
BitTradeOne
\持っている人もこれからの人も!/
Shige zone
TTM ミートアップ
開発者による技術解説
2023.04.22
浅草橋 技研ベース
参加無料
配信アリ
Powered by Shigezone
詳細・参加は
Conpass で!

今日の目標

・TTM8の構成を知る
・拡張IOをつくれるようになる

製品

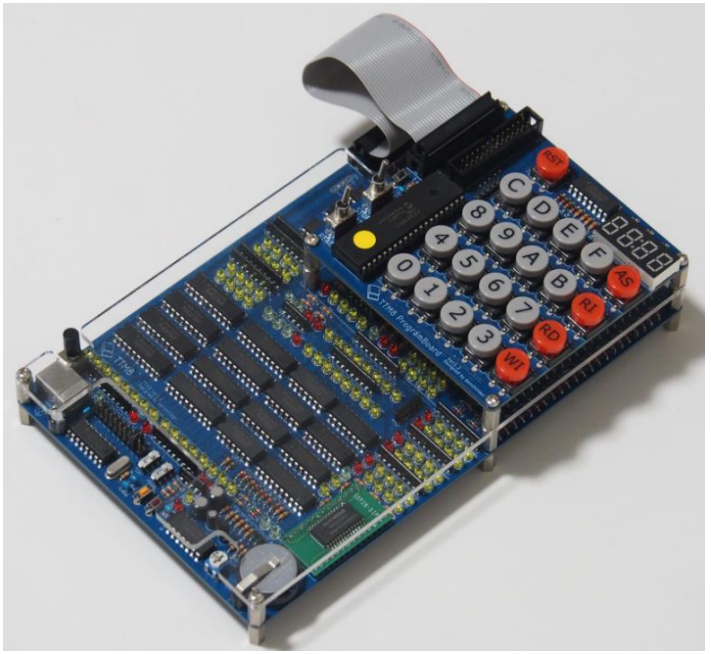
みやこ電子工房
MIYAKO DENSHI KOBO

BTOオンラインショップ、Shigezone、他電子工作ショップにて発売中

組み立て

みやこ電子工房
MIYAKO DENSHI KOBO

ひたすらはんだづけ！



IC数：44個
LED数：118個
はんだづけ時間：5～6時間

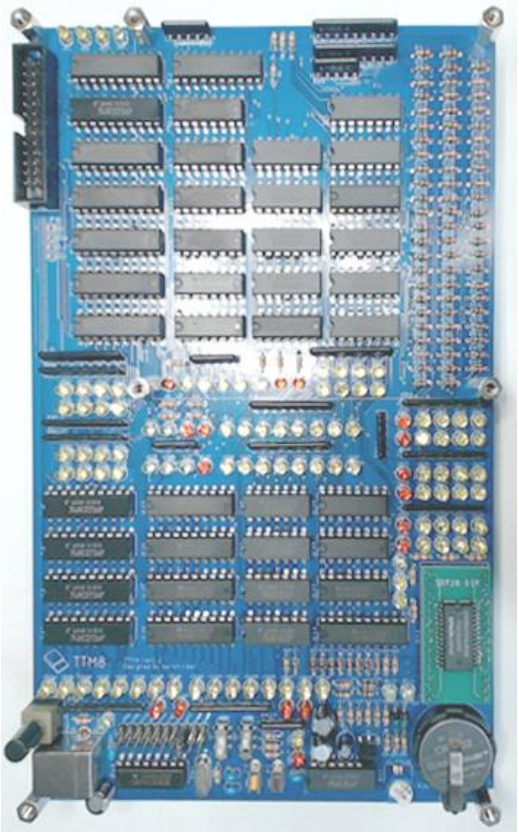
スペック表

みやこ電子工房
MIYAKO DENSHI KOBO

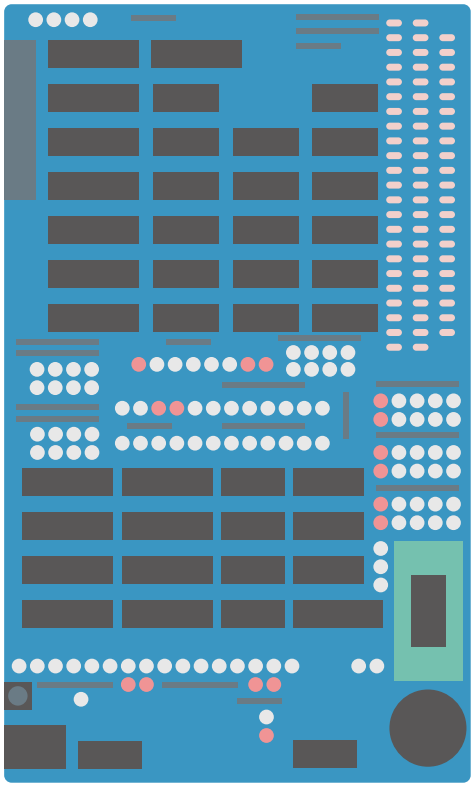
電源電圧	5V
消費電流	1A以下
bit数	8bit
命令種類	16種類
総レジスタ数	10個
汎用レジスタ数	2個
1プログラムにおける最大命令数	64命令
スタックおよびユーザ操作用RAM領域	32Byte
クロック	段階的に250kHzから244Hz、低周波クロック、手動クロック
入出力	なし(拡張して任意のI/Oを付与できる)

