタセーブエリア SP(L) | FFB4 | |
| FFE1 | CPUレジスタセーブエリア PC(H) | | システムリザーブエリア |
| FFE0 | CPUレジスタセーブエリア PC(L) | F800 | |
| FFDF | CPUレジスタセーブエリア IX(H) | | ユーザ用スタック・プログラムエリア |
| FFDE | CPUレジスタセーブエリア IX(L) | F7FF | |
| FFDD | CPUレジスタセーブエリア IY(H) | | ユーザ用スタック・プログラムエリア |
| FFDC | CPUレジスタセーブエリア IY(L) | | ユーザ用スタック・プログラムエリア |
| FFDB | CPUレジスタセーブエリア A' | 83F8 | |
| FFDA | CPUレジスタセーブエリア F' | | TK80モニタ作業エリア (ND80Zモニタでは使用されない) |
| FFD9 | CPUレジスタセーブエリア B' | 83F7 | |
| FFD8 | CPUレジスタセーブエリア C' | | TK80モニタ作業エリア (ND80Zモニタでは使用されない) |
| FFD7 | CPUレジスタセーブエリア D' | | TK80モニタ作業エリア (ND80Zモニタでは使用されない) |
| FFD6 | CPUレジスタセーブエリア E' | | TK80モニタ作業エリア (ND80Zモニタでは使用されない) |
| FFD5 | CPUレジスタセーブエリア H' | 83C7 | |
| FFD4 | CPUレジスタセーブエリア L' | | ユーザ用プログラムエリア |
| FFD3 | CPUレジスタセーブエリア I | 83C6 | |
| FFD2 | CPUレジスタセーブエリア R | | ユーザ用プログラムエリア |
| FFD1 | トレース、レジスタダンプ制御フラグ | | ユーザ用プログラムエリア |
| FFD0 | I/Oモード制御フラグ | | ユーザ用プログラムエリア |
| FFCF | レジスタモード制御フラグ | 8000 | |
| | | | ユーザ用プログラムエリア |

(4)RSTジャンプテーブル

システムワークエリアの中に、RST7～RST1ジャンプテーブルがあります。

これはユーザーがプログラム中でRST命令を使ったり、あるいは割り込み処理を行ったときに、ユーザー領域にジャンプさせるためのものです。

割り込みには通常はRST7を使うのですが、TK80モニタ、ND80ZモニタではRST7をステップ動作に使用しているためにユーザーが使うことはできません。

ユーザーに開放されているのはRST6～RST2です。

RST1はモニタリエントリアドレス(0051、0851)へジャンプします。

モニタリエントリはスタックポインタを再設定しますが、7セグメントLED表示などはクリアされません。

RST命令のエントリアドレスは0000～0038の間のアドレスで8バイトごとに置かれています。

ND80Z3. 5システムROMではそのアドレスはTK80モニタROMの領域なので、そこにユーザーが任意のジャンプ命令などを自由に書き込むことはできません。

そのための対策として、ROMに置かれている本来のRST命令のエントリアドレスには、上のメモリマップにあるRAMのアドレスへのジャンプ命令が書かれています。

たとえばアドレス0010はRST2のエントリアドレスですが、ND80Z3. 5モニタROMの0010には次のように書かれています。

ユーザープログラムの中でRST2命令が実行されると(あるいは割込みによってRST2が実行されると)、RAMのFBD番地にジャンプします。

ユーザープログラムの先頭で、FFBD~FFBFにユーザーが希望するRST2の処理ルーチンへのジャンプ命令を書き込むようにしておくことによって、ユーザーがRST命令を利用することができるようになります。

たとえば8200にジャンプさせたい場合には、FFBDにC3を、FFBEに00を、FFBFに82を書き込んでからRUNさせます。(あるいはユーザープログラムの先頭に次の命令を書いておきます)

```
3EC3 MVI A, C3
32BDFF STA $FFBD
210082 MVI H, $8200
22BEFF SHLD $FFBE
```

なお0000スタートのTK80モニタプログラムが選択されているときには、FFBA~FFCEのジャンプテーブルではなくて、83D1~83DFのジャンプテーブルを使います。

0000スタートのTK80モニタプログラムは起動時にFFBA~FFCEのジャンプテーブルに、83D1~83DFのジャンプテーブルへのジャンプ命令を書き込みます。

TK80モニタプログラムが選択されているときには、83D1~83DFにユーザープログラムへのジャンプ命令を書くことでRST命令を利用することができるようにするためです。

