ロジックICで創る
CPUキット TTM8

取扱説明書

目次

1	概要	3
2	TTM8 ついて	3
3	インターフェイス	5
4	ユニット	8
5	LED	12
6	プログラムボード	18
7	操作説明	20
8	命令セット	24
9	プログラム方法	28
10	拡張レジスタ	29
11	スタートアップ	30

1　概要

本文書は”ロジック IC で創る CPU キット TTM8”の取扱説明書です。TTM8 を扱う前に必ずお読みください。初めて扱う方はまず項目 11 のスタートアップからお読みください。キットで CPU 本体の TTM8 と TTM8 にプログラムを書き込むための TTM8PB(TTM8 Program Board) がセットになっております。本書ではこれら二つについて取り扱います。

本書では TTM8 を一個の製品として取り扱い、内部構成については触れません。より内部構成を含めたより詳しい情報は別売りの解説本をご検討ください。

2　TTM8 ついて

TTM8 は計算機の内部構造を学習するための工作キットで、ロジック IC のみの組み合わせで動作する CPU です。8bitCPU であり汎用性のある命令を 16 種類実行可能な実用性のある CPU となっています。プログラムは最大 64 命令から構成することができます。命令の組み方によって様々な動作をさせることできます。搭載された RAM はバックアップ電池にて内容が保存されているため電源を落としても書き込んだプログラムは保持されます。

メインターゲットはこれから CPU の内部構造を学んでいきたいと考える学生や社会人です。さらに、この CPU はユーザーが自由に拡張 IO を増設できるように設計しております。熟練者やより学びを深めたいユーザー様には自ら拡張基板を製作して頂き、制御対象を TTM8 で制御するというような使い方も想定しております。

2.1　主要機能

表 1: TTM8：スペック

電源電圧	5V
消費電流	1A 以下
bit 数	8bit
命令種類	16
総レジスタ数	10
汎用レジスタ数	2
1 プログラムにおける最大命令数	64 命令
スタックおよびユーザー操作用 RAM 領域	32Byte
クロック	ジャンパーピンとスイッチにて選択 段階的に 250kHz から 244Hz, 低周波クロック, 手動クロック
入出力	なし (拡張して任意の IO を付与できる)

電源

TTM8の電源は5Vです。LEDが多く実装されている影響で比較的多くの電流を消費するのでパソコンのUSBポートからの給電はお勧めしません。1A以上のACアダプターあるいはモバイルバッテリーから給電してください。

クロック

TTM8にはクロックジェネレータが載っているので外部からクロックを供給する必要はありません。

クロック発振源は3種類あります。高周波クロックと低周波クロックと手動クロックです。

高周波クロックは4MHzクリスタルから分周した安定したクロックを提供します。低周波クロックは抵抗とコンデンサで簡易的なクロックです。このクロックは目で見える程度の速度で発振します。CPUの動きがLEDのチカチカで見たい時はこちらを選択します。また低周波クロックはボリュームで発振速度を調整できます。手動クロックはボタンを押すことで手動でクロックを供給するものです。1ステップずつ動作させたい時に使用します。

命令

TTM8は16種類の命令を持っています。詳細は命令セットの項目を参照ください。

命令は3つの区分に分かれています。オペコード部、ストアアドレス部、ロードアドレス部です。これがそれぞれ1byte分の領域を使用して3byte分でRAMに書き込みます。RAMに書き込む時はロードアドレス部、ストアアドレス部、オペコード部の順番に書き込みます。

RAM

TTM8にはRAMが搭載されております。アドレス空間は16進数で00からFFの8bit分です。

00からBFまでが実行するプログラムとして命令を保存する領域です。前述したように命令は3byte分のアドレスを用いるためこの192バイトの領域では最大64個の命令でプログラムを組むことができます。

C0からCFの領域は内部レジスタの領域です。内部レジスタへアクセスするときはこの領域の決まったアドレスへアクセスします。たとえばCレジスタはC2へDレジスタはC3へアクセスします。

D0からDFの領域は外部レジスタの領域です。拡張ポートから外部へレジスタを拡張したときに割り振られるアドレスはこの領域からです。拡張したレジスタがどのアドレスになるかは拡張ボードの仕様によります。

E0からFFの領域はスタックの領域です。リセット時にスタックポインタはFFを示しています。これは次にスタックにプッシュした時に格納するアドレスを示しています。スタックにプッシュするごとにFFからFEとアドレスの下方へ順に格納されていきます。最大32個までプッシュできます。またこの領域は変数の格納先としても使用することも想定しています。スタックがまだ使用していない任意のアドレスを変数として扱う事ができます。スタックが値を上書きする可能性があるので変数を使用する時はE0やE1などの下方の領域から使用するのがおすすめです。

スタック

TTM8にはスタックが搭載されています。スタックを扱う命令はPUSH,POP,CALL,RETURNの4つです。スタックポインタはリセット時に16進数でFFです。スタックされるごとにスタックポインタはインクリメントしてアドレスの下方に順に格納されていきます。スタックを搭載していることで一時的にデータの格納をすることができます。

入出力

TTM8には入出力が備わっておりません。CPUとメモリに加えてIOを備える事でコンピュータとい

う事ができます。ですから IO を備えていない TTM8 は CPU ということになります。その代わりに TTM8 が持っているレジスタには LED をぶら下げているので常に値をモニターすることができます。さらに拡張ポートに IO を拡張してコンピュータとすることができます。

3 インターフェイス

この節では TTM8 を操作するためのインターフェイスを取り扱います。図 1 に TTM8 の写真を示します。

1-拡張ポート

拡張ポートは TTM8 にレジスタを拡張する時に使用します。また TTM8 にプログラムを書き込むためにも使用します。書き込み用の基板であるプログラムボードはこの拡張ポートに接続して使用します。

2-電源スイッチ

TTM8 の電源スイッチです。押し込まれた状態で電源が供給されます。電源は 5V を投入してください。

3-電源投入口 (USB-B コネクタ)

電源投入口です。5V を投入してください。USB は通常、電圧 5V なので故意でもなければ高電圧が印加される心配はありませんが、意識的に電源電圧は確認した下さい。

LED が多く実装されている関係で、電流を比較的多く消費します。ゆえにパソコンの USB ポートから電源を供給する事はお勧めしません。モバイルバッテリーや USB ポートのついた AC アダプターを電源として用いる事をお勧め致します。

この電源投入口は通信には使用しません。

4-クロック周波数切り替えピンヘッダ

高周波クロックの周波数切り替え用のピンヘッダです。4MHz のクリスタルから分周して生成したクロックをこのピンヘッダにジャンパーピンを縦向きにを用いる事で選択します。

選択できる周波数は基板左から 250k,125k,62.5k,31.3k,15.6k,7.81k,3.91k,977,488,244Hz です。なおクロックを使用するときはクロック切り替えスライドスイッチで高周波クロックを選択して下さい。クロックを切り替える時は電源を落とした状態で切り替えてください。

5-クロック切り替えスライドスイッチ

左手のスイッチは 4MHz クリスタルから分周した高精度の高周波クロックと R、C で発振した低周波クロックを切り替えます。奥へスライドすると高周波クロック、手前へスライドすると低周波クロックです。

右手のスイッチは手動クロックと自動クロック (高周波クロックと低周波クロック) を切り替えます。奥へスライドすると自動クロック、手前へスライドすると手動クロックです。

クロックの切り替えは電源を落とした状態で切り替えてください。たとえば 250kHz のクロックを選択する時はピンヘッダの一番左にジャンパーピンを左側のスライドスイッチを奥に、右側のスライドスイッチを奥に位置させます。

6-手動クロックとリセットボタン

左手の黄色のボタンは手動クロックのためのボタンです。クロック切り替えスライドスイッチを手動クロックにしている状態でこのボタンを押下することで手動でクロックを送ることができます。ボタンの後段にはローパスフィルタをかましているのでチャタリングの心配はありません。

右手の赤色のボタンはリセットボタンです。このボタンを長押しする事で CPU 全体にリセットがかかります。

7-低周波クロック調整ボリューム

低周波クロックの発振速度をこのボリュームを回すことで調整できます。低周波クロックを使用するときはクロック切り替えスライドスイッチを低周波クロックに設定します。

8-RAM バックアップ用電池

RAM 内に保存されたデータを保持するためのバックアップ用電池です。CR2032 を取り付けます。この電池のおかげで TTM8 の電源を落とした時でも RAM には電源が供給されるため値が保持され次に TTM8 を使うときにも前回のプログラムが保持されている状態となります。この電池は定期的に交換してください。