構造の図式化

みやこ電子工房
MIYAKO DENSHI KOBO



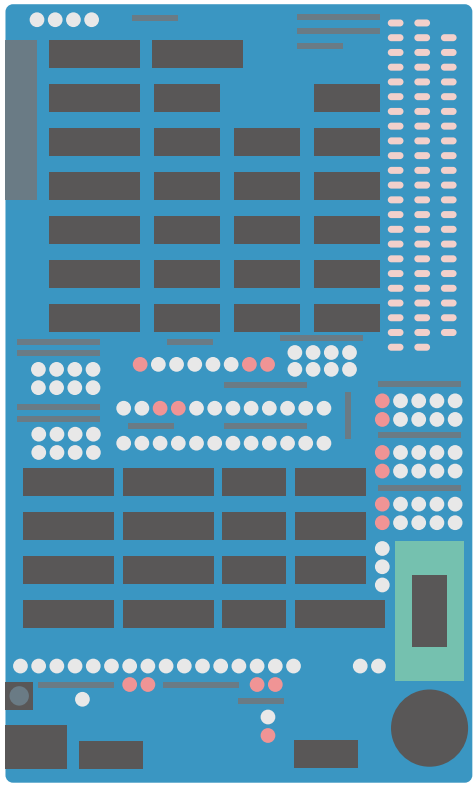
写真



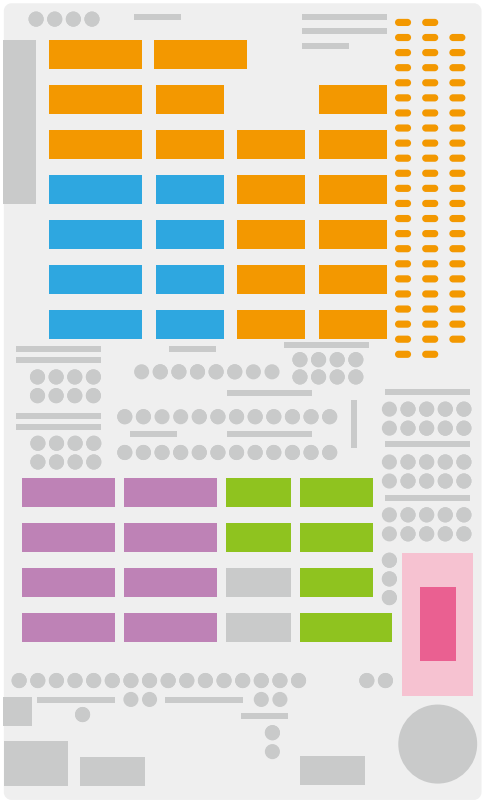
だいたい図にする

構造の図式化

みやこ電子工房
MIYAKO DENSHI KOBO

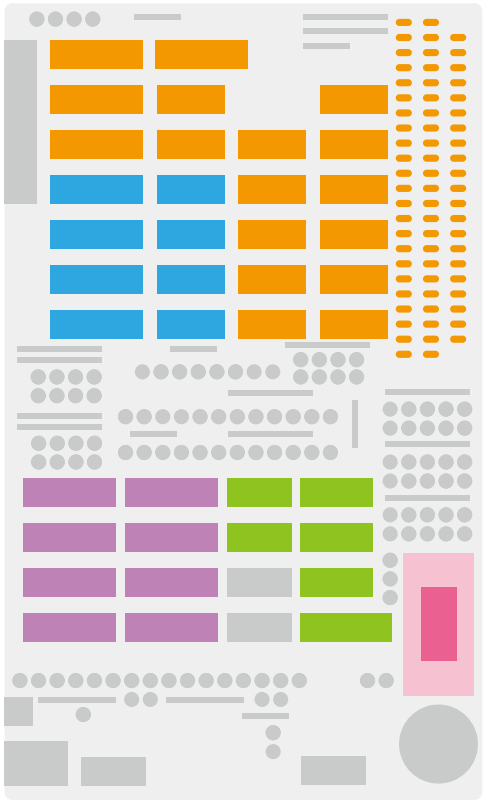


だいたい図にする

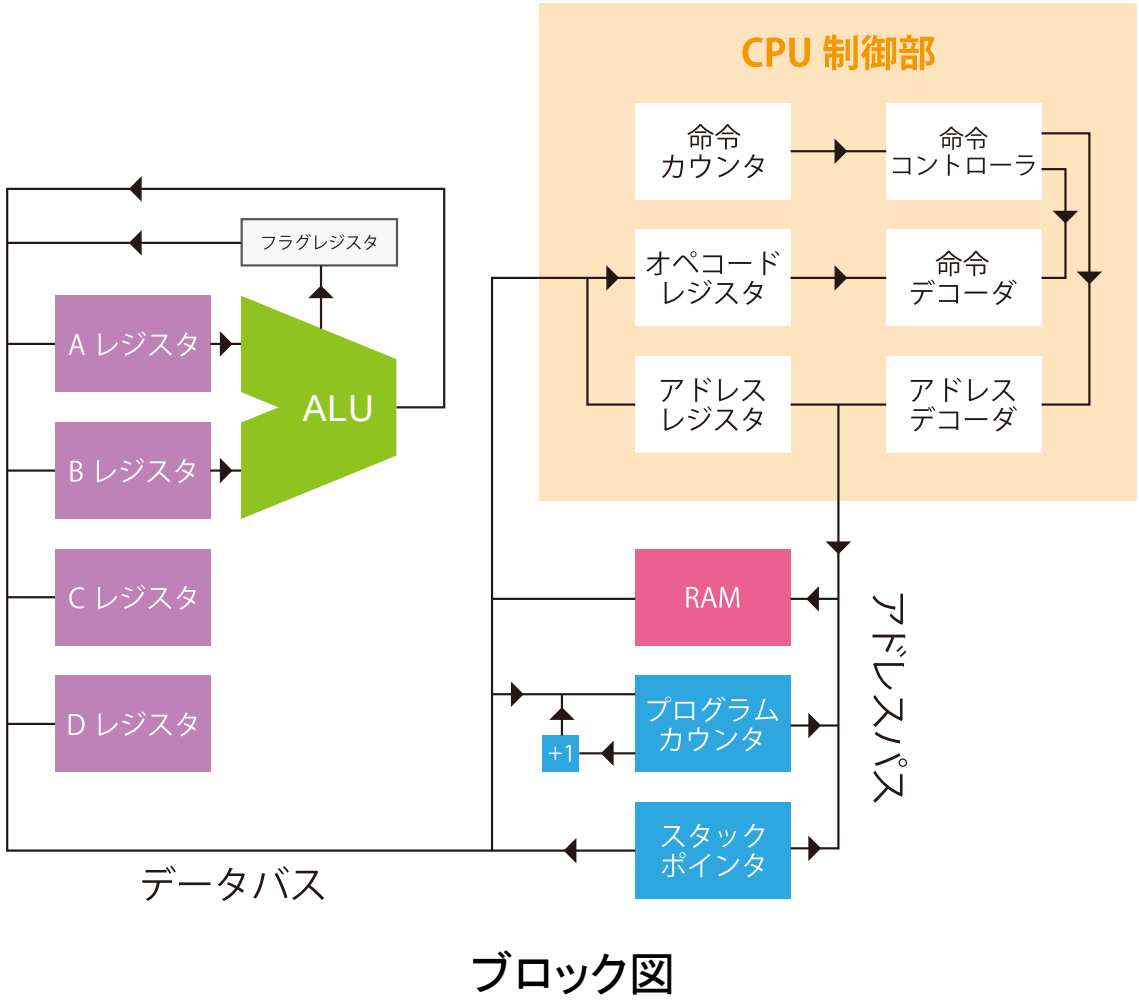


役割で色分け

構造の図式化



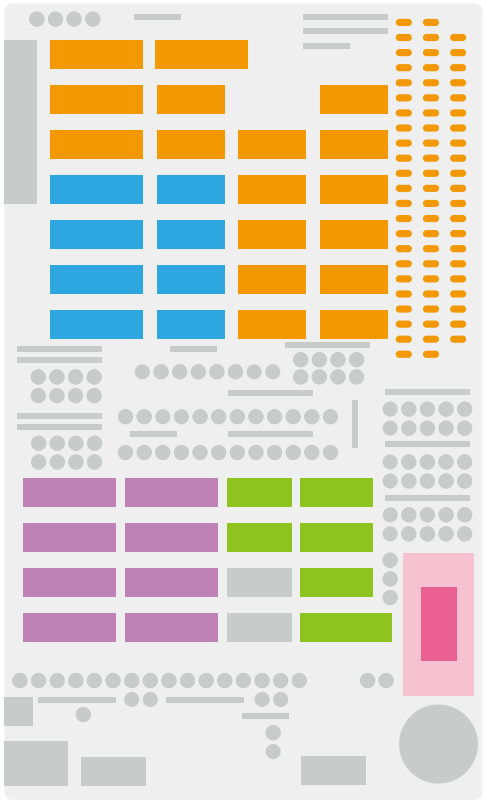
役割で色分け

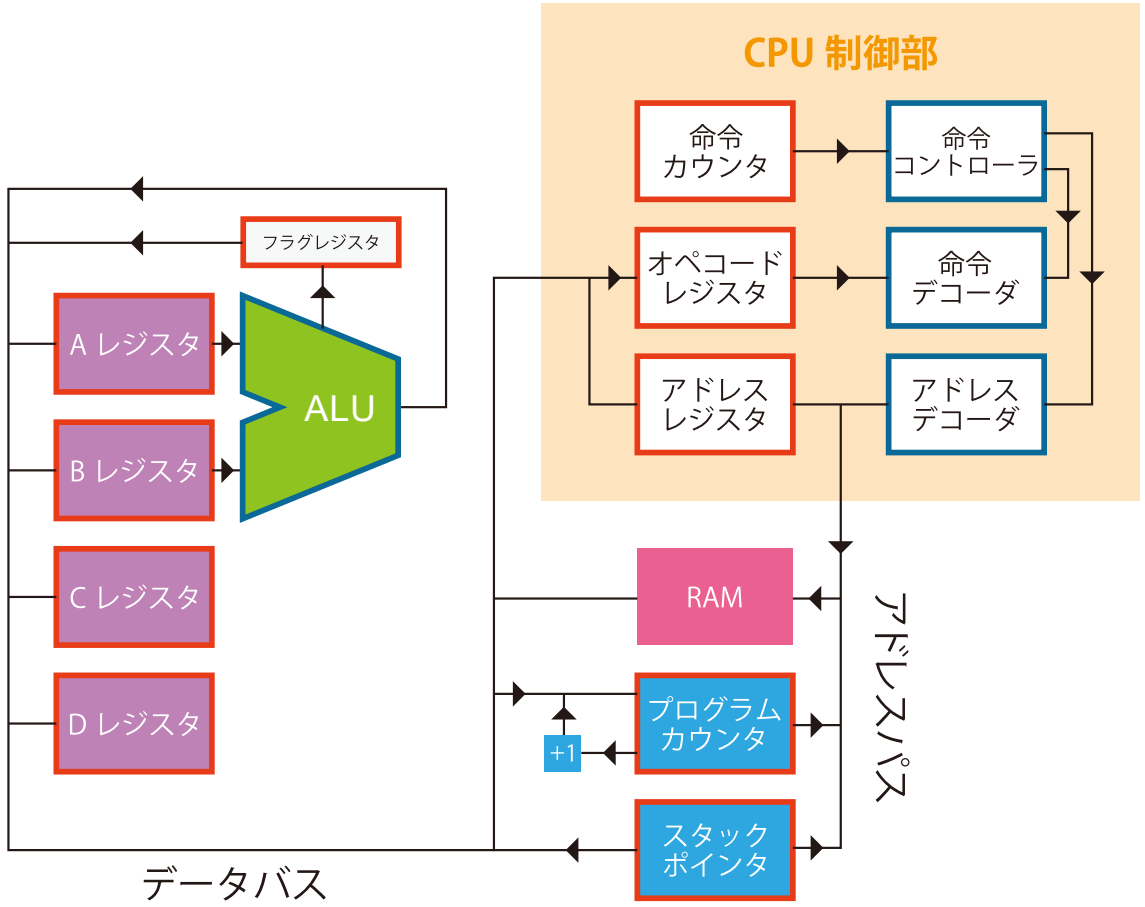


ブロック図

CPUは 組み合わせ回路 と 順序回路

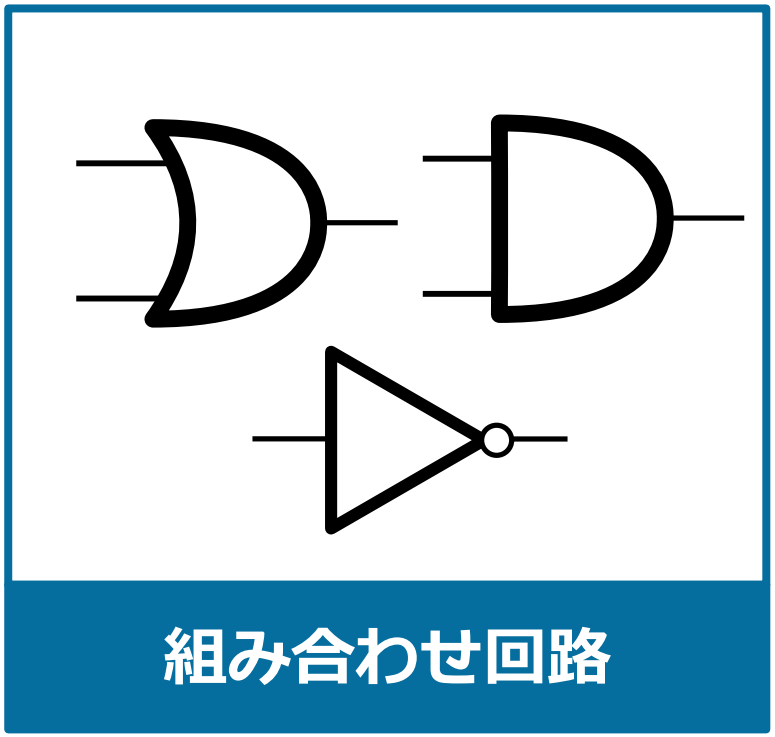
みやこ電子工房
MIYAKO DENSHI KOBO





CPUは 組み合わせ回路 と 順序回路

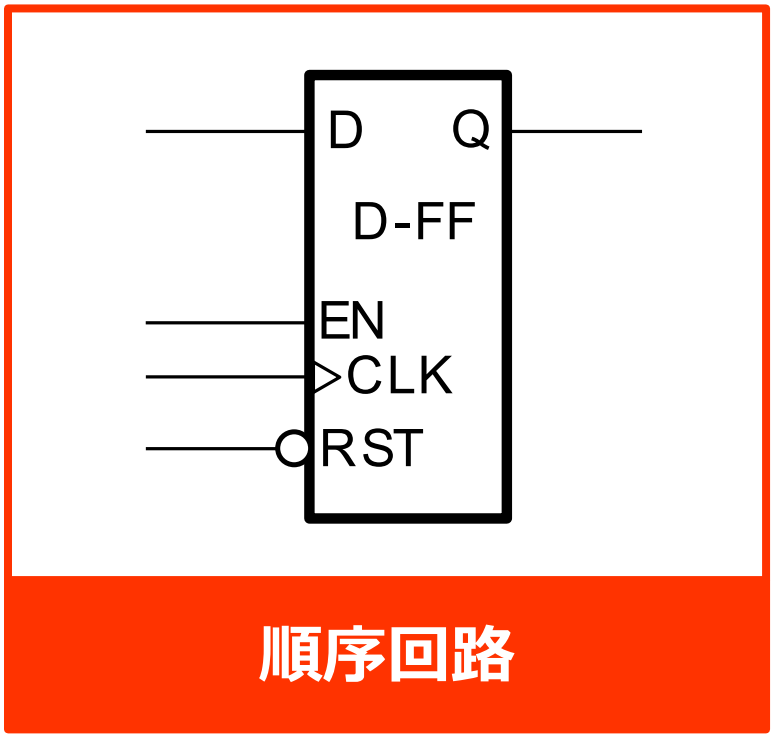
みやこ電子工房 MIYAKO DENSHI KOBO



組み合わせ回路

現在の入力によって
出力が決まる回路

ALU、デコーダ



順序回路

現在・過去の入力によって
出力が決まる回路