TK80モニタプログラムを選択しているときに、FFBA~FFCEを書き換えてRST命令を利用することも出来ませんが、ここでわざわざ83D1~83DFへジャンプするようにしているのは、TK80用に書かれたプログラムでRST命令を利用している場合には、83D1~83DFのジャンプテーブルを利用していますから、そのようなプログラムをできるだけ変更しないでそのまま使えるようにするためです。

なお、ジャンプテーブルのうち、RST1のアドレス(FFBA~FFBC)を書き換えると、ユーザープログラムでRST1命令(16進コードCF)を実行しても、モニタプログラムのリエントリアドレス(0051、0851)に戻れなくなります。

またRST7のアドレス(FFCC~FFCE)を書き換えると、モニタプログラムのステップ機能やブレイク機能、トレース機能が動作しなくなります。

そのような場合でもRESET(MON)キーを押せば初期状態に戻ります。

[9]モード2の割込みジャンプテーブル(KL5C80A12CPU対象)

ZBK-V3BASICはZ80のモード2の割込みをサポートしています。

ここで説明するモード2の割込みはハードウェア割込みです。

上で説明したRST0~RST7もハードウェア割込みとして使うこともできますが、このシステムではRST0~RST7命令を実行することで行なわれるソフトウェア割込みと定義づけています。

RST0~RST7はここで説明するモード2の割込み(ハードウェア割込み)とは別のものと考えてください。

モード2の割込みは複数の割込み処理プログラムをそれぞれの割込み要求元に合わせて選択実行できるように工夫されたものです。割込みプログラムはメモリのどのアドレスにでも自由に置くことができます。そしてそのアドレス(16ビット、2バイト)を割込み番号順にメモリの特定アドレスに(テーブル全体が上位8ビットアドレスが共通になるように)並べて格納しておきます(これを割込みテーブルと呼びます)。テーブルの置かれたアドレスの上位8ビットはレジスタに記憶させます。そして割込み番号(00、02、04、…、FE)を割込み元になるZ80ファミリ-ICに与えます。割込み元はCPUに割込み要求を出すとき、この与えられた割込み番号をCPUに伝えます(特定のタイミングでデータバスに送り出す)。CPUは受け取った割込み番号を下位バイト、レジスタの内容を上位バイトとして合成し、要求元に対応する割込みプログラムの置かれた割込みテーブルアドレスを算出します。

計算上は割込みテーブルには128個のプログラムアドレスを配置することができますが、ND80Kモニタでは16個分を確保しており、KL5C8012割込みコントローラのサポートする16レベルの割込みに合わせてあります。

割込みテーブルはRAMのFEE0H~FEFFHに置かれています。RAMですからテーブルを書き換えることによってユーザーの用意する割込みプログラムを実行させることが可能になります。割込みテーブルはKL5C8012の割込みコントローラの働きに合わせてありKL5C8012の特定機能割込みに割り当てられています。割込みテーブルには割込みプログラムアドレスを下位アドレス、上位アドレスの順に書き込みます。

[割込みテーブル]

| アドレス | 割込レベル | 割り込み要求元 |
|------|--------|-----------------------|
| FEFE | IR[15] | タイマ/カウンタBチャンネル2OUTS出力 |
| FEFC | IR[14] | タイマ/カウンタBチャンネル1OUTS出力 |
| FEFA | IR[13] | タイマ/カウンタBチャンネル0OUTS出力 |
| FEF8 | IR[12] | タイマ/カウンタAチャンネル1OUT出力 |
| FEF6 | IR[11] | タイマ/カウンタAチャンネル0OUT出力 |

| | | |
|------|--------|-------------------------------|
| FEF4 | IR[10] | USART(シリアルインタフェース) TXEMPTY出力 |
| FEF2 | IR[9] | USART(シリアルインタフェース) RXRDY出力 |
| FEF0 | IR[8] | USART(シリアルインタフェース) TXRDYPIN出力 |
| FEEE | IR[7] | 外部入力P07/IR7 |
| FEEC | IR[6] | 外部入力P06/IR6 |
| FEEA | IR[5] | 外部入力P05/IR5 |
| FEE8 | IR[4] | 外部入力P04/IR4 |
| FEE6 | IR[3] | 外部入力P03/IR3 |
| FEE4 | IR[2] | 外部入力P02/IR2 |
| FEE2 | IR[1] | 外部入力P01/IR1 |
| FEE0 | IR[0] | 外部入力P00/IR0 |