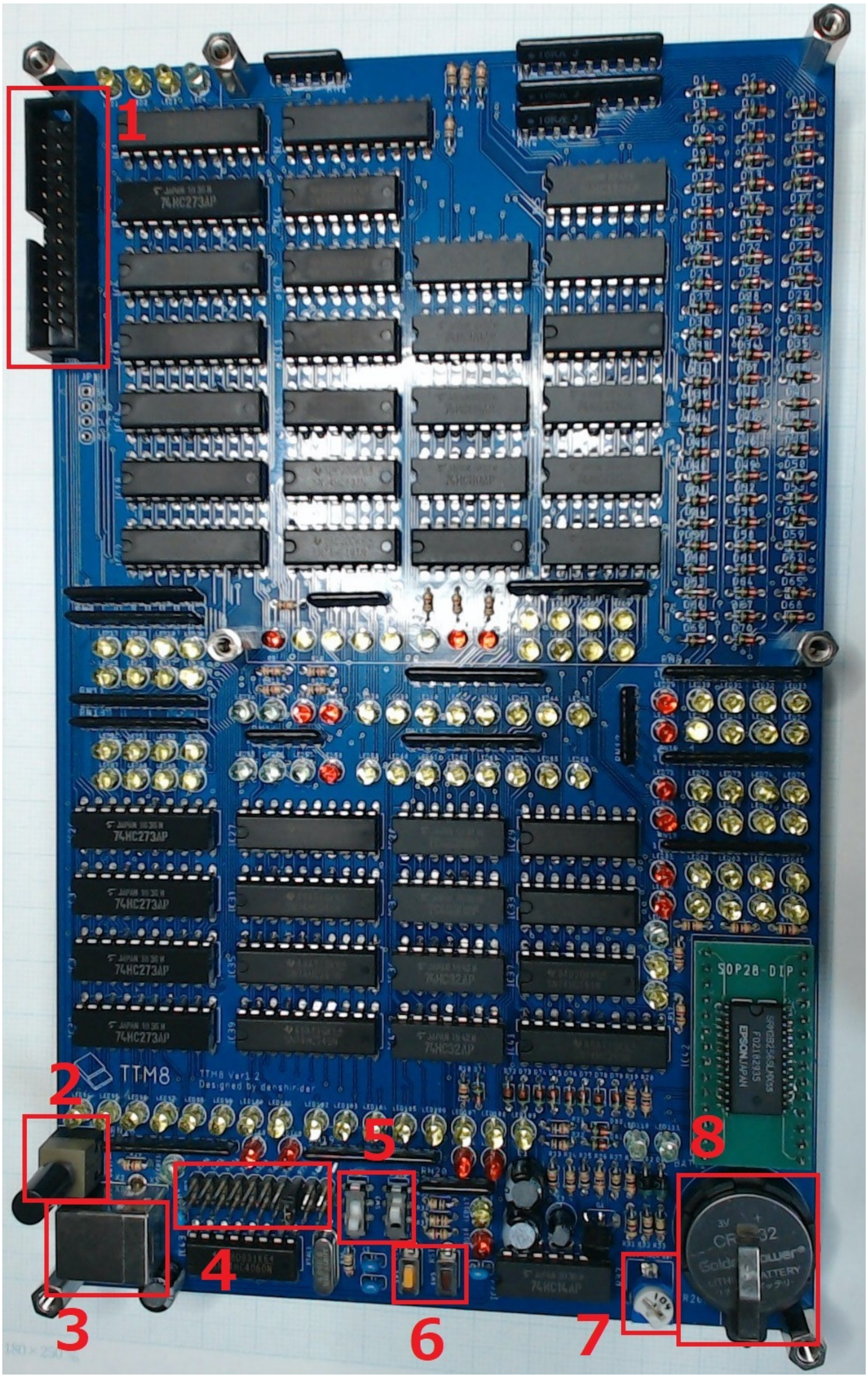


図 1: インターフェイス図解

4　ユニット

この節では TTM8 の各ユニットについて記載します。図 2 には TTM8 のブロック図を図 3 には実際の写真と共にユニットの位置を示します。

1-命令カウンタ

命令カウンタは CPU 制御部の一番根本に位置するレジスタです。TTM8 はまずこのカウンタが動作を開始するところから始まります。TTM8 の命令は 7 クロックで 1 命令が完了します。つまり 1 命令あたり 7 クロックとなります。命令カウンタはそれぞれのクロックでどんな動きをするか決定し 0 から 6 までカウントを繰り返します。

2-命令コントローラ

命令コントローラは命令カウンタの値を参照して実際の動作を決定するユニットです。命令カウンタの値により 7 種類の動作を切替え各ユニットの動作をコントロールします。

命令カウントが 0 の時の動き、1 の時の動き、2 の時の動き、〜6 の時の動きがあり、それぞれ違っています。0〜6 までの動作合わせて一個の命令を実行できます。

3-オペコードレジスタ

オペコードレジスタは命令全体のうちオペコードを格納するためのレジスタです。命令カウンタが 4 の時の次のクロックでオペコードが RAM からオペコードレジスタに格納されます。

4-命令デコーダ

命令を解釈するデコーダです。オペコードレジスタからオペコード (命令) を受け、これを解釈し各ユニットに動作の指示する TTM8 の中核の一つです。

5-アドレスレジスタ

アドレスレジスタは命令全体のうちストアアドレス、ロードアドレスの示す先にアクセスするために一時的に格納するためのレジスタです。またストアアドレスの部分がイミディエイトデータとなるような命令ではこのレジスタにイミディエイトデータが一時的に格納されます。

6-アドレスデコーダ

アドレスレジスタに格納されているアドレスを解釈して示されたアドレス先に読み取りや書き込み信号を送るためのデコーダです。

7-RAM

プログラムやスタックのデータを格納するためのメモリです。揮発性の SRAM ですがバックアップ電池を搭載しているのでデータは保持されます。アドレスバスのアドレスを参照してデータバスに出力するかあるいはデータバスのデータを書き込みます。

8-プログラムカウンタ

プログラムカウンタは今実行中のプログラムの番地を示したカウンタです。命令を RAM から取得する時はこのプログラムカウンタを参照します。

1 命令はオペコード、ストアアドレス、ロードアドレスの 3 バイトを使用するので、命令カウンタが一巡する (1 命令を実行) 毎にカウントが 3 進みます。

9-スタックポインタ

スタックポインタは次にスタックへ PUSH する時にデータが格納されるアドレスを示すレジスタです。

PUSH されるとデクリメントされ、POP されるとインクリメントされます。

10-A **レジスタ**

A レジスタは ALU が演算するためのデータを格納するレジスタの一つです。命令全体のうちロードアドレスで示された先のデータが格納されます。

11-B **レジスタ**

B レジスタは ALU が演算するためのデータを格納するレジスタの一つです。命令全体のうちストアアドレスで示された先のデータが格納されます。

12-ALU

A レジスタの値と B レジスタの値を元に演算をするユニットです。TTM8 では全加算器を ALU として使用しています。ADD、SUB、CMP 命令で使用されます。

13-C **レジスタ**

汎用のレジスタです。MVIC 命令を用いるとイミディエイトデータをこの C レジスタに格納することができます。

14-D **レジスタ**

汎用のレジスタです。MVID 命令を用いるとイミディエイトデータをこの D レジスタに格納することができます。

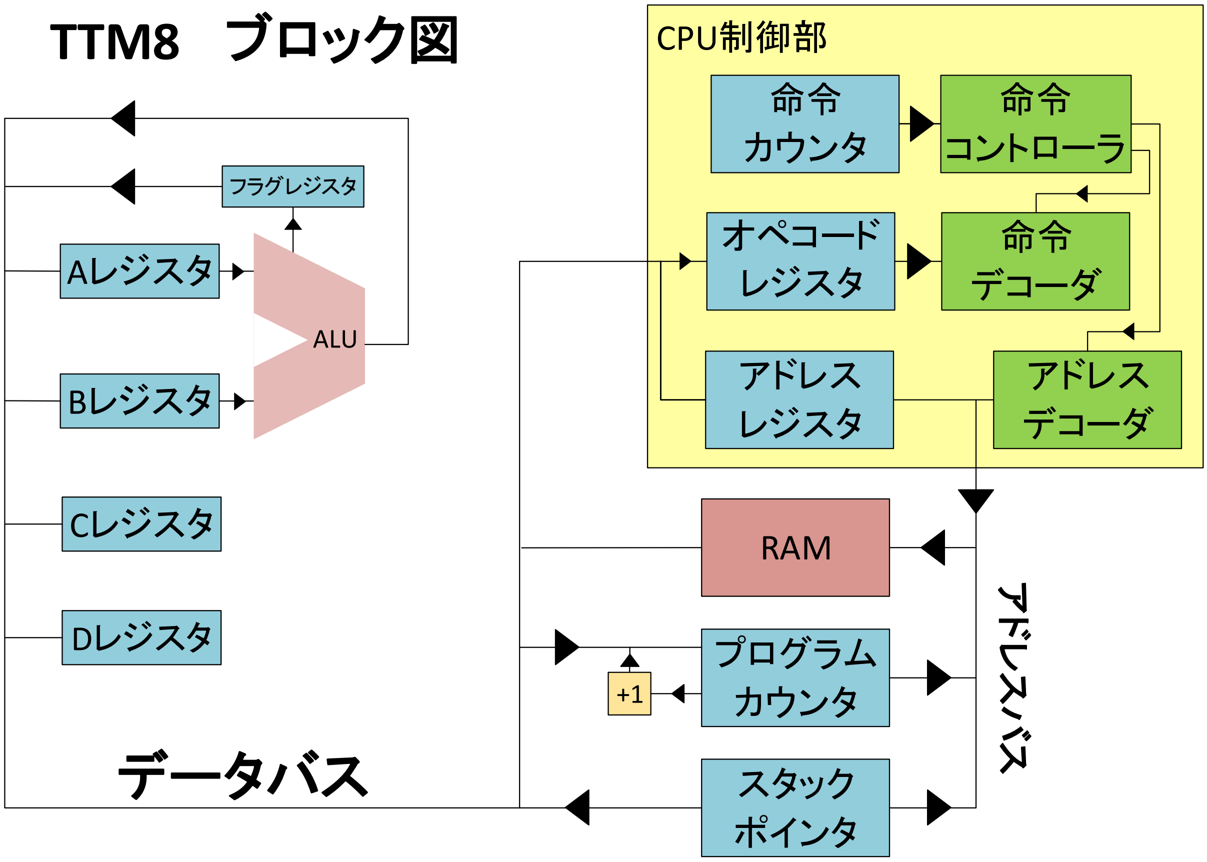

TTM8　ブロック図
CPU制御部
命令
カウンタ
命令
コントローラ
フラグレジスタ
オペコード
レジスタ
命令
デコーダ
Aレジスタ
ALU
Bレジスタ
アドレス
レジスタ
アドレス
デコーダ
Cレジスタ
RAM
Dレジスタ
プログラム
カウンタ
+1
アドレスバス
データバス
スタック
ポインタ

図 2: ブロック図

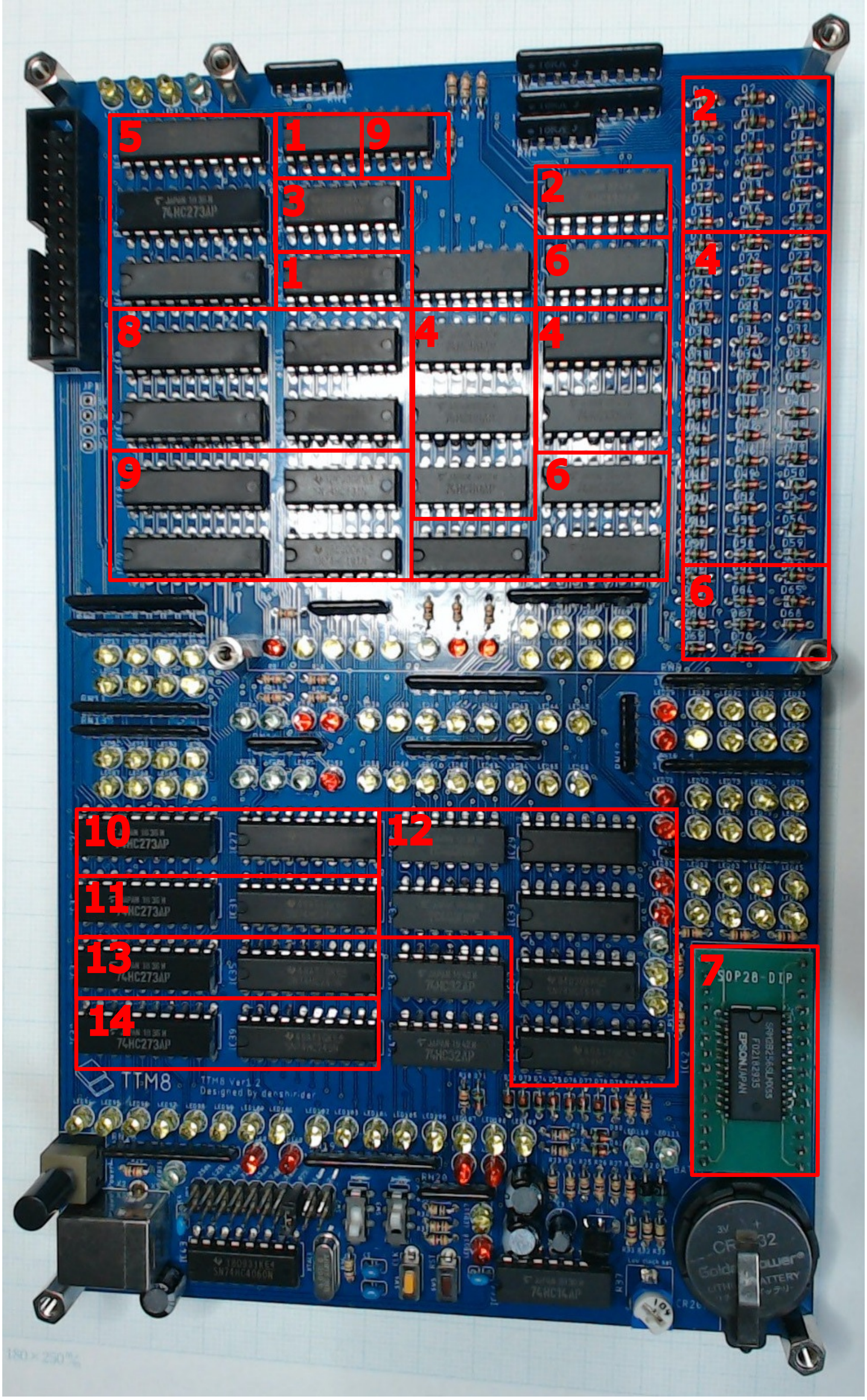


図 3: ユニット図

5 LED

TTM8 は搭載しているレジスタなどに LED を接続することで動作が視覚的に理解できるように工夫されています。本節ではそれぞれの LED が何を示すものであるか記載します。図 5 に LED のマップを示しします。LED があるところには各色でマークをつけました。複数の LED である値を示すものは一つにまとめています。