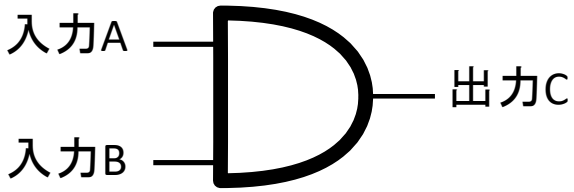
レジスタ

CPUは **組み合わせ回路** と 順序回路

みやこ電子工房 MIYAKO DENSHI KOBO

入力に対して決まった変換を出力する回路

AND回路



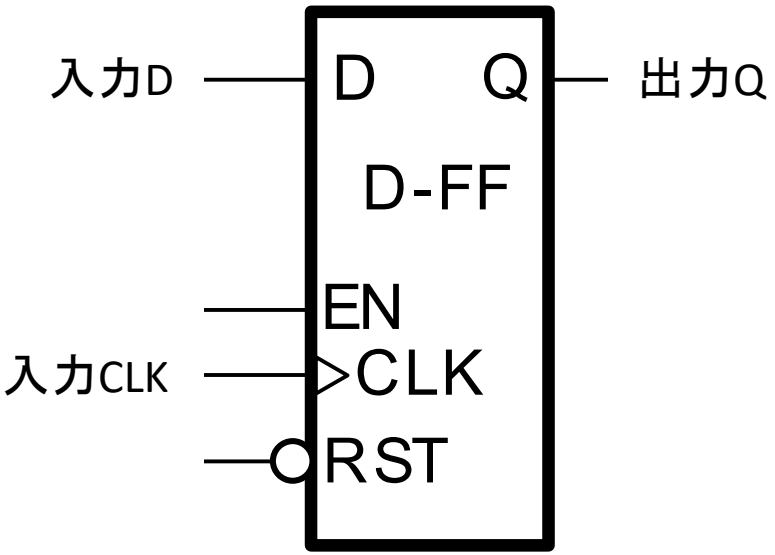
AND条件に当てはまる
＝入力が両方とも「1」　→出力「1」

そのほかの場合　→出力「0」

入力A	入力B	出力C
0	0	0
0	1	0
1	0	0
1	1	1

前の入力によって結果が変わる回路

D-FF回路



入力D	入力 CLK	出力Q
0	↑	0
1	↑	1
X	0	Q_n
X	1	Q_n

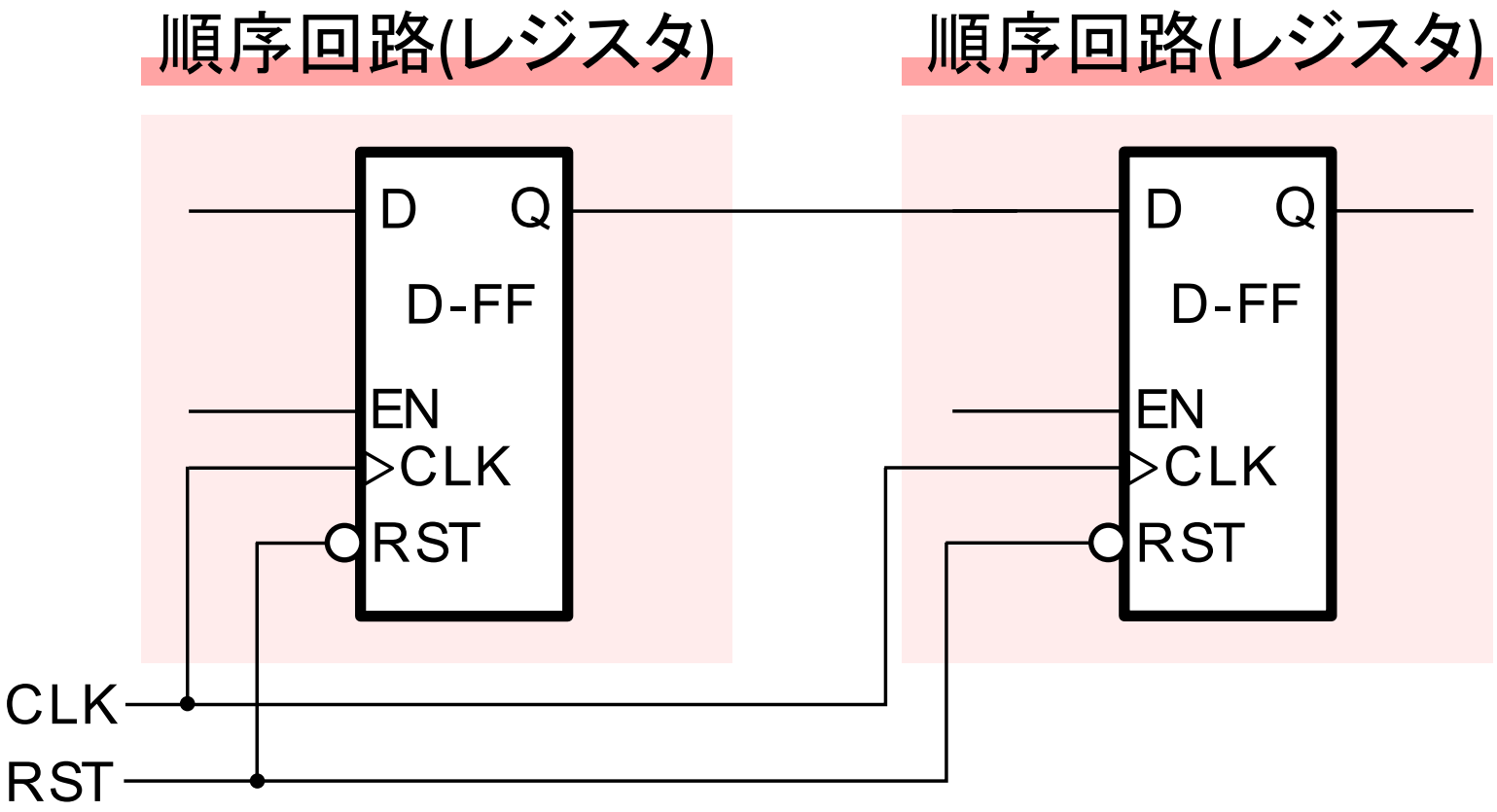
Q_n：前の入力D

※簡単のため真理値表省略

CPUは 組み合わせ回路 と 順序回路

みやこ電子工房 MIYAKO DENSHI KOBO

基本は　順序回路→順序回路　のコピー

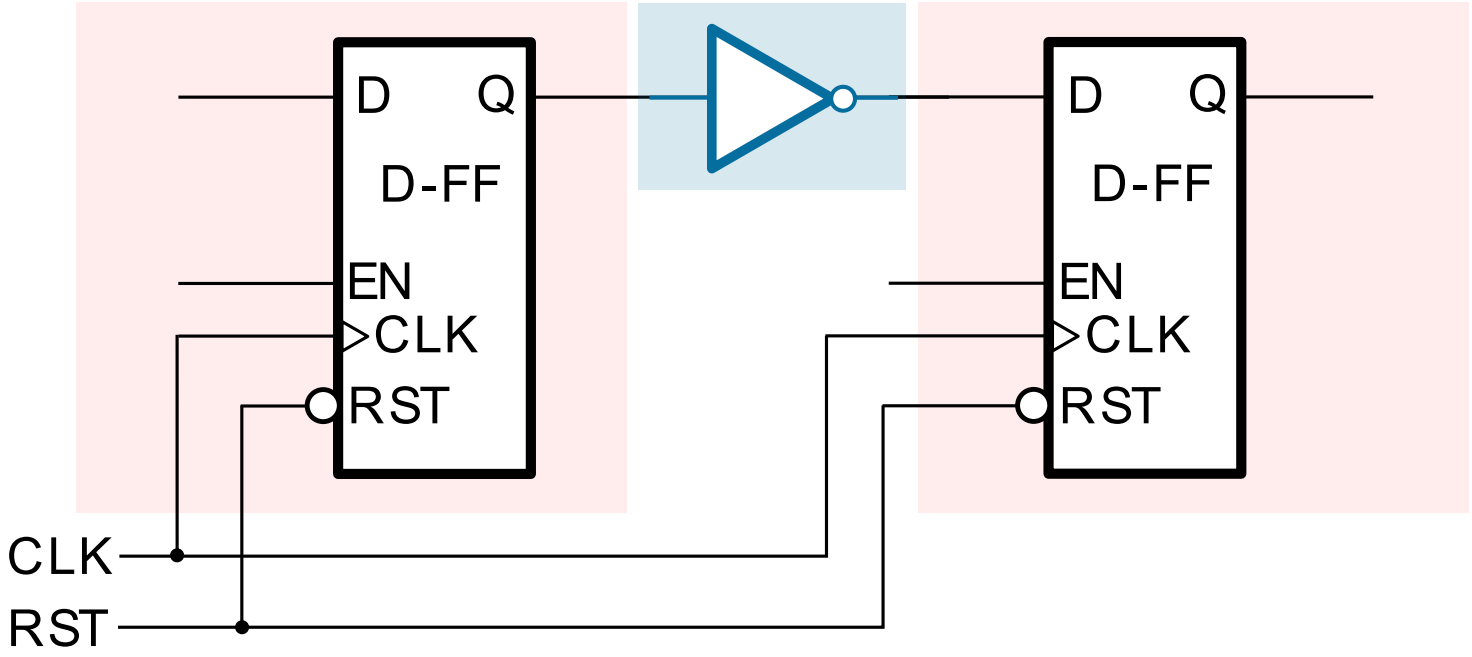


CPUは **組み合わせ回路** と **順序回路**

みやこ電子工房 MIYAKO DENSHI KOBO

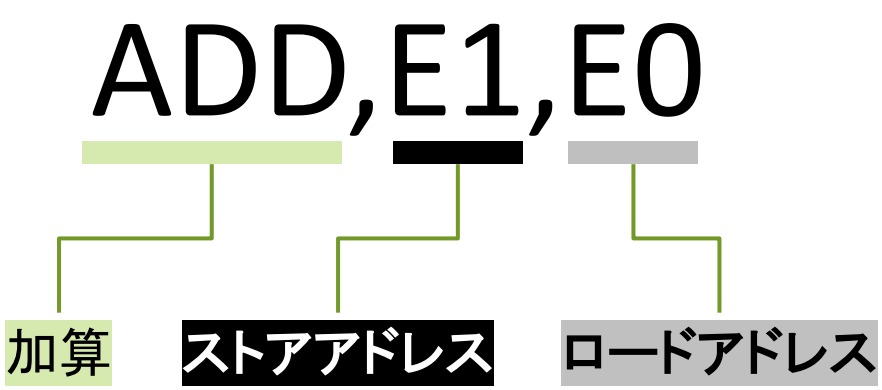
順序回路　→　組み合わせ回路　→　順序回路
にすればデータを変換してコピーできる

順序回路(レジスタ)　組み合わせ回路　順序回路(レジスタ)