[注意1]

タイマ/カウンタBチャンネル1はZBK-V3BASICではKL5C8012内蔵シリアルインターフェース(RS232C)のクロックとして使用しています。ユーザーがZBK-V3BASICの232C機能を利用するときはチャンネル1をその他の用途に使うことはできません。

[注記]

IR[1](外部入力P01)はユーザー用外部割込み信号として拡張バスコネクタに接続端子を配置してあります。

[10]I/Oマップ

以下のI/Oアドレスに割り付けられている各I/Oインターフェースのうち、アドレス80~83の82C55以外のほとんどはモニタプログラムの制御に関係しています。

特にそのうち出力の出力に割り当てられているI/Oアドレスに対して不用意にデータを出力すると、モニタプログラムの動作が影響を受けることになりますから、注意してください。

そのような場合でもRESET(MON)キーを押せば初期状態に戻ります。

アドレス 00~7F 未使用

アドレス 80~83 IC8 I/OポートLSI 82C55
 80 Aポート
 81 Bポート
 82 Cポート
 83 コントロールワード

アドレス 84~8F 未使用

アドレス 90~93 未使用

アドレス 94~97 8ビット入力および4ビット出力ポート
 (入力)
 ビット0~3 PIC18F14K50データ
 ビット4~6 PIC18F14K50コントロール
 ビット7 ディップスイッチNo.1 (TK80/ND80Zモニタ選択)
 (出力)
 ビット0~3 PIC18F14K50データ

アドレス 98~9B 8ビット出力ポート
 ビット0~ビット2 PIC18F14K50コントロール
 ビット3 DATA出カイナーブルメディスイナーブル
 ビット4 7セグメントLED表示DMA制御
 ビット5 スピーカー出力
 ビット6 未使用
 ビット7 未使用

アドレス 9C~9F 3ビット出力ポート
 ビット0~2 5x5キーマトリクスラインセレクト (出力)

アドレス 9C~9F 8ビット入力ポート

アドレス A0～FF 未使用

[11]プログラム、データの保存

(1)メモリバックアップ

ROMに書かれたプログラムは電源を切っても消えないで残っていますが、RAMに書かれたデータやプログラムは電源を切ると失われてしまいます。

しかしND80KL/86はRAMをボード上のボタン電池でバックアップしていますから、電源を切ってもRAMに書かれたプログラムやデータは消えないでそのまま残っています。

ただプログラムミスや操作ミスでプログラムが暴走したりすると、RAMに書かれたデータやプログラムは破壊されてしまうことがあります。

そのような場合にそなえて、RAM上で作成したプログラムやデータは、USBケーブルでパソコンに接続して、パソコンのハードディスクに保存しておくことをおすすめします。

(2)USB接続によるプログラム、データの保存

TK80モニタプログラム、ND80Zモニタプログラムには、USBケーブルでパソコンと接続して、パソコンのハードディスクにプログラムやデータを保存したり、逆にパソコンのハードディスクに保存してあるプログラムやデータを、ND80KL/86のRAMにロードすることができます。

実際にUSB通信を行うインターフェース部分はZ80Aのプログラムではなくて、ボード上のPIC18F14K50が行います。

USB接続でプログラムやデータのSAVE、LOADを行うためには、あらかじめパソコン側で送信、受信のためのプログラムを準備しておく必要があります。

パソコン側でのUSB接続の準備については「USB接続説明書」を参照してください。

ND80KL/86の操作は、TK80モニタプログラムとND80Zモニタプログラムとでは、キーの操作が異なっています。

それぞれの具体的な操作の仕方については、「TK80モニタプログラム操作説明書」または「ND80Zモニタプログラム操作説明書」の「4章 プログラムのSAVE、LOAD」を参照してください。

[12]RS232Cインターフェース

TK80モニタプログラム、ND80Zモニタプログラムには、ユーザープログラムやデータをRS232Cインターフェースを使って外部に送出したり、外部から受信したりする機能があります。

実際にRS232C通信を行うインターフェース部分はシステムROMのプログラムではなくて、ボード上のPIC18F14K50が行います。

(1)仕様

RS232Cインターフェースの仕様は以下の通りです。

ボーレート 9600ボー、4800ボー、2400ボー（ディップスイッチにより選択）
データ長 8ビット
スタートビット 1
ストップビット 1
パリティビット なし
通信方式 調歩同期式（非同期式）
データ 8ビットバイナリ
コネクタ 9ピンDSUB（メス）