黄色枠で囲まれているレジスタの値を示す LED は点灯していれば 1 で消灯していれば 0 となります。左側が上位ビットで右側が下位ビットです。二段になっている場合は奥側が上位ビット手前側が下位ビットを示します。レジスタ操作系の信号を示すその他の LED はこの限りではありませんが基本的にその信号が有効となる時に点灯します。

図 4: LED が示すビット

1-命令カウンタ

3bit のレジスタ命令カウンタを示します。命令カウンタは TTM8 の制御部司令塔です。このカウンタ値により命令処理が切り替わります。

2-命令カウンタ-Enable

命令カウンタの ENABLE 信号です。点灯していれば命令カウンタが無効でカウントはせず CPU の動作が停止します。プログラムを書き込む時などは命令カウンタは停止させるので点灯します。消灯していれば命令カウンタは有効で CPU は動作をしている状態です。

3-アドレスバス

TTM8 のアドレスバスをモニターするための LED です。

4-データバス

TTM8 のデータバスをモニターするための LED です。

5-オペコードレジスタ-Write

オペコードレジスタへの書き込み信号があれば点灯します。

6-オペコードレジスタ

オペコードレジスタを示します。オペコードレジスタは命令全体のうちオペコードを格納するためのレジスタです。オペコードレジスタの先には命令デコーダが接続されています。

7-アドレスレジスタ-アドレスバス出力

アドレスレジスタの値をアドレスバスに出力するときに点灯します。

8-アドレスレジスタ-Write

アドレスレジスタへの書き込み信号があれば点灯します。命令全体のうちストアアドレス、ロードアドレスは一旦アドレスレジスタに格納されアドレスレジスタの先にあるアドレスデコーダによりアドレスを解析し示されたデータを読み取ります。この一旦格納するという時に点灯します。

9-アドレスレジスタ-Read

アドレスレジスタへの読み込み信号があれば点灯します。命令のうち即時データを C、D レジスタに格納する MVIC、MVID において即時データは一時的にアドレスレジスタに格納されているのでこれを読み込むときに点灯します。

10-アドレスレジスタ

アドレスレジスタを示します。アドレスレジスタは命令全体のうちストアアドレス、ロードアドレス、あるいは即時データを格納するレジスタです。アドレスレジスタの先にはアドレスデコーダが接続されています。

11-プログラムカウンタ-Enable

プログラムカウンタの Enable 信号です。点灯しているときにはプログラムカウンタは動作しておりクロックにしたがってカウントアップします。

12-プログラムカウンタ-アドレスバス出力

プログラムカウンタの値をアドレスバスに出力する時に点灯します。プログラムカウンタは実行するべき命令のアドレスが書かれているレジスタです。まずプログラムカウンタをアドレスバスに出力するところから命令実行が始まります。

13-プログラムカウンタ-Write

プログラムカウンタへの書き込み信号があれば点灯します。プログラムカウンタへの書き込みがあるという事は、それはすなわちジャンプ命令をさします。

14-プログラムカウンタ-Read

プログラムカウンタへの読み込み信号があれば点灯します。CALL 命令ではプログラムカウンタへの読み込み信号を発して値をスタックに保存します。

15-プログラムカウンタ

プログラムカウンタを示します。プログラムカウンタは実行中の命令のアドレスが書かれているレジスタです。クロックにしたがってカウントアップし順番に命令を実行する要となります。

16-スタックポインタ-Enable

スタックポインタの Enable 信号です。点灯しているときにはスタックポインタは動作しておりクロックにしたがってカウントアップあるいはカウントダウンします。

17-スタックポインタ-アドレスバス出力

スタックポインタの値をアドレスバスに出力する時に点灯します。

18-スタックポインタ-カウント方向

スタックポインタのカウント方向を示す信号です。点灯しているときはカウントアップの方向、消灯しているときはカウントダウンの方向をしめします。スタックポインタの Enable 信号がなければカウントはしません。

19-スタックポインタ-Read

スタックポインタへの読み込み信号があれば点灯します。

20-スタックポインタ

スタックポインタを示します。スタックポインタはスタックに PUSH するときに格納されるアドレスを示します。リセット値は FF です。スタックは FF から格納を開始し PUSH されるごとに FE,FD とカウントダウンされていきます。POP するとカウントアップします。

21-A レジスタ-Read

A レジスタへの読み込み信号があれば点灯します。ロードアドレスが示した先のデータは A レジスタに格納されます。MOV 命令では A レジスタに格納されているデータを読み込み格納先へ送ります。

22-A レジスタ-Write

A レジスタへの書き込み信号があれば点灯します。ロードアドレスが示した先のデータは A レジスタに格納されます。この時書き込み信号が発されこの LED は点灯します。

23-A レジスタ

A レジスタを示します。A レジスタの先には ALU が接続されており ALU は A レジスタの値と B レジスタの値を元に計算します。命令全体のうちロードアドレスが示す先のデータが A レジスタに格納されます。計算の必要がない命令では A レジスタからストアアドレスが示す先にデータを送ります。

24-B レジスタ-Read

B レジスタへの読み込み信号があれば点灯します。

25-B レジスタ-Write

B レジスタへの書き込み信号があれば点灯します。ストアアドレスが示した先のデータは B レジスタに格納されます。この時書き込み信号が発されこの LED は点灯します。

26-B レジスタ

B レジスタを示します。B レジスタの先には ALU が接続されており ALU は A レジスタの値と B レジスタの値を元に計算します。命令全体のうちストアアドレスが示す先のデータが B レジスタに格納されます。

27-フラグレジスタ-Read

フラグレジスタへの読み込み信号があれば点灯します。TTM8 にはフラグレジスタの値によって処理が変わるジャンプ命令がありますが、この時フラグレジスタのデータはデータバスを通らないのでこの LED は点灯しません。

28-加算器出力

ALU の計算結果出力信号があれば点灯します。ADD、SUB の計算結果を出力する必要がある時に点灯します。

29-加算器計算値

ALU である加算器の計算結果を示します。A レジスタの値と B レジスタの値の足し算結果、あるいは B レジスタの値から A レジスタの値を引いた値が示されます。なお計算結果がマイナスの数になる場合 8bit での 2 の補数表現で表します。

30-加算器-引き算信号

ALU である加算器で減算を行う場合点灯します。この LED が点灯している状態では加算器計算結果は減算をした値になります。

31-フラグレジスタ

フラグレジスタを示します。フラグレジスタは ALU の計算結果に基づいてセットされるフラグを格納するレジスタです。フラグが更新される命令は ADD、SUB、CMP 命令の 3 つです。キャリーフラグとゼロフラグがあります。手前側の LED がキャリーフラグで奥側の LED がゼロフラグです。フラグ

が立ったときに点灯します。このフラグを参照して処理が変わるジャンプ命令として JC、JNC、JZ、JNZ があります。

32-C レジスタ-Write

C レジスタへの書き込み信号があれば点灯します。

33-C レジスタ-Read

C レジスタへの読み込み信号があれば点灯します。

34-C レジスタ

C レジスタを示します。C レジスタは汎用のレジスタで計算を行う際に用いるレジスタです。

35-D レジスタ-Write

D レジスタへの書き込み信号があれば点灯します。

36-D レジスタ-Read

D レジスタへの読み込み信号があれば点灯します。

37-D レジスタ

D レジスタを示します。D レジスタは汎用のレジスタで計算を行う際に用いるレジスタです。

38-RAM 出力

RAM への出力信号があれば点灯します。点灯するときは RAM からデータバスに出力がある時です。命令の読み込みやスタックに格納したデータを POP する時などに点灯します。

39-RAM 書き込み

RAM への書き込みがあれば点灯します。スタックに PUSH する時や変数を RAM 上に用意する時に点灯します。外部からプログラムを書き込む時も RAM への書き込みを行いますがこの場合この LED は点灯しません。

40-クロック

クロックとともに点滅する LED です。クロックが High のときは点灯し Low の時は消灯します。RAM へのプログラムの書き込みを行う際はクロックは命令カウンタの動作とともに停止します。