命令の構成　ADD命令を例として

※TTM8独自の命令記法です



ロードアドレスが示す先の内容＋ストアアドレスが示す先の内容

→ストアアドレスが示す先に格納する

アドレスマップ

みやこ電子工房
MIYAKO DENSHI KOBO

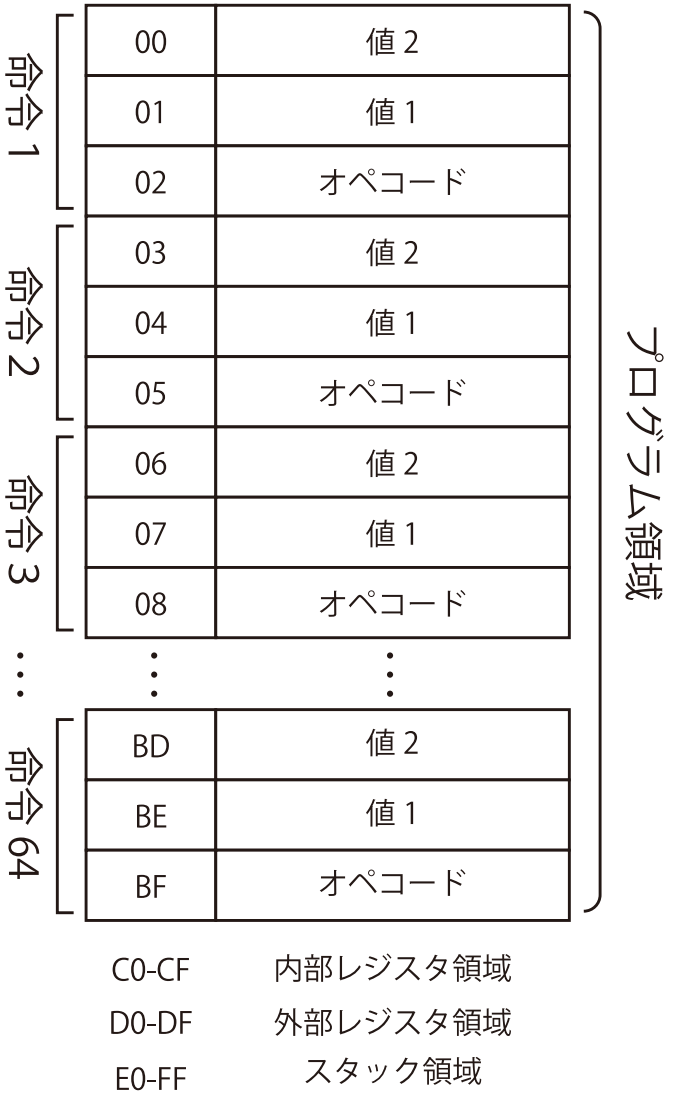
ADD,E1,E0

オペコード,値1,値2

TTM8のアドレスは00～FF
(16進数)