(2)接続ケーブル

RS232Cでの通信をおこなうときは、ND80KL/86の9ピンDSUBコネクタ（CN1）に232Cケーブルを接続します。

ND80KL/86のDSUBコネクタは9ピンメス型コネクタですから、接続ケーブルは9ピンオス型タイプになります。

RS232Cの接続はこちらのRXD(受信データ)を相手のTXD(送信データ)に、こちらのTXDを相手側のRXDに接続する(クロス接続方式)と、こちらのRXDを相手のRXDに、こちらのTXDを相手側のTXDに接続する(ストレート接続方式)とがあります。

いずれの接続方式にするかはこちら側と相手側の仕様によって決まります。

相手側のコネクタも9ピンの場合と25ピンの場合があって、さらにそのコネクタがオス型の場合とメス型の場合があります。

一般に双方のコネクタの形式に合ったRS232Cケーブルを用いることで、クロス接続かストレート接続かも合うことになります。

ND80KL/86のDSUB9ピンコネクタはWindowsパソコンの9ピンコネクタとはオスメスが逆になっています。

Windowsパソコンの232C9ピンコネクタはオス型になっています。

この場合の接続はストレート接続になります。

ND80KL/86の9ピンコネクタの端子信号は、相手(Windowsパソコン)側のコネクタの端子信号と同じ信号名になっています。

たとえば2番ピンは、WindowsパソコンではRXDですがND80KL/86の9ピンコネクタの2番ピンも信号名はRXDです。

しかしND80KL/86の回路では、232CDSUB9ピンコネクタの2番ピンにはRX(受信)ではなくてTX(送信)ラインが配線されています。

ですからこちら(ND80KL/86)のRXDと相手(Windowsパソコン)のRXDをストレートに接続することで、正しく通信が行われることになります。

[注記]ND80KL/86組立キットにはRS232Cケーブルは付属していません。

目的の用途に合った接続ケーブルを用意してください。

ND80KL/86のRS232CインターフェースはRXDとTXDとGNDの3本のラインしか接続しない、タレ流し式の接続です。

3本しか結線しませんから、コネクタさえ用意すれば、ケーブルを自作することも可能です(RS232C接続用9ピンDSUBコネクタの端子接続図は、次の「Ⅹ. コネクタ端子接続図」にあります)。

(3) タレ流し式の接続

ND80KL/86のRS232C通信は、ハードウェアハンドシェイクを行わない、タレ流し式の通信を行います。

タレ流し式ですから、こちらが受信する場合に相手が余り高速でデータを送ってくると、データを受信しそこねてしまいます。

また逆にこちらが送信の場合、相手があまり処理に時間がかかると、相手側が受信エラーになってしまいます。

しかし接続が簡単なので一般にパソコンではこのような簡易型の接続方法がよく用いられます。

ND80KL/86では、相手側がタレ流し式の結線になっていない場合でも、タレ流し式の送受信が行われるようにするため、DTR(4番ピン)とDSR(6番ピン)、RTS(7番ピン)とCTS(8番ピン)を基板内部でショートしてあります。