41-リセット

リセット信号とともに点灯する LED です。点灯していればリセットがかかっている状態です。

42-電源 LED

TTM8 に電源が投入されているときに点灯する LED です。視覚的に電源投入状態がわかります。

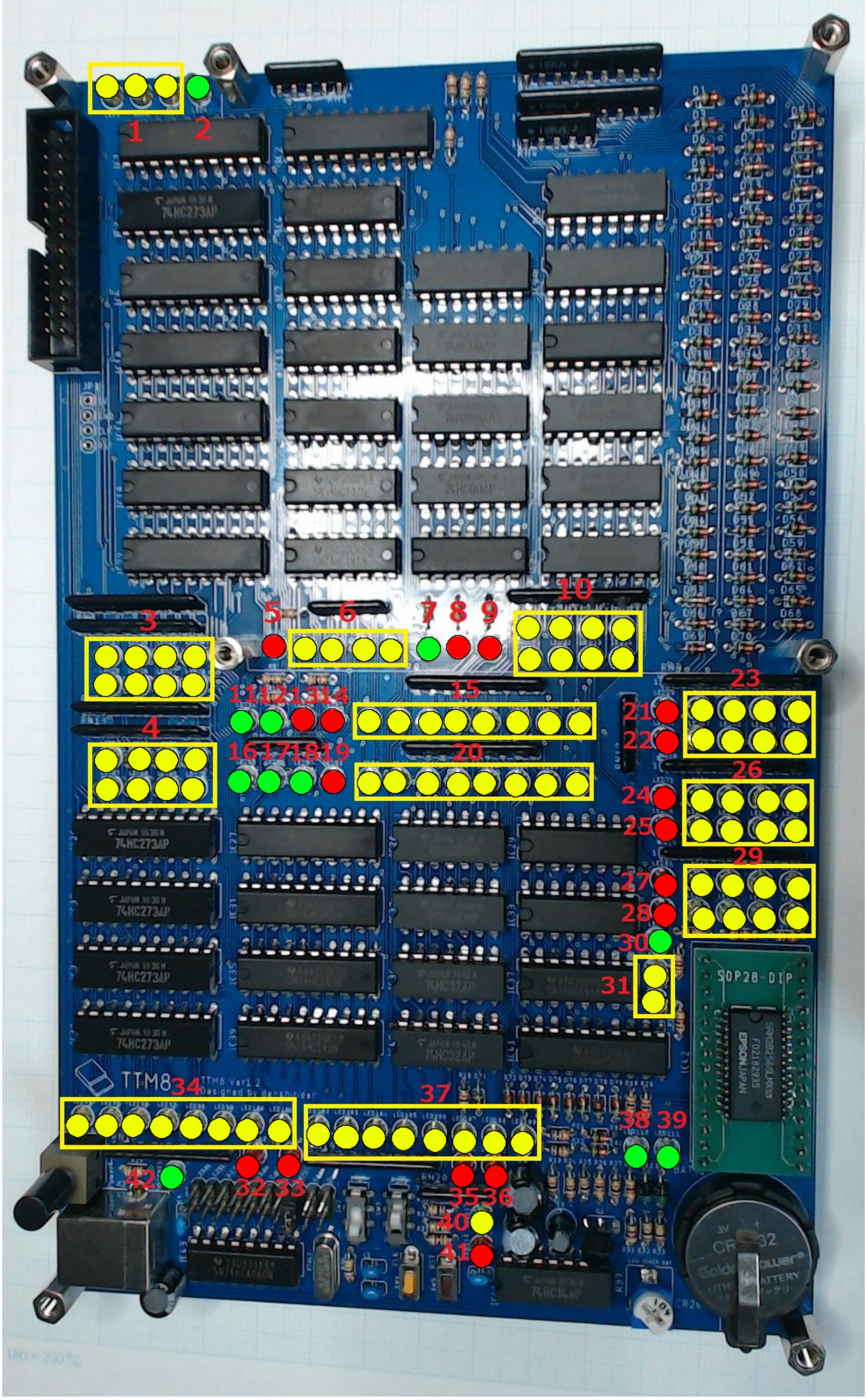

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42

図 5: LED 図解

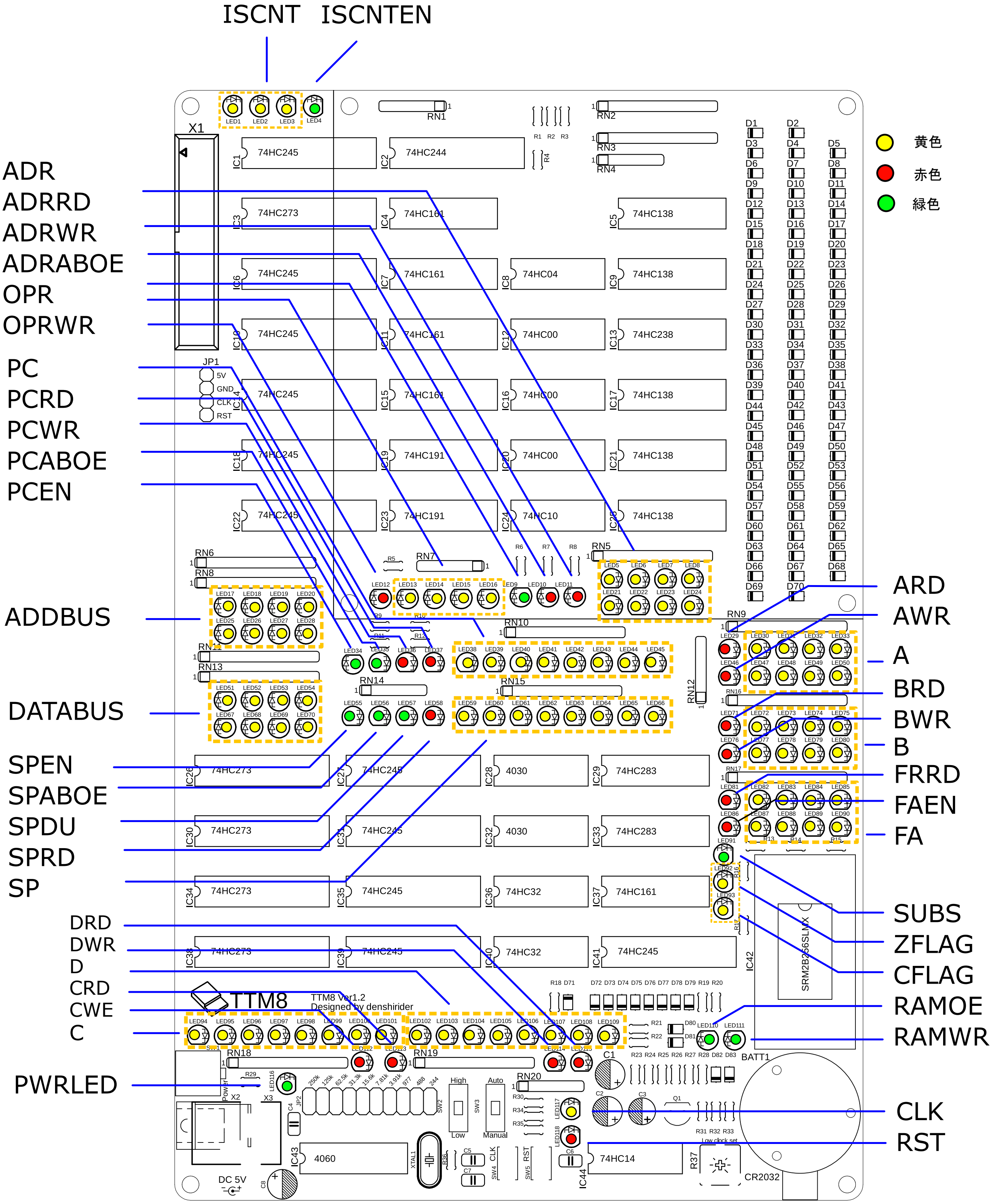

ISCNT
ISCNTEN
ADR
ADRRD
ADRWR
ADRABOE
OPR
OPRWR
PC
PCRD
PCWR
PCABOE
PCEN
ADDBUS
DATABUS
SPEN
SPABOE
SPDU
SPRD
SP
DRD
DWR
D
CRD
CWE
C
PWRLED
黄色
赤色
緑色
ARD
AWR
A
BRD
BWR
B
FRRD
FAEN
FA
SUBS
ZFLAG
CFLAG
RAMOE
RAMWR
CLK
RST
TTM8
TTM8 Ver1.2
Designed by denshirider

図 6: LED 早見図

6　プログラムボード

この節では TTM8 へプログラムを書き込むための TTM8 プログラムボードを説明します。TTM8 本体のみではプログラムを書き込むことができません。TTM8 の拡張ポートと TTM8 プログラムボードの拡張ポートを接続します。図 8 に TTM8 プログラムボードの写真とともに各部を示します。

1-拡張ポート

TTM8 本体と接続するための拡張ポートです。接続には 26 ピンリボンケーブルを用います。2 ポートあり、拡張ボードを次々に接続できるようになっております。どちらに接続してもかまいません。

2-プログラムボードリセットボタン

プログラムボードの処理を行っている PIC マイコンのリセットボタンです。基本的に操作する必要はありません。

3-START/STOP トグルスイッチ

TTM8 の動作の START/STOP を切り替えることができます。START 状態ではプログラムボードは TTM8 に対しては受動的な動作しかしません。STOP 状態ではプログラムボードは TTM8 に対して命令カウンタの停止とクロックの停止を要求します。プログラムを書き込む時には STOP 状態にします。START/STOP トグルスイッチを奥に倒すと START となり、手前に倒すと STOP となります。

4-WRITE/READ トグルスイッチ

プログラムの WRITE/READ を切り替えることができます。START/STOP トグルスイッチの操作により TTM8 を STOP 状態になっているときにこの WRITE/READ 切り替えは有効です。WRITE モードは RAM 上に値を書き込みをする時のモードです。READ は RAM 内の値を参照するモードです。WRITE/READ トグルスイッチを奥に倒すと WRITE となり、手前に倒すと READ となります。

5-リセットボタン

TTM8 上にあるリセットボタンと同様の動作をするリセットボタンです。長押しすることで TTM8 にリセットをかけることができます。

6-7 セグメント LED

プログラムを書き込む時に値を確認するための 7 セグメント LED です。4 桁になっており左手の 2 桁がアドレス値を示し、右手側の 2 桁がデータ値を示します。いずれも 16 進数で表示します。TTM8 が動作状態ではそれぞれアドレスバス、データバスの値を示し続けます。

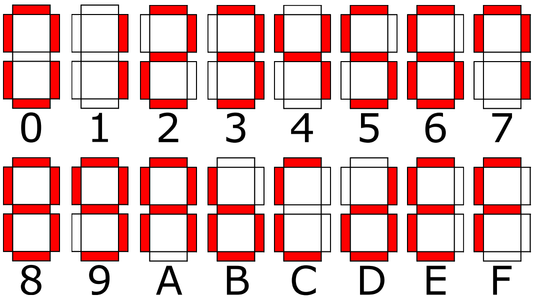


図 7: 7 セグメントと数値との対応表

7-キーパッド

プログラムを書き込む時や RAM 内を参照する時などにデータを入力するためのキーパッドです。0 から F までの 16 進数のキーパッドです。図 8 に示すような順で並んでいます。

8-機能キー

プログラムの書き込み等を実行するキーです。奥側から AS(ADDRESS SET)、RI(READ INCREMENT)、RD(READ DECREMENT)、WI(WRITE INCREMENT) キーです。AS キー：WRITE モード時 7 セグのデータ部に表示されている値をアドレス値にセットする。