00～BF:プログラム領域
命令が64個まで格納できる。

3アドレス使って1命令。
値2→値1→オペコードの順に格納していく。



簡単なADD命令を見てみる

みやこ電子工房
MIYAKO DENSHI KOBO

①ADD命令

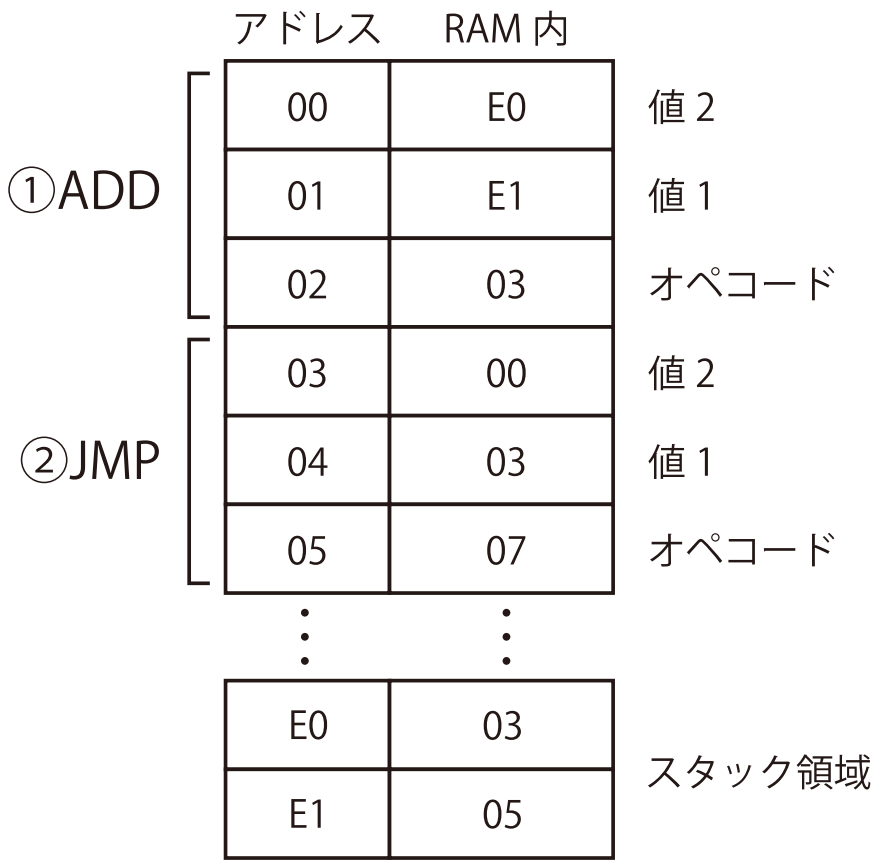
ADD,E1,E0

②JMP命令

JMP,03,--

①E1（05）にE0（03）を加算、E1に

②JMP命令-JMP命令のループ



簡単なADD命令を見てみる

みやこ電子工房
MIYAKO DENSHI KOBO

①ADD命令

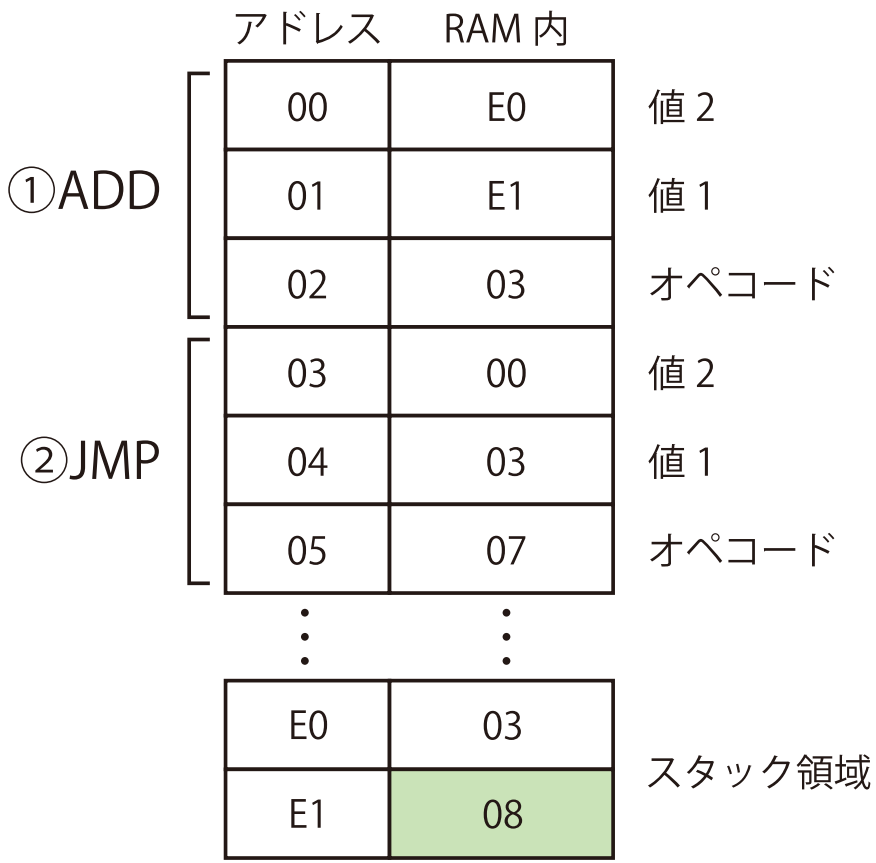
ADD,E1,E0

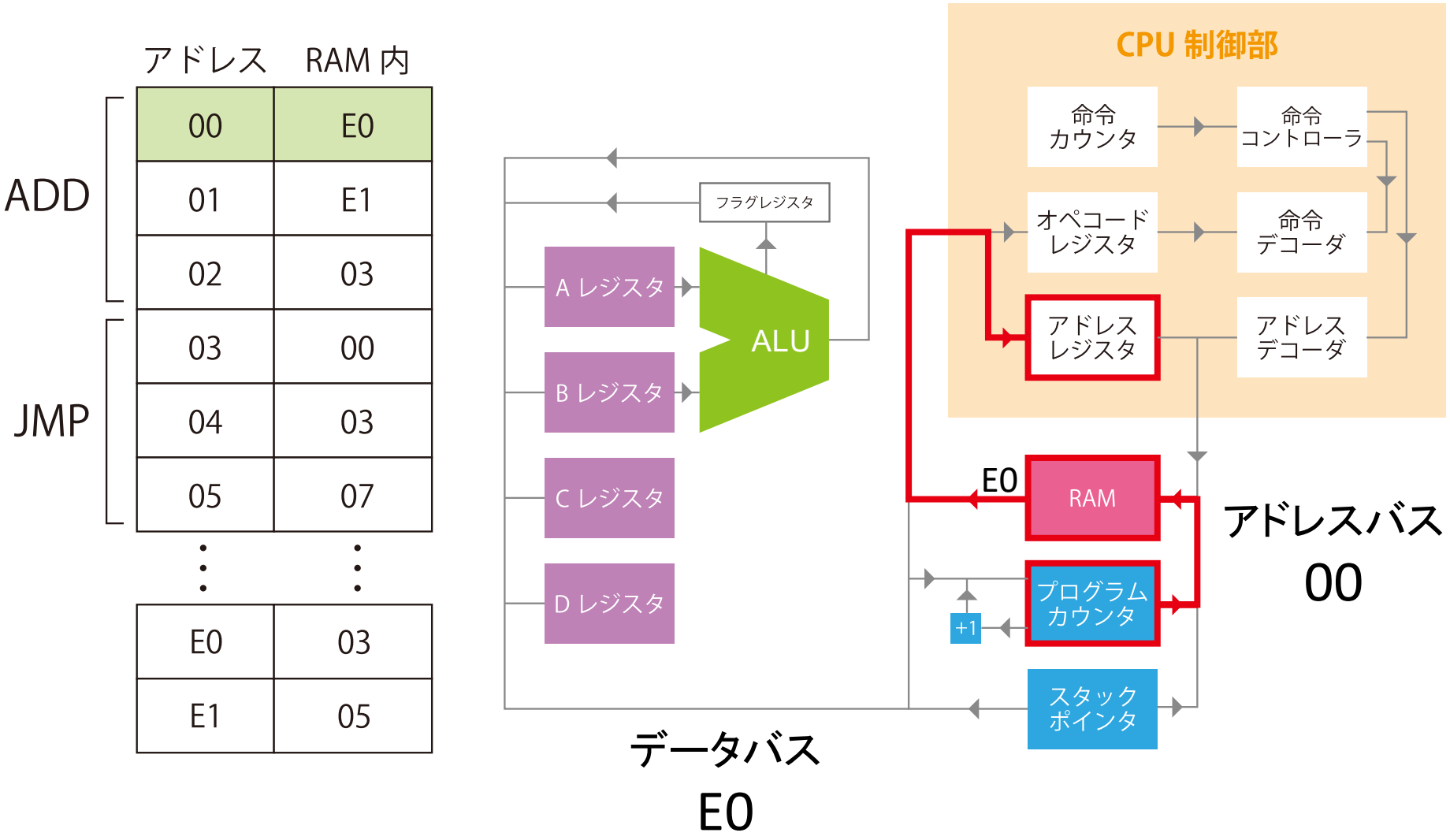
②JMP命令

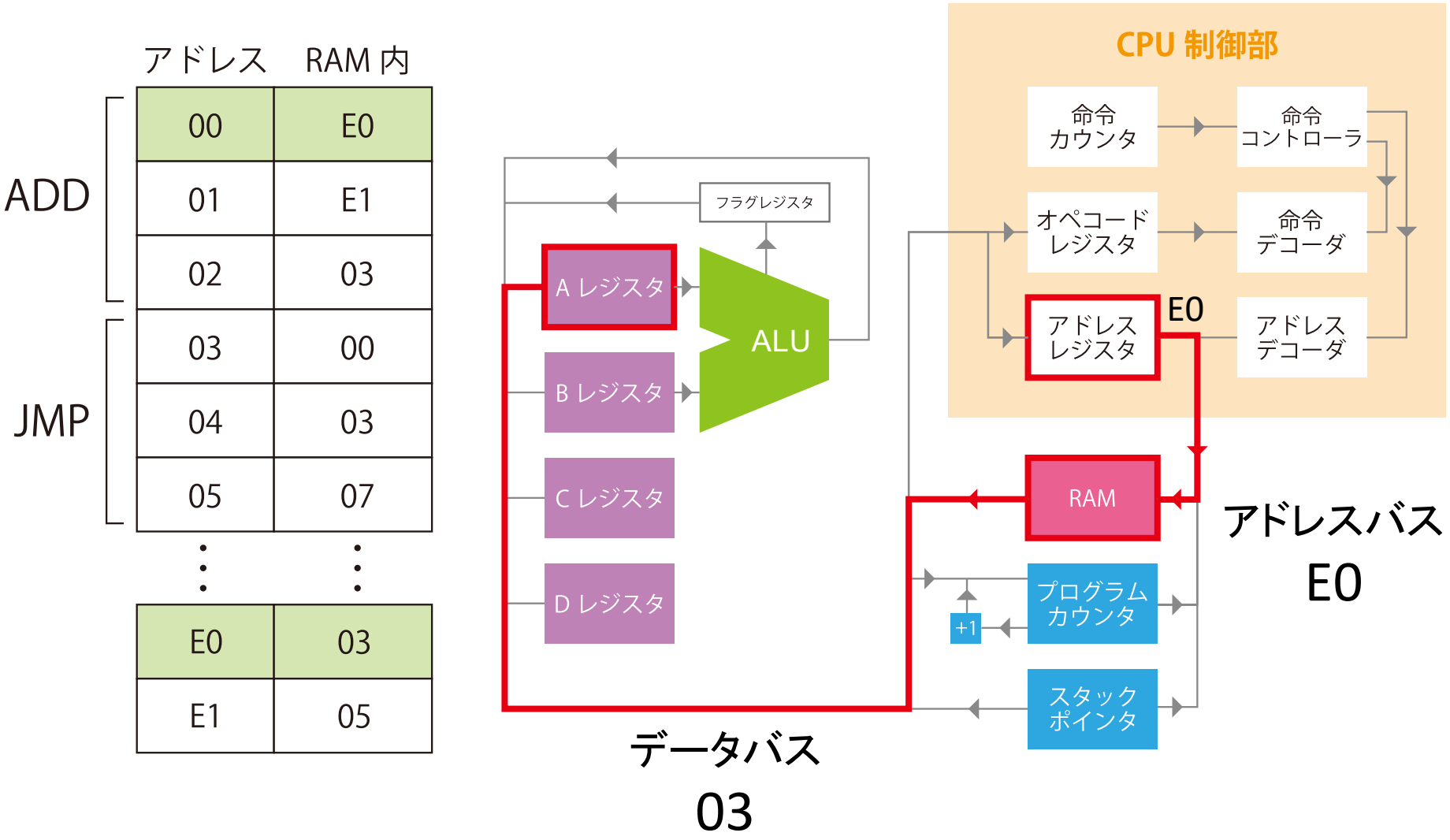
JMP,03,--

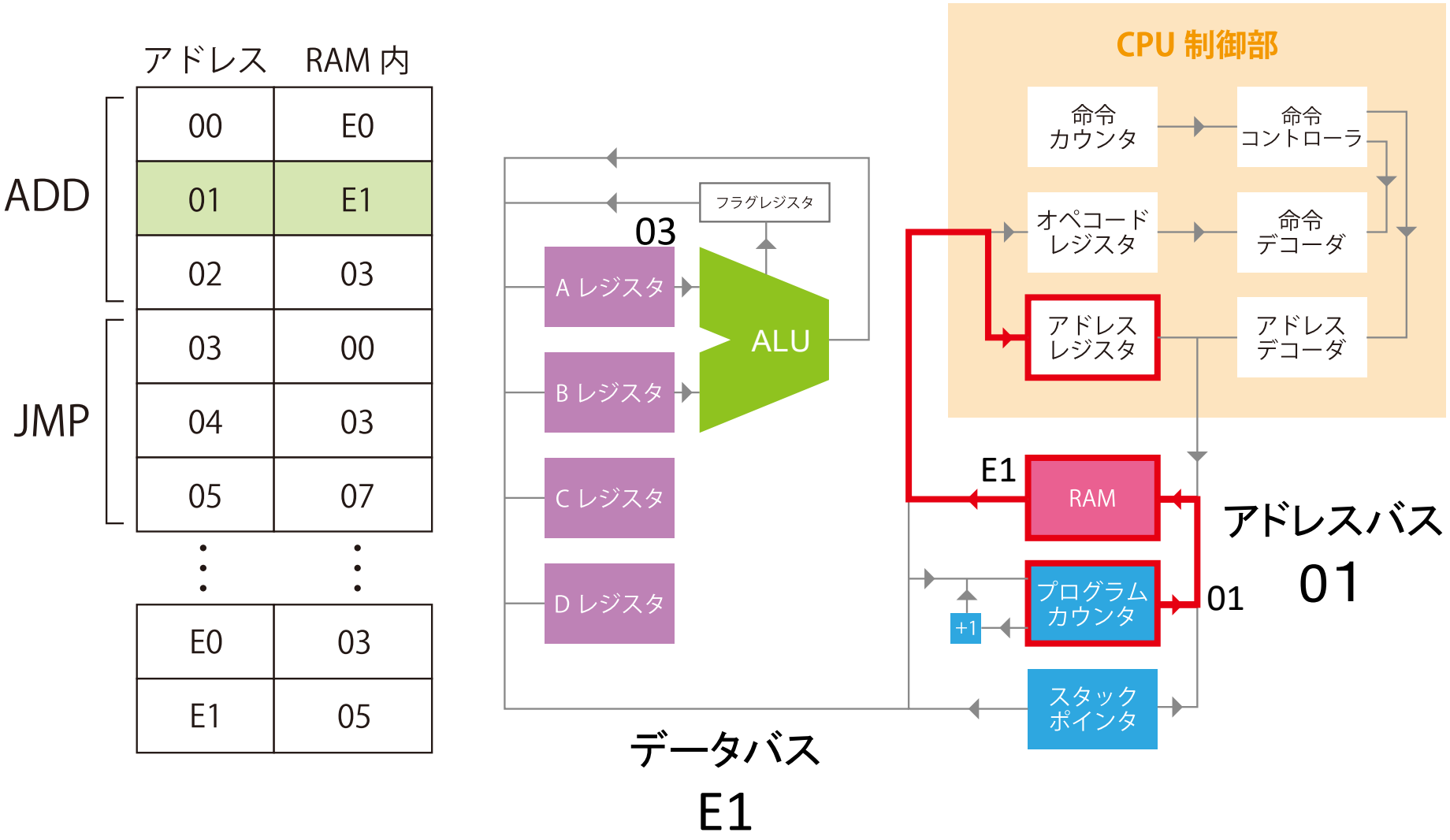
①E1（05）にE0（03）を加算、E1に
→E1が08になった

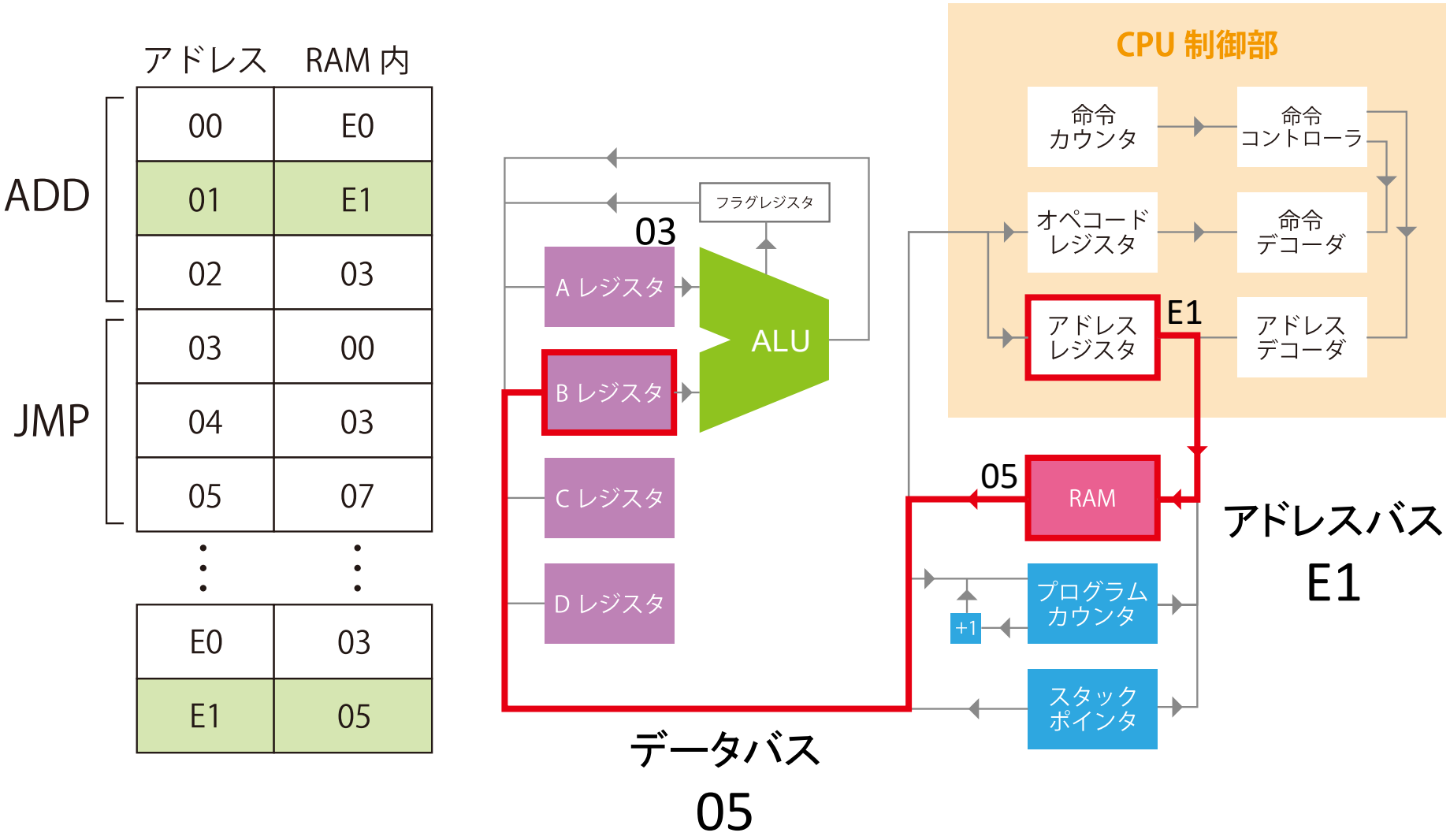
②JMP命令-JMP命令のループ
→プログラムが停止した