(4) ボーレート

ND80KL/86のRS232C通信は、ハード、ソフト上の都合で、ボーレートは、9600ボー、4800ボー、2400ボーのいずれかから選択します。

データの仕様は、8ビット、ノンパリティ、スタートビット1、ストップビット1のみです。

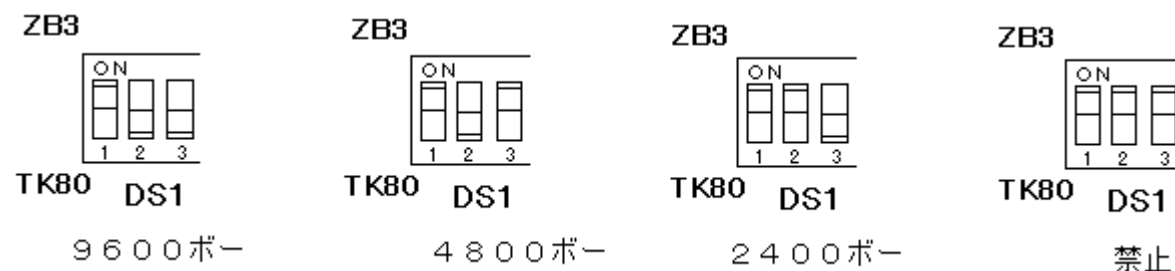
ボーレートは、ディップスイッチDS1のNo.2、No.3で設定します。

No.2 OFF、No.3 OFF 9600ボー

No.2 OFF、No.3 ON 4800ボー

No.2 ON、No.3 OFF 2400ボー

No.2 ON、No.3 ON 禁止



[注意]No.2とNo.3をともにONにしてはいけません。

ボーレートの設定はリセットまたは電源をONにしたときにだけ行われます。

電源をONにして、操作している途中でボーレートを変更するために、ディップスイッチを変更した場合には、一度リセットスイッチを押して、リセットしてください。

(5)通信プログラム

オリジナルのTK80モニタプログラムにはRS232Cのための通信機能はありません。

ND80KL/86はTK80モニタプログラムが選択されているときでも簡単にRS232C通信が行えるように、モニタプログラムROMにRS232C送信、受信プログラムを書いています。

簡単なプログラムを書いて、RS232C送信、受信プログラムにアクセスすることで、RS232C通信を行うことができます。具体的な使い方は、「TK80モニタプログラム操作説明書」の「7章 RS232C通信」を参照してください。

ND80Zモニタプログラムでは、簡単なキー操作でRS232C通信を行うことができます。

[*(I/O)][2(SI)]と続けて押すことでRS232C送信を行うことができます。

また[*(I/O)][3(SO)]と続けて押すことでRS232C受信を行うことができます。

具体的な使い方は、「ND80Zモニタプログラム操作説明書」の「4章 プログラム、データのSAVE、LOAD」を参照してください。

(6)受信エラー

受信エラーが発生すると、LEDにError. . . と表示されます。

通常発生するエラーはフレーミングエラーです。

ボーレートが違っているときに発生するエラーです。

通信状態が悪くてスタートビット、ストップビットの位置がおかしい場合にもエラーになります。

また、ND80KL/86が先に受信状態になっていなければならないのに、それ以前にデータが送信されてきたときもエラーになります。

その場合のエラーはオーバーランエラーか受信バッファオーバーフローエラーです。

発生した受信エラーの種類は、アドレスFFB8に入れています。

[F][F][B][8][ADRSSET]と操作することで、そのエラーの種類を知ることができます。

| | |
|------------------|----|
| フレーミングエラー | 04 |
| オーバーランエラー | 02 |
| 受信バッファオーバーフローエラー | 01 |

[13]コネクタ端子接続図

(1)CN1 RS232Cコネクタ(9pinDSUBコネクタメス)

| ピンNo. | 信号名 |
|-------|-------------|
| 1 | GND |
| 2 | RXD(送信) *注記 |
| 3 | TXD(受信) *注記 |
| 4 | ボード上でピン6と接続 |
| 5 | GND |
| 6 | ボード上でピン4と接続 |
| 7 | ボード上でピン8と接続 |
| 8 | ボード上でピン7と接続 |
| 9 | — |

*注記

9ピンメス型コネクタはパソコン側とストレートケーブルで接続することを前提にしているために、端子の信号名の表記は、接続する相手側の信号名をそのまま使うことが慣例になっているようです。

端子2のRXDは本来は受信端子を示しますが、ND80KL/86の回路では、信号送信ラインが接続されています。同様に端子3のTXDは本来は送信端子を示しますが、ND80KL/86の回路では、信号受信ラインが接続されています。

(2)CN9 82C55入出力コネクタ (26pinフラットケーブルコネクタ)

| | | | |
|-------|----|----|-----|
| ▷ PA5 | 1 | 2 | PA4 |
| PA7 | 3 | 4 | PA6 |
| PA2 | 5 | 6 | PA3 |
| PA0 | 7 | 8 | PA1 |
| +5V | 9 | 10 | GND |
| PC6 | 11 | 12 | PC7 |
| PC4 | 13 | 14 | PC5 |
| PC1 | 15 | 16 | PC0 |
| PC3 | 17 | 18 | PC2 |
| PB6 | 19 | 20 | PB7 |
| PB4 | 21 | 22 | PB5 |
| PB2 | 23 | 24 | PB3 |
| PB0 | 25 | 26 | PB1 |