RI、RD キー： 7 セグのアドレス部に表示されている値をインクリメント、デクリメントします。

WI キー：WRITE モード時 7 セグのデータ部に表示されている値をアドレス部にセットされている先に書き込み、アドレス値をインクリメントします。

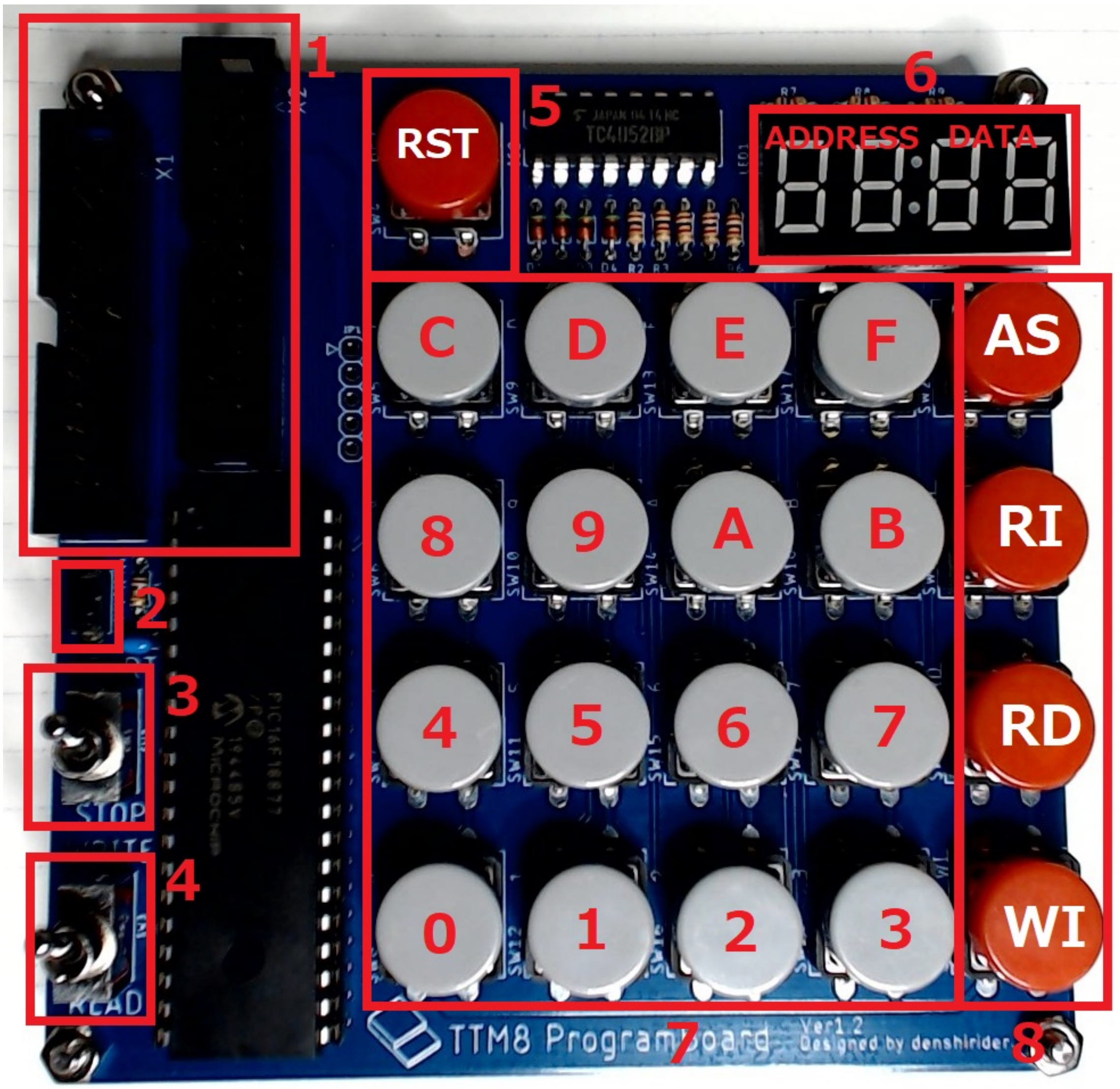


図 8: プログラムボード

7 操作説明

7.1 バックアップ電池をセットする

TTM8 に RAM のバックアップ電池をセットします。CR2032 のプラス極を上面にしてセットします。バックアップ電池は定期的に交換してください。

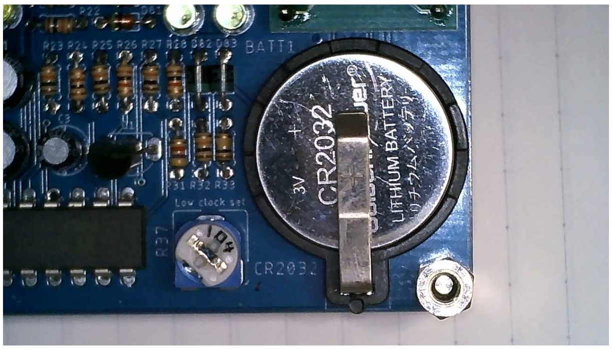


図 9: バックアップ電池

7.2 クロックを設定する

クロックは電源が OFF の時に設定してください。高周波クロック切り替えピンヘッダとクロック切り替えスライドスイッチでクロックを設定します。

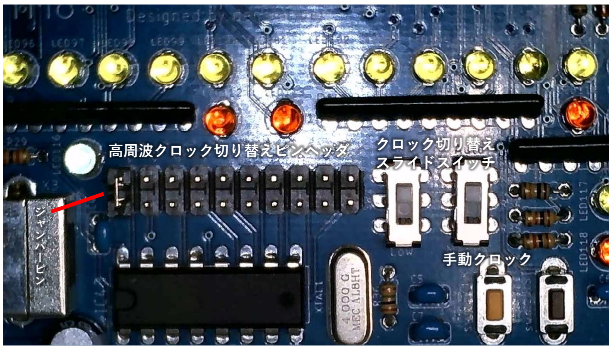


図 10: クロックの設定

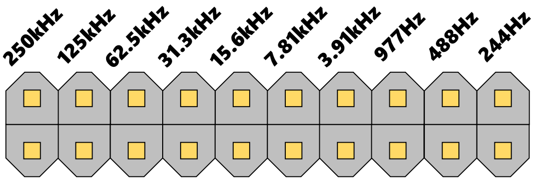


図 11: 高周波クロック切り替えピンヘッダ

手動クロックに設定する

右側のスライドスイッチを手前側にした状態にします。左側のスライドスイッチやピンヘッダの状態は問いません。手動クロックに設定すると手動クロックボタンでクロックを発する事ができるようになります。自身の任意のタイミングでクロックを発せるのでデバックに大変有効です。この手動クロックにはローパスフィルタを介しているのでチャタリングの心配はありません。

低周波クロックに設定する

左側のスライドスイッチを手前側へ、右側のスライドスイッチを奥側にします。ピンヘッダの状態は問いません。低周波クロックは人間の目に見える範囲でクロックを発するためのモードです。

TTM8 では各レジスタに LED を配置して CPU の動作状態を監視できるように設計しています。あまりに高周波だとこの LED はすべて点灯した状態に見えてしまうので動作状態を監視することができません。低周波クロックはこのために用意しています。

低周波クロックの周波数は TTM8 の右手前に配置している低周波クロック調整ボリュームにて調整することができます。微細な調整はできませんがここで程よい周波数に調整してください。なおこの周波数の調整は電源をつけた状態でも可能です。

高周波クロックに設定する

左側のスライドスイッチを奥側へ、右側のスライドスイッチを奥側にします。さらに高周波クロック切り替えピンヘッダで周波数を決定します。ジャンパーピンを奥と手前でショートする形で設定します。左側から 250k,125k,62.5k,31.3k,15.6k,7.81k,3.91k,977,488,244Hz です。

この高周波クロックは 4MHz のクリスタルから分周したクロックになりますので正確な周波数を設定することができます。実用的な動作をさせる時にこの高周波クロックに設定するとよいでしょう。ジャンパーピンが取り外しづらい場合ピンセットを用いてください。

7.3 プログラムボードと接続する

26 ピンリボンケーブルで TTM8 とプログラムボードを接続します。TTM8 の左奥にある拡張ポートのボックスヘッダに接続します。切り欠きがあるのでこれを合わせて接続します。プログラムボードは TTM8 と並べておくか、スペーサを利用して積み上げるようにおいてもよいです。

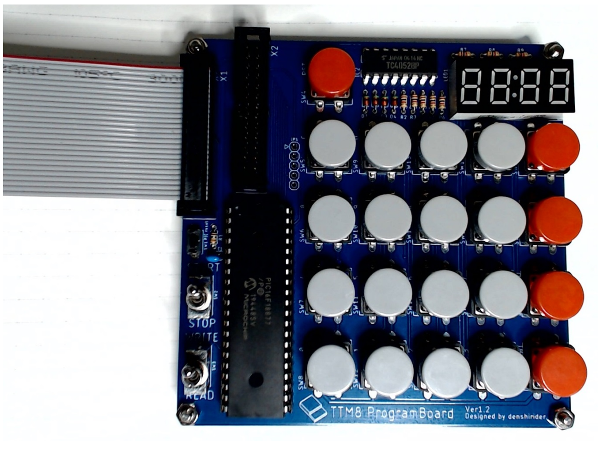


図 12: プログラムボードにリボンケーブルを接続する

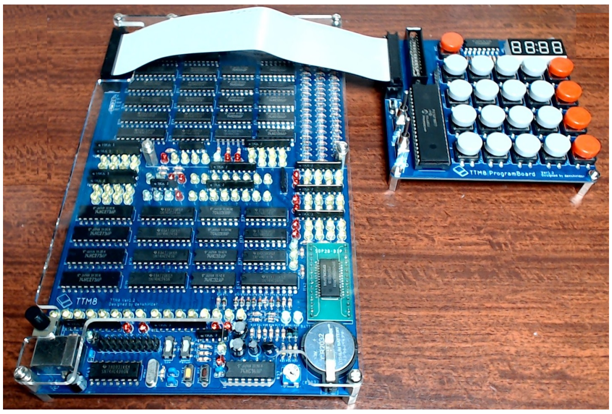


図 13: プログラムボードを接続する 1

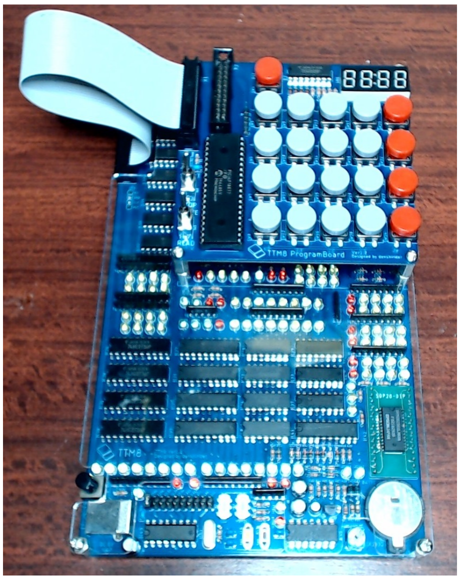


図 14: プログラムボードを接続する 2

7.4 電源を投入する

電源は TTM8 の左手前にある USB-B コネクタから投入します。USB-B であるのは確実に 5V を投入するためです。TTM8 では比較的 LED がたくさんついているので消費電流がそれなりにあります。なのでパソコンの USB ポートから給電することはお勧めしません。モバイルバッテリーや AC アダプターを使用してください。

USB-B ポートに電源を投入したら電源スイッチで電源を入れます。電源スイッチは押し込んで ON です。

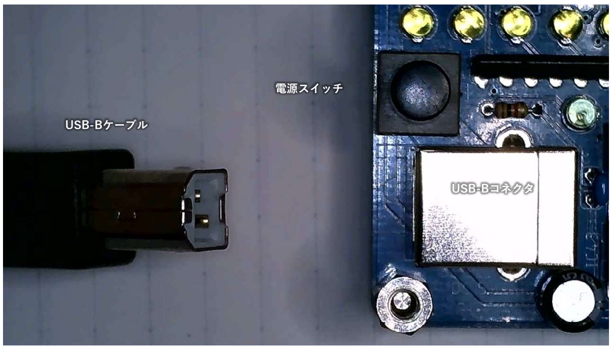


図 15: バックアップ電池交換

7.5 リセットする

TTM8 にリセットをかけると TTM8 に搭載されているレジスタが初期化されます。プログラムカウンタも初期化されるためプログラムが最初から走ることになります。RAM 内のデータは書き変わりません。リセッ

トボタンを長押しすることでリセットがかかります。リセットボタンは TTM8 上のリセットボタンかプログラムボードを接続している場合プログラムボード上にもリセットボタンがあります。