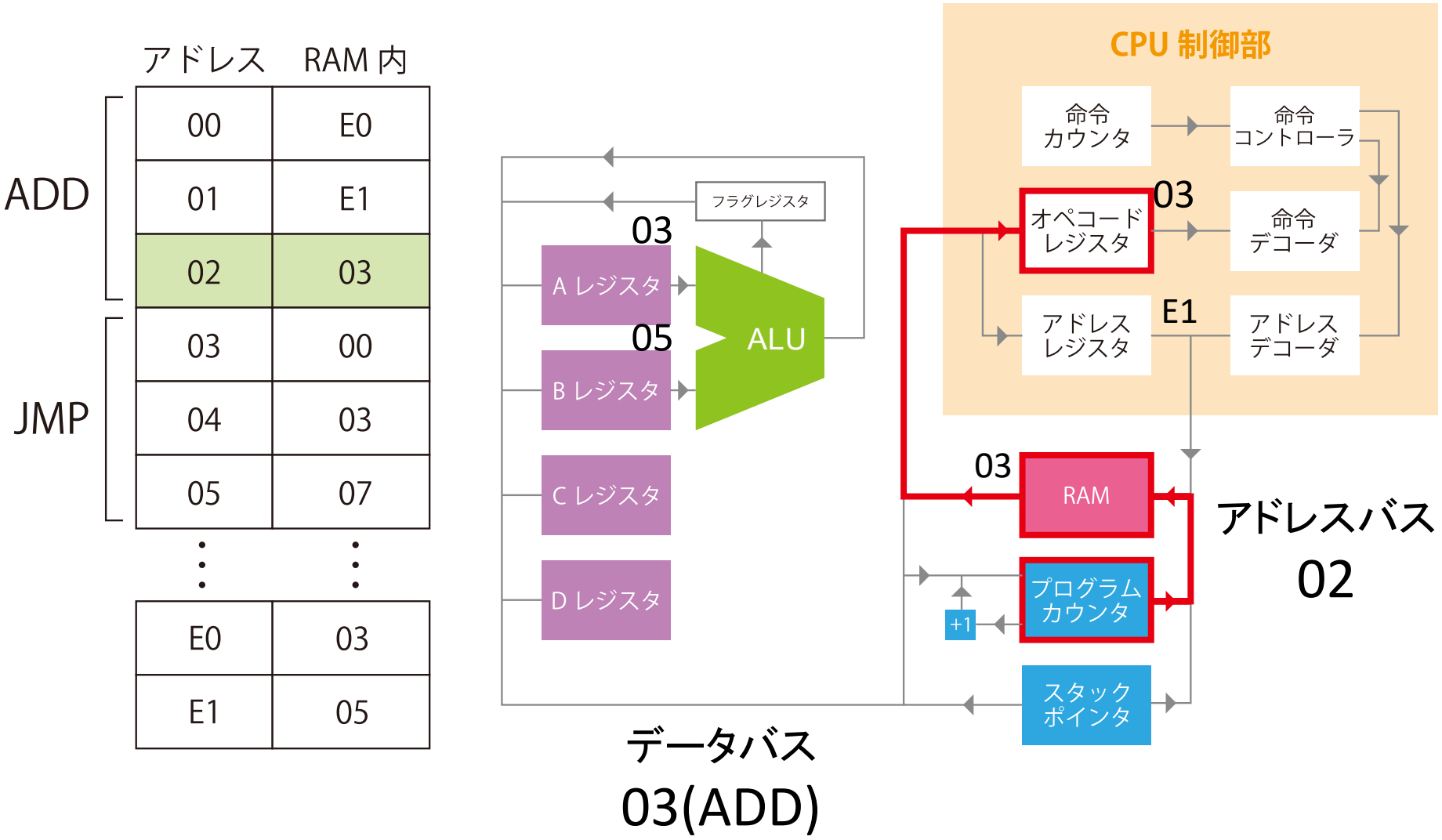


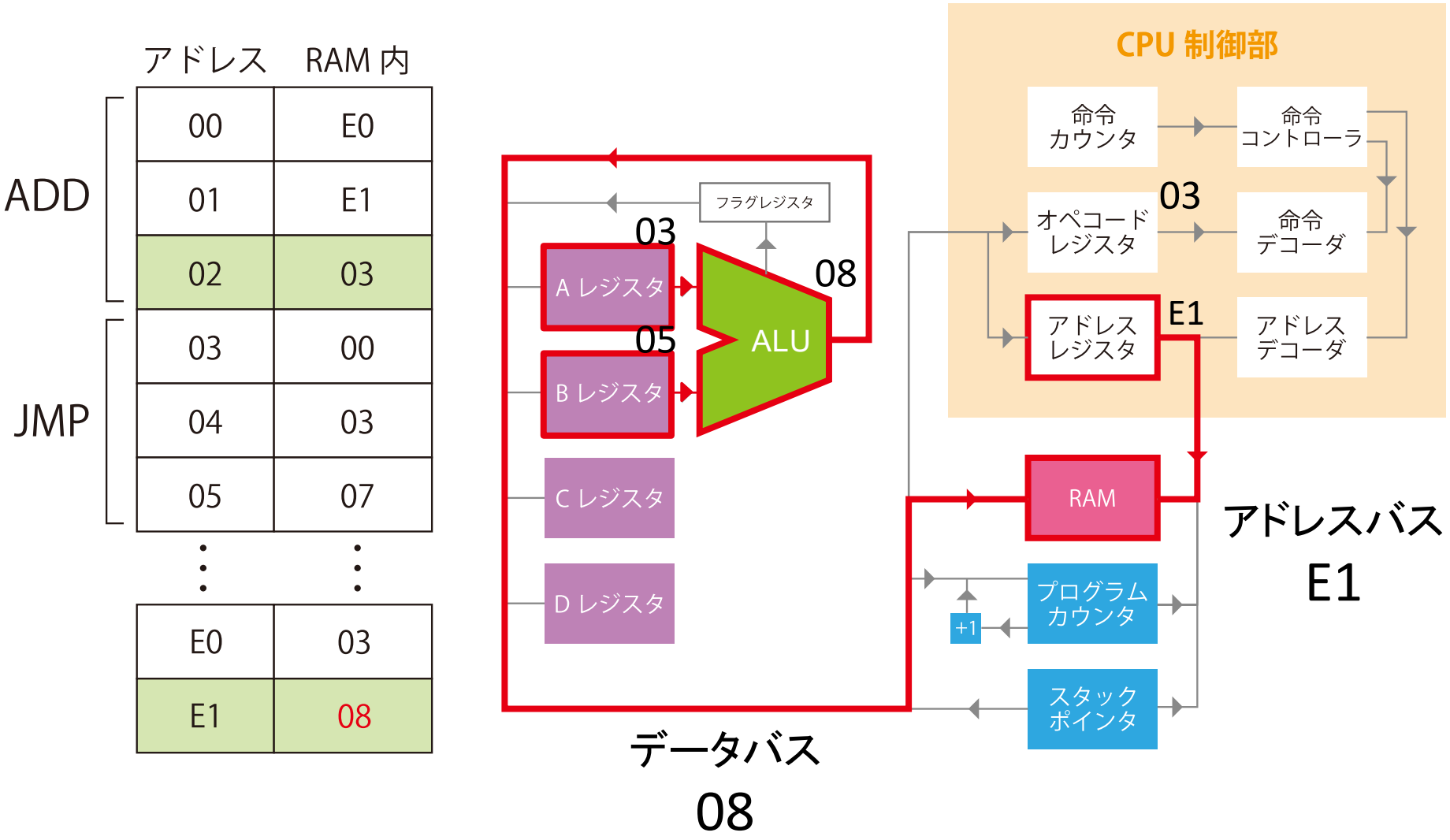

アドレス
RAM 内
ADD
JMP
00
E0
01
E1
02
03
03
00
04
03
05
07
E0
03
E1
05
CPU 制御部
命令
カウンタ
命令
コントローラ
オペコード
レジスタ
命令
デコーダ
アドレス
レジスタ
アドレス
デコーダ
フラグレジスタ
A レジスタ
B レジスタ
C レジスタ
D レジスタ
ALU
E0
RAM
アドレスバス
00
プログラム
カウンタ
+1
スタック
ポインタ
データバス
E0

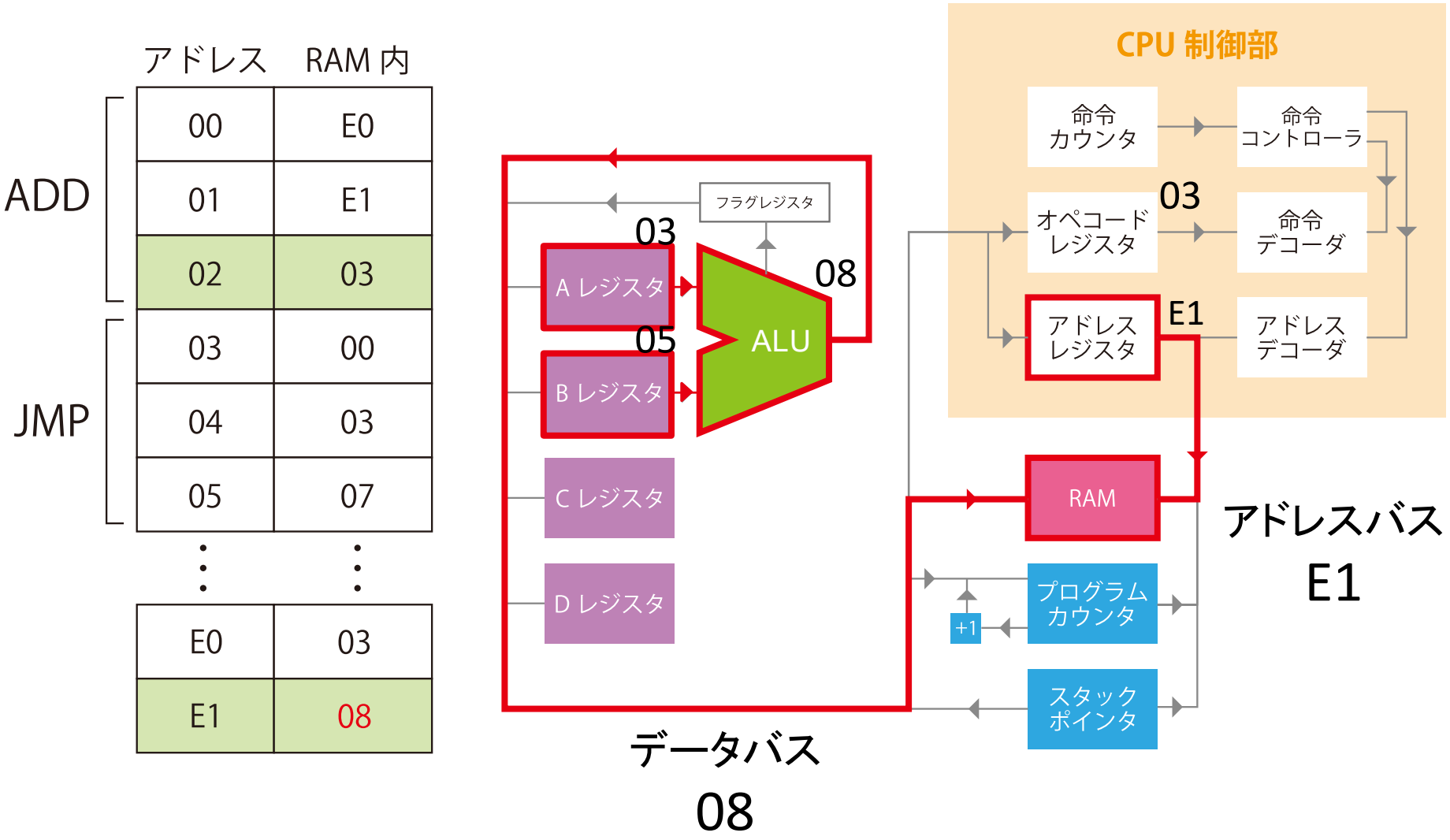

アドレス
RAM 内
ADD
JMP
00
E0
01
E1
02
03
03
00
04
03
05
07
E0
03
E1
05
CPU 制御部
命令
カウンタ
命令
コントローラ
オペコード
レジスタ
命令
デコーダ
アドレス
レジスタ
E0
アドレス
デコーダ
フラグレジスタ
A レジスタ
ALU
B レジスタ
C レジスタ
D レジスタ
RAM
プログラム
カウンタ
+1
スタック
ポインタ
アドレスバス
E0
データバス
03


アドレス
RAM 内
ADD
JMP
00
E0
01
E1
02
03
03
00
04
03
05
07
E0
03
E1
05
CPU 制御部
命令
カウンタ
命令
コントローラ
オペコード
レジスタ
命令
デコーダ
アドレス
レジスタ
アドレス
デコーダ
フラグレジスタ
03
A レジスタ
B レジスタ
C レジスタ
D レジスタ
ALU
E1
RAM
アドレスバス
01
プログラム
カウンタ
01
+1
スタック
ポインタ
データバス
E1


アドレス
RAM 内
ADD
JMP
00
E0
01
E1
02
03
03
00
04
03
05
07
E0
03
E1
05
CPU 制御部
命令
カウンタ
命令
コントローラ
オペコード
レジスタ
命令
デコーダ
アドレス
レジスタ
E1
アドレス
デコーダ
フラグレジスタ
03
A レジスタ
ALU
B レジスタ
C レジスタ
D レジスタ
05
RAM
プログラム
カウンタ
+1
スタック
ポインタ
アドレスバス
E1
データバス
05


アドレス
RAM 内
ADD
JMP
00
E0
01
E1
02
03
03
00
04
03
05
07
E0
03
E1
05
CPU 制御部
命令
カウンタ
命令
コントローラ
03
オペコード
レジスタ
命令
デコーダ
アドレス
レジスタ
E1
アドレス
デコーダ
フラグレジスタ
03
A レジスタ
05
B レジスタ
ALU
C レジスタ
D レジスタ
03
RAM
プログラム
カウンタ
+1
スタック
ポインタ
アドレスバス
02
データバス
03(ADD)


アドレス
RAM 内
ADD
JMP
00
E0
01
E1
02
03
03
00
04
03
05
07
E0
03
E1
08
CPU 制御部
命令
カウンタ
命令
コントローラ
03
オペコード
レジスタ
命令
デコーダ
E1
アドレス
レジスタ
アドレス
デコーダ
フラグレジスタ
03
A レジスタ
08
ALU
05
B レジスタ
C レジスタ
D レジスタ
RAM
プログラム
カウンタ
+1
スタック
ポインタ
アドレスバス
E1
データバス
08


アドレス
RAM 内
ADD
JMP
00
E0
01
E1
02
03
03
00
04
03
05
07
E0
03
E1
08
CPU 制御部
命令
カウンタ
命令
コントローラ
03
オペコード
レジスタ
命令
デコーダ
E1
アドレス
レジスタ
アドレス
デコーダ
フラグレジスタ
03
A レジスタ
08
ALU
05
B レジスタ
C レジスタ
D レジスタ
RAM
プログラム
カウンタ
+1
スタック
ポインタ
アドレスバス
E1
データバス
08

命令は

MOV
コピーする

ADD、SUB、CMP
加算、減算、比較

JMP、JC、JNC、JZ、JNZ
ジャンプ命令

PUSH、POP
スタックを操作

などなど16種類

TTM8　命令セット

命令は3つの要素で構成される。(命令、値1、値2)である。命令によっては値1や値2のみ使用するものがあるが、その場合使用しない値にはダミーを置く。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される。なおメモリには値2、値1、命令の順番で格納される。
下記の命令詳細に用いている略語は表下部に記述。←は値のコピーを示す。両辺が同じ場合更新を示す。[]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。

命令	機械語	ニーモニック	命令構成	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]

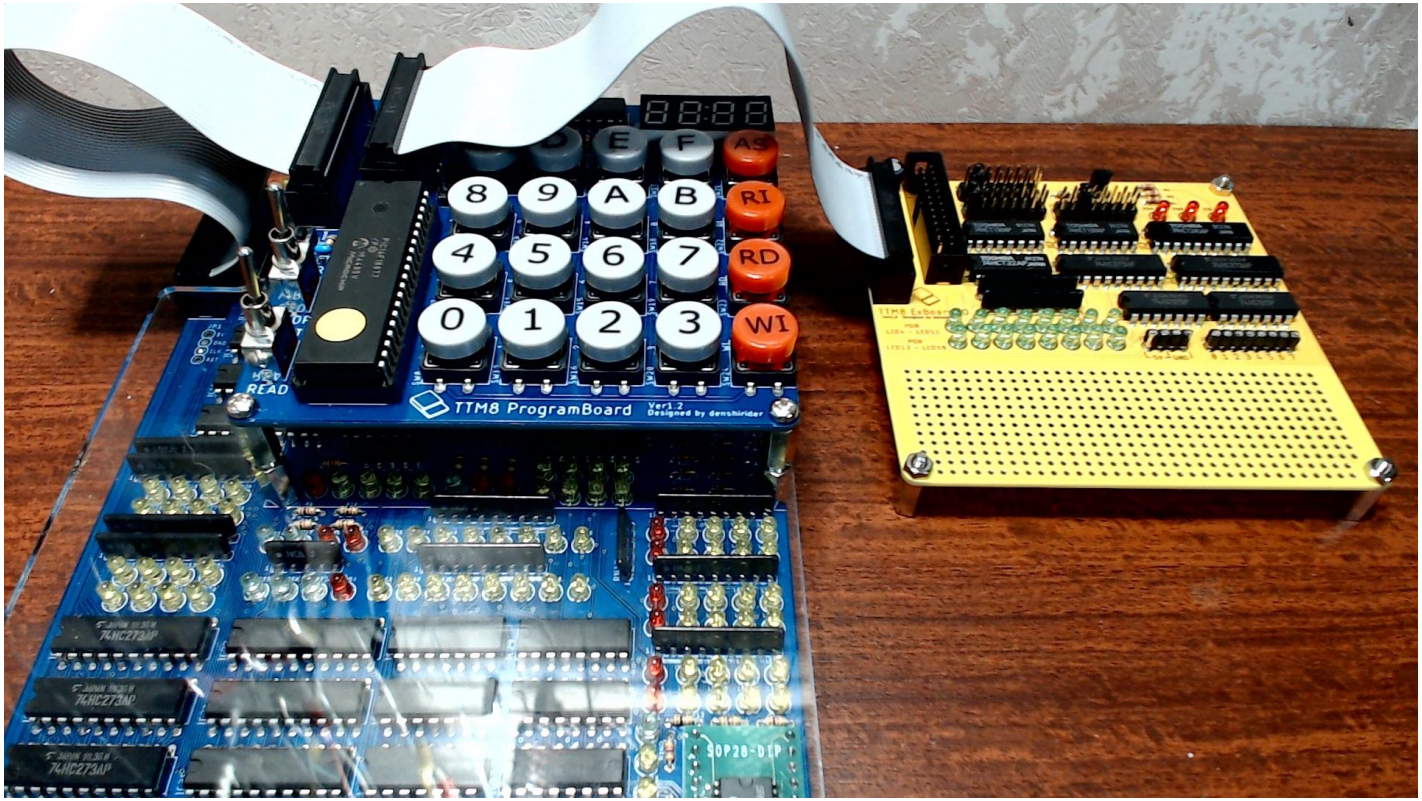
STA　ストアアドレス　LDA　ロードアドレス　IM　即時データ
C　Cレジスタ　D　Dレジスタ　FR　フラグレジスタ　PC　プログラムカウンタ　SP　スタックポインタ

7セグメント表示
と
DIPスイッチ入力

ワークショップ

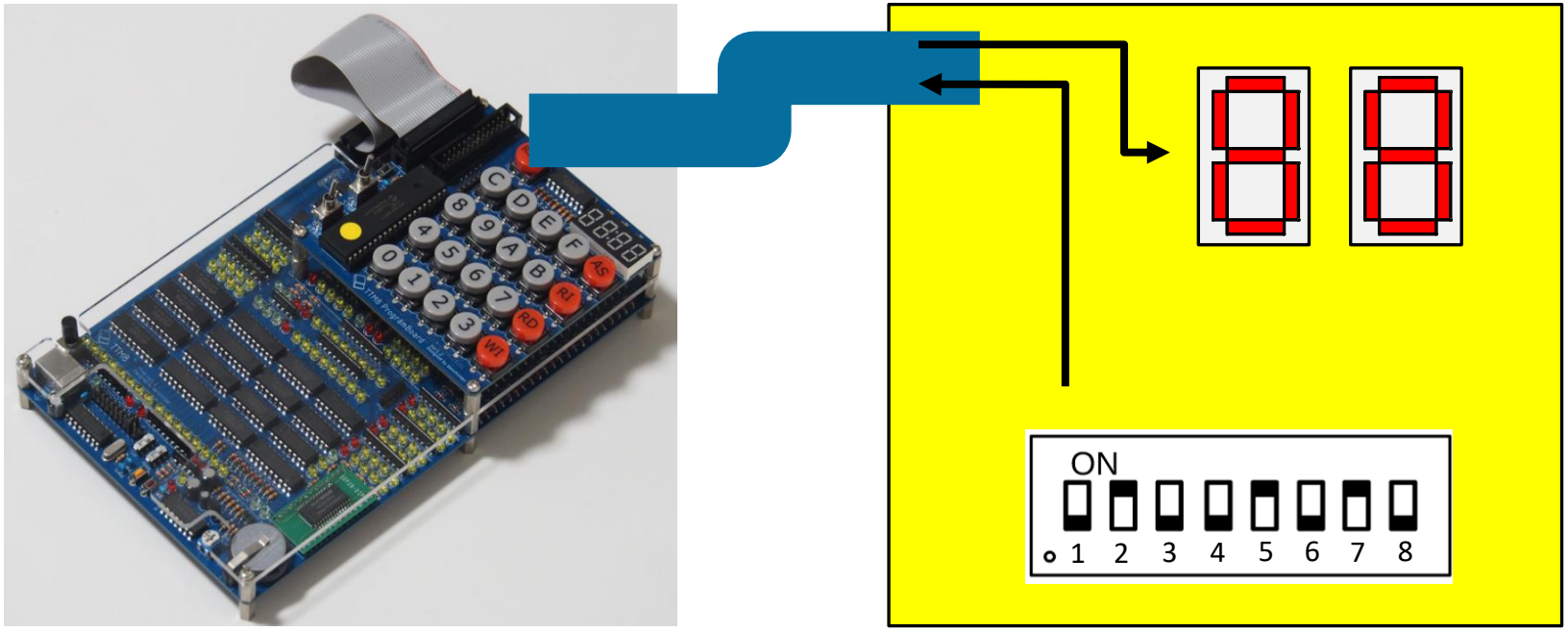
みやこ電子工房
MIYAKO DENSHI KOBO

TTM8はIOを拡張できます



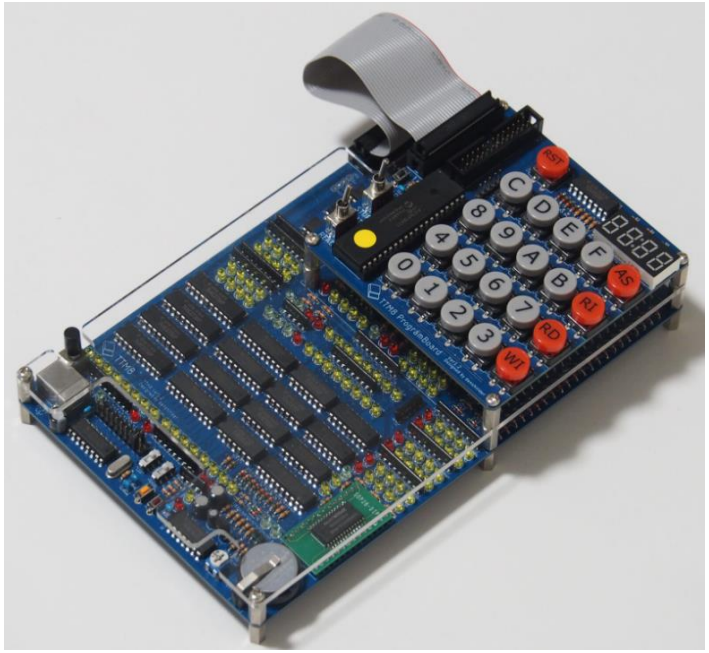
ワークショップ

みやこ電子工房
MIYAKO DENSHI KOBO

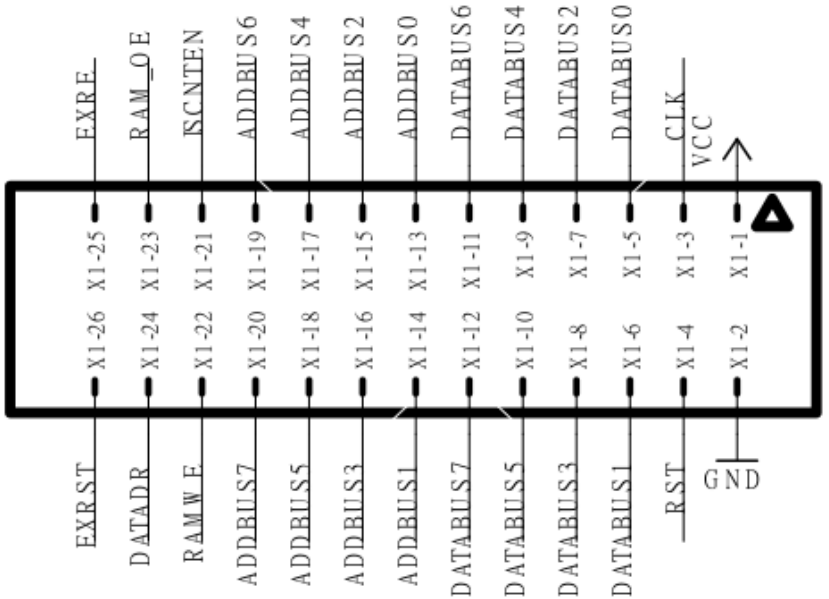


ワークショップ

みやこ電子工房
MIYAKO DENSHI KOBO

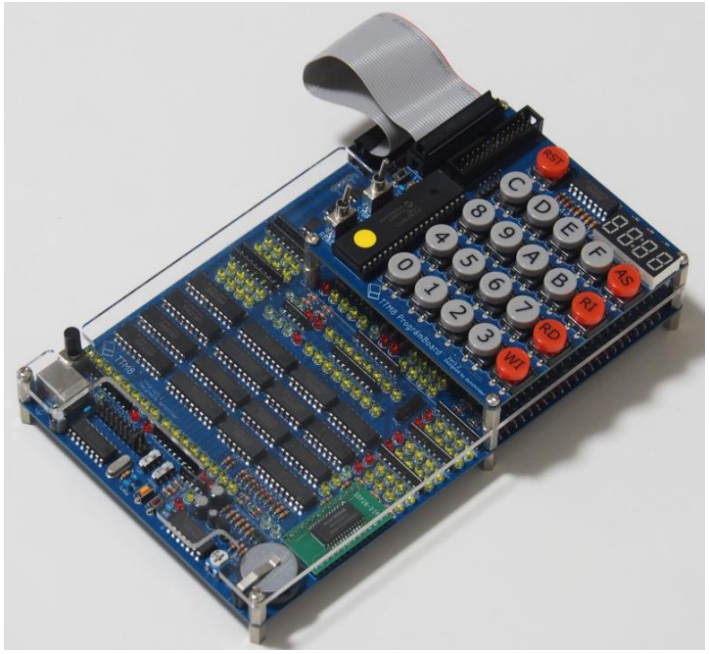


拡張ポート

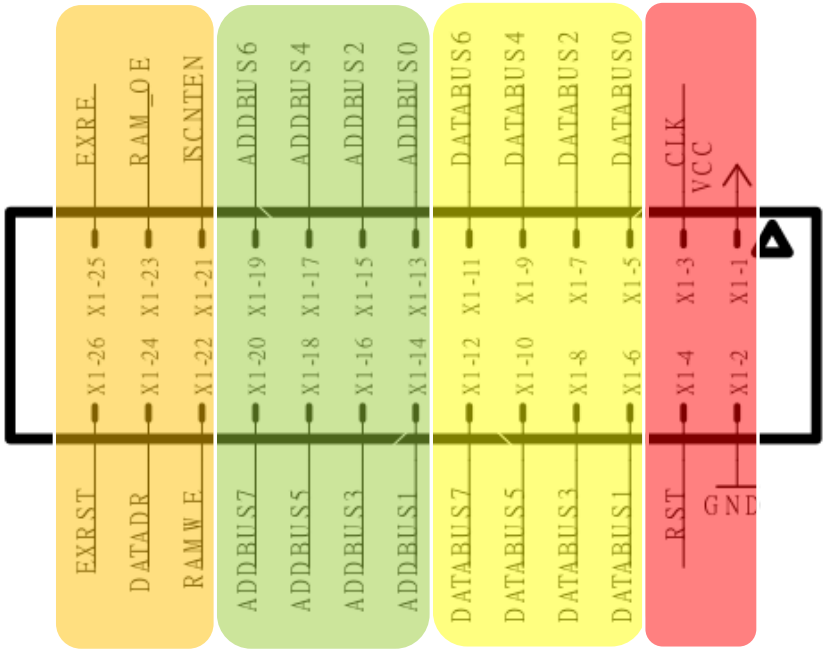


ワークショップ

みやこ電子工房
MIYAKO DENSHI KOBO



拡張ポート

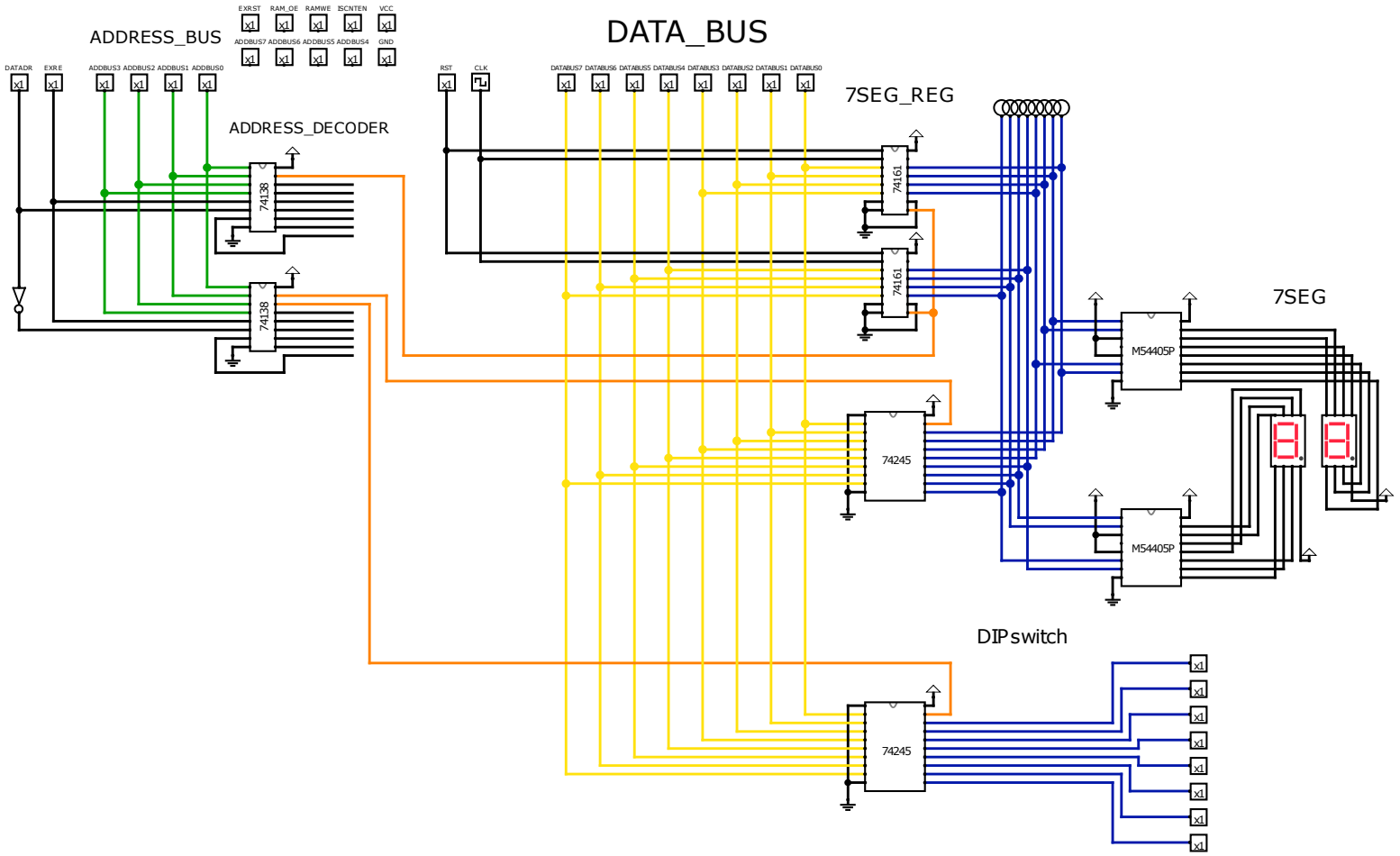


アドレスバス データバス

ワークショップ

みやこ電子工房
MIYAKO DENSHI KOBO

TTM8_EX_7seg & DIPswitch



みやこ電子工房

MIYAKO DENSHI KOBO