(3) CN7 I/O増設用バスコネクタ(26pinフラットケーブルコネクタ)

| | | | |
|-----------|----|----|----------|
| ▷ RESETIN | 1 | 2 | GND |
| +5V | 3 | 4 | INT |
| GND | 5 | 6 | — |
| VRAMS* | 7 | 8 | RESETOUT |
| D0 | 9 | 10 | D1 |
| D2 | 11 | 12 | D3 |
| D4 | 13 | 14 | D5 |
| D6 | 15 | 16 | D7 |
| A0 | 17 | 18 | A1 |
| A2 | 19 | 20 | A3 |
| A4 | 21 | 22 | A5 |
| A6 | 23 | 24 | A7 |
| IORD | 25 | 26 | IOWR |

* VRAMSはAM188(KL5C80A12はNC)

(4) CN4 増設用バスコネクタ(16pinフラットケーブルコネクタ)

| | | | |
|-------|----|----|-------|
| ▷ A8 | 1 | 2 | A9 |
| A10 | 3 | 4 | A11 |
| A12 | 5 | 6 | A13 |
| A14 | 7 | 8 | A15 |
| MEMRD | 9 | 10 | MEMWR |
| — | 11 | 12 | — |
| A16 | 13 | 14 | A17 |
| A18 | 15 | 16 | A19 |

(5) CN5 RAMボード接続用20pinヘッダ

▷

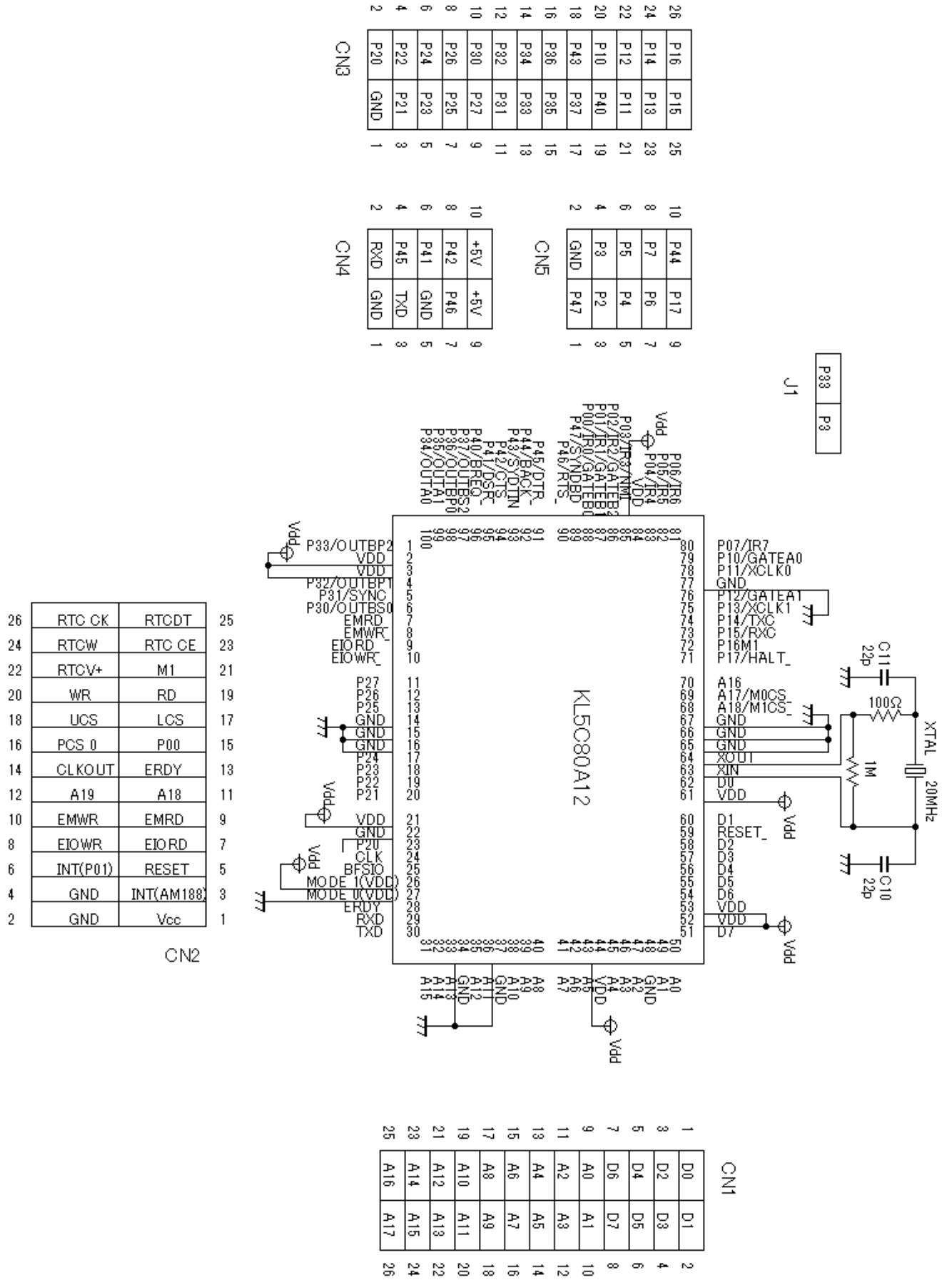
| | | | |
|--------|----|----|--------|
| GND | 1 | 2 | GND |
| RAMCE | 3 | 4 | RAMRD |
| A9 | 5 | 6 | A8 |
| A11 | 7 | 8 | A10 |
| A13 | 9 | 10 | A12 |
| A15 | 11 | 12 | A14 |
| A17 | 13 | 14 | A16 |
| RAMWR | 15 | 16 | A18 |
| — | 17 | 18 | — |
| RAMVCC | 19 | 20 | RAMVCC |