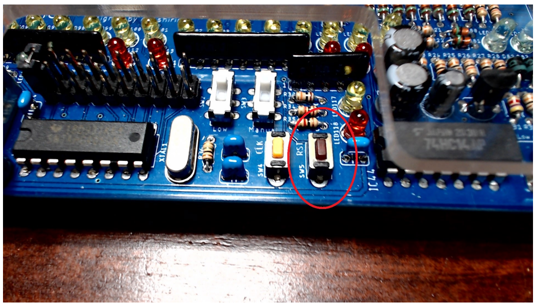


図 16: リセットボタン

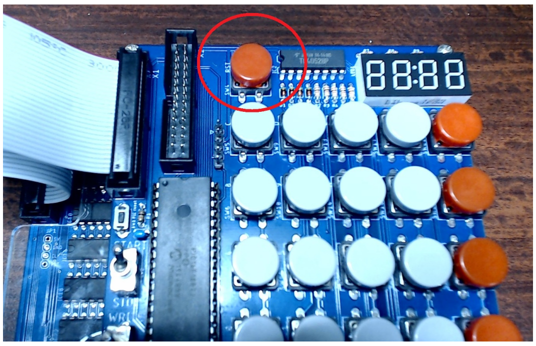


図 17: プログラムボード上のリセットボタン

7.6　START/STOP 切り替え

プログラムボード上の START/STOP トグルスイッチを操作することで TTM8 の動作の START/STOP を切り替えることができます。START 状態ではプログラムボードは TTM8 に対しては受動的な動作しかしません。STOP 状態ではプログラムボードは TTM8 に対して命令カウンタの停止とクロックの停止を要求します。プログラムを書き込む時には STOP 状態にします。START/STOP トグルスイッチを奥に倒すと START となり、手前に倒すと STOP となります。

7.7　WRITE/READ 切り替え

プログラムボード上の WRITE/READ トグルスイッチを操作することで、プログラムの WRITE/READ を切り替えることができます。START/STOP トグルスイッチの操作により TTM8 が STOP 状態になっているときにこの WRITE/READ 切り替えは有効です。WRITE モードは RAM 上に値を書き込みをする時のモードです。READ は RAM 内の値を参照するモードです。WRITE/READ トグルスイッチを奥に倒すと WRITE となり、手前に倒すと READ となります。

図 18: トグルスイッチ

8　命令セット

この節では TTM8 の命令セットについて解説します。表 2 に命令の一覧を表にして示します。後に各命令について詳細を解説致します。

命令は 3 つの要素で構成されています。(オペコード、値 1、値 2) です。命令によっては値 1 や値 2 のみ使用するものもありますが、その場合使用しない値にはダミーを置きます。必ず連続した 3 つのアドレス先に格納されプログラムカウンタは命令毎に +3 されます。

注意として、メモリには値 2、値 1、オペコードの順番で格納する必要があるので気を付けてください。表 2 の命令詳細に用いている略語は下部に記述。

”←”は値のコピーを示す。

”-”はその命令では参照されない値、ダミーデータを置く。

”()”で示すものは括弧内をアドレスとして用いて、その示す先。

STA：ストアアドレス

LDA：ロードアドレス

IM：即時データ

C：C レジスタ

D：D レジスタ

FR：フラグレジスタ

PC：プログラムカウンタ

SP：スタックポインタ

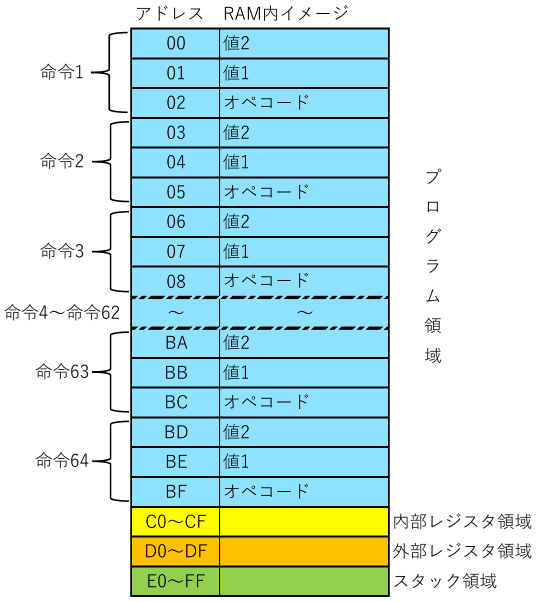


図 19: RAM 内のプログラム置き方イメージ

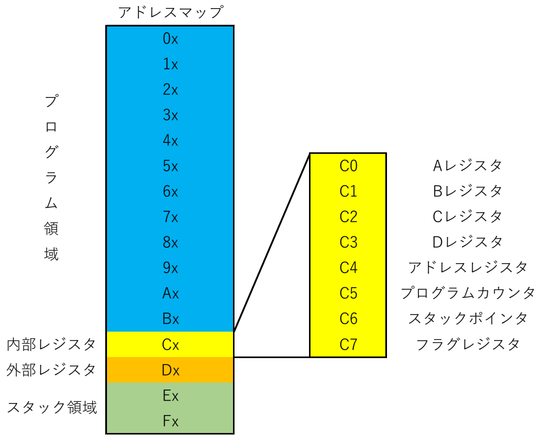


図 20: アドレスマップ

表 2: TTM8：命令セット

命令	機械語	ニーモニック	命令構成 オペコード, 値 1, 値 2	詳細
move	0	MOV	MOV, STA, LDA	ストアアドレス、ロードアドレスを指定して、ロードアドレス先の内容をストアアドレス先にコピーする。 ロードアドレス先の内容は変化しない。 (STA) ← (LDA)
move immediate C register	1	MVIC	MVIC, IM, –	即時データを C レジスタに格納する。 C ← IM
move immediate D register	2	MVID	MVID, IM, –	即時データを D レジスタに格納する。 D ← IM
addition	3	ADD	ADD, STA, LDA	ストアアドレス、ロードアドレスを指定して、それぞれの内容を加算しストアアドレス先に格納する。 ロードアドレス先の内容は変化しない。 フラグレジスタが更新される。 (STA) ← (STA) + (LDA) FR ← FR
subtraction	4	SUB	SUB, STA, LDA	ストアアドレス、ロードアドレスを指定して、ストアアドレス先の内容からロードアドレス先の内容を減算した値をストアアドレス先に格納する。 ロードアドレス先の内容は変化しない。 フラグレジスタが更新される。 (STA) ← (STA) - (LDA) FR ← FR
Comparison	5	CMP	SUB, STA, LDA	ストアアドレス、ロードアドレスを指定して、ストアアドレス先の内容からロードアドレス先の内容を減算した値でもってフラグレジスタを更新する。 ストアアドレス先の内容、ロードアドレス先の内容は変化しない。 フラグレジスタが更新される。 FR ← FR
address jump	6	ADJP	ADJP, –, LDA	ロードアドレスを指定して、ロードアドレス先の内容をプログラムカウンタに格納する。 ロードアドレス先の内容は変化しない。 PC ← (LDA)
jump	7	JMP	JMP, IM, –	即時データをプログラムカウンタに格納する。 PC ← IM
jump carry	8	JC	JC, IM, –	キャリーフラグが 1 の時即時データをプログラムカウンタに格納する。 PC ← IM (Cflag == 1)
jump not carry	9	JNC	JNC, IM, –	キャリーフラグが 0 の時即時データをプログラムカウンタに格納する。 PC ← IM (Cflag == 0)
jump zero	A	JZ	JZ, IM, –	ゼロフラグが 1 の時即時データをプログラムカウンタに格納する。 PC ← IM (Zflag == 1)
jump not zero	B	JNZ	JNZ, IM, –	キャリーフラグが 0 の時即時データをプログラムカウンタに格納する。 PC ← IM (Cflag == 0)
push	C	PUSH	PUSH, –, LDA	ロードアドレスを指定して、ロードアドレス先の内容をスタックポインタが示す先に格納する。 格納後スタックポインタはデクリメントされる。 ロードアドレス先の内容は変化しない。 (SP) ← (LDA) SP ← SP-1
pop	D	POP	POP, STA, –	スタックポインタをインクリメントする。 インクリメント後、指定したストアアドレス先にスタックポインタが示す先の内容を格納する。 ストアアドレスには内部レジスタのみ指定できる。 SP ← SP+1 (STA) ← (SP)
call	E	CALL	CALL, IM, –	プログラムカウンタの内容をスタックポインタが示す先に格納する。 格納後スタックポインタはデクリメントされ、即時データをプログラムカウンタに格納する。 プッシュされるプログラムカウンタの内容は内部処理の関係上、現在実行中のアドレス (CALL 命令が書かれているアドレス) より +1 された値となる。 (SP) ← PC SP ← SP-1 PC ← IM
return	F	RET	RET, –, –	スタックポインタをインクリメントする。 インクリメント後、プログラムカウンタにスタックポインタが示す先の内容を格納する。 SP ← SP+1 PC ← (SP)

8.1 MOV

MOV 命令は一番基本的な命令でデータのコピーをする命令です。値 1 にストアアドレスを値 2 にロードアドレスを指定します。ロードアドレスで示した先の内容をストアアドレスで示した先へコピーします。汎用レジスタである C レジスタや D レジスタ、外部レジスタのデータのやり取り、スタック領域を変数として使用して任意のアドレスとのやり取りに使用します。

8.2 MVIC、MVID

MVIC、MVID は即時データを直接 C レジスタ、D レジスタへ格納する命令です。値 1 に即時データを指定します。値 2 にはダミーを置きます。命令実行すると MVIC では即時データが C レジスタへ格納されます。MVID では D レジスタへ格納されます。TTM8 は基板を拡張しなければ入力ポートがありませんので、たとえば TTM8 で足し算をする時、計算の元となる値はこの命令で CPU が扱える形にします。

8.3 ADD、SUB、CMP

ADD、SUB、CMP はすべて ALU を使用する演算系の命令です。ストアアドレスとロードアドレスを指定します。これらの命令では命令実行に伴いフラグレジスタが更新されます。演算結果にキャリーが発生した場合 C(キャリー) フラグが立ちます。演算結果が 00 になった場合 Z(ゼロ) フラグが立ちます。

ADD ではストアドレスで示した先の内容とロードアドレスで示した先の内容を加算した値をストアアドレスで示した先へ格納します。SUB ではストアドレスで示した先の内容からロードアドレスで示した先の内容を減算した値をストアアドレスで示した先へ格納します。

CMP は値を比較するための演算です。演算動作としては SUB と同様です。しかし演算結果を格納することはありません。フラグレジスタのみ更新されます。CMP 命令後に Z フラグが立っていればストアドレスで示した先の内容とロードアドレスで示した先の内容が一致していた事が分かります。さらに C フラグを見れば大小関係が分かります。命令後 C フラグが立っていれば (STA)>(LDA) の関係となり C フラグが下がっていれば (STA)<(LDA) の関係になります。

8.4 ADJP

ADJP はアドレス値で指定するジャンプ命令となります。値 1 にはダミーを置きます。値 2 にロードアドレスを指定します。命令を実行することでロードアドレスで示した先の内容がプログラムカウンタへ格納されプログラムのジャンプを実現します。

8.5 JMP、JC、JNC、JZ、JNZ

JMP、JC、JNC、JZ、JNZ は即時データを指定してジャンプを行う命令です。値 1 にはジャンプ先のアドレスを即時データで指定します。値 2 にはダミーを置きます。JMP 命令は指定した即時データをプログラムカウンタに格納します。プログラムカウンタの値が変わることによって実行される命令が変わるのでジャンプ命令が実現できます。

その他の命令はフラグの状態によってジャンプするか否かが決まります。JMP 命令と同じようにプログラ

ムカウンタに即時データを格納する事でジャンプします。計算結果によってプログラムの分岐を行う場合に使用します。JC では C フラグが立っている時、JNC では C フラグが下がっている時、JZ では Z フラグが立っている時、JNZ では Z フラグが下がっているときにそれぞれジャンプします。

8.6 PUSH、POP

PUSH、POP はスタックを操作する命令です。PUSH 命令では値 1 にダミーを置き、値 2 にロードアドレスを指定します。PUSH 命令を実行するとロードアドレスで示した先の内容がスタックに PUSH されます。その後にスタックポインタがデクリメントされます。スタックポインタがデクリメントされることで、データを次々にスタックすることを実現しています。

POP 命令では値 1 にストアアドレスを指定して、値 2 にはダミーを置きます。POP 命令を実行するとまずスタックポインタがインクリメントされスタックに最後に PUSH したデータのアドレスを示すようにします。その後にそのスタックポインタの示す先の内容をストアアドレスで示した先にコピーします。PUSH と POP ではスタックポインタがカウントするタイミングが違います。

8.7 CALL、RET

CALL、RET はスタックを操作してサブルーチンの実行などに使用する命令です。プログラム中で分岐を行う事は前述のフラグレジスタを参照したジャンプ命令で実現可能です。しかしながらジャンプ命令では一度ジャンプした後に元々実行中であった番地まで戻ることはできません。CALL 命令はこれを可能にした命令です。

CALL 命令では値 1 にはジャンプ先のアドレスを即時データで指定して、値 2 にはダミーを置きます。JMP 命令と PUSH 命令を組み合わせた命令でこの CALL 命令が書かれている番地の次の命令の番地をスタックに PUSH して同時に即時データをプログラムカウンタに格納してジャンプを実現します。するとスタックには戻り先の番地が格納されています。RET 命令は POP 命令とほぼ同じ動作をするもので、POP 命令との違いは格納先がプログラムカウンタに固定いるところです。RET 命令は値 1、値 2 ともにダミーを置きます。CALL 命令でジャンプした先で処理が終わった後に RET 命令を実行することで元の実行中だった番地に戻ることができます。

9 プログラム方法

プログラムを TTM8 に書き込む時は拡張ポートにプログラムボードを接続します。RAM の各番地に 1 バイトずつ書き込みます。この時書き込みデータは 16 進数で指定します。プログラムボードはいくつかの機能キーと 16 進数に対応した 0〜F までキーパッドがあります。これらのキーの詳細についてはプログラムボードの項目をご確認ください。プログラム方法の手順は以下の通りです。

9.1 命令の書き込み番地

命令セットの項目で書いたように TTM8 の命令は一つで 3 バイト分使用します。プログラムの領域は 00 番地から BF 番地の 192 バイトなので $192 = 3 \times 64$ となり最大 64 命令まで書き込めます。TTM8 にリセットをかけるとプログラムカウンタは 00 にリセットされますからプログラムは 00 番地から開始されます。命令は 00 番地から 3 バイトずつで書き込んで行きましょう。00 番地、03 番地、06 番地・・という具合。図 19 参照。

9.2 RAM に書き込む

1-TTM8 の動作を停止する

START/STOP 切り替えトグルスイッチを手前側に倒して TTM8 の動作を停止します。

2-WRITE モードに設定する

WRITE/READ 切り替えトグルスイッチを奥側に倒して WRITE モードに設定します。

3-アドレスを入力する

データを書き込むアドレスを入力します。キーパッドにて 16 進数で入力します。キーパッドから入力したアドレス値は 7 セグ LED のデータ部分に表示されます。

4-アドレスをセットする

データを書き込むアドレスをセットします。アドレス値を入力したら機能キーのうち ADDRESS SET キーを押下することで 7 セグ LED のデータ部分にある値が 7 セグ LED のアドレス部分に遷移してアドレスがセットされます。

5-データを書き込む

書き込みたいアドレスがセットされた状態でデータを 16 進数で入力します。入力されたデータは 7 セグ LED のデータ部分に表示されます。後に機能キーのうち WI キーを押下する事でセットされたアドレス値に 7 セグ LED のデータ部分に表示されたデータを書き込みます。書き込みが終了すると同時に自動的にアドレス値がインクリメントされますので連続でデータを書くことができます。

9.3 RAM 内のデータを参照する

1-TTM8 の動作を停止する

START/STOP 切り替えトグルスイッチを手前側に倒して TTM8 の動作を停止します。

2-READ モードに設定する

WRITE/READ 切り替えトグルスイッチを手前側に倒して READ モードに設定します。

3-アドレスを入力する

データを参照したいアドレスをキーパッドで入力します。入力したアドレス値は 7 セグ LED のアドレス部分に表示されます。自動的にアドレス値にセットされた先の RAM の内容が 7 セグ LED データ部分に表示されます。

4-データを順番に参照する

機能キーのうち RI キーでセットされているアドレス値がインクリメントされます。連続で押すことで次々に内容を参照できます。RD キーではセットされているアドレス値がデクリメントされます。

10 拡張レジスタ

TTM8 は拡張ポートからレジスタを拡張して周辺機能を拡張する事ができます。拡張レジスタは D0〜DF までのアドレスに割り振られます。最大 16 個まで拡張できます。拡張は専用の拡張ボードを接続するか、ユーザー様が自身で回路を作成し拡張することも想定しています。拡張レジスタを自作する手法は別売りの解説本にて解説しています。

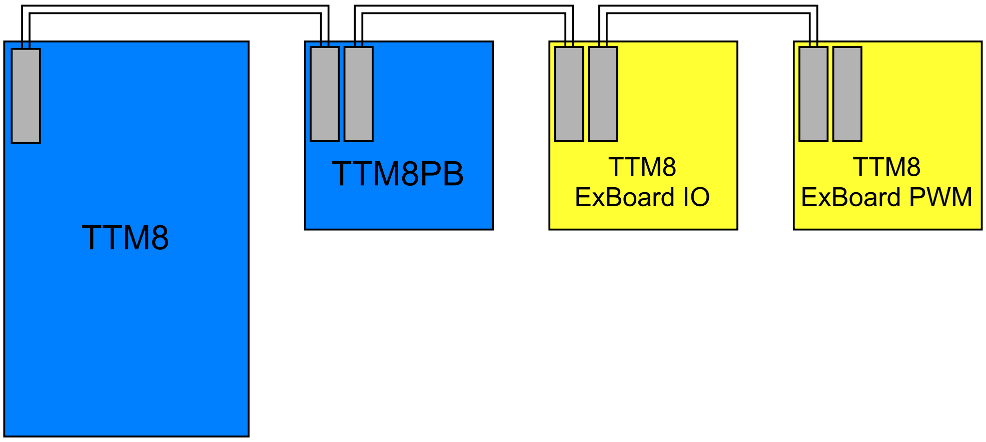


図 21: 数珠繋ぎで IO を拡張するイメージ

11　スタートアップ

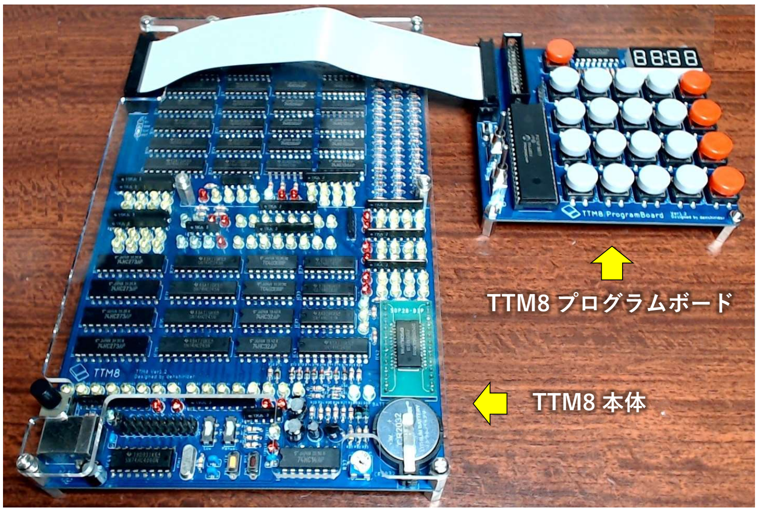


図 22: TTM8 と TTM8PB

本項では初めて TTM8 を触っていただく方に向けて細かい仕様は置いておいてまず動作させてみることを目標に解説します。

11.1　追加で用意するもの

本商品に付属しているもの以外にも用意するものがいくつかあります。以下にリストします。

電源
: TTM8 に給電する電源を用意してください。USB ポートがついた AC アダプターかモバイルバッテリをお勧めします。

USB-B ケーブル
: TTM8 に給電するためのケーブルです。

組み立てに必要な道具
: TTM8 は組み立てキットです。ご自身ではんだづけして完成させる必要があります。そのために必要な道具をご用意ください。詳細は組み立て説明書を参考にしてください。

11.2　TTM8 および TTM8 プログラムボードを組み立てる

TTM8 は組み立てキットです。ご自身ではんだづけして完成させる必要があります。詳細は別紙、組立説明書を参考にしてください。最後に RAM のバックアップ電池を取り付けてください。項目 7.1 を参照してください。

11.3　手動クロックを設定する

電源投入するまえにクロックの設定をします。手動クロックに設定してください。項目 7.2 を参照してください。

11.4　TTM8 とプログラムボードを接続する

TTM8 の拡張ポート使ってプログラムボードを接続します。項目 7.3 を参照してください。

11.5　電源を投入する

TTM8 に電源を投入します。初めて電源を投入するときはよくよくはんだづけを確認してください。テスターを用意して VCC と GND が短絡していないかは最低限確認しておくことをお勧めします。拡張ポートの 1 ピンが VCC で 2 ピンが GND となっています。電源投入は項目 7.4 を参照してください。

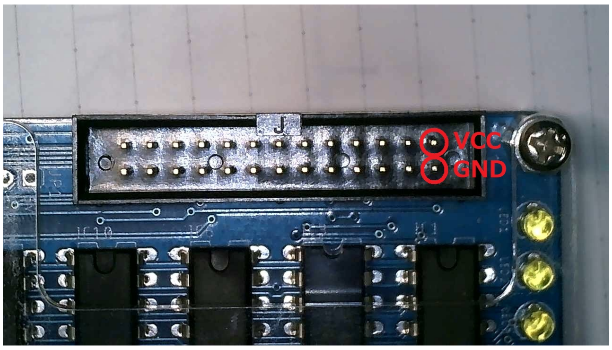


図 23: 拡張ポートの電源ライン

11.6　TTM8 の動作を停止して書き込みモードにする

プログラムを書き込む準備をします。TTM8 が動作している状態では書き込むことができません。TTM8 に接続されたプログラムボード上の START/STOP トグルスイッチを操作して TTM8 の動作を停止します。後にプログラムボードを書き込みモードにします。WRITE/READ トグルスイッチを操作して書き込みモードにします。操作は項目 7.7 を参照してください。

11.7　サンプルプログラムを書き込む

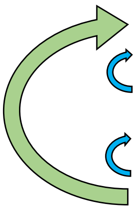
サンプルプログラムを実際に RAM に書き込んでいきます。サンプルプログラムとして D レジスタと C レジスタの LED を使って流れる光を作るプログラムを用意しました。動作としましては D レジスタの最下位 bit から 1bit ずつ光がながれていき C レジスタの最上位 bit まで光が進むとまた D レジスタ最下位に戻るというような動きをします。RAM に書き込む方法は項目 9 を参照してください。

以下に一部具体的に方法を示します。下記のデータ部とはプログラムボード上の 7 セグメント LED 右側 2 桁です。アドレス部とは同じく左側 2 桁です。プログラムを図 24、動作のフローチャートを図 25、RAM に

書き込む値を図 26、動作イメージを図 27 にそれぞれ示します。

1. TTM8 の動作を停止して書き込みモードにする
2. キーパッドの 0 を二回押す
   データ部に 00 が入力される
3. 機能キーの AS キーを押す
   アドレス部に 00 が遷移する
4. アドレス 00 には何もかきこまないので機能キーの WI キーを押す
   セットされているアドレスがインクリメントされ 01 がセットされる
5. キーパッドの 0、1 キーを順に押す
   データ部に 01 が入力される
6. 機能キーの WI キーを押す
   アドレス 01 にデータ 01 が書き込まれアドレスが 02 にセットされる
7. キーパッドの 0、2 キーを順に押す
   データ部に 02 が入力される
8. 機能キーの WI キーを押す
   アドレス 02 にデータ 02 が書き込まれアドレスが 03 にセットされる
9. キーパッドの C、3 キーを順に押す
   データ部に C3 が入力される
10. 機能キーの WI キーを押す
    アドレス 03 にデータ C3 が書き込まれアドレスが 04 にセットされる
11. キーパッドの C、3 キーを順に押す
    データ部に C3 が入力される
12. 機能キーの WI キーを押す
    アドレス 04 にデータ C3 が書き込まれアドレスが 05 にセットされる
13. キーパッドの 0、3 キーを順に押す
    データ部に 03 が入力される
14. 機能キーの WI キーを押す
    アドレス 05 にデータ 03 が書き込まれアドレスが 06 にセットされる
    ・
    ・
    ・

このように図 26 を参考にして次々にアドレス 00 からアドレス 14 までデータを書き込んでください。



命令番号	アドレス10進	アドレス16進	命令	ストア	ロード	動作
0	0	00	mvid	1		Dレジスタに01を格納する
1	3	03	add	DR	DR	DレジスタにDレジスタ＋Dレジスタを格納する
2	6	06	jnc	03		キャリーフラグが立っていなければ03番地へジャンプする
3	9	09	mvic	01		Cレジスタに01を格納する
4	12	0C	add	CR	CR	CレジスタにCレジスタ＋Cレジスタを格納する
5	15	0F	jnc	0x0C		キャリーフラグが立っていなければ0C番地へジャンプする
6	18	12	jmp	0x00		00番地へジャンプする

図 24: 流れる光のプログラム

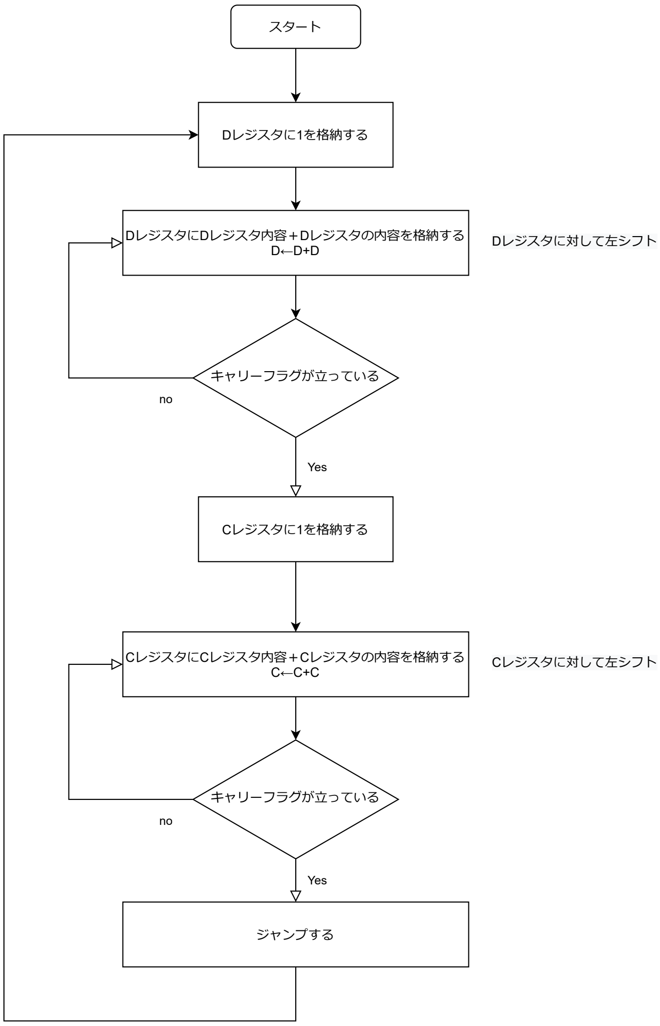


図 25: 流れる光 フローチャート

アドレス	書き込む値
00	-
01	01
02	02
03	C3
04	C3
05	03
06	-
07	03
08	09
09	-
0A	01
0B	01
0C	C2
0D	C2
0E	03
0F	-
10	0C
11	09
12	-
13	00
14	07

図 26: 流れる光のプログラム　書き込む値

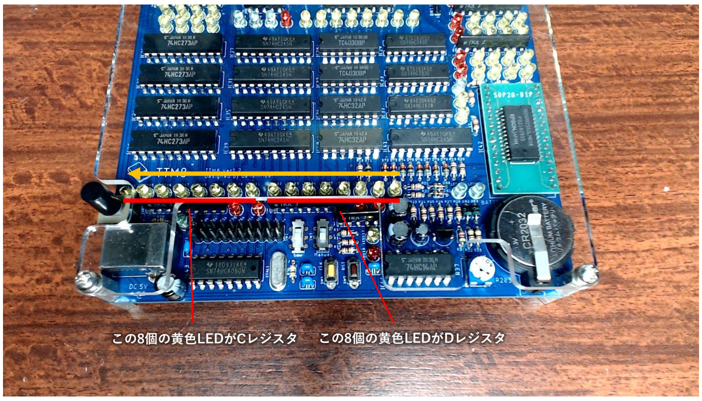


図 27: 流れる光のプログラム動作イメージ

11.8 サンプルプログラムを確認する

サンプルプログラムを書き込んだら書き込んだデータが正しいか確認しましょう。プログラムボードのWRITE/READ トグルスイッチを手前に倒して READ モードにします。READ モードでは RAM 内のセットされているアドレスの内容がデータ部に表示されます。機能キーの RI キー、RD キーでセットされているアドレスをインクリメント、デクリメントできます。これを用いてサンプルプログラムが正しく入力されているか確認してください。また間違いがあれば正しい値に修正してください。以下にアドレス 0E に誤って C2 を書き込んでしまっていた場合の修正手順を具体的に示します。

1. サンプルプログラムの書き込みが終了する。
   最後の WI キーが押されアドレス部に 15 がセットされているの状態
2. WRITE/READ トグルスイッチを手前に倒し READ モードにする。
3. キーパッドの 0 キーを二回押してアドレス 00 にセットして内容を確認する
   アドレス 00 の内容が参照されている状態
4. 機能キーの RI キーを押してアドレス 01 にセットし内容を確認する
   アドレス 01 の内容が参照されている状態
5. 機能キーの RI キーを押してアドレス 02 にセットし内容を確認する
   アドレス 02 の内容が参照されている状態
   ・
   ・
   ・
6. 機能キーの RI キーを押してアドレス 0E にセットし内容を確認する
   アドレス 0E の内容が参照されている状態
7. アドレス 0E の内容が誤って C2 であることを確認する
8. WRITE/READ トグルスイッチを奥に倒し WRITE モードにする
9. キーパッドの 0、3 キーを順に押して正しい値 03 を入力する
10. 機能キーの WI キーを押して正しい値を書き込む
    アドレス 0E にデータ 03 が書き込まれアドレスが 0F にセットされる

11. WRITE/READ トグルスイッチを手前に倒し READ モードにする。
12. 機能キーの RD キーを押してアドレスを 0E にセットして内容を確認する
    アドレス 0E の内容が参照されている状態
13. アドレス 0E に正しい値が書かれている事を確認する
14. 機能キーの RI キーを押してアドレス 0F にセットし内容を確認する
    アドレス 0F の内容が参照されている状態
    ・
    ・
    ・
15. すべてのアドレス先を確認して正しく書き込まれている事を確認する

11.9　TTM8 の動作を開始する

プログラムが正しく入力できているか確認できたら TTM8 を動作させてみます。正しく動作しない場合はもう一度プログラムの確認を行ってください。あるいはハードウェアでのミスの可能性もあります。はんだづけが失敗している箇所がないか確認してください。以下に具体的な手順を示します。

1. サンプルプログラムの内容が正しく書き込まれていることを確認する
2. プログラムボードの START/STOP トグルスイッチを奥に倒して TTM8 を動作状態にする
   クロックが手動クロックに設定しているため動きはしません。
3. プログラムボード上のリセットボタンか本体のリセットボタンを長押ししてリセットをかける
   念のためリセットをかけます。リセット信号を示す赤色の LED が点灯すればリセットがかかっている状態です。
4. 手動クロックボタンを次々に押していきプログラムを走らせる。
5. D レジスタ、C レジスタの内容を示す LED を確認して流れる光のサンプルプログラムが正しく動作している事を確認する。

11.10　クロックを自動クロックに変える

手動クロックで動作を確認できたら自動クロックで動かしてみます。クロックを変える時は一度 TTM8 の電源を落とします。今回クロックは 244Hz に設定してみましょう。以下に具体的な手順を示します。

1. 手動クロックでプログラムが正しく動作していることを確認する
2. 電源を落とすため電源スイッチを押す
3. クロックを高周波クロックの 244Hz に設定する。手順は項目 7.2 を参照。
4. 電源スイッチを押して再び電源を入れる。
5. 自動的に TTM8 が動作し流れる光のプログラムが動作している事を確認する。

11.11　TTM8 を楽しむ

ここまで動作できたらまずは安心して大丈夫だと思います。お疲れ様でした。プログラムを自由に組んでTTM8 をお楽しみください。

改訂履歴

ver1.0	初版
ver1.1	アドレスマップ追加
ver1.2	全体文言修正、誤記修正

著作権表示

発行: みやこ電子工房
著者: 電子ライダー
twitter:@denshirider
email:denshirider@gmail.com
Copyright©2023 電子ライダー

みやこ電子工房
MIYAKO DENSHI KOBO