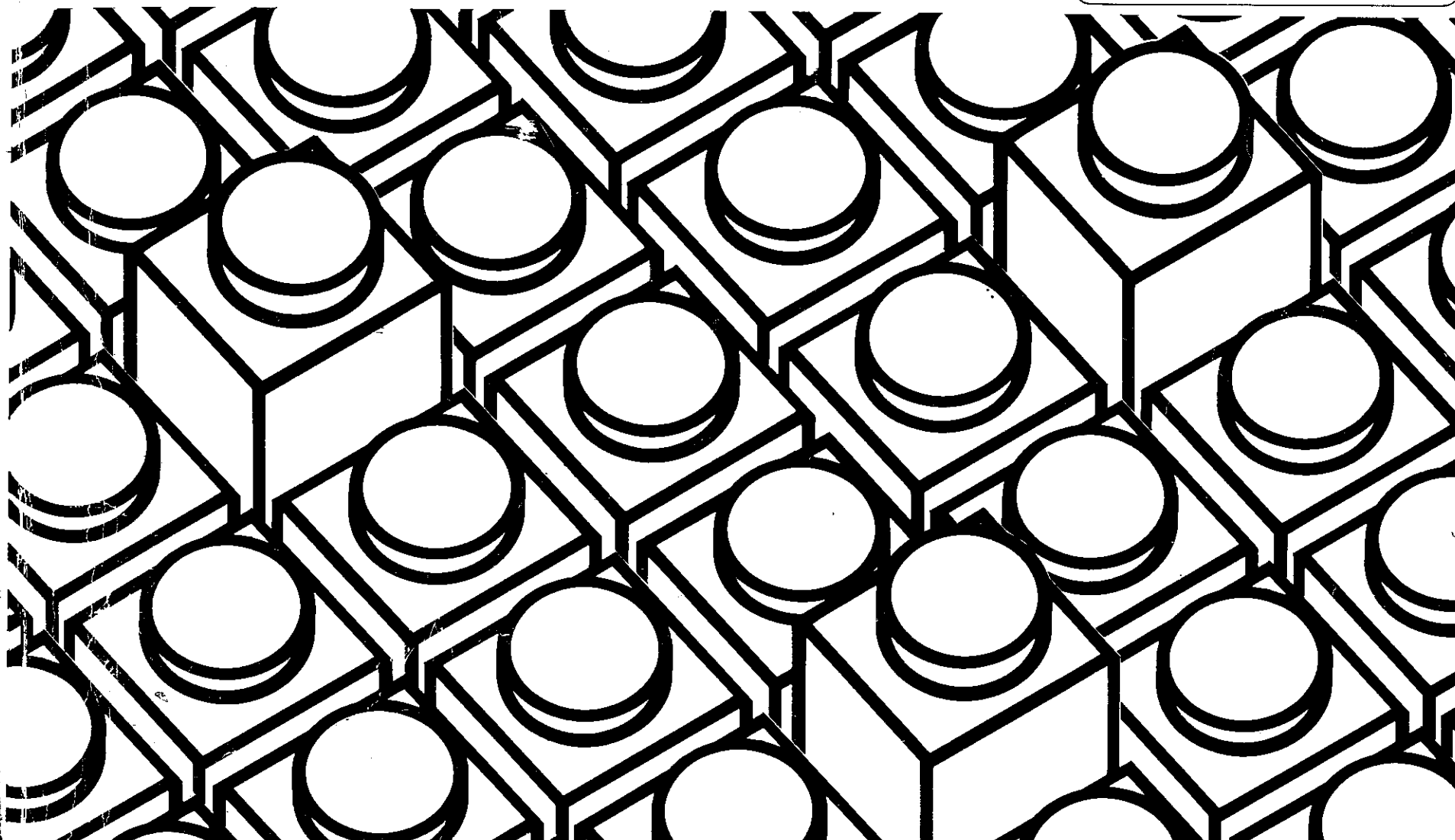


学研電子フロッツ

FX-SYSTEM

FX-マイコンR-165

プログラム集



はじめに

電子ブロックは、エレクトロニクスの基礎から応用、そしてマイクロコンピュータの原理から応用までを、楽しい遊びを通じてマスターすることができる、電気実験セットです。基礎実験では、ブロックの並べかえで、抵抗、コンデンサ、トランジスタ等のはたらきや、電子回路の基礎知識を、楽しい実験を通じて身につけるよう編集してあります。そして、FX-マイコンでは、ブロックを並べかえても、できないような実験が、コンピュータへのプログラムの書き込み、書きかえで簡単にできることがわかり、コンピュータのすばらしさを知ることができます。特に電子ブロックFX-マイコンでは、マイコンゲームを楽しんだあと、自分で作ったプログラムで、コンピュータを君の命令でコントロールすることができるようにしてあります。計算システムや人工衛星をコントロールしているコンピュータが、今や私達の身近なエアコン、マシン等に、組み込まれるようになりました。しかし、そのほとんどが、コンピュータが組み込まれている機械を使うというだけで、コンピュ

ータがどのような原理で働いているか、理解できないと思います。このような時代に、電子ブロックFX-マイコンで、楽しい遊びを通じて、コンピュータに頼り、年齢に応じてマイコンのプログラムに挑戦してみてください。すぐに理解できない所も、何回も挑戦して行くうちに、きっと理解できると思いますので、このFX-マイコンを長く愛用してください。

もくじ

●FX-マイコン 各部の名称とつかいかた	4	No. 12 16進数を小さい順にくり返し表示する実験	30
●コンピュータの言葉について	6	No. 13 16進数の奇数を小さい順にくりかえし表示	32
●マイコンゲームについて	7	No. 14 16進数の偶数を小さい順に一回だけ表示	33
No. 1 電子オルガン	7	No. 15 10進数を小さい順に一回だけ表示	34
No. 2 自動演奏	8	No. 16 10進数の奇数を小さい順に一回だけ表示	35
No. 3 音階当てゲーム	16	No. 17 10進数の偶数を小さい順にくりかえし表示	36
No. 4 もぐらたたき	17	No. 18 10進数を大きい順にくりかえし表示	37
No. 5 テニスゲーム	18	No. 19 16進数を大きい順に一回だけ表示	38
No. 6 タイマー	19	No. 20 数字キーを離すと止まる電子サイコロ	40
No. 7 モールス電文の自動発信	20	No. 21 数字キーを押すと止まる電子サイコロ	41
●FX-マイコンのプログラムテクニック	21	(2) 第2郡の命令について	42
No. 8 TIA命令とAO命令による数字LEDの点燈	22	※ CAL命令のはたらき(その1)	42
(1) 第1郡の命令について	23	※ CH命令のはたらき(その2)	42
※ TIA命令のはたらき	23	No. 22 TIY, AIY, AM命令を使った実験	43
※ AO命令のはたらき	23	No. 23 MA命令によるメモリー内データの逐次表示	44
No. 9 CH命令とJUMP命令を使った2数表示の実験	24	※ TIY命令のはたらき	44
※ CH命令のはたらき(その1)	25	※ AM命令のはたらき	44
※ JUMP命令のはたらき(その1)	25	※ AIY命令のはたらき	44
No. 10 KA命令によるキー入力の数字LED表示	26	No. 24 M+命令によるキー入力の数を加える実験	45
※ KA命令のはたらき	27	※ MA命令のはたらき	45
※ JUMP命令のはたらき(その2)	27	※ M+命令のはたらき	45
※ STEPモードとRUNモードについて(その1)	27	No. 25 M-命令によるキー入力の数を引いた実験	46
No. 11 AIA命令による数字の加算表示	28	※ M-命令のはたらき	46
※ AIA命令のはたらき	29	No. 26 CAL TIMR命令を使った実験	47
※ STEPモードとRUNモードについて(その2)	29	※ CAL TIMRのはたらき	47

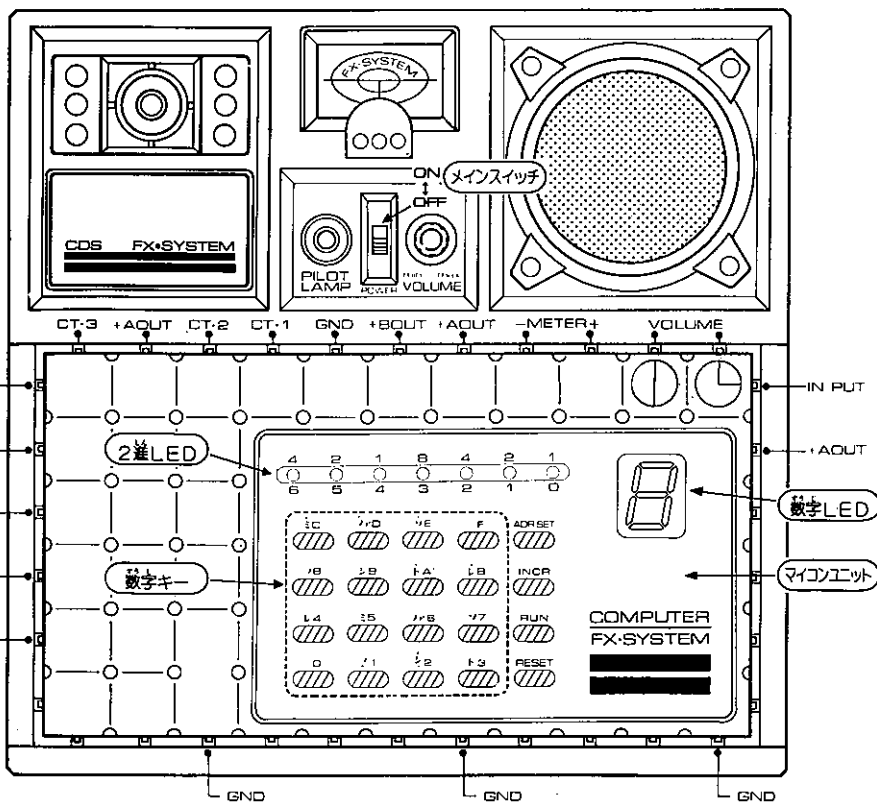
もくじ

No 27	メモリにオール0をストアする実験	48	※ CAL RSTO命令のはたらき	73	
No 28	メモリに0~Fをストアする実験	49	No 47	CAL DSPR命令を使った実験	74
No 29	メモリにF~0をストアする実験	50	※ CAL DSPR命令のはたらき	74	
No 30	10進数アップカウンタの実験	51	No 48	CAL CMPL命令を使った実験	75
No 31	キーONで働く、10進ダウンカウンタの実験	52	※ CAL CMPL命令のはたらき	75	
No 32	1桁+1桁の16進加算の実験(その1)	53	No 49	CIA命令を使った実験	76
No 33	1桁+1桁の16進加算の実験(その2)	54	※ CIA命令のはたらき	76	
No 34	A~Fの16進数を10進数に変える実験	56	No 50	CY命令を使った実験	78
No 35	0~Fの16進数を10進数に変える実験	57	※ CY命令のはたらき	78	
No 36	1桁+1桁の10進加算の実験	58	No 51	2進LEDを右から左へ逐次点燈	79
No 37	1桁-1桁の16進減算と10進補正の実験	59	No 52	2進LEDを右から左へ一つおきに逐次点燈	80
No 38	1桁+1桁の16進加算と10進補正の実験	60	No 53	2進LEDを左から右へ一つおきに逐次点燈	81
No 39	1桁×1桁の16進乗算の実験	62	No 54	2進LEDを左から右へ点滅しながら移動	82
No 40	1桁×1桁の10進乗算の実験	64	No 55	2進LEDを右左に往復点燈移動(その1)	83
No 41	1桁-1桁の10進減算の実験	66	No 56	2進LEDを右左に往復点燈移動(その2)	84
No 42	1桁÷1桁の除算の実験	67	No 57	2進LEDを中央より両端へ点燈移動	85
(3) 第3郡の命令について		68	No 58	2進LEDを両端より中央へ点燈移動	86
※ CIY 命令のはたらき		68	No 59	50~57番地より58~5F番地への移動	87
※ CAL SETR命令のはたらき		68	No 60	メモリ内の5以下の数がいくつあるか表示する実験	88
No 43	CIY, CAL SETR 命令を使った実験	69	No 61	メモリ内のデータ合計を表示する実験	89
No 44	CAL RSTR 命令を使った実験	70	No 62	メモリ内のデータ平均値を表示する実験	90
※ CAL RSTR命令のはたらき		70	No 63	2桁+2桁の16進加算の実験	92
No 45	CAL SHTS命令を使った実験	71	No 64	2桁-2桁の16進減算の実験	94
※ CAL SHTS命令のはたらき		71	No 65	2桁×1桁の16進乗算の実験	96
No 46	CAL RSTO命令を使った実験	72	No 66	2桁÷1桁の16進除算の実験	98

も く じ

(4) 第4部の命令について.....	100	No 83 メトロノーム.....	126
No 67 CAL SUND命令を使った実験.....	100	No 84 16進数の偶数が奇数を当てるゲーム.....	127
※ CAL SUND命令のはたらき.....	100	No 85 上か下かゲーム.....	128
No 68 CAL ERRS命令を使った実験.....	101	No 86 落とし穴ゲーム.....	129
※ CAL ERRS命令のはたらき.....	101	No 87 運動神経ゲーム.....	130
No 69 CAL LONS命令を使った実験.....	102	No 88 数字ゲーム.....	132
※ CAL LONS命令のはたらき.....	102	No 89 神経衰弱ゲーム.....	134
No 70 CAL SIFT命令を使った実験.....	103	No 90 加算ゲーム.....	135
※ CAL SIFT命令のはたらき.....	103	No 91 数当てゲーム.....	136
No 71 CAL DEM+とCAL ENDSの実験.....	104	No 92 ターゲットゲーム.....	138
※ CAL DEM+命令のはたらき.....	104	No 93 16進加算ゲーム.....	140
※ CAL ENDS命令のはたらき.....	105	No 94 ガンファイターゲーム.....	142
No 72 CAL DEM-命令を使った実験.....	106	No 95 スロットルマシン.....	144
※ CAL DEM-命令のはたらき.....	107	No 96 数字の記憶力テスト.....	146
No 73 CAL CHNG命令を使った実験.....	108	No 97 0~9までの乱数をメモリへストア.....	148
※ CAL CHNG命令のはたらき.....	108	No 98 音階当てゲーム.....	150
No 74 メモリ内のデータを小さい順に並べる実験.....	110	No 99 ドレミファ音の乱数をメモリへストア.....	152
No 75 2桁×1桁の10進乗算.....	112	No100 もぐらたたき.....	154
No 76 2桁÷1桁の10進除算.....	114	● CAL INPT命令の実験.....	156
No 77 メモリ内の合計を求める.....	116	※ CAL INPT命令のはたらき.....	157
No 78 メモリ内の平均値を求める.....	118	● SW1によるカウントとSW2によるストア.....	159
No 79 データ入力によるモールス自動発信.....	120	● 自作プログラム表.....	162
No 80 7分59秒までのデフリメントタイマー.....	122		
No 81 ドレミファ音付き2進LEDの点灯移動.....	124		
No 82 モールス、エレキ.....	125		

FX-マイコン 各部分の名称とつがい




メインスイッチ …… メインスイッチをONにするとマイコンが作動し、データ等を書き込むことができます。OFFにすると書き込んだデータ等がすべて消えます。尚じ実験を長時間実験しない場合メインスイッチはOFFにしておきましょう。

2進LED …… 7つのLEDの点灯や消えている状態の、色々な組み合わせで、主に、コンピュータのアドレスを、2進数で表現したり、LEDゲームにも利用します。

数字LED …… 主に、メモリのデータを16進数で表現したり、プログラムの実行結果を表現したりします。

数字キー …… 主に、アドレス番地や、データ数値の入力キーに碰ったり、あらかじめ組まれている、プログラムの指定にも使います。

ADR SET (アドレスセットキー) …… アドレスセットキー、と順に押すことにより、アドレスを指定したり、指定したアドレスの、データを読み出したりする。

INCR (インクリメントキー) …… データの書きこみや、読み出しのときに押す

RUN (ランキー) …… インプットしたプログラムや、あらかじめ組まれているプログラムを実行させるときに押す。

RESET (リセットキー) …… このキーを押すことにより、最初のアドレスにもどり、コンピュータは次の命令を待つ状態になります。

●コンピュータの言葉について



日本人は「ひらがな」や「カタカナ」を使います。

おはよう

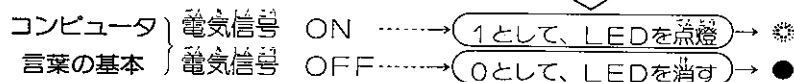


外国人は「アルファベット」を使います。

GOOD MORNING

コンピュータは、電気信号の(ON)と(OFF)の組み合わせで、命令やデータを理解します。

コンピュータは

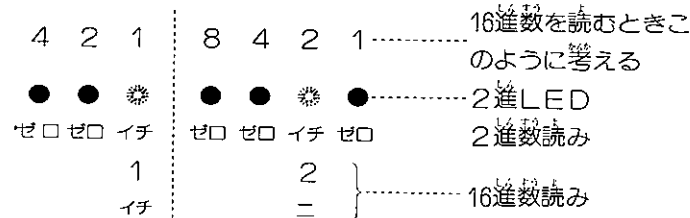


このLEDを、たとえば●●●●●●●●●●のように、組み合わせコンピュータの言葉を作り、これを2進数、16進数、10進数になおして理解します。

皆さんが、いつも使っている数は、9から10に進むときに、ケタ上げされるので10進数といいます。これに対して1から2に進むときに、ケタ上げされて10（イチ、ゼロ）となる数を2進数といい、15から16に進むときにケタ上げされて10（イチ、ゼロ）となる数を16進数といいます。コンピュータはおもにこの16進数を使います。

2進LED	2進数表現	16進数表現	10進数表現
●●●●●●●●	0000000	00	0
●●●●●●●○	0000001	01	1
●●●●●●○●	0000010	02	2
●●●●●○●●	0000011	03	3
●●●●○●●●	0000100	04	4
●●●○●●●●	0000101	05	5
●●○●●●●●	0000110	06	6
●●●●●●●●	0000111	07	7
●●●●○●●●	0001000	08	8
●●●○●●●●	0001001	09	9
●●○●●●●●	0001010	0A	10
●●●●○●●●	0001011	0B	11
●●●○●●●●	0001100	0C	12
●●○●●●●●	0001101	0D	13
●●●●○●●●	0001110	0E	14
●●●○●●●●	0001111	0F	15
●●○●●●●●	0010000	10	16
●●●●○●●●	0010001	11	17

16進数を使うのは、2進LEDで表示される、2進数を読むのに、とても便利だからです。2進数を読むとき、下から4つの単位で区切って16進数で読むと便利です。



※この16進数の12は、（イチ、二）と読みます。

★ マイコンゲームについて

マイコンゲームの実験は、次の7つです。

- NO1 電子オルガン
- NO2 音楽の自動演奏
- NO3 音階あてゲーム
- NO4 モグラたたきゲーム
- NO5 テニスゲーム
- NO6 タイマー
- NO7 モールス電文の自動発信



では、これからのマイコンゲームを楽しみながら、マイコンのはたらきと、コンピュータのことは理解して行きましょう。

FX-マイコンは、小、中学生の皆さんに、マイコンとは、どんなはたらきをするものかを、マイコンゲームの遊び方、ジュニア、マイコンのプログラムの作り方、書き込み方、実行のさせ方を通じて知ってもらうために作った、小さな、コンピュータです。

わからないところは、皆さんが上の学年に進むと共に、だんだんとわかってきますから、たえず、コンピュータのプログラムに挑戦してみてください。

君は、いつのまにか、コンピュータを立派にマスターしている自分に気がつくことでしょう。

No. 1 電子オルガン

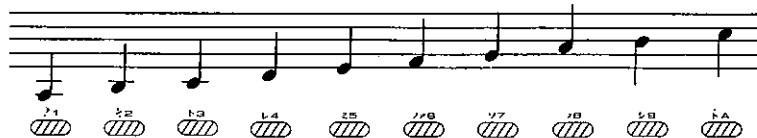
①① ~ ①①①までの14の音階で演奏を楽しんでください。

A. メインスイッチをONにします。

B. ①①① ①①① ①①① キーを順に押し実行すると、

電子オルガンのプログラムがスタートします。

次の順序でキーを押しながら音をたしかめてください。



※キーを押している間だけ音がでます。



さあ、この電子オルガンを使って、皆さんの好きな曲を弾いてください。

※数字キーの値を、マイコンのプログラムで音階に変えています。

C. 実験が終わったらすぐ同じ実験をしない場合
メインスイッチをOFFにしてください。

このゲームでの、コンピュータのことは

キー1—ラ	6—ファ	B—レ
2—シ	7—ソ	C—ミ
3—ド	8—ラ	D—ファ
4—レ	9—シ	E—ソ
5—ミ	A—ド	OFF—無音

No. 2 自動演奏



(アドレス)(データの書き込み)

(2進LED-アドレス表)

(LED)

次の曲は、みなさんもよく知っている「さくら さくら」です。
 音符データを書き込んで自動演奏を楽しんでください。

日本古謡

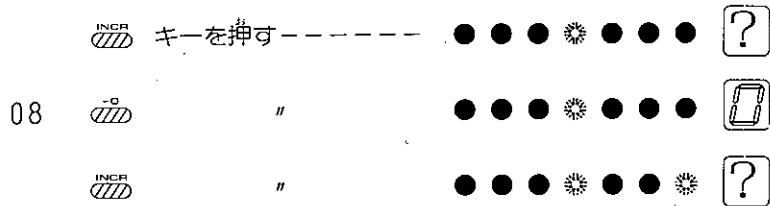
ラ ラ シ ラ ラ シ ラ シ ド シ ラ シ ラ ファ
 ミ ド ミ ファ ミ ミ ド シ ラ シ ド シ ラ シ ラ ファ
 ミ ド ミ ファ ミ ミ ド シ ラ ラ シ ラ ラ シ

押すキー									
RESET	-----								?
00	キーを押す		●	●	●	●	●	●	A
INCR	-----		●	●	●	●	●	●	?
01	"		●	●	●	●	●	●	8
INCR	-----		●	●	●	●	●	●	?
02	"		●	●	●	●	●	●	3
INCR	-----		●	●	●	●	●	●	?
03	"		●	●	●	●	●	●	8
INCR	-----		●	●	●	●	●	●	?
04	"		●	●	●	●	●	●	3
INCR	-----		●	●	●	●	●	●	?
05	"		●	●	●	●	●	●	9
INCR	-----		●	●	●	●	●	●	?
06	"		●	●	●	●	●	●	7
INCR	-----		●	●	●	●	●	●	?
07	"		●	●	●	●	●	●	F

A 簡単な音符データの書き込み

メインスイッチをONにして次にキーを押してください。

※ RESET キーや INCR キーを押したときの、数字LEDの「?」表示は、
 どんな数字が出ていてもよいです。

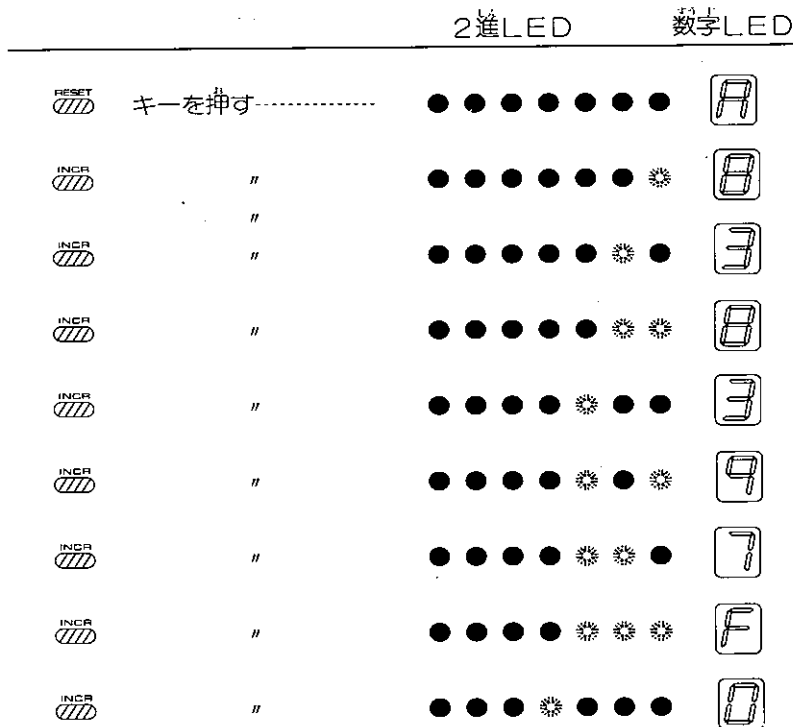


これは、「さくら さくら」の、の音符をコンピュータのこぼに置したデータです。音符データとしては次の表のようになります。

アドレス(番地)	2進LED	データ (命令コード)
00	●●●●●●●●	A-----曲のテンポ
01	●●●●●●●★	8-----音階
02	●●●●●●●★	3-----長さ }
03	●●●●●●●★	8-----音階
04	●●●●●●●★	3-----長さ }
05	●●●●●●●★	9-----音階
06	●●●●●●●★	7-----長さ }
07	●●●●●●●★	F-----終止符
08	●●●●●●●★	0-----くり返し演奏

B データの読み出し

書き込んだデータを次のようにして読み出します。上の表とコンピュータから読み出される、プログラムのアドレス(2進LED)と、データ(数字LED)の表示を見比べて、プログラムどおりに書き込まれていることを確かめます。



このように、次々に、キーを押すと、プログラムのアドレスが、1つずつ進み、2進LEDがアドレスを、数字LEDがデータを表示します。


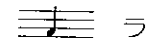
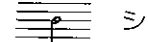
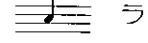
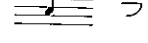
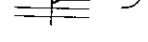
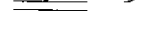


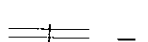
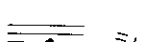
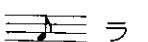

※正しく書き込まれていないときは、キーを押して、始めから書き直します。

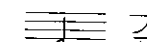
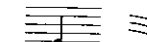
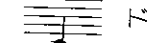
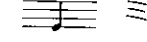
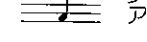
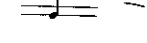
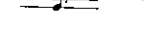


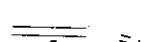
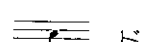
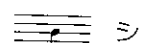
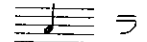
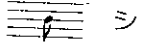
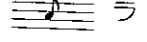

C 自動演奏開始

正しく書き込まれていることを、確認した後、次のようにキーを押してください。

キー キー キーを、順々に押し実行すると、「さくら さくら」……………を、繰り返し演奏します。

自動演奏のデータの書き込みと、演奏のさせ方がわかったら、次のデータを書き込んで「さくら さくら」の曲を楽しんでください。
データの書き込み方は、次のページにあります。

アドレス	2進LED	データ
00	●●●●●●●●	A----曲のテンポ
01	●●●●●●●○	8----音階 }  ラ
02	●●●●●●●○	3----長さ }  ラ
03	●●●●●●●○	8----音階 }  ラ
04	●●●●●●●○	3----長さ }  ラ
05	●●●●●●●○	9----音階 }  シ
06	●●●●●●●○	7----長さ }  シ
07	●●●●●●●○	8----音階 }  ラ
08	●●●●●●●○	3----長さ }  ラ
09	●●●●●●●○	8----音階 }  ラ
0A	●●●●●●●○	3----長さ }  ラ
0B	●●●●●●●○	9----音階 }  シ
0C	●●●●●●●○	7----長さ }  シ
0D	●●●●●●●○	8----音階 }  ラ
0E	●●●●●●●○	3----長さ } ラ
0F	●●●●●●●○	9----音階 } シ
10	●●●●●●●○	3----長さ } シ
11	●●●●●●●○	A----音階 } ラ
12	●●●●●●●○	3----長さ } ラ
13	●●●●●●●○	9----音階 } シ
14	●●●●●●●○	3----長さ } シ
15	●●●●●●●○	8----音階 } ラ
16	●●●●●●●○	3----長さ } ラ
17	●●●●●●●○	9----音階 } シ
18	●●●●●●●○	1----長さ } シ
19	●●●●●●●○	8----音階 } ラ
1A	●●●●●●●○	1----長さ } ラ


1B	●●●●●●●○	6----音階 }  ラ
1C	●●●●●●●○	7----長さ }  ラ
1D	●●●●●●●○	5----音階 }  ラ
1E	●●●●●●●○	3----長さ }  ラ
1F	●●●●●●●○	3----音階 }  ラ
20	●●●●●●●○	3----長さ }  ラ
21	●●●●●●●○	5----音階 }  ラ
22	●●●●●●●○	3----長さ }  ラ
23	●●●●●●●○	6----音階 }  ラ
24	●●●●●●●○	3----長さ }  ラ
25	●●●●●●●○	5----音階 }  ラ
26	●●●●●●●○	3----長さ }  ラ
27	●●●●●●●○	5----音階 }  ラ
28	●●●●●●●○	1----長さ }  ラ
29	●●●●●●●○	3----音階 }  ラ
2A	●●●●●●●○	1----長さ }  ラ
2B	●●●●●●●○	2----音階 } ラ
2C	●●●●●●●○	7----長さ } ラ
2D	●●●●●●●○	8----音階 } ラ
2E	●●●●●●●○	3----長さ } ラ
3F	●●●●●●●○	9----音階 } シ
30	●●●●●●●○	3----長さ } シ
31	●●●●●●●○	A----音階 } ラ
32	●●●●●●●○	3----長さ } ラ
33	●●●●●●●○	9----音階 } シ
34	●●●●●●●○	3----長さ } シ
35	●●●●●●●○	8----音階 } ラ
36	●●●●●●●○	3----長さ } ラ
37	●●●●●●●○	9----音階 } シ
38	●●●●●●●○	1----長さ } シ
39	●●●●●●●○	8----音階 } ラ
3A	●●●●●●●○	1----長さ } ラ








アドレス	2進LED	データ
3B	●●●●●●●●	6---音階 } フ
3C	●●●●●●●●	7---長さ } フ
3D	●●●●●●●●	5---音階 } ミ
3E	●●●●●●●●	3---長さ } ミ
3F	●●●●●●●●	3---音階 } ド
40	●●●●●●●●	3---長さ } ド
41	●●●●●●●●	5---音階 } ミ
42	●●●●●●●●	3---長さ } ミ
43	●●●●●●●●	6---音階 } フ
44	●●●●●●●●	3---長さ } フ
45	●●●●●●●●	5---音階 } ミ
46	●●●●●●●●	3---長さ } ミ
47	●●●●●●●●	5---音階 } ミ
48	●●●●●●●●	1---長さ } ド
49	●●●●●●●●	3---音階 } ド
4A	●●●●●●●●	1---長さ } ド
4B	●●●●●●●●	2---音階 } シ
4C	●●●●●●●●	7---長さ } シ
4D	●●●●●●●●	8---音階 } ラ
4E	●●●●●●●●	3---長さ } ラ
4F	●●●●●●●●	8---音階 } ラ
50	●●●●●●●●	3---長さ } ラ
51	●●●●●●●●	9---音階 } シ
52	●●●●●●●●	7---長さ } シ
53	●●●●●●●●	8---音階 } ラ
54	●●●●●●●●	3---長さ } ラ
55	●●●●●●●●	8---音階 } ラ
56	●●●●●●●●	3---長さ } ラ
57	●●●●●●●●	9---音階 } シ
58	●●●●●●●●	7---長さ } シ


59	●●●●●●●●	0---無音
5A	●●●●●●●●	F---長さ
5B	●●●●●●●●	F---終止符
5C	●●●●●●●●	0---繰返し演奏

D 「さくら さくら」のデータの書き込み


「さくら さくら」の音符データを見ながら、00番地から5C番地までに、データを書き込んでください。









 キーを押してから、次のようにキーを順に押します。

       ---以下略、

上記のように、データ表を見ながら、次々に数字キーと、 キーを交互に押して全部書き込んでください。

E データの読み出しと、書き間違いのチェック

 キーを順々に押しながら、プログラムのアドレス（2進LED）と、データ（数字LED）の表示を見比べて、プログラムどおりに書き込まれているか確かめます。もし表のデータと違っている箇所があれば、そのアドレスに自印を付けておいてください。













00		キーを押す---	●●●●●●●●	
01		"	●●●●●●●●	
02		"	●●●●●●●●	
03		"	●●●●●●●●	

F データの訂正

少ないデータを書き違えたときは、前のように、始めから書き直すことができますが、長いデータの一部を書き違えたときは、(自印を付けた箇所) 沢の方法で違っている所だけを書き直します。







(例) アドレスの03番地に3が書き込まれているとき。

(正しいデータは8)

	キーを押す		} データを間違えたアドレス番地の番号。
	"		
	"		
	"		} 03番地に正しいデータ(8)を書き込む。
	"		
	"		




G 自動演奏開始

正しく書き込まれていることが、確かめられたら、沢のようにキーを押します。「さくら さくら」の曲が、くり返し、くり返し、演奏されます。

	キーを押す	-----	
	"	-----	
	"	-----	

※ 数字LEDが何も表示されていない状態

H 自動演奏のストップと再開




曲の途中で、演奏を止めたいときは、キーを演奏が止まるまで押してください。もう一度演奏したい場合は、キー、キーを押すと、再び演奏を始めます。

I 自動演奏のテンポの変更

演奏のテンポは、00番地をF、E、D、C-----2、1、0と小さい数字にしていくと早くなります。

では、沢のように演奏のテンポを書き換えてみてください。









(例) 00番地のAを、5に変えてみます。


キー、キー、キーを順に押してください。

書き換えましたら、Gの方法で、再び演奏させてください。早いテンポで、「さくら さくら」の曲が演奏されます。00番地の内容をいろいろかえて、実験してみてください。

J 自動演奏を一回のみで止めたい時

曲のデータの最後(さくら さくら)の曲では、5C番地)の0をFに、沢のように書き換えます。

	キーを押す	● ● ● ● ● ● ● ●	
	"	● ● ● ● ● ● ● ●	
	"	● ● ● ● ● ● ● ●	
	"	● ● ● ● ● ● ● ●	




 キーを押す



 "

"



続いて、(   キーを押して) 自動演奏を開始させます。「さくら さくら」の曲は、一回だけ演奏してとまります。

このゲームのコンピュータのことは、「NO1、電子オルガン」の音階データのほかに、
テンポ、データ 0~F はやい0 ←————→ Fおそい
休符、データ = 0
音階の長さデータ 次の表を見てください。
終止符、データ=F (くり返し演奏のときは、Fの次に0を書く、一回のみのときは、Fの次にFを書く)

〈音階や休符の長さのデータ〉

音階と休符の長さのデータは、0~Fの16段階で16分音符(♪)の長さを1とした場合、データは1を引いた0になります。この方法で、データは次のとおりです。

音階及び休符	長さ	データ
16分音符(♪)	-----1	0
16分休符(♯)	-----1	0
8分音符(♪)	-----2	1
8分休符(♯)	-----2	1
付点8分音符(♪)	-----3	2
4分音符(♪)	-----4	3
4分休符(♯)	-----4	3
付点4分音符(♪)	-----6	5
2分音符(♪)	-----8	7
2分休符(-)	-----8	7

付点2分音符(♪)	-----12	B
全音符(○)	-----16	F
全休符(-)	-----16	F

K 自動演奏データの書き方

皆さんの好きな曲を、次の方法で自動演奏データに置いて、自動演奏を楽しんでください。

- ① 自動演奏データは、00番地から5F番地までに書く。
- ② 00番地には、自動演奏のテンポ、データを書く。
- ③ 奇数番地に、音階又は、休符のデータを書く。
- ④ 偶数番地に、音階又は、休符の長さのデータを書く。
- ⑤ 5D番地以内に、②、③、④、③、④、③、④ --- ③、④の順序で自動演奏データを書き終え(5D番地より長い曲は、切りの良いところで止めます)最後の音階(又は休符)データの長さの次にF(終りのしるし)を書く、その次に、繰り返し演奏をさせたいときは0を書き、一回のみで演奏を止めたいときは、Fを書く。

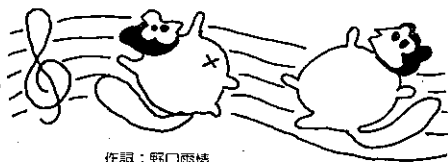
では、次の曲を、上の「自動演奏のデータの書き方」に従って、データに置いてみてください。
音は、楽譜の後に書いてありますから、正しくデータに置せたか確かめてください。

しょうじょうじ たのき 証城寺の狸ばやし

日本音楽著作権協会(出)許諾第8100488号

♩=88

作詞：野口雨情
作曲：中山晋平



mf

しょうじょうじょう しょうじょうじ しょうじょうじの にわは

うつつきよだ みんなでて さいさいさい

おーらの ともだち ポンポコポンノ ポン

【自動演奏のデータ】

アドレス	2進LED	データ
00	●●●●●●●●	9---曲のテンポ ♩=88
01	●●●●●●●●	3---音階
02	●●●●●●●●	3---長さ }
03	●●●●●●●●	5---音階
04	●●●●●●●●	3---長さ }
05	●●●●●●●●	7---音階
06	●●●●●●●●	1---長さ }
07	●●●●●●●●	5---音階
08	●●●●●●●●	1---長さ }
09	●●●●●●●●	3---音階
0A	●●●●●●●●	1---長さ }
0B	●●●●●●●●	0---休符 }
0C	●●●●●●●●	1---長さ }
0D	●●●●●●●●	7---音階
0E	●●●●●●●●	1---長さ }
0F	●●●●●●●●	5---音階
10	●●●●●●●●	1---長さ }

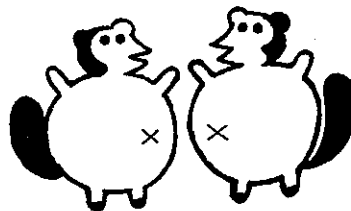
11	●●●●●●●●	4---音階		レ
12	●●●●●●●●	1---長さ		ト
13	●●●●●●●●	3---音階		レ
14	●●●●●●●●	1---長さ		ト
15	●●●●●●●●	4---音階		レ
16	●●●●●●●●	1---長さ		ト
17	●●●●●●●●	5---音階		ミ
18	●●●●●●●●	1---長さ		ソ
19	●●●●●●●●	7---音階		ソ
1A	●●●●●●●●	1---長さ		ソ
1B	●●●●●●●●	0---休符		
1C	●●●●●●●●	1---長さ		レ
1D	●●●●●●●●	5---音階		ミ
1E	●●●●●●●●	3---長さ		ミ
1F	●●●●●●●●	7---音階		ソ
20	●●●●●●●●	3---長さ		ソ
21	●●●●●●●●	A---音階		ト
22	●●●●●●●●	1---長さ		ト
23	●●●●●●●●	C---音階		ミ
24	●●●●●●●●	1---長さ		ミ
25	●●●●●●●●	B---音階		レ
26	●●●●●●●●	1---長さ		レ
27	●●●●●●●●	A---音階		ト
28	●●●●●●●●	1---長さ		ト
29	●●●●●●●●	7---音階		ソ
2A	●●●●●●●●	2---長さ		ソ
2B	●●●●●●●●	7---音階		ソ
2C	●●●●●●●●	0---長さ		ソ
2D	●●●●●●●●	8---音階		ラ
2E	●●●●●●●●	1---長さ		ラ
2F	●●●●●●●●	7---音階		ソ
30	●●●●●●●●	1---長さ		ソ

アドレス	2進LED	データ
31	●●●●●●●●	A --- 音階
32	●●●●●●●●	1 --- 長さ
33	●●●●●●●●	A --- 音階
34	●●●●●●●●	1 --- 長さ
35	●●●●●●●●	A --- 音階
36	●●●●●●●●	1 --- 長さ
37	●●●●●●●●	0 --- 休符
38	●●●●●●●●	1 --- 長さ
39	●●●●●●●●	6 --- 音階
3A	●●●●●●●●	1 --- 長さ
3B	●●●●●●●●	5 --- 音階
3C	●●●●●●●●	1 --- 長さ
3D	●●●●●●●●	4 --- 音階
3E	●●●●●●●●	1 --- 長さ
3F	●●●●●●●●	3 --- 音階
40	●●●●●●●●	1 --- 長さ
41	●●●●●●●●	4 --- 音階
42	●●●●●●●●	1 --- 長さ
43	●●●●●●●●	5 --- 音階
44	●●●●●●●●	1 --- 長さ
45	●●●●●●●●	7 --- 音階
46	●●●●●●●●	1 --- 長さ
47	●●●●●●●●	7 --- 音階
48	●●●●●●●●	1 --- 長さ
49	●●●●●●●●	A --- 音階
4A	●●●●●●●●	1 --- 長さ
4B	●●●●●●●●	8 --- 音階
4C	●●●●●●●●	0 --- 長さ
4D	●●●●●●●●	7 --- 音階
4E	●●●●●●●●	0 --- 長さ
4F	●●●●●●●●	5 --- 音階
50	●●●●●●●●	1 --- 長さ



51	●●●●●●●●	4 --- 音階
52	●●●●●●●●	1 --- 長さ
53	●●●●●●●●	3 --- 音階
54	●●●●●●●●	3 --- 長さ
55	●●●●●●●●	0 --- 休符
56	●●●●●●●●	3 --- 長さ
57	●●●●●●●●	F 終止符
58	●●●●●●●●	0 くり返し演奏

※ このデータでは、最後をくり返し演奏にしています。


正しくデータに値せましたか？ それでは、この曲を、「さくら さくら」の曲と同じ方法で、データを書き込んで、演奏させてみましょう。





No. 3 おんかいあ音階当てゲーム

音階当てゲームは、基本音〈ド〉とその後に続いて出る音を、音階キー（～)を押し、音階を当てていくゲームです。
このゲームを始める前に、No.1の“電子オルガン” No.2の“自動演奏”で、音階を正しく聞き分けられるようにしておいてください。

A ゲームのスタート


メインスイッチをONにして、キーを押ししてください。

キーを押し、キーを押すと、

初めに基本音〈ド〉

そして、その次に〈ラ〉から〈ソ〉までの音階の内の1つの音が出て、音階当てゲームがスタートします。


B 初めに出了る音は、必ず〈ド〉の音ですから、

まず、キーを押すと、キーが押されるのと同時に〈ド〉の音が出ます。


次に、〈ド〉の音に続いて出たと感じる音のキーを押ししてください。

正解ですと、キーが押されるのと同時に押した音が出た後、
更に新しい音が1つ加わって、3つの音が出ますから、
次のCへ進みます。

未正解ですと、“ピュー・ピュー”というエラー音が出て、得点“0”
(最初の〈ド〉の音は点数にならない)が表示されます。

もう一度ゲームを行うときを押すと再スタートです。音階当てに挑戦してみましょう。

C 正解して、〈ド〉の音に続いて2つの音階が出ましたら、

とその次に出了る音のキーを押し、次に、新しく加わった音階に挑戦します。

3つとも正解しますと、更に1つ新しい音が加わり、Dへ進みます。

—16— ここで未正解ですと、エラー音が出て、得点“1”が表示されます。

D 正解が続くと、前に出た音に1つつ新しい音を加えられるので、音階を当てていってください。

最初の〈ド〉の音の後、10音全部正解した場合は、
終了音が出て、
得点“A”(10点)が表示されます。

途中で未正解になると、

エラー音が出て、

正解した数(最初の〈ド〉の音は点数にならない)が得点となつて、表示されます。

キミは得点とれましたか？

キミの音感には友達よりもよかったですか。

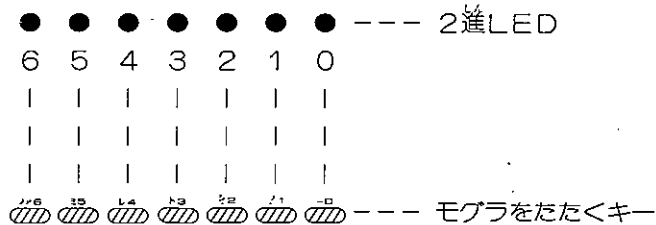
何回も挑戦してみてください。



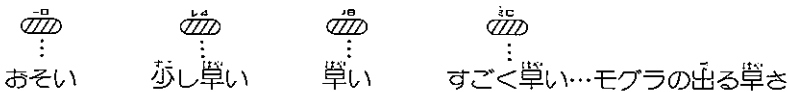
No. 4 もぐらたたき



7つの2進LEDをモグラが出て来る穴とし、2進LEDが点灯したときをモグラが出て来たことにして、数字キーを使い、もぐらたたきを行います。モグラは、全部で10匹出て、ゲームは終了し、うまくたたきあてた点数の、数字がLEDに表示されます。



- ゲームをスタートさせると、2進LEDが、点灯したり消えたり、します。この2進LEDが点灯しているうちに、それぞれの数字に合った数字キーを押すと得点になります。2進LEDが点灯しているあいだに数字キーをうまく押すと、ポツという正解音が出ます。もしはずれた場合は、音は出ません。
- このゲームでは、キーの後に、、、、キーの内どれかを押すことにより、モグラの出る早さをきめることができます。



A ゲームのスタート

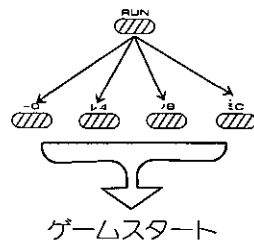
メインスイッチをONにする。

キーを押す -----

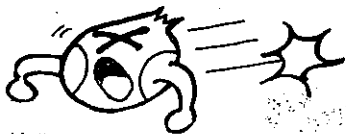
キーを押すと、モグラの出る早さが、おそい状態でゲームがスタートします。

- モグラは、横に並んだ7つの2進LEDのどこか、ひとつに出て来ます。この点灯したLEDの下に書いてある、数字と同じキーを、す早く押して、得点します。モグラが10匹出て、ゲーム終了すると、ピロピロ音が出て、得点表示します。得点表示は、16進数ですので、10点のばあいを表示します。

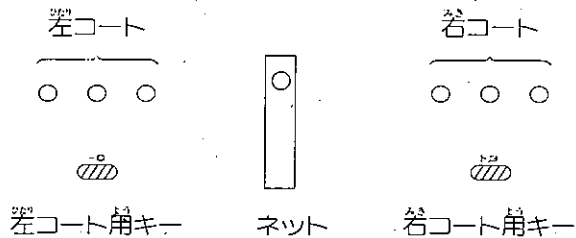
- B ゲームをもう一度行うとき、キーの後に、モグラの出る早さを指定すると、ゲームは再スタートします。



No. 5 テニスゲーム

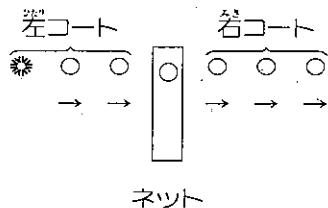


このゲームは、光るLEDをテニスボール、数字キーをラケットとして考えた、2人で楽しむゲームです。



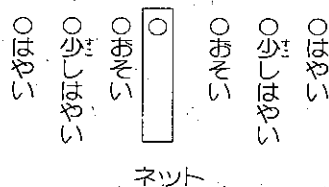
●左コートの入は、キーを、右コートの入は、キーをそれぞれ、ラケットとして使います。光るLEDを打ち返す時、このキーを押して、すぐにはなします。キーを押し続けていると、返球できません。

●ゲームをスタートさせると、左コートのLEDが光り、キーのサーブで、ゲーム開始になります。



●キーを押すと、LEDが光りながら、右コートへ移動します。右コートの入は、自分のコートに、LEDが入って来たら、キーを押して、すぐにはなすと、「ポツ」という音がして、LEDを打ち返せます。このような、キー操作をくりかえして、相手がミスをするまで打ち返します。ミスをするときピロピロ音が出ます。

●次のように、自分のコートのLEDを打ち返すタイミングで、相手コートへの返球スピードが変化します。

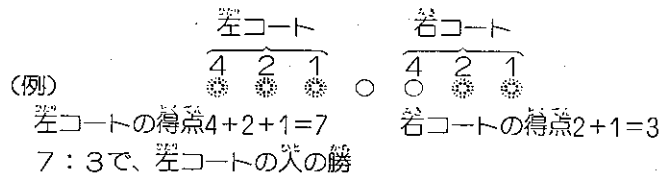


●次のミスをする時、相手の得点になります。自分のコート以外で、キーを押すと、から振りになる。(キーの二度押しはできません)

キーを押し続けているとミスになる。
光るLEDが自分のコートで、打ち返せなかった時

●どちらかが、ミスをした時、ミスをした側の端にLEDが止まりますので、光っているコートの入のサーブで、ゲーム開始です。

●ゲーム終了、どちらかの得点が、7点になると終了音(ピロピロ音)が出て、得点を2進LEDが表示します。得点は次のように計算します。



A ゲームのスタートのしかた

、、-----キーを順に押しとゲームスタート

B もう一度、同じゲームをする場合 キーを押します。

No. 6 タイマー

この実験は、あらかじめ、自分できめた時間（最大7分60秒まで）に、
・エンド音を出す実験です。

- たとえば、3分25秒のタイマーをセットしたいとき、

RESET キーを押す ----- ● ● ● ● ● ● ● ● [?]

↑D ----- ● ● ● ● ● ● ● ● [3]

INCR ----- ● ● ● ● ● ● ● ● [?]

↑D ----- ● ● ● ● ● ● ● ● [2]

INCR ----- ● ● ● ● ● ● ● ● [?]

↑S ----- ● ● ● ● ● ● ● ● [5]

INCR ----- ● ● ● ● ● ● ● ● [?]

↑E ----- ● ● ● ● ● ● ● ● [E]

RUN キーを押すと、 ● ● ● ● ● ● ● ● [5]
タイマーがスタートします。

※数字LEDの表示説明中 [?] は、どのような数でもかまわない数字の意味です。

↑D キーを押すと、タイマーがスタートし、およそ1秒ごとに、ポツポツという音と、ともに、残り時間を表示してゆきます。

- 残り時間の読みかた

4 2 1 8 4 2 1 [5] (3分25秒の例)

3分 25秒

10秒後

4 2 1 8 4 2 1 [5]

3分 15秒

さらに1分後

4 2 1 8 4 2 1 [5]

2分 15秒



このように、どんどん減少して行き、

● ● ● ● ● ● ● ● [0] になると、エンド音(ピロピロ...)がでます。

- 再びタイマーをセットするとき、分、秒をセットして ↑E ↑D キーでスタートさせてください。

No. 7 モールス電文の自動発信

この実験は、モールス電文をマイコンデータに置して、インプットすると、自動発信する実験です。

●モールス符号表

Aイ	..	ソ	----	ア	-----	6
□	Pツ	サ	-----	7
Bハ	Qネ	キ	-----	8
Cニ	Rナ	...	ユ	-----	9
Dホ	...	Sラ	...	メ	-----	0
Eエ	.	Tム	-	ミ	-----		欧文
ト	Uウ	...	シ	-----		?
Fチ	□	ヱ	-----		ピリオド
Gリ	---	ノ	ヒ	-----		???
Hヌ	オ	モ	-----		和文
I	..	Vク	セ	-----		濁点 ..
ル	-----	Wヤ	ス	-----		半濁点
Jヲ	-----	Xマ	ン	-----		長音
Kワ	---	Yケ	1	-----		区切点
Lカ	Zフ	2	-----		段落
Mヨ	--	コ	-----	3		(括弧
Nタ	..	エ	-----	4)括弧
OL	---	テ	5		

	送信 コード		送信 コード		送信 コード
..	0	字の区切りと・	8	無音の区切り1	A
.-	1	字の区切りと-	9	無音の区切り2	B
-.	4	・と語の区切り	3	終り	E
--	5	-と語の区切り	7	くり返し	F
・と字の区切り	2	語の区切りと・	C		
-と字の区切り	6	語の区切りと-	D		

●モールス電文を打つ場合、字の簡は、3点分の時間をあけ（1点は、モールスの・の長さ）、語と語の簡は、7点あける約束になっています。

例として タロウ、という文字を、コードで表わしてみると、

タ □ ウ
 .. (字の区切り) .-.- (字の区切り) .-.- (語の区切り) ... モールスコード
 4 8 4 6 0 7 マイコンコード

もう一つ例として、TOM、という文字を、コードで表わしてみると、

T 0 M
 .. (字の区切り) .-.- (字の区切り) .-.- (語の区切り) ... モールスコード
 6 5 6 5 A マイコンコード

●モールス発信の早さをきめるデータ

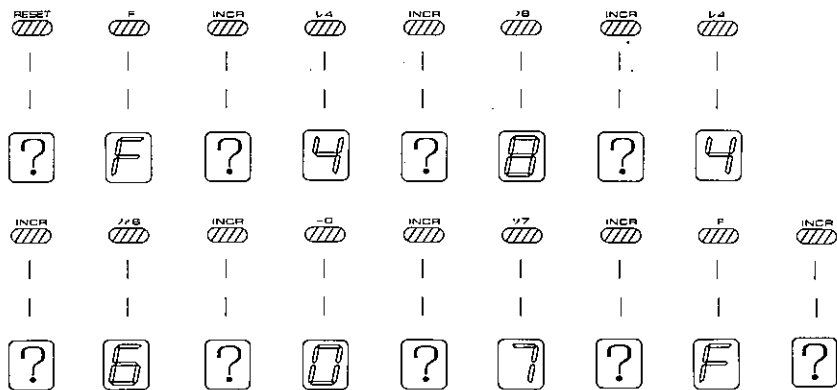
0-----7、8-----F

←はやい おそい→

モールス電文の命令コードを、マイコンに書き込む前に、早さをきめるデータを入れます。

●データの書き込み

RESET、F、INCR……とキーを押して、タロウのデータを入れながら、各数字LEDの表示を確認してください。



●書き込んだデータの確認をしてください。

RESET INCR INCR……と次々に、INCRキーを押して行くと、



このようにデータが、書き込まれていれば正しいです。

●RESET、F、RUNで、命令を実行すると、タロウのモールス音がくりかえし出て来ます。

●モールス音を止める場合、RESETキーを押しているときと止まります。さらに、同じモールス音を出す場合、F、RUNキーを押すと、音が止まります。

●モールス発信音の早さを変るとき、00番地へ書き込まれているF、を変えると、早さが変わります。Fを5にかえるには、

RESET F F INCR……とキーを押すと[F]となりますから、ここに、F INCR キーを押すと、早さをきめるデータが変わり、モールス発信音が、Fのときより早く発信されます。RESET F RUNでたしかめてください。

FX-マイコンのプログラムテクニック

マイコンゲームで楽しく遊んだら、こんどは、FX-マイコンを、君の命令でコントロールしてみましょう。

これから、FX-マイコンの命令語と、プログラムテクニックを、マスターして、コンピュータに、君が作ったプログラムを書き込んで、このFX-マイコンを楽しく遊んでください。その楽しい目的のために、まずFX-マイコンの命令語の命令記号や、命令コード、そのはたらきなどをおぼえてください。別紙の命令記号一覧表にしたがって、だんだんに、おぼえて行きましょう。また命令記号のはたらきなどは、各実験の所で、くわしく説明してありますから、くりかえし実験して、各命令記号の意味を理解してください。



No. 8 TIA命令とAO命令による数字LEDの点灯

数字LEDを点灯させる命令は、このFX-マイコンでは、AO命令だけです。

この実験は、TIA命令でAレジスタに数字を入れ、これを数字LEDに出力して点灯させる、基本的なプログラムの勉強です。

●プログラム例（数字LEDに5を点灯させる）

アドレス	命令記号	命令コード	
00	TIA	8	Ar (Aレジスタ) に数字5を入れる。 Arの内容<5>をOポートへ出力して、数字LEDを点灯させる。
01	<5>	5	
02	AO	1	

Ⓐ プログラムの書き込み

上のプログラムを見て、4の方法で、00番地から02番地までに、プログラムを書き込みます。

Ⓑ メインスイッチをONにして、次の順序で、キーを押してください。

	2進LED	数字LED(データ表示)	
押すキー	(アドレス表示)		
	● ● ● ● ● ● ● ●		} 00番地にTIA命令(8)を書き込む、アドレスは次へ進む。
	● ● ● ● ● ● ● ●		
	● ● ● ● ● ● ● ●		
	● ● ● ● ● ● ● ●		

押すキー



} 01番地



} 02番地



Ⓒ 上のよう書き込んだら、書き込んだデータを読み出して、確かめてみてください。

押すキー



Ⓓ 書き込んだ命令を、次のような操作で実行させます。

押すキー



} 1命令自実行 (TIA命令)



} 2命令自実行 (AO命令)

キーを押したあとで、数字LEDに“5”が表示されました。これで、この命令が実行されたことがわかるでしょう。



数字LEDに何も表示されず、消えている状態のことです。



プログラムを書き込む前にメモリに書き込まれていたデータ数字でどんな数字を表示してもよい。

① 01番地の命令コード〈5〉を〈2〉に変更する場合。

次の順序でキーを押して書き換えてください。

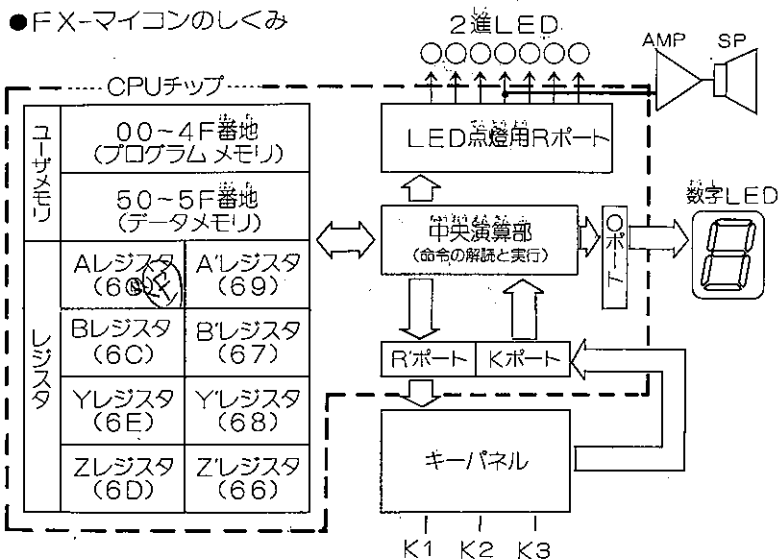


に続いて、書き換えた命令を、次の操作で実行します。



2 を表示します。 データ書き換えまでの表示は、 5

●FX-マイコンのしくみ

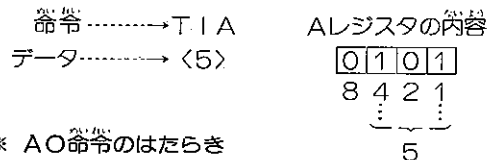


(1) 第1群の命令について

第1群の命令は、KA、AO、CH、TIA、AFA、JUMP命令の6種類の命令グループです。この第1群の実験は、Aレジスタと、Bレジスタが、コンピュータの中で、どのようなはたらきをするか、プログラムを通じて、テストします。Aレジスタは、コンピュータの中心となるレジスタで、数字LEDを点灯するとき、キー入力を読み込むとき、計算をするときなど、いつもAレジスタを使います。Bレジスタは、Aレジスタの補助レジスタで、Aレジスタの内容を一時的に保管して、Aレジスタのはたらきを助けます。

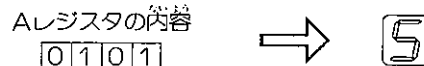
※ TIA命令のはたらき

TIA命令は、下の図のように、プログラム用紙二行で書き、二行目に書いた数は、この命令を実行するとAレジスタに書き込まれます。



※ AO命令のはたらき

AO命令は、下の図のように、Aレジスタの内容を数字LEDに表示する命令です。プログラムで命令を実行すると、TIA命令などで、Aレジスタに書き込まれていた数字が、数字LEDに表示されます。



この命令が実行されても、Aレジスタの内容は変わりません。

No. 9 CH命令とJUMP命令を使った2数表示の実験

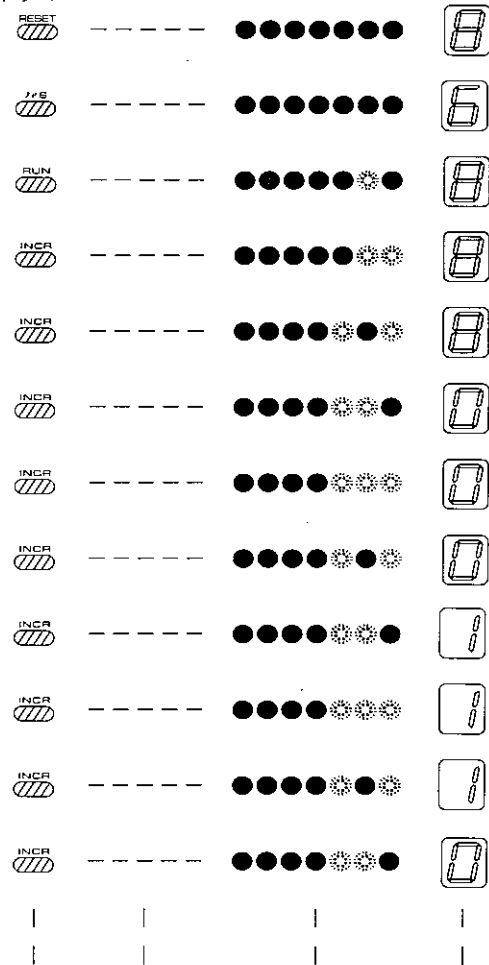
0と1の2つの数字を交互に数字LEDに表示したいときには、Aレジスタのみでは、表示できません。こうしたときには、CH命令とJUMP命令を加えて、次のようなプログラムを作ります。

●プログラム例(数字LEDに0と1を交互に点灯させる)

アドレス	命令記号	命令コード	
00	TIA	8	Aレジスタに数字1を入れる Aレジスタの1をBレジスタに入 れる。
01	<1>	1	
02	CH	2	
03	TIA	8	Aレジスタに数字0を入れる。
04	<0>	0	
05	AO	1Aレジスタの内容を表示
06	CH	2ArとBrの内容を交換する。
07	JUMP	F	プログラムの実行を05番地に戻 し、くり返し実行させる。
08	<0>	0	
09	<5>	5	

- Ⓐ プログラムの書き込み
00番地から、09番地までに、上のプログラムを書き込んでください。
- Ⓑ プログラムの確認と訂正を行なってください。
- Ⓒ 書き込んだプログラムを次のような操作で実行と確認をします。

押すキー



TIA命令実行

CH命令実行

TIA命令実行

AO命令実行

CH命令実行

JUMP命令実行

AO命令実行

CH命令実行

JUMP命令実行

AO命令実行

このあと、INCRキーを3回押すごとに、CH命令、JUMP命令、AO命令がくり返され、数字LEDの表示する数字が、“0”から“1”、“1”から“0”へと変わります。

CH命令で、AレジスタとBレジスタの内容が、1と0、0と1に変わり、JUMP命令で、プログラムが05番地に戻ってくり返し実行されることなど、CH命令とJUMP命令のはたらきを、ここで理解してください。

TIA命令と、AO命令のはたらきは、前の実験と同じです。

● CH命令のはたらき (その1)

CH命令は、下の図のように、AレジスタとBレジスタの内容を交換する命令です。プログラムで、この命令を実行すると、Aレジスタの内容がBレジスタに入り、Bレジスタの内容が、Aレジスタに入ります。

▲CH命令実行前

AレジスタにTIA命令で1が入っているとす。

Bレジスタには、直接数字を入れる命令がないので、未定。

Aレジスタ	Bレジスタ
1	未定

▲CH命令実行後

Aレジスタ	Bレジスタ
未定	1

命令実行後、TIA命令でAレジスタに0を入れる。

Aレジスタ	Bレジスタ
0	1

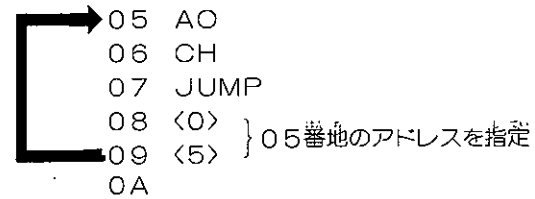
▲次に、CH命令を実行すると

Aレジスタ	Bレジスタ
1	0

と、CH命令を実行するたびに、Aレジスタの内容とBレジスタの内容が、入れかわります。

● JUMP命令について (その1)

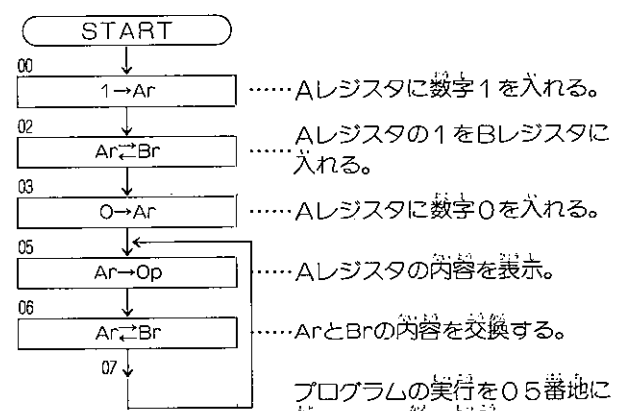
JUMP命令は、プログラムの実行順序を変える命令です。つぎのように、プログラム用紙に3行で書き、プログラムでこの命令を実行させると、2行目(上位)と3行目(下位)で指定したアドレスへ、プログラムの実行が移ります。



上のJUMP命令がないと、プログラムは、アドレスの0A番地以降へどんどん進んでいってしましますが、このJUMP命令で、プログラムの実行は、05番地と09番地の間の命令をくり返し実行し、0A番地以降の命令が実行されることはありません。

アドレス	命令記号
00	TIA
01	<1>
02	CH
03	TIA
04	<0>
05	AO
06	CH
07	JUMP
08	<0>
09	<5>

(フローチャート)



プログラムの実行を05番地に戻し、くり返し実行させる。

プログラムが実行されるようすを、プログラムの命令記号と比較しながら、図や式を使って表わすと、上の図になります。

この図をフローチャートといい、プログラムの流れ(実行順序)がよくわかるように図にしたものです。このフローチャートにしたがって、プログラムを書きます。

No. 10 KA命令によるキー入力の数字LED表示

数字キー から を押したとき、数字キーの値（0からF）をレジスタに入れる命令は、このFX-マイコンでは、KA命令だけです。

この実験は、KA命令で、数字キー から、 の内の一つのキーが押されたとき、レジスタに押された数字キーの値を入れ、数字キーが押されないときは、KA命令の次のJUMP命令で00番地にプログラムに戻して、数字キーが押されるのを待ちます。数字キーを押すと、KA命令の次のJUMP命令は実行されず、その次のAO命令で、レジスタに入った数字キーの値を数字LEDに表示します。

●プログラム

アドレス	命令記号	命令コード	
00	KA	0	} KA命令（数字キーの読み込み命令）の後、数字キーを押さないと、00番地に戻ってキーが押されるのを待つ。
01	JUMP	F	
02	<0>	0	
03	<0>	0	} 数字LEDにArの値が表示される。
04	AO	1	
05	JUMP	F	} 次に数字キーを押した時、その時の数字が表示できるように、再び00番地へ戻る。
06	<0>	0	
07	<0>	0	

Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ プログラムの実行

書き込んだプログラムを、次の順序で実行してください。

押すキー



全命令を実行 (RUNモード)

キーを押した時、2進LEDのアドレス表示が01番地を表示していますが、これは、数字キーが押されるまで01番地から04番地へは進まず、00番地に戻って、キー入力を待っているのです。

※ この時2進LEDは、00番地と01番地を交互に表示しているのですが、非常に早い速度で点滅をくり返しているため、私達の目には、普通より少し明るさが落ちる程度で、ずっと01番地を表示しているように見えるのです。

次に、表示させたい数字のキー（例えば“6”）を押します。

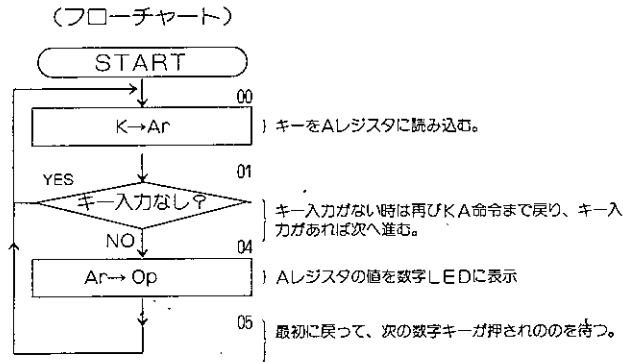


キー入力があると、00番地に戻らずに、次の命令（04番地）に進み、キーを押した数が、数字LEDに表示されます。表示されると、再び最初（00番地）に戻って、次のキー入力を待ちます。キーを離すと2進LEDは、01番地を示します。（※印 参照）

他の数字キーを押してみてください。キーを押すたびにその数を表示させることができます。

命令記号

KA
JUMP
<0>
<0>
AO
JUMP
<0>
<0>



キー入力があると、00番地に戻らずに次の命令(04番地)に進み、キーを押した数が数字LEDに表示されます。表示されると、再び最初(00番地)に戻って、次のキー入力を待ちます。キーを離すと、2進LEDは01番地を示します。(※印 参照)

他の数字キーを押してみてください。キーを押すたびにその数を表示させることができます。

● KA命令について

KA命令は、数字キーの値をレジスタに入れる命令です。

キーが押されていないときには、キーが押されていないことを知らせる信号(実行フラグ)を1にして、次のJUMP命令を実行させて、キー入力をさせるようにします。

キーが押されたときに、数字キーの値をレジスタに入れると共に、キーが押されたことを知らせる。実行フラグを0にして、次のJUMP命令は実行させず、その次の命令の実行に移ります。

● JUMP命令について(その2)

KA命令を実行すると、実行フラグが1か0になっています。実行フラグが1の場合は、JUMP命令を実行します。実行フラグが0の場合は、JUMP命令を実行せずに次の命令に移ります。

No10の場合、キー入力がないときは実行フラグは1となりJUMP命令を実行してキー入力を待つことになります。

キー入力があると、実行フラグが0となり、JUMP命令は実行せず04番地に進みます。05番地のJUMP命令は、前のAO命令で実行フラグが1になるから、いつもJUMPします。

実行フラグが1か0になる命令は、KA、M+、M-、AIA、AIY、CIA、CIYの7命令です。その他の命令はすべて、実行フラグが必ず1となり無条件でJUMP命令を実行します。

● STEPモードとRUNモードについて(その1)

No8とNo9の実験では、プログラムの実行を、

RESET STEP RUN キー.....

と、次々に押して命令を実行させ、(これをSTEPモードという)

No10の実験では、プログラムの実行を、

RESET STEP RUN キーを押して命令を実行させました(これを

RUNモードという)

STEPモードは、STEPキーを押したとき、まず最初の1命令が実行され、その次にSTEPキーを1回押すごとに、1命令づつ命令を実行していくので、STEPモードといえます。

RUNモードは、RUNキーを押すと、プログラムに書かれている命令を、速いスピードで、すべての命令をつぎつぎに実行して行きます。

STEPモードとRUNモードは、STEPキーの前に押す数字キーによって異なります。STEPキーはSTEPモードを、RUNキーはRUNモードを指定します。プログラムによって、STEPモードとRUNモードを選んで実行させます。(次の実験のあとで詳しく説明します。)

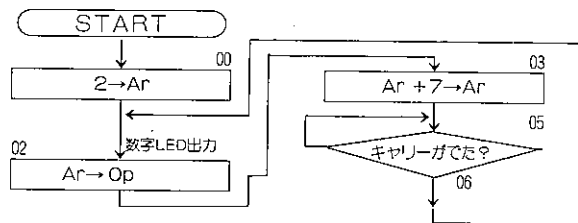
No. 11 AIA命令による数字の加算表示

Aレジスタは、コンピュータの中で、いろいろな働きをする、重要なレジスタです。AIA命令は、Aレジスタに一定の数を加え、Aレジスタの内容を変化させる命令です。この命令は実行フラグを1か0にする命令ですから、JUMP命令のはたらき（その2、27P）の説明をふりがえりながら実験を進めてください。

●プログラム

アドレス	命令記号	命令コード	
00	TIA	8	} Aレジスタに2が入る。
01	<2>	2	
02	AO	1	…Aレジスタの値が表示される。
03	AIA	9	} Aレジスタの値に7を加えた数が Aレジスタに入る
04	<7>	7	
05	JUMP	F	} キャリーが出た時（Fをこえた時） は、プログラムが停止した状態になる。
06	<0>	0	
07	<5>	5	
08	JUMP	F	} キャリーが出ない時（0～Fの間） は、02番地に戻って、表示、加算 がくり返される。
09	<0>	0	
0A	<2>	2	

(フローチャート)



Ⓐ プログラムの書き込み

Ⓑ プログラムの確認と訂正。

Ⓒ プログラムの実行。

書き込んだプログラムを、次の操作で実行させます。

押すキー



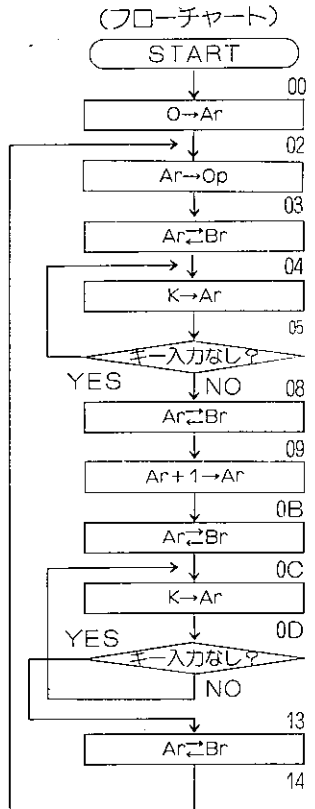
No. 12 16進数を小さい順にくり返し表示する実験

No.8 ~ No.11のプログラムは、第1群の命令語のはたらきを知るための、テスト・プログラムでした。

しかし、これからのプログラムは、ある目的を、どのようなプログラムで実行させるかが、テーマとなります。

●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<0>	0
02	AO	1
03	CH	2
04	KA	0
05	JUMP	F
06	<0>	0
07	<4>	4
08	CH	2
09	AIA	9
0A	<1>	1
0B	CH	2
0C	KA	0
0D	JUMP	F
0E	<1>	1
0F	<3>	3
10	JUMP	F
11	<0>	0
12	<C>	C
13	CH	2
14	JUMP	F
15	<0>	0
16	<2>	2



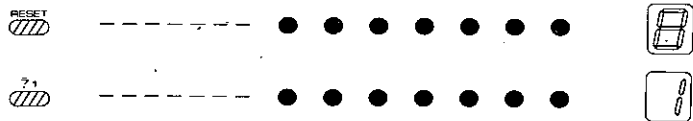
*フローチャートの右肩の数(00等)は、アドレスを表わしている。

- ① 初めは“0”を表示し、14番地の命令、「JUMP02」で、A0命令に戻ったときは、Aレジスタに1を加えた数が、次々に表示される。
- ② キーが押されたことだけを確かめ、Aレジスタの内容が、KA命令で変わらないように、キー入力の前にBレジスタに一時しまっておき、キー入力を確認した後、Bレジスタから再びAレジスタに戻しています。
また、キー入力がないときは、KA命令に戻って、キー入力があるまでKA命令とJUMP命令をくり返し実行し、次のアドレス08に進まない。
- ③ キー入力があると、Aレジスタの内容に1を加え、次のキー入力チェックで、Aレジスタの内容が変わらないようにBレジスタに一時しまっておく。
- ④ ここでは、キーを押し続けていると次の命令に進まず、キーを離したとき次の命令に進むようにしています。(この命令によって、キーを押して離したときに、数字LEDの数字が1つずつ進んで、表示される)
- ⑤ キーを押すのをやめたとき、Bレジスタの内容(③で、Aレジスタに1を加えた数)をAレジスタに戻して、02番地まで戻って、その値を表示させるプログラムに進む。

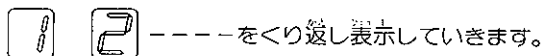
Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ 命令を次のように実行させてください。

押すキー



RESET キーを押すと、数字LEDは、0を表示し、次に | - | までのキー（どのキーでもよい）を押して、離すたびに、数字LEDは、1つつ加わって表示されます。



Ⓒ 初めてプログラムを勉強される皆さんには、だいぶむづかしいプログラムですが、数字キーを押すたびに、数字LEDにいろいろな数字を表示する一つの型として覚えてください。

一つのプログラムの形を覚えると、そのプログラムを少しずつ変更することにより、目的の似ている違ったプログラムを作ることができます。

プログラムの勉強は、いろいろな目的に応じたプログラムの形をより多く覚えて、自分の目的とするプログラムが作れるようになることです。

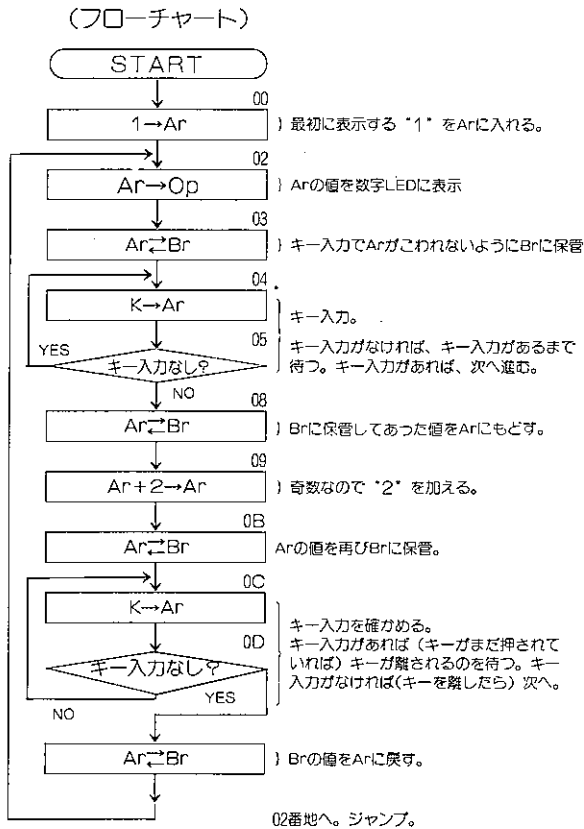
№13から№17の実験は、そのテクニックの勉強です。

No. 13 16進数の奇数を小さい順にくりかえし表示



●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<1>	1
02	A0	1
03	CH	2
04	KA	0
05	JUMP	F
06	<0>	0
07	<4>	4
08	CH	2
09	AIA	9
0A	<2>	2
0B	CH	2
0C	KA	0
0D	JUMP	F
0E	<1>	1
0F	<3>	3
10	JUMP	F
11	<0>	0
12	<C>	C
13	CH	2
14	JUMP	F
15	<0>	0
16	<2>	2



Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ 命令の実行。

RESET 1000 RUN キーを順に押してください。

1000 キーを押したとき、1000を表示し、次に数字キーを押して、離すたびに、



---と、奇数をくり返し表示します。

このプログラムやフローチャートと、No.12のプログラムやフローチャートを見くらべてください。違っているところは、01番地のTIA命令の値が、0であったのが、1に、0A番地のAIA命令の値が、1から2に変わっただけです。下の図を見て考えてください。



差 --- 1 1 1 1 ----- 1 1



差 --- 2 2 2 2 ----- 2 2

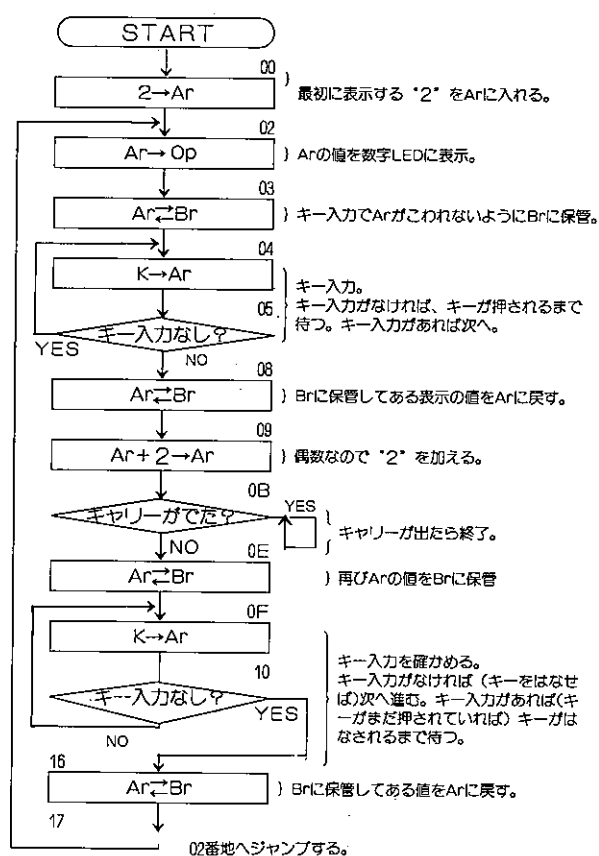
No. 14 16進数の偶数を小さい順に一回だけ表示



●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<2>	2
02	AO	1
03	CH	2
04	KA	0
05	JUMP	F
06	<0>	0
07	<4>	4
08	CH	2
09	AIA	9
0A	<2>	2
0B	JUMP	F
0C	<0>	0
0D		B
0E	CH	2
0F	KA	0
10	JUMP	F
11	<1>	1
12	<6>	6
13	JUMP	F
14	<0>	0
15	<F>	F
16	CH	2
17	JUMP	F
18	<0>	0
19	<2>	2

(フローチャート)



No. 12 と No. 13 の実験を理解して実行された皆さんは、左のプログラムとフローチャートを見れば、そのはたらきやプログラムが変わったところが、すぐにわかりますね。

- Ⓐ プログラムの書き込みと、確認をしてください。
- Ⓑ 命令の実行。 キーを順に押してください。

キーを押したとき 2 を表示し、次に数字キーを押して、離すたびに、偶数表示をします。1 回のみ表示しますので、E の表示の後でストップします。



して、E のあとでストップします。

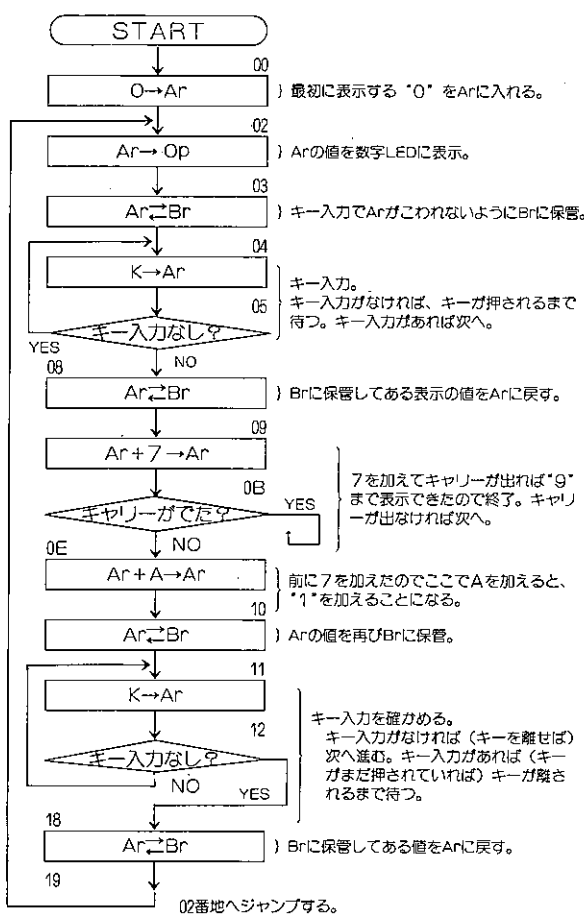
No. 15 10進数を小さい順に一回だけ表示



●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<0>	0
02	AO	1
03	CH	2
04	KA	0
05	JUMP	F
06	<0>	0
07	<4>	4
08	CH	2
09	AIA	9
0A	<7>	7
0B	JUMP	F
0C	<0>	0
0D		B
0E	AIA	9
0F	<A>	A
10	CH	2
11	KA	0
12	JUMP	F
13	<1>	1
14	<8>	8
15	JUMP	F
16	<1>	1
17	<1>	1
18	CH	2
19	JUMP	F
1A	<0>	0
1B	<2>	2

(フロチャート)



- Ⓐ プログラムの書き込みと、確認をしてください。
- Ⓑ 命令の実行。RESET キーの順に押してください。

RUN キーを押したとき0を表示し、次から数字キーを押すと+1して、いきます。表示が9になったらストップします。

Ar+7 でArが9になったかチェックをしています。次の Ar+A は Ar+7 と、Ar+A で、Ar+1 とするためにつけてあります。

〈例〉

$$Ar=5 \text{ --- } 0101 \text{ --- } 0101$$

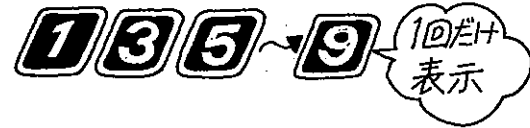
$$\begin{array}{cccc} & & & 8421 \\ & & & \vdots \\ & & & 0101 \\ & & & \vdots \\ & & & 41 \\ & & & 5 \end{array}$$

$$\begin{array}{r} Ar+7 \quad 0101 \\ +) 0111 \\ \hline 1100 = Ar \end{array}$$

$$\begin{array}{r} Ar+A \quad 1100 \\ +) 1010 \\ \hline 0110 = Ar \end{array}$$

という計算で、Arは6となり5+1が計算されたこととなります。

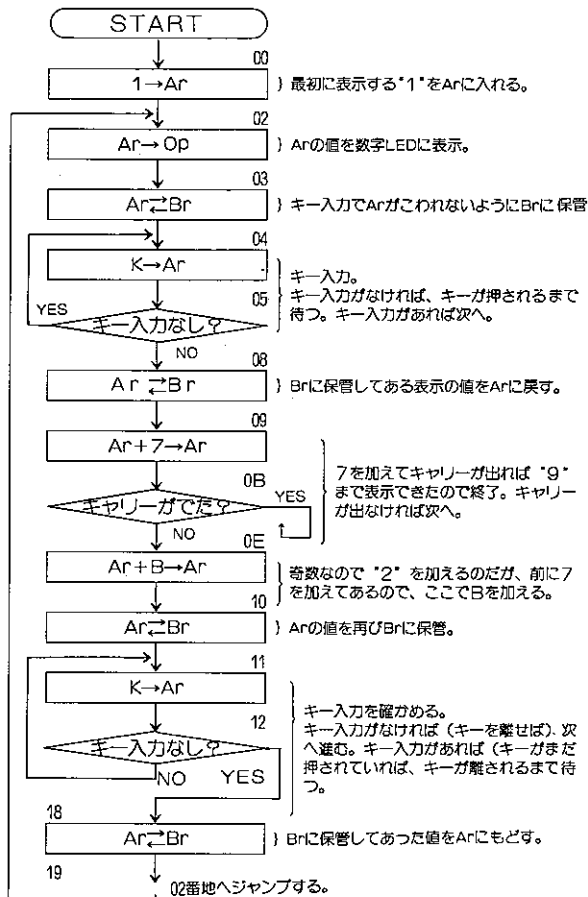
No. 16 10進数の奇数を小さい順に一回だけ表示



●プログラム

アドレス	命令記号	命令コード
00	T I A	8
01	<1>	1
02	AO	1
03	CH	2
04	KA	0
05	JUMP	F
06	<0>	0
07	<4>	4
08	CH	2
09	A I A	9
0A	<7>	7
0B	JUMP	F
0C	<0>	0
0D		B
0E	A I A	9
0F		B
10	CH	2
11	KA	0
12	JUMP	F
13	<1>	1
14	<8>	8
15	JUMP	F
16	<1>	1
17	<1>	1
18	CH	2
19	JUMP	F
1A	<0>	0
1B	<2>	2

(フローチャート)



Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ 命令の実行。RESET キーの順に押してください。

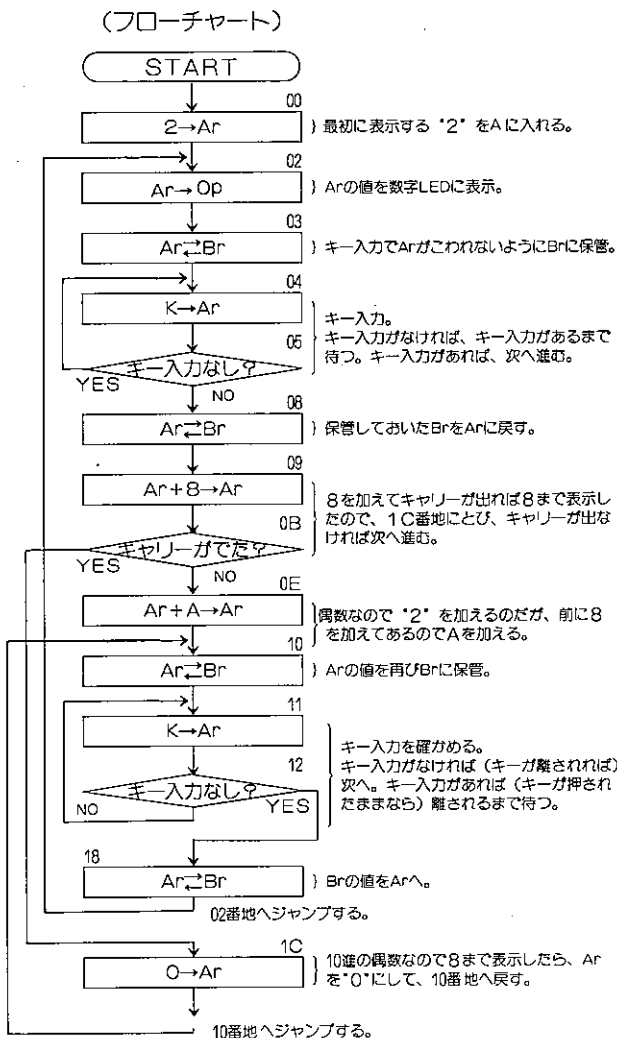
10進数の奇数表示です。今まで使ったテクニックの組合せですから、フローチャートのみでわかるはず。考えてみましょう。

RUN キーを押したとき1を表示し、次から数字キーを押すと+2していきます。表示が9になったらストップします。

No. 17 10進数の偶数を小さい順にくりかえし表示

●プログラム

アドレス	命令記号	命令コード
00	T I A	8
01	<2>	2
02	A O	1
03	CH	2
04	KA	0
05	JUMP	F
06	<0>	0
07	<4>	4
08	CH	2
09	A I A	9
0A	<8>	8
0B	JUMP	F
0C	<1>	1
0D	<C>	C
0E	A I A	9
0F	<A>	A
10	CH	2
11	KA	0
12	JUMP	F
13	<1>	1
14	<8>	8
15	JUMP	F
16	<1>	1
17	<1>	1
18	CH	2
19	JUMP	F
1A	<0>	0
1B	<2>	2
1C	T I A	8
1D	<0>	0
1E	JUMP	F
1F	<1>	1
20	<0>	0



Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ 命令の実行。RESET キーの順に押し続けてください。

10進数の偶数表示です。[Ar+8] でくり返しチェックをしています。

その次の [Ar+A] の組合せで [Ar+2] のかわりをしています。

No.14とNo.15の組合せです。わからない場合は、No.14や、15をもう一度やり直しましょう。

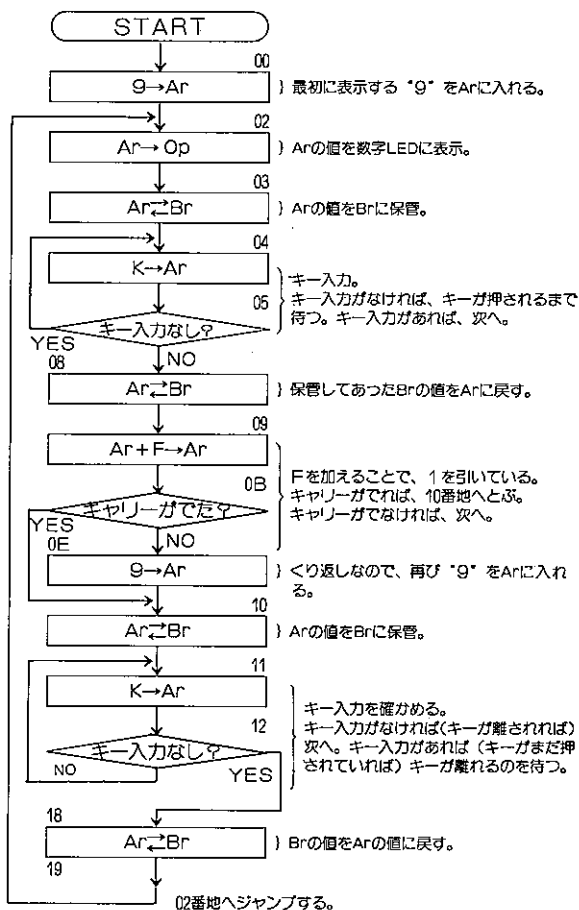
RUN キーを押したとき 2 を表示し、次から数字キーを押すと +2 していき、偶数をくりかえし表示します。

No. 18 10進数を大きい順にくりかえし表示

●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<9>	9
02	AO	1
03	CH	2
04	KA	0
05	JUMP	F
06	<0>	0
07	<4>	4
08	CH	2
09	AIA	9
0A	<F>	F
0B	JUMP	F
0C	<1>	1
0D	<0>	0
0E	TIA	8
0F	<9>	9
10	CH	2
11	KA	0
12	JUMP	F
13	<1>	1
14	<8>	8
15	JUMP	F
16	<1>	1
17	<1>	1
18	CH	2
19	JUMP	F
1A	<0>	0
1B	<2>	2

(フローチャート)



(A) プログラムの書き込みと、確認をしてください。

(B) 命令の実行、 キーの順に押してください。

これは、10進数9~0までの表示で、最初の数字LEDの表示と、繰返し表示のテクニックです。

を使い Ar が 0 のとき

でキャリーは出ませんので、この時 Ar を 9 にしてやればくりかえし表示します。

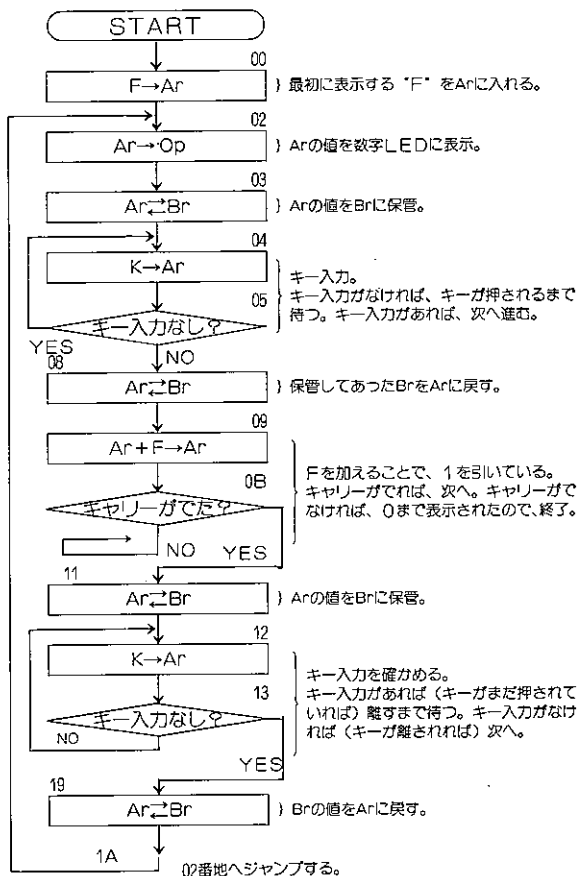
キーを押したとき 9 を表示し、次から数字キーを押すと -1 していき、9~0 までをくりかえし表示します。

No. 19 16進数を大きい順に一回だけ表示

●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<F>	F
02	AO	1
03	CH	2
04	KA	0
05	JUMP	F
06	<0>	0
07	<4>	4
08	CH	2
09	AIA	9
0A	<F>	F
0B	JUMP	F
0C	<1>	1
0D	<1>	1
0E	JUMP	F
0F	<0>	0
10	<E>	E
11	CH	2
12	KA	0
13	JUMP	F
14	<1>	1
15	<9>	9
16	JUMP	F
17	<1>	1
18	<2>	2
19	CH	2
1A	JUMP	F
1B	<0>	0
1C	<2>	2

(フローチャート)



左のプログラムやフローチャートとNo12のプログラムやフローチャートを見くらべて下さい。違っているところは、まず、01番地のTIA命令の値が0であったのが、Fに、0A番地のAIA命令の値が1であったのが、Fに変わっています。これは、No12が、キーを押すごとに表示される数が、1ずつ増加されるのに対し、No19は、数が1ずつ減少するので、Fを加えることによって、簡潔的に1を引いていることとなります。

これを式で表してみると、

$$\begin{array}{r}
 \text{(2進数)} \quad \text{(16進数)} \\
 F + F \longrightarrow 1111 \dots\dots F \\
 +) 1111 \dots\dots F \\
 \hline
 \text{キャリー} \longrightarrow 11110 \dots\dots E (F-1) \\
 \\
 E + F \longrightarrow 1110 \dots\dots E \\
 | \quad +) 1111 \dots\dots F \\
 | \quad \hline
 | \quad 11101 \dots\dots D (E-1) \\
 | \quad | \quad | \\
 | \quad | \quad | \quad \text{となります。}
 \end{array}$$




このように、Fを加えた答は、Fを加える前のArの値が、0以外はすべてキャリーがでます。09番地から10番地までのJUMP命令は、Arが0のときだけキャリーがでません。


$$\begin{array}{r}
 0 + F \longrightarrow 0000 \dots\dots 0 \\
 +) 1111 \dots\dots F \\
 \hline
 \bigcirc 1111 \dots\dots F \\
 \text{キャリーはでません。}
 \end{array}$$

そして、No12と違い、ここでは、AIA命令と、CH命令の間に、このJUMP命令を入れることによって、0まで表示したら、そこでストップするようにしています。

Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ 命令の実行

   キーを順に押してください。

 キーを押したとき **F** を表示し、次に、数字キーを押して、離すたびに、

E D C B A 9 8 7 6 5

4 3 2 1 0 ---と、1つつがさく表示


し、0でとまります。

(このあとは、どの数字キーを押しても0しか表示しません。)

No12～No19までの実験を理解するために、一番大切なことは、No12のフローチャートの右に書いてある説明の②で、キー入力の前にCH命令を使って、AレジスタをBレジスタに保管し、キー入力後に、再びBレジスタから、Aレジスタに内容を戻すことです。

その後、説明③と④で、A I A命令で、Aレジスタに一定の数字を加えたあと、CH命令で、またAレジスタの内容をBレジスタへ保管して、キーが離されたことをチェックし、キーが押しつづけている間は、次のプログラムへ進まないようにするという、プログラム・テクニックです。

特に、プログラムの警い初めは、この④のプログラム・テクニックをうっかり忘れて、④のプログラムを省略して、0C番地のKA命令のところで、JUMP 02 としがちです。

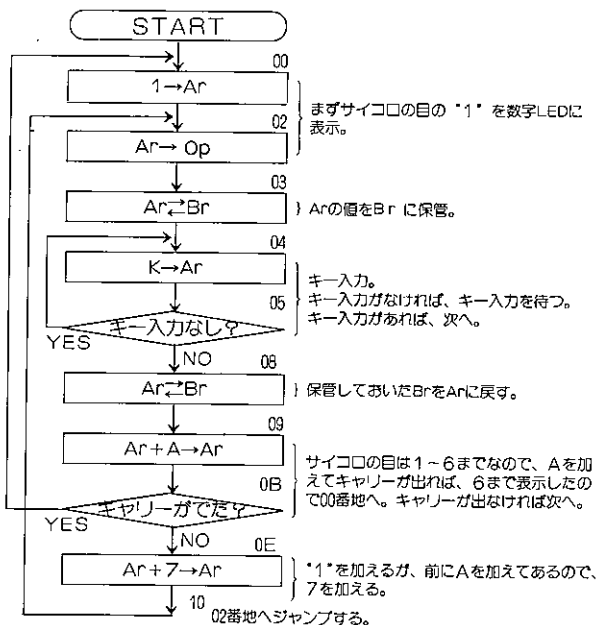
このようなプログラム・ミスをする、数字キーを一回押ただけで、数字LEDは、0、1、2、3、4………F、0、1、2………を早い速度で表示してしまい、数字LEDは、 を点灯しているように見え、数字キーをはなしたときに止まりますが、0、1、2、3、4………Fのうちの一つを表示してしまいます。

No. 20 数字キーを離すと止まる電子サイコロ

●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<1>	1
02	AO	1
03	CH	2
04	KA	0
05	JUMP	F
06	<0>	0
07	<4>	4
08	CH	2
09	AIA	9
0A	<A>	A
0B	JUMP	F
0C	<0>	0
0D	<0>	0
0E	AIA	9
0F	<7>	7
10	JUMP	F
11	<0>	0
12	<2>	2

(フロチャート)



(A) プログラムの書き込みと、確認をしてください。

(B) キーの順に押してください。

キーを押した後、数字LEDは1を表示していますね。キーを押してみましよう。表示が早く変わります。キーをはなせば表示はとまりません。

09番地の $Ar+A$ で、6をこえたかどうかチェックしています。6をこえていないばあい、 $Ar+7$ で1づつふやして表示しています。

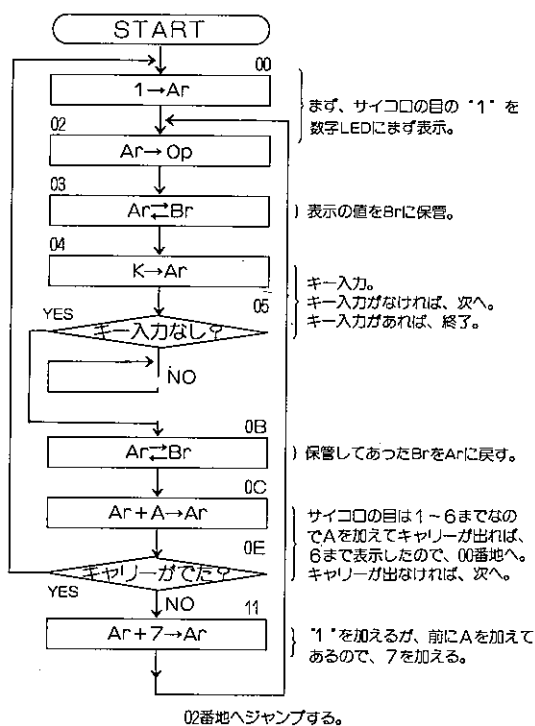


No. 21 数字キーを押すと止まる電子サイコロ

●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<1>	1
02	AO	1
03	CH	2
04	KA	0
05	JUMP	F
06	<0>	0
07		B
08	JUMP	F
09	<0>	0
0A	<8>	8
0B	CH	2
0C	AIA	9
0D	<A>	A
0E	JUMP	F
0F	<0>	0
10	<0>	0
11	AIA	9
12	<7>	7
13	JUMP	F
14	<0>	0
15	<2>	2

(フロチャート)



まず、サイコロの目の「1」を数字LEDにまず表示。

表示の値をBrに保存。

キー入力。
キー入力がない場合は、次へ。
キー入力があれば、終了。

保管してあったBrをArに戻す。

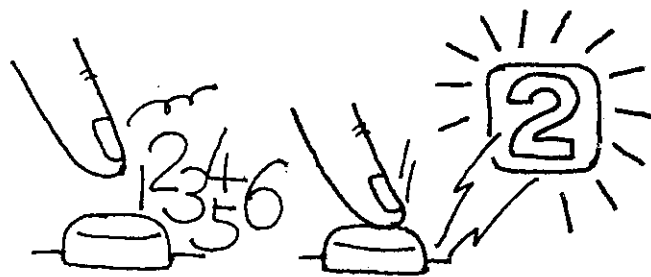
サイコロの目は1～6までなのでAを加えてキャリーが出れば、6まで表示したので、00番地へ。キャリーが出なければ、次へ。

「1」を加えるが、前にAを加えているので、7を加える。

Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ RESET 71 RUN キーの順に押してください。

RUN キーを押した後、表示は早く変わります。ここで、RESET ~ RUN のキーを押してください。表示はストップします。



(2) 第2群の命令について

第2群の命令は、AM、MA、M+、M-、TIY、AIY、CAL TIMR命令の7種類の命令グループです。この第2群の命令グループは、Yレジスタとデータメモリを主に使います。

Yレジスタは、次の表のように、その値をきめることにより、データメモリのアドレスも指定できます。本文中、メモリと書かれているのはこのデータメモリのことです。

Yレジスタの値	データメモリのアドレス	Yレジスタの値	データメモリのアドレス
0	50	8	58
1	51	9	59
2	52	A	5A
3	53	B	5B
4	54	C	5C
5	55	D	5D
6	56	E	5E
7	57	F	5F

上のアドレスは、プログラムを書くときのアドレスと同じで、特に、50～5F番地を、データメモリとして割り当て、プログラムでデータを書き込んだり、順序を並べかえたり、読み出ししたりするときに使えます。

Yレジスタは、上のデータメモリのアドレス指定のほかに、Aレジスタとは別の機能を持ったレジスタとしても使われます。

※ CAL 命令について (その1)

CAL TIMR (コールタイマー) 命令のCAL (コール) は、あらかじめ用意されているサービスプログラムをコール (呼び出す) して、そのプログラムを実行してから、CAL 命令の次の命令に移ります。

CAL 命令は、このCAL TIMR命令を含めて、次の16種類のサービスプログラムが用意されています。

命令コード		命令記号	命令コード		命令記号
1行目	2行目		1行目	2行目	
E	0	CAL RSTO	E	8	CAL ERRS
E	1	CAL SETR	E	9	CAL SHTS
E	2	CAL RSTR	E	A	CAL LONS
E	3	CAL INPT	E	B	CAL SUND
E	4	CAL CMPL	E	C	CAL TIMR
E	5	CAL CHNG	E	D	CAL DSPR
E	6	CAL SIFT	E	E	CAL DEM-
E	7	CAL ENDS	E	F	CAL DEM+

JUMP 命令と同じようにCAL 命令は実行フラグが1のとき実行され、0のとき実行されません。

※ CH命令について (その2)

第1群の命令でCH命令の説明をしました。わすれた人はもう一度、CH命令の説明を読み返してみてください。AレジスタとBレジスタを交換する命令ですね。実はCH命令には、もう一つ働きがあります。YレジスタとZレジスタの交換をします。

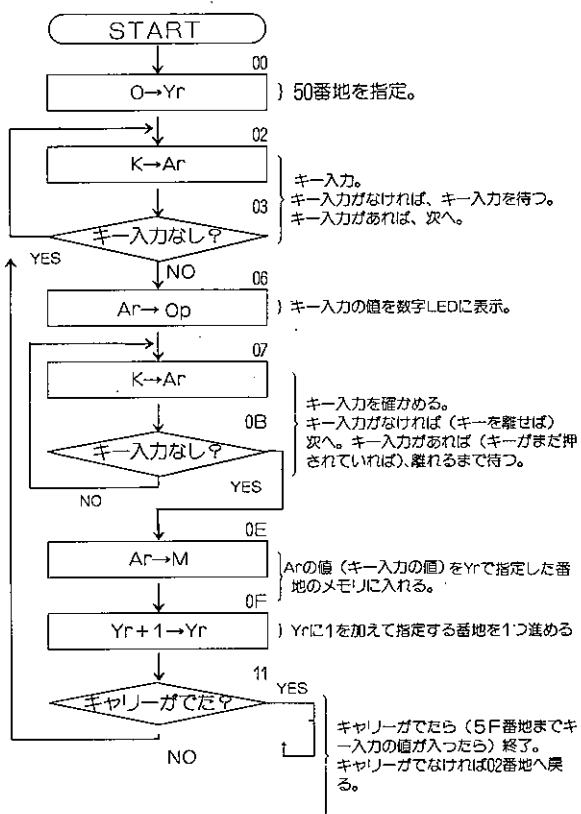
Aレジスタ ↔ Bレジスタ 同時に変わるので注意
Yレジスタ ↔ Zレジスタ しましょう

No. 22 TIY, AIY, AM命令を使った実験

●プログラム

アドレス	命令記号	命令コード
00	TIY	A
01	<0>	0
02	KA	0
03	JUMP	F
04	<0>	0
05	<2>	2
06	AO	1
07	KA	0
08	JUMP	F
09	<0>	0
0A	<E>	E
0B	JUMP	F
0C	<0>	0
0D	<7>	7
0E	AM	4
0F	AIY	B
10	<1>	1
11	JUMP	F
12	<1>	1
13	<1>	1
14	JUMP	F
15	<0>	0
16	<2>	2

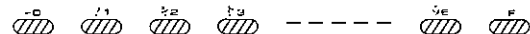
(フローチャート)



◆このプログラムは、キー入力後のJUMP命令を二通り使っています。一つは、キーが押されるまで待つJUMPでもう一つは、キーがはなれるまで待つJUMPがあります。二通りのJUMPを使うことによって、一つづつ、二重押しのないキー入力ができます。16回の数字キー入力をくり返すことによって、このプログラムは終了します。

- Ⓐ プログラムの書き込みと、確認をしてください。
- Ⓑ RESET、、、 キーを順に押してください。
- Ⓒ キー入力を、16回 (0~F) 押しします。

〈例〉メモリに書き込みたい値 (0~F) キーを押す。



- Ⓓ メモリの値を50番地から読み出してみましょう。

	キーを押してください	● ● ● ● ● ● ● ●	
	"	● ● ● ● ● ● ● ●	
	"	● ● ● ● ● ● ● ●	
	"	● ● ● ● ● ● ● ●	
	"	● ● ● ● ● ● ● ●	
	"	● ● ● ● ● ● ● ●	
	"	● ● ● ● ● ● ● ●	
	"	● ● ● ● ● ● ● ●	

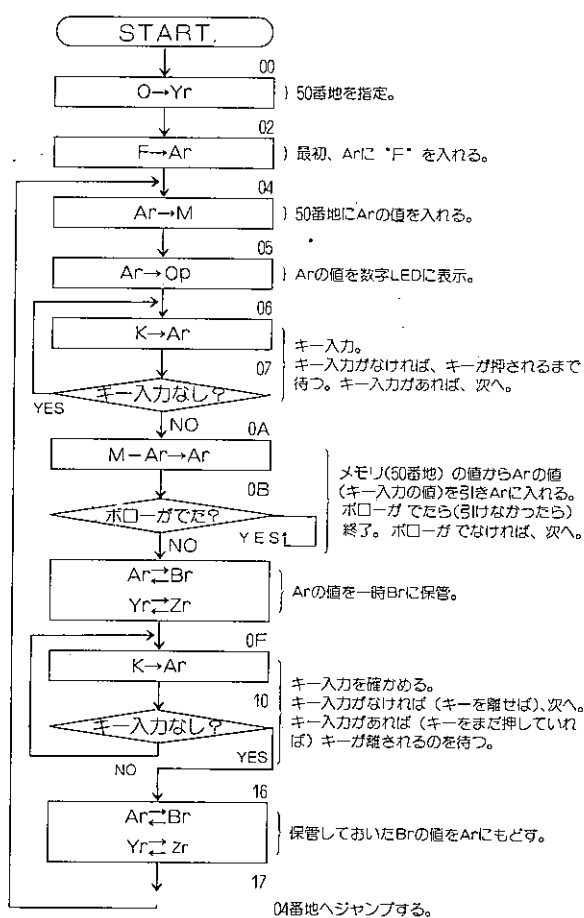
で前ページ表のように50番地を指定し、数字キーの値をストアします。 で一つづつYレジスタの値をふやしているため、50~5F番地まで数字キーの値がストアされます。

No. 25 M-命令によるキー入力の数を引きいた実験

●プログラム

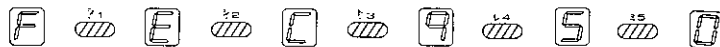
アドレス	命令記号	命令コード
00	TIY	A
01	<0>	0
02	TIA	8
03	<F>	F
04	AM	4
05	AO	1
06	KA	0
07	JUMP	F
08	<0>	0
09	<6>	6
0A	M-	7
0B	JUMP	F
0C	<0>	0
0D		B
0E	CH	2
0F	KA	0
10	JUMP	F
11	<1>	1
12	<6>	6
13	JUMP	F
14	<0>	0
15	<F>	F
16	CH	2
17	JUMP	F
18	<0>	0
19	<4>	4

(フローチャート)



●No.24は、M+を使って加算をしましたが、この実験では、M-を使って減算(ひき算)をしましょう。

- Ⓐ プログラムの書き込みと、確認をしてください。
- Ⓑ キーを順に押し実行します。
- Ⓒ 初めはFを表示しますが、0~Fのキーを押すことによって、表示データはキー入力の値で、減算されます。キー入力の値をいろいろ変えて実験してみてください。0まで行くと終わります。



※ M-命令について

フローチャートでは、M-Ar→Arと表現してあります。メモリの値からAレジスタの値をひいて、その答をAレジスタに入れる命令です。メモリよりもAレジスタの方が大きい場合は引くことができません。この引けない時をポロ- (ケタ下げ) といいます。

この時の実行フラグは、

ポロ-の出る場合 → 実行フラグ1

ポロ-の出ない場合 → 実行フラグ0

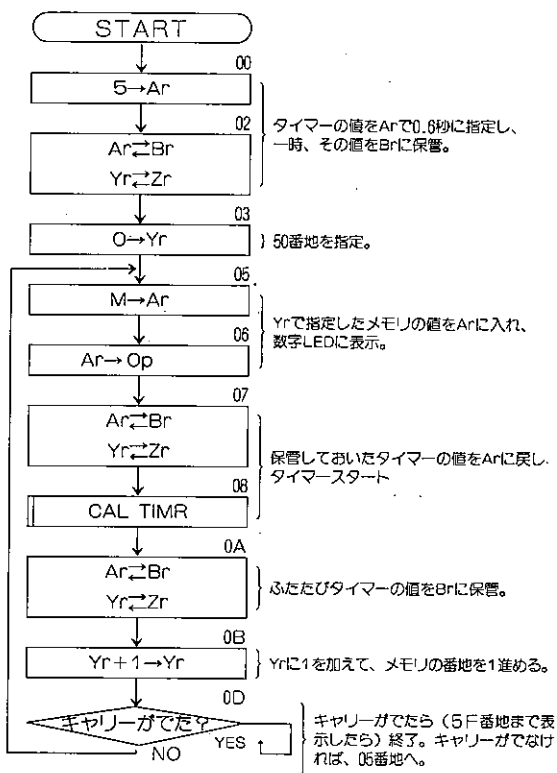
となります。この命令も、M+命令と同じようにJUMP命令やCAL命令に影響します。

No. 26 CAL TIMR^{めいれい}命令を使った実験^{つか} ^{じっけん}

●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<5>	5
02	CH	2
03	TIY	A
04	<0>	0
05	MA	5
06	AO	1
07	CH	2
08	[CAL	E
09	TIMR	C
0A	CH	2
0B	AIY	B
0C	<1>	1
0D	JUMP	F
0E	<0>	0
0F	<D>	D
10	JUMP	F
11	<0>	0
12	<5>	5

(フローチャート)



Ⓐ プログラムの書き込みと、確認をしてください。

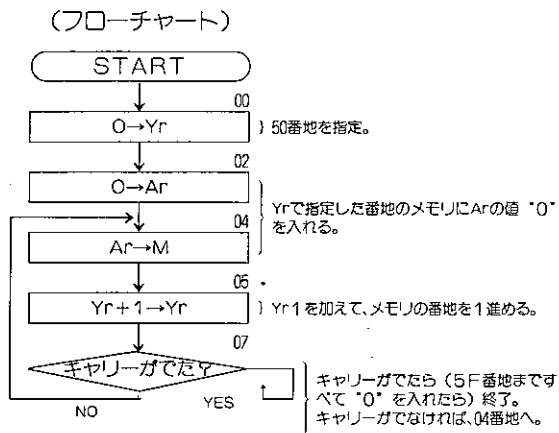
Ⓑ メモリ50から5Fまでに、0~Fを書き込みます。

〈例〉

No. 27 メモリにオール0をストア(しまいこむ)実験 じっけん

●プログラム

アドレス	命令記号	命令コード
00	T I Y	A
01	<0>	0
02	T I A	8
03	<0>	0
04	A M	4
05	A I Y	B
06	<1>	1
07	JUMP	F
08	<0>	0
09	<7>	7
0A	JUMP	F
0B	<0>	0
0C	<4>	4



メモリの50~5Fのすべてに "0" を書き込むプログラムです。

- Ⓐ プログラムの書き込みと、確認をしてください。
- Ⓑ キーの順に押してください。
- Ⓒ メモリに書き込まれているか、チェックしてみましょう。

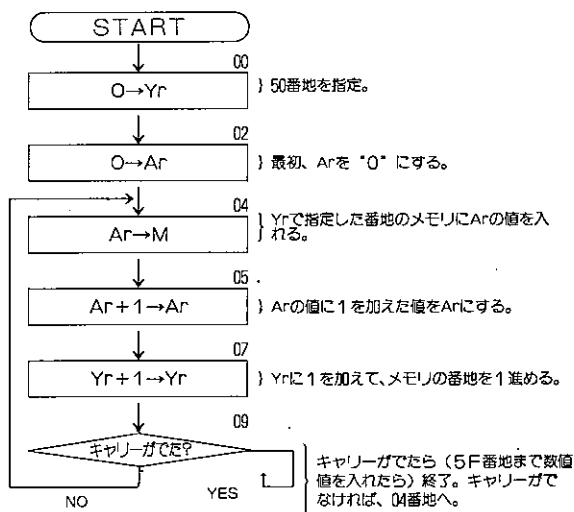
	キーを押してください	● ● ● ● ● ● ● ●	
	キーを押してください	● ● ● ● ● ● ● ●	
	"	● ● ● ● ● ● ● ●	
	"	● ● ● ● ● ● ● ●	
	"	● ● ● ● ● ● ● ●	
⋮	⋮	⋮	⋮
	"	● ● ● ● ● ● ● ●	

No. 28 メモリに0~Fをストアする実験 じっけん

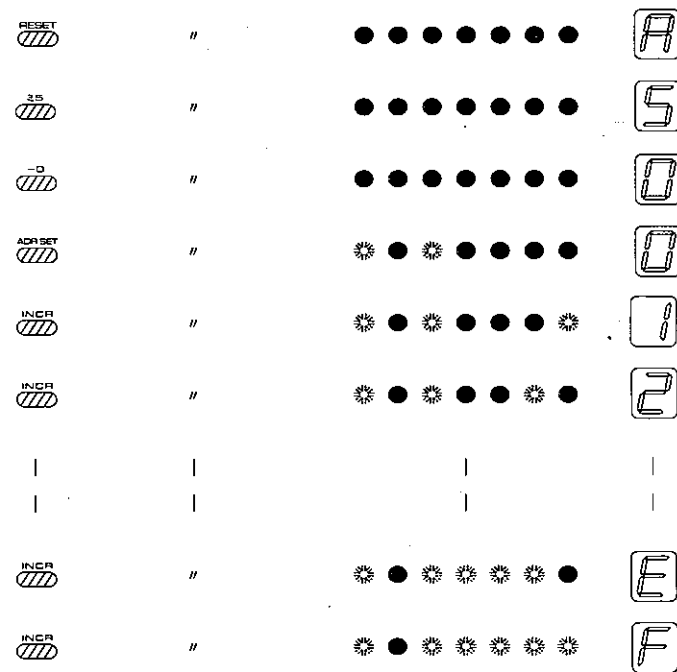
●プログラム

アドレス	命令記号	命令コード
00	T I Y	A
01	<0>	0
02	T I A	8
03	<0>	0
04	A M	4
05	A I A	9
06	<1>	1
07	A I Y	B
08	<1>	1
09	JUMP	F
0A	<0>	0
0B	<9>	9
0C	JUMP	F
0D	<0>	0
0E	<4>	4

(フローチャート)



押すキー。



●メモリ50~5Fに0~Fを順に書き込むプログラムです。

- (A) プログラムの書き込みと、確認をしてください。
- (B) キーの順に押してください。
- (C) メモリを読み出してみましょう。

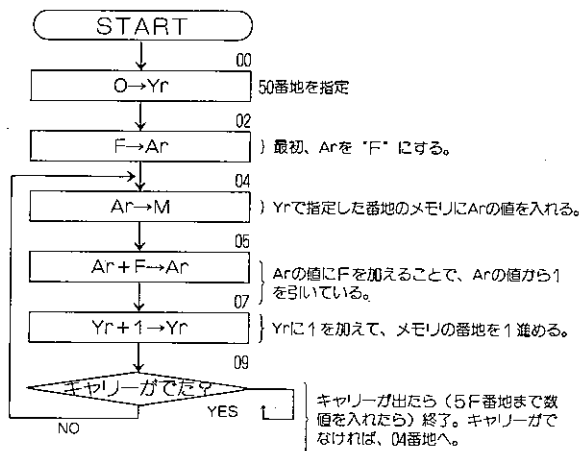
●No.27に、 が加わっています。これでメモリに書き込むデータを+1ずつ増していきます。

No. 29 メモリにF~0をストアする実験 じっけん

●プログラム

アドレス	命令記号	命令コード
00	TI Y	A
01	<0>	0
02	T I A	8
03	<F>	F
04	AM	4
05	A I A	9
06	<F>	F
07	A I Y	B
08	<1>	1
09	JUMP	F
0A	<0>	0
0B	<9>	9
0C	JUMP	F
0D	<0>	0
0E	<4>	4

(フローチャート)



押すキー

	"	● ● ● ● ● ● ● ● ● ●	
	"	● ● ● ● ● ● ● ● ● ●	
	"	● ● ● ● ● ● ● ● ● ●	
	"	● ● ● ● ● ● ● ● ● ●	
	"	● ● ● ● ● ● ● ● ● ●	
	"	● ● ● ● ● ● ● ● ● ●	
	"	● ● ● ● ● ● ● ● ● ●	

● メモリ50~5FにF~0を順に書き込むプログラムです。

(A) プログラムの書き込みと、確認をしてください。

(B) キーの順に押してください。

(C) メモリを読みだしてみましょう。

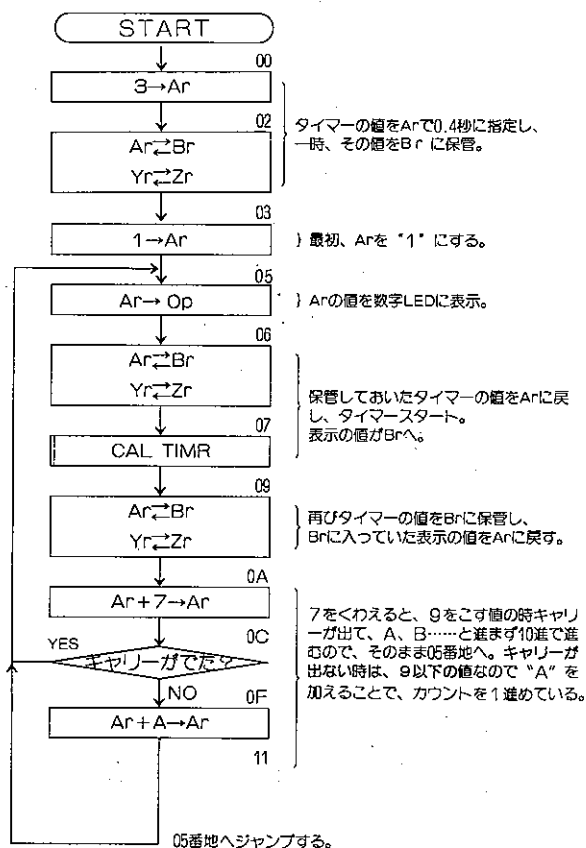
● で、メモリに書き込むデータを、Fから順に1つづつ小さくしていきます。

No. 30 ^{しんすう}10進数アップ・カウンタの実験 ^{じっけん}

●プログラム

アドレス	命令記号	命令コード
00	T I A	8
01	<3>	3
02	CH	2
03	T I A	8
04	<1>	1
05	AO	1
06	CH	2
07	CAL	E
08	TIMR	C
09	CH	2
0A	A I A	9
0B	<7>	7
0C	JUMP	F
0D	<0>	0
0E	<5>	5
0F	A I A	9
10	<A>	A
11	JUMP	F
12	<0>	0
13	<5>	5

(フローチャート)



● 表示は1、2———8、9、0、1と変化しくり返します。

Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ キーを順に押してください。

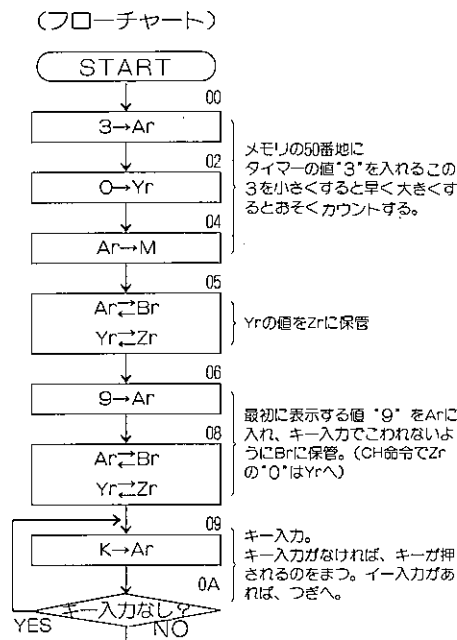
Ⓒ ---

No. 31 キーONで働く、10進ダウンカウンタの実験

●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<3>	3
02	TIY	A
03	<0>	0
04	AM	4
05	CH	2
06	TIA	8
07	<9>	9
08	CH	2
09	KA	0
0A	JUMP	F
0B	<0>	0
0C	<9>	9
0D	KA	0
0E	JUMP	F
0F	<1>	1
10	<4>	4
11	JUMP	F
12	<0>	0
13	<D>	D
14	CH	2
15	AO	1
16	CH	2
17	MA	5
18	CAL	E
19	TIMR	C
1A	AM	4
1B	CH	2
1C	AIA	9

アドレス	命令記号	命令コード
1D	<F>	F
1E	JUMP	F
1F	<1>	1
20	<5>	5
21	TIA	8
22	<9>	9
23	JUMP	F
24	<1>	1
25	<5>	5



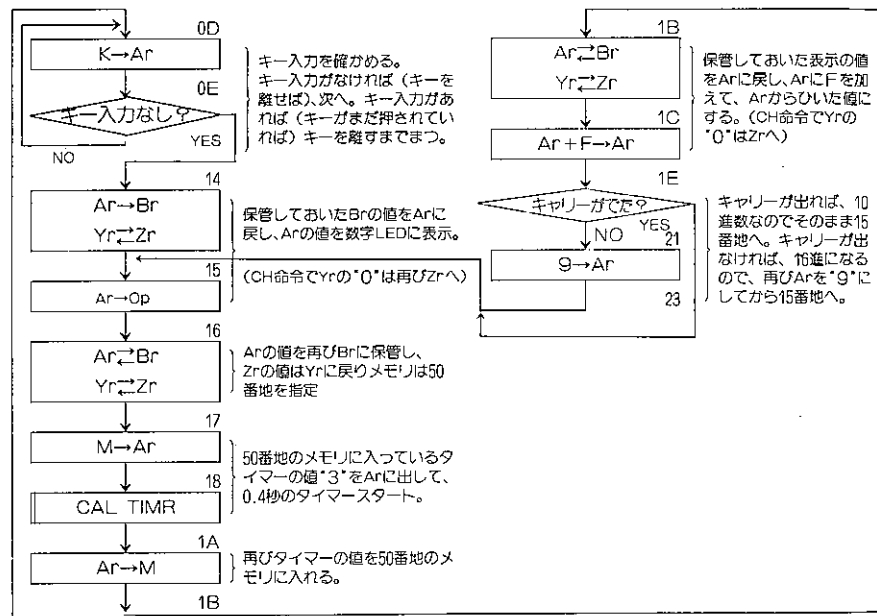
● キー入力をスタートボタンとして、ダウンカウンタを組んでみましょう。

① プログラムの書き込みと、確認をしてください。

② キーを順に押してください。

● キー入力 (0~Fのどのキーでもよい) を押して、離れたときから10進ダウンカウントが始まります。

③ ----- -----

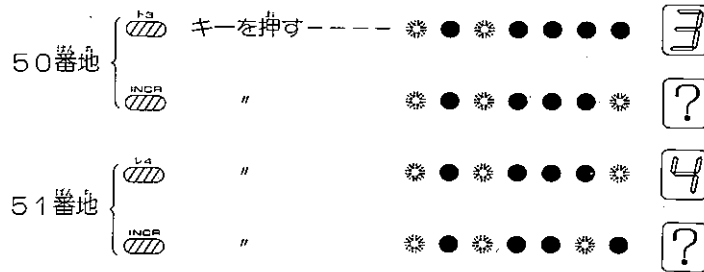
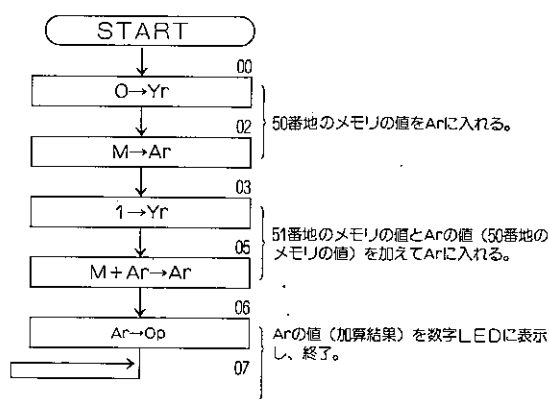


No. 32 1桁+1桁の16進加算の実験(その1)

●プログラム

アドレス	命令記号	命令コード
00	TIY	A
01	<0>	0
02	MA	5
03	TIY	A
04	<1>	1
05	M+	6
06	AO	1
07	JUMP	F
08	<0>	0
09	<7>	7

(フローチャート)



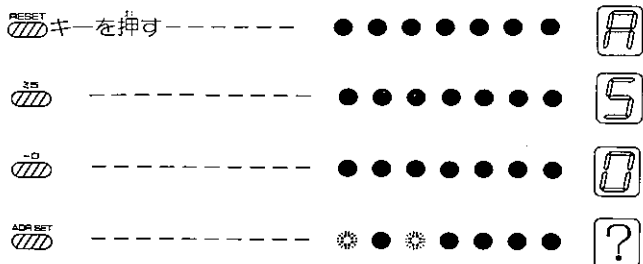
●コンピュータで計算をしてみましょう。
メモリ50番地+51番地の16進計算をして、答えは数字LEDに表示します。

- Ⓐ プログラムの書き込みと、確認をしてください。
- Ⓑ メモリ50とメモリ51にデータを書き込みます。(0~Fの値)
〈例〉メモリ50に<3>。メモリ51に<4>のデータを書き込みます。

- Ⓒ キーを順に押してください。
- Ⓓ 答えが、数字LEDに表示されます。

$$3 + 4 = \boxed{7}$$

アドレス 50 51
○ + △ = →数字LEDに表示

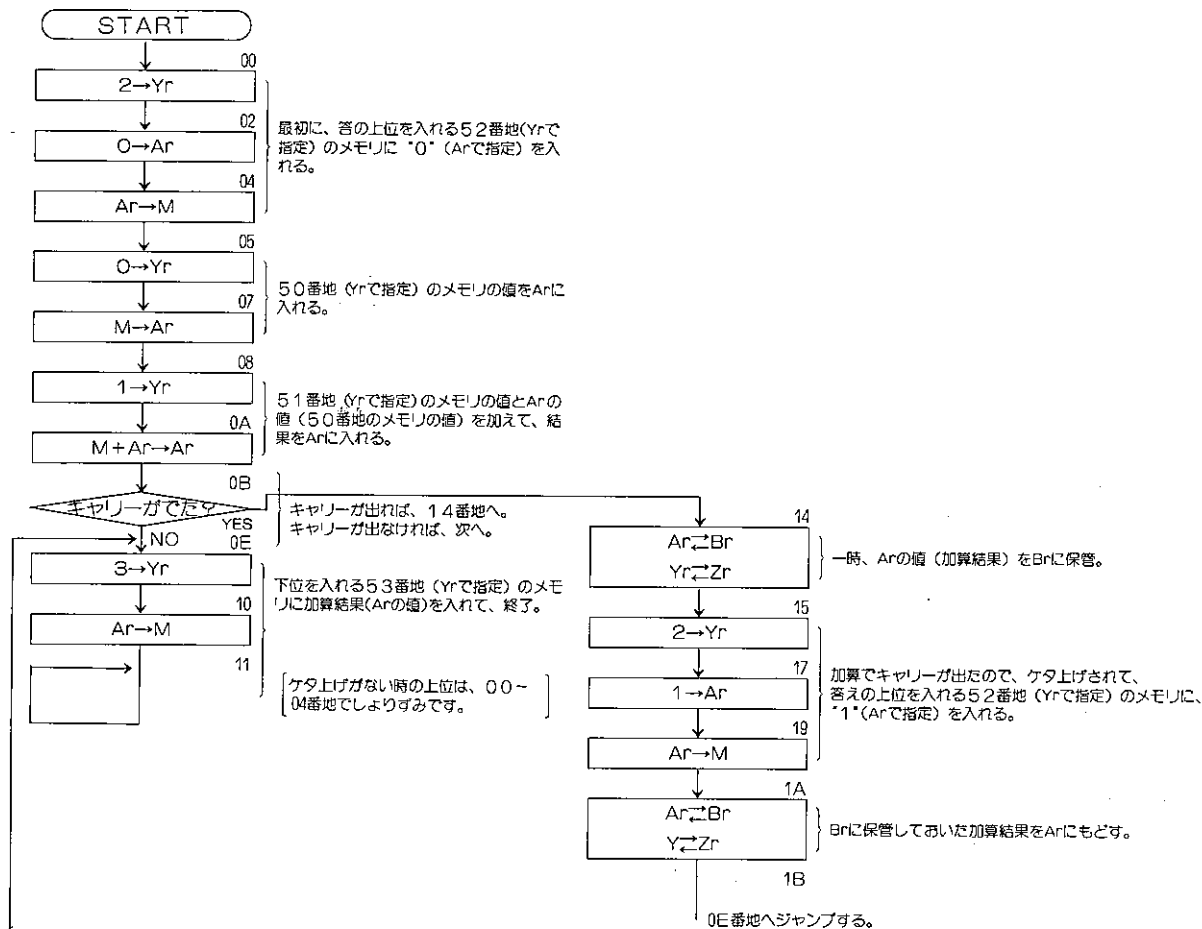


No. 33 けた けた しん か さん じっけん 1桁+1桁の16進加算の実験(その2)

●プログラム

アドレス	命令記号	命令コード
00	TIY	A
01	<2>	2
02	TIA	8
03	<0>	0
04	AM	4
05	TIY	A
06	<0>	0
07	MA	5
08	TIY	A
09	<1>	1
0A	M+	6
0B	JUMP	F
0C	<1>	1
0D	<4>	4
0E	TIY	A
0F	<3>	3
10	AM	4
11	JUMP	F
12	<1>	1
13	<1>	1
14	CH	2
15	TIY	A
16	<2>	2
17	TIA	8
18	<1>	1
19	AM	4
1A	CH	2
1B	JUMP	F
1C	<0>	0
1D	<E>	E

(フローチャート)













- No32は、1ケタの答えしか出ませんでした。この実験では、答えを2ケタにしてみました。

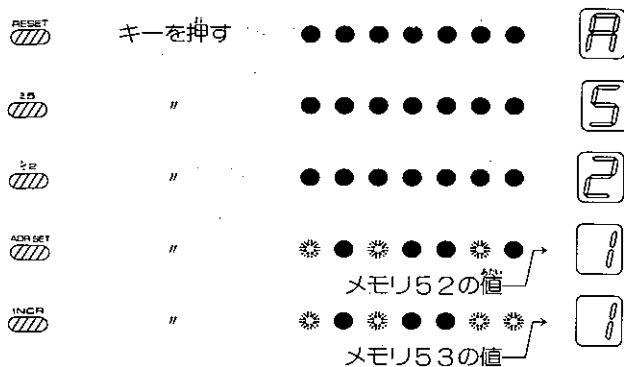
答えは、メモリ52（上位）53（下位）に書き込まれます。

アドレス 50 51 52 53

$$\bigcirc + \triangle = \boxed{} \boxed{}$$

----->2ケタ

- (A) プログラムの書き込みと、確認をしてください。
- (B) メモリ50とメモリ51にデータを書き込みます。(8+9)
 (例)        キーを押す
- (C)    キーを順に押してください。
- (D) 答えを確かめてみましょう。答えは、52番地と53番地のメモリに入るようにプログラムしてありますので、次のように読出します。例の答えは1（上位） 1（下位）の16進数で数字LEDに表示します。（11(16進数)=17(10進数)）



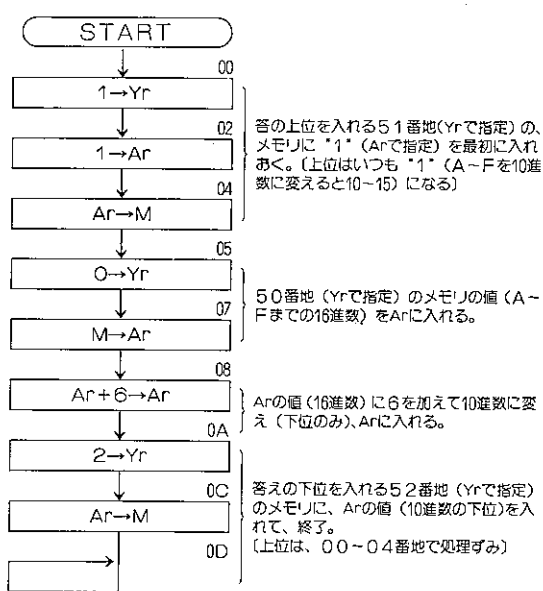
- $M + Ar \rightarrow Ar$ でキャリーがあった場合はArを一時保管しておき、Arを使ってメモリ52にデータ1を書き込んだ後、保管しておいたArの値をメモリ53に書き込みます。キャリーがなかったときはメモリ53にArの値を書き込みます。

No. 34 A~Fの16進数を10進数に変える実験

●プログラム

アドレス	命令記号	命令コード
00	TIY	A
01	<1>	1
02	TIA	8
03	<1>	1
04	AM	4
05	TIY	A
06	<0>	0
07	MA	5
08	AIA	9
09	<6>	6
0A	TIY	A
0B	<2>	2
0C	AM	4
0D	JUMP	F
0E	<0>	0
0F	<D>	D

(フローチャート)

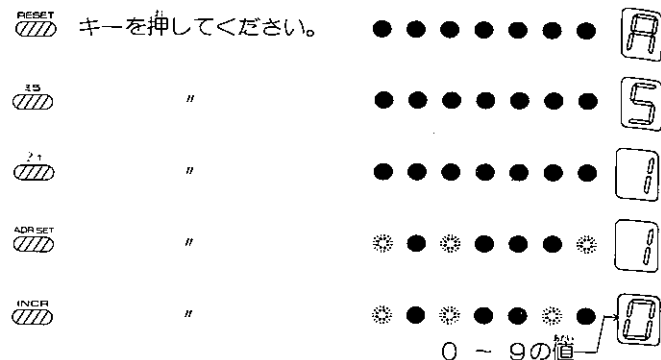


答の上位を入れる51番地(Yrで指定)のメモリに“1”(Arで指定)を最初に入れおく。(上位はいつも“1”(A~Fを10進数に変えると10~15)になる)

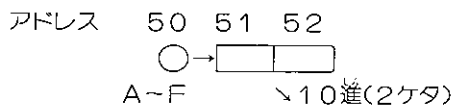
50番地(Yrで指定)のメモリの値(A~Fまでの16進数)をArに入れる。

Arの値(16進数)に6を加えて10進数に変え(下位のみ)、Arに入れる。

答の下位を入れる52番地(Yrで指定)のメモリに、Arの値(10進数の下位)を入れて、終了。
(上位は、00~04番地で処理済み)



●メモリ50の値に6を加えて10進数にしています。これでA~Fのデータに6を加えることによって、10進数の下のケタになることがわかりますね!! A=10、B=11---F=15となりますから、上のケタ(メモリ51)は、必ず1となります。



● 16進数を10進数に変換するプログラムです。ここでは簡単にするために、16進数をA~Fに限って考えてみましょう。答えは、51、52番地に入ります。

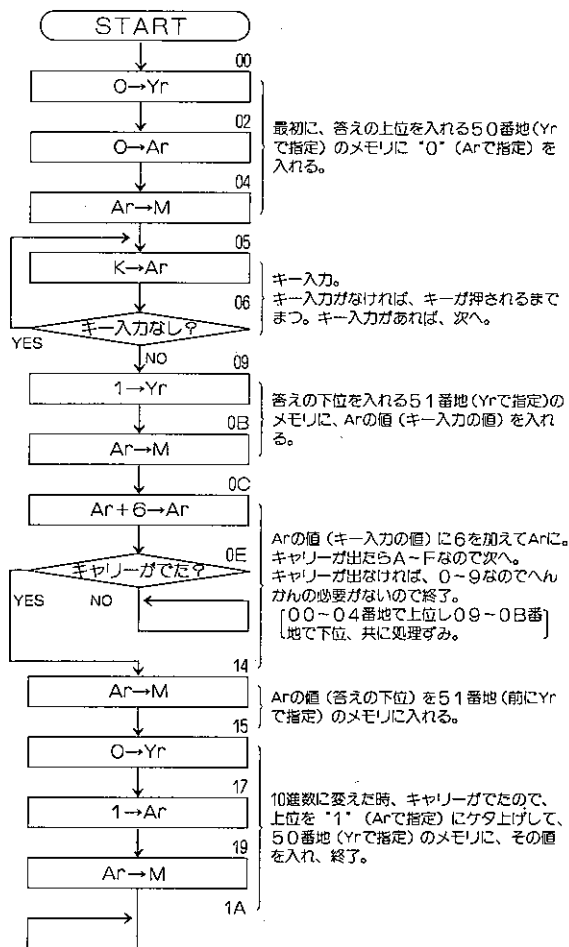
- (A) プログラムの書き込みと、確認をしてください。
- (B) メモリ50にデータ(A~F)を書き込みます。
 (例) キーを順に押してください。
- (C) キーを順に押してください。
- (D) 10進数に、置かれているか確認します。

No. 35 O~Fの16進数を10進数に変える実験

●プログラム

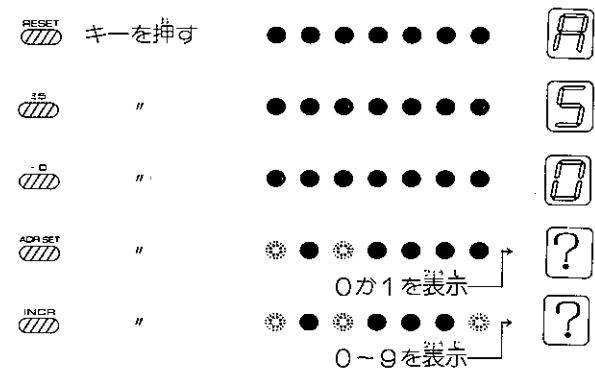
アドレス	命令記号	命令コード
00	T I Y	A
01	<0>	0
02	T I A	8
03	<0>	0
04	A M	4
05	K A	0
06	JUMP	F
07	<0>	0
08	<5>	5
09	T I Y	A
0A	<1>	1
0B	A M	4
0C	A I A	9
0D	<6>	6
0E	JUMP	F
0F	<1>	1
10	<4>	4
11	JUMP	F
12	<1>	1
13	<1>	1
14	A M	4
15	T I Y	A
16	<0>	0
17	T I A	8
18	<1>	1
19	A M	4
1A	JUMP	F
1B	<1>	1
1C	<A>	A

(フローチャート)

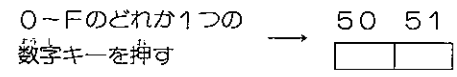


● No.34の応用として、キー入力の値(16進数0~F)を10進数に置してみましょう。16進数が0~Fまでであるために、0~9、A~Fの判断が必要となります。答えは50,51番地に入ります。

- Ⓐ プログラムの書き込みと、確認をしてください。
- Ⓑ キーを順に押してください。
↑入力番つきRUNモード
- Ⓒ 10進数に、直したい値のキー入力を1回押します。(0~Fのどれかのキーを1回押す)
- Ⓓ 10進数に、置っているか確かめてみましょう。



● A~Fまでの判断は、**Ar+6**で行います。キャリーが出ない場合は、そのままかまいません。キャリーが出たときは、10進数をメモリに書き込むプログラムに進みます。

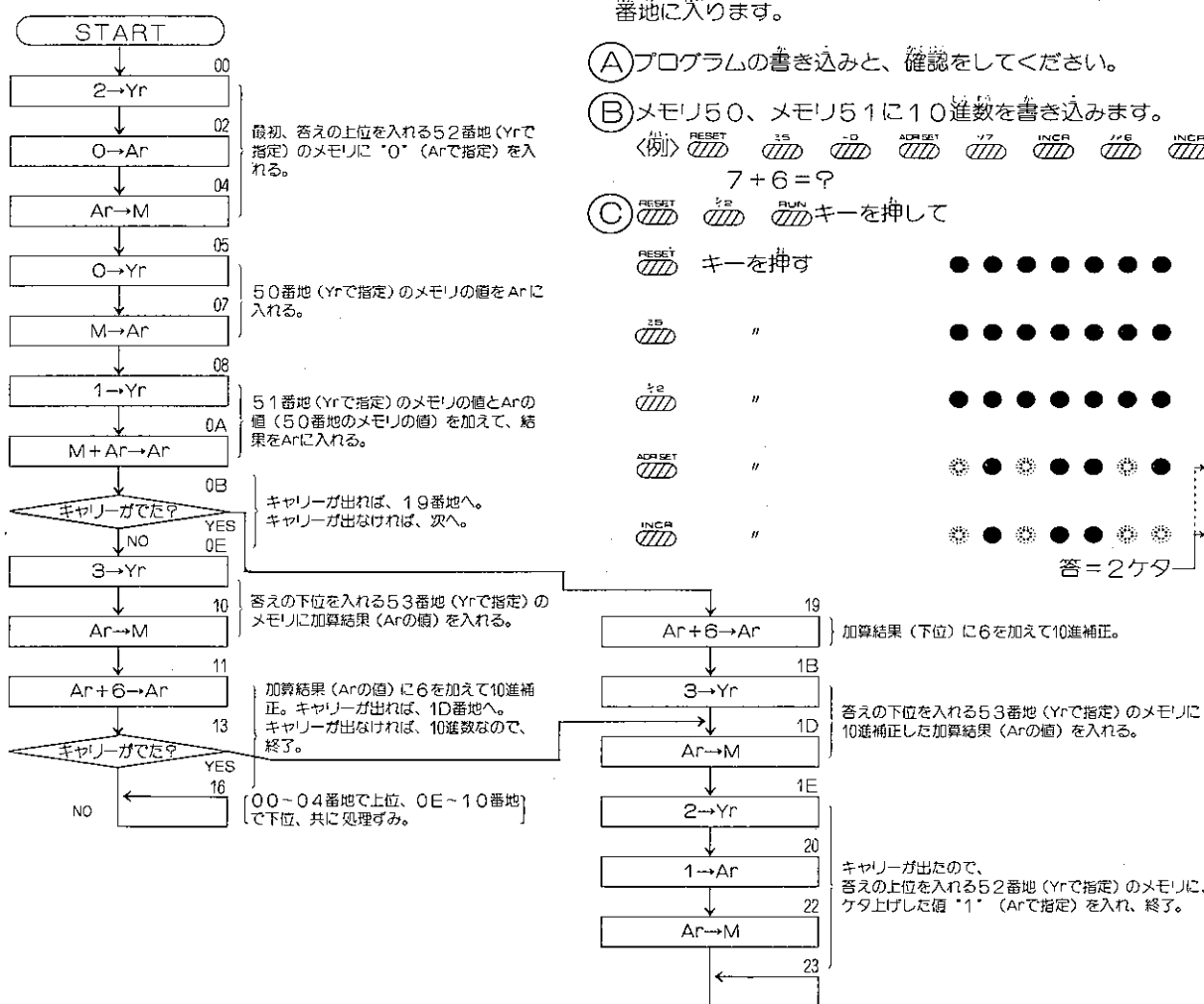


No. 36 けた けた しん か さん じっけん 1桁+1桁の10進加算の実験

●プログラム

アドレス	命令記号	命令コード
00	TIY	A
01	<2>	2
02	TIA	8
03	<0>	0
04	AM	4
05	TIY	A
06	<0>	0
07	MA	5
08	TIY	A
09	<1>	1
0A	M+	6
0B	JUMP	F
0C	<1>	1
0D	<9>	9
0E	TIY	A
0F	<3>	3
10	AM	4
11	AIA	9
12	<6>	6
13	JUMP	F
14	<1>	1
15	<D>	D
16	JUMP	F
17	<1>	1
18	<6>	6
19	AIA	9
1A	<6>	6
1B	TIY	A
1C	<3>	3
1D	AM	4
1E	TIY	A
1F	<2>	2
20	TIA	8
21	<1>	1
22	AM	4
23	JUMP	F
24	<2>	2
25	<3>	3

(フローチャート)



● メモリを使った、10進加算のプログラムを作りましょう。
ケタ数は1ケタ+1ケタで、答えは2ケタとなり52、53番地に入ります。

(A) プログラムの書き込みと、確認をしてください。

(B) メモリ50、メモリ51に10進数を書き込みます。

例) 7+6=?

(C) キーを押して

キーを押す

" " " " " " "

" " " " " "

" " " " " "

" " " " "

" " " " "

答=2ケタ

A

5

2

1

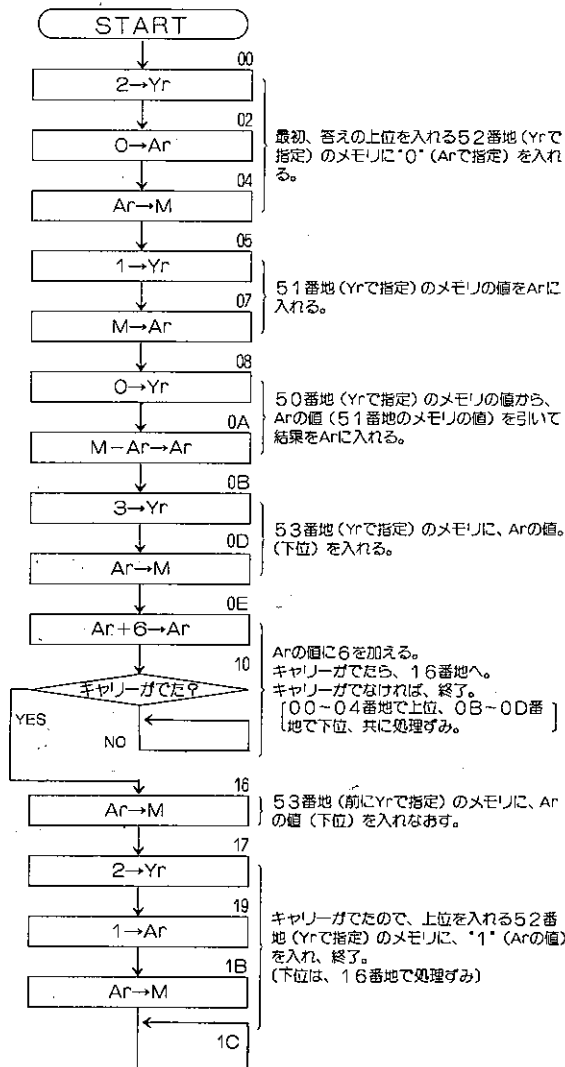
3

No. 37 けた けた しんげんさん しんほせい じっけん 1桁-1桁の16進減算と10進補正の実験

●プログラム

アドレス	命令記号	命令コード
00	TIY	A
01	<2>	2
02	TIA	8
03	<0>	0
04	AM	4
05	TIY	A
06	<1>	1
07	MA	5
08	TIY	A
09	<0>	0
0A	M-	7
0B	TIY	A
0C	<3>	3
0D	AM	4
0E	AIA	9
0F	<6>	6
10	JUMP	F
11	<1>	1
12	<6>	6
13	JUMP	F
14	<1>	1
15	<3>	3
16	AM	4
17	TIY	A
18	<2>	2
19	TIA	8
1A	<1>	1
1B	AM	4
1C	JUMP	F
1D	<1>	1
1E	<C>	C

(フローチャート)



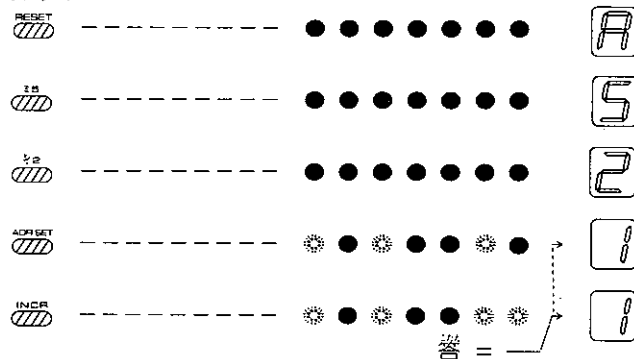
● 16進数を書いた減算をし、その答えを10進数に直し答えは、52、53番地に入ります。

(A) プログラムの書き込みと、確認をしてください。

(B) メモリ50、メモリ51に16進数を書き込みます。(メモリ50をメモリ51より、大きくします)

〈例〉

(C) キーを順に押してください。



◆ $Ar+6 \rightarrow Ar$ と、その次の で10進補正しています。

アドレス 50 51 52 53

○ - △ =

16進数 答えは10進数(2ケタ)

No. 38 けた けた しん か さん しん ほ せい じっけん 1桁+1桁の16進加算と10進補正の実験

●プログラム

アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	TIY	A	27	<2>	2
01	<2>	2	28	TI A	8
02	TI A	8	29	<1>	1
03	<0>	0	2A	AM	4
04	AM	4	2B	JUMP	F
05	TIY	A	2C	<2>	2
06	<0>	0	2D		B
07	MA	5	2E	AM	4
08	TIY	A	2F	TIY	A
09	<1>	1	30	<2>	2
0A	M+	6	31	TI A	8
0B	JUMP	F	32	<2>	2
0C	<1>	1	33	AM	4
0D	<9>	9	34	JUMP	F
0E	TIY	A	35	<3>	3
0F	<3>	3	36	<4>	4
10	AM	4	37	A I A	9
11	A I A	9	38	<6>	6
12	<6>	6	39	AM	4
13	JUMP	F	3A	A I A	9
14	<2>	2	3B	<6>	6
15	<0>	0	3C	JUMP	F
16	JUMP	F	3D	<4>	4
17	<1>	1	3E	<2>	2
18	<6>	6	3F	JUMP	F
19	TIY	A	40	<2>	2
1A	<3>	3	41	<F>	F
1B	A I A	9	42	AM	4
1C	<6>	6	43	TIY	A
1D	JUMP	F	44	<2>	2
1E	<3>	3	45	TI A	8
1F	<7>	7	46	<3>	3
20	AM	4	47	AM	4
21	A I A	9	48	JUMP	F
22	<6>	6	49	<4>	4
23	JUMP	F	4A	<8>	8
24	<2>	2			
25	<E>	E			
26	TIY	A			

加算の答			
16進数	→	10進数	16進数 → 10進数
0~9	→	0~9	14~19 → 20~25
A~F	→	10~15	1A~1D → 26~29
10~13	→	16~19	1E → 30

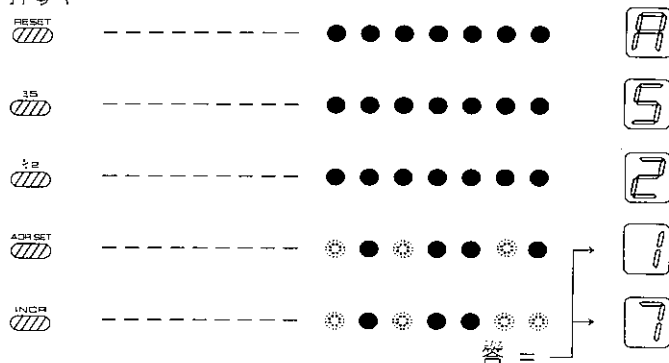
● たしざんの10進補正を試みましょう。

(A) プログラムの書き込みと、確認をしてください。

(B) メモリ50とメモリ51にデータを書き込みます。答えは、52、53番地に入ります。
 (例) メモリ50 <8> 51 <9> $8 + 9 = ?$



(C) キーを順に押してください。
 押すキー



◆ 16進の加算をしてから、10進補正をしています。分岐がたくさんできますので注意しましょう。