(6) CN6 RAMボード接続用20pinヘッダ

▷

| | | | |
|----|----|----|----|
| D1 | 1 | 2 | D0 |
| D3 | 3 | 4 | D2 |
| D5 | 5 | 6 | D4 |
| D7 | 7 | 8 | D6 |
| A1 | 9 | 10 | A0 |
| A3 | 11 | 12 | A2 |
| A5 | 13 | 14 | A4 |
| A7 | 15 | 16 | A6 |
| — | 17 | 18 | — |
| — | 19 | 20 | — |

[14]KL5C80A12CPUボードコネクタ端子接続図

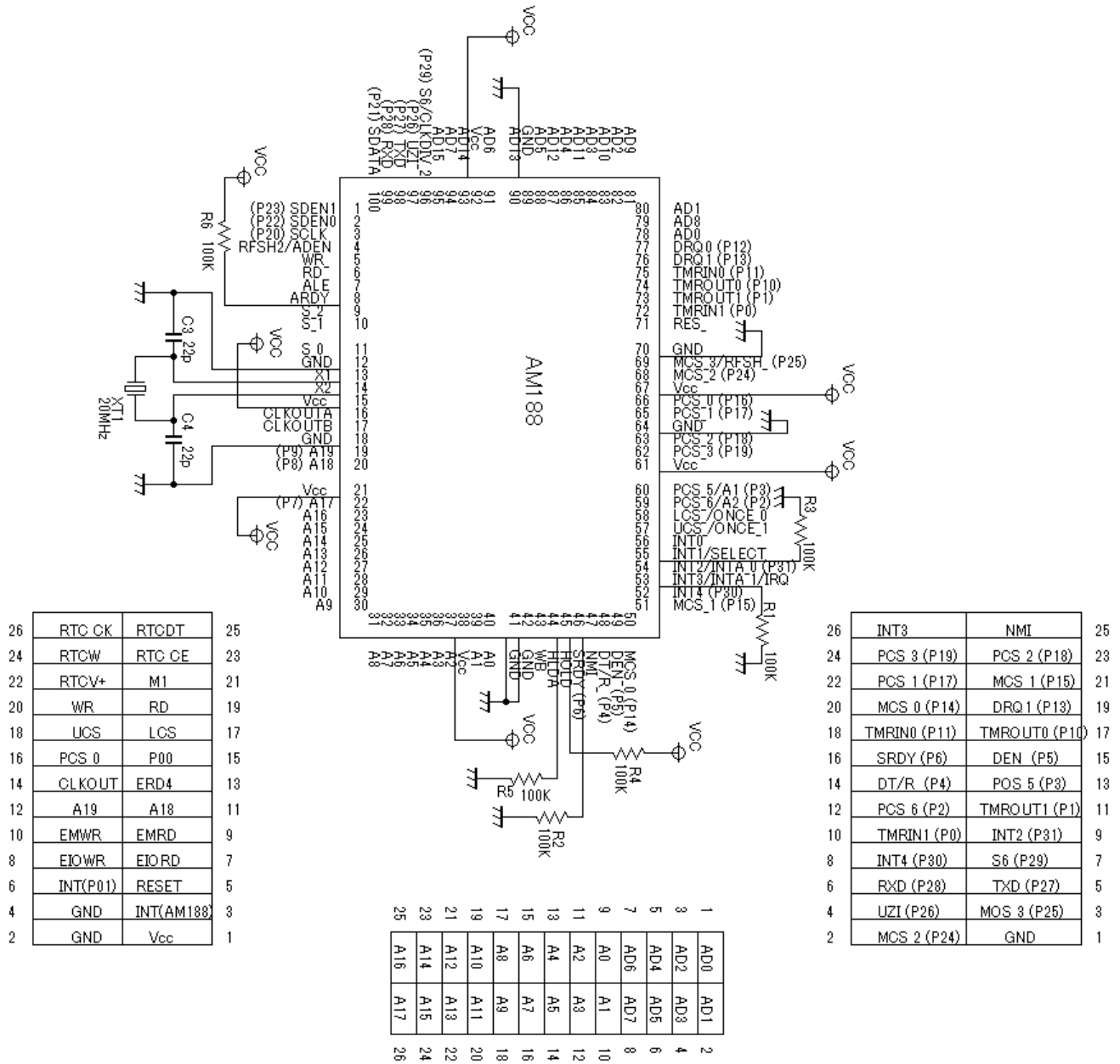


J1は将来の機能拡張用です。

[15]AM188CPUボードコネクタ端子接続図

| | | | |
|----|-----|-----|---|
| 10 | +5V | +5V | 9 |
| 8 | | | 7 |
| 6 | | GND | 5 |
| 4 | | TXD | 3 |
| 2 | RXD | GND | 1 |

| | | | |
|----|-------------|-------------|----|
| 16 | DRQ0 | | 15 |
| 14 | SDATA (P21) | SDEN1 (P23) | 13 |
| 12 | SDEN0 (P22) | SOLK (P20) | 11 |
| 10 | RFSH2 | ALE | 9 |
| 8 | ARDY | S.2 | 7 |
| 6 | S.1 | S.0 | 5 |
| 4 | HIDA | HOLD | 3 |
| 2 | GND | Vcc | 1 |



| | | | |
|----|----------|------------|----|
| 26 | RTC CK | RTCDT | 25 |
| 24 | RTCW | RTC CE | 23 |
| 22 | RTCV+ | M1 | 21 |
| 20 | WR | RD | 19 |
| 18 | UCS | LCS | 17 |
| 16 | PCS 0 | P00 | 15 |
| 14 | CLKOUT | ERD4 | 13 |
| 12 | A19 | A18 | 11 |
| 10 | EMWR | EMRD | 9 |
| 8 | EIOWR | EIORD | 7 |
| 6 | INT(P01) | RESET | 5 |
| 4 | GND | INT(AM188) | 3 |
| 2 | GND | Vcc | 1 |

| | | | |
|----|-----|-----|----|
| 1 | AD0 | AD1 | 2 |
| 3 | AD2 | AD3 | 4 |
| 5 | AD4 | AD5 | 6 |
| 7 | AD6 | AD7 | 8 |
| 9 | A0 | A1 | 10 |
| 11 | A2 | A3 | 12 |
| 13 | A4 | A5 | 14 |
| 15 | A6 | A7 | 16 |
| 17 | A8 | A9 | 18 |
| 19 | A10 | A11 | 20 |
| 21 | A12 | A13 | 22 |
| 23 | A14 | A15 | 24 |
| 25 | A16 | A17 | 26 |

| | | | |
|----|--------------|---------------|----|
| 26 | INT3 | NMI | 25 |
| 24 | PCS 3 (P19) | PCS 2 (P18) | 23 |
| 22 | PCS 1 (P17) | MCS 1 (P15) | 21 |
| 20 | MCS 0 (P14) | DRQ 1 (P13) | 19 |
| 18 | TMRIN0 (P11) | TMROUT0 (P10) | 17 |
| 16 | SRDY (P6) | DEN (P5) | 15 |
| 14 | DT/R (P4) | POS 5 (P3) | 13 |
| 12 | PCS 6 (P2) | TMROUT1 (P1) | 11 |
| 10 | TMRIN1 (P0) | INT2 (P31) | 9 |
| 8 | INT4 (P30) | S6 (P29) | 7 |
| 6 | RXD (P28) | TXD (P27) | 5 |
| 4 | UZI (P26) | MOS 3 (P25) | 3 |
| 2 | MCS 2 (P24) | GND | 1 |

[memo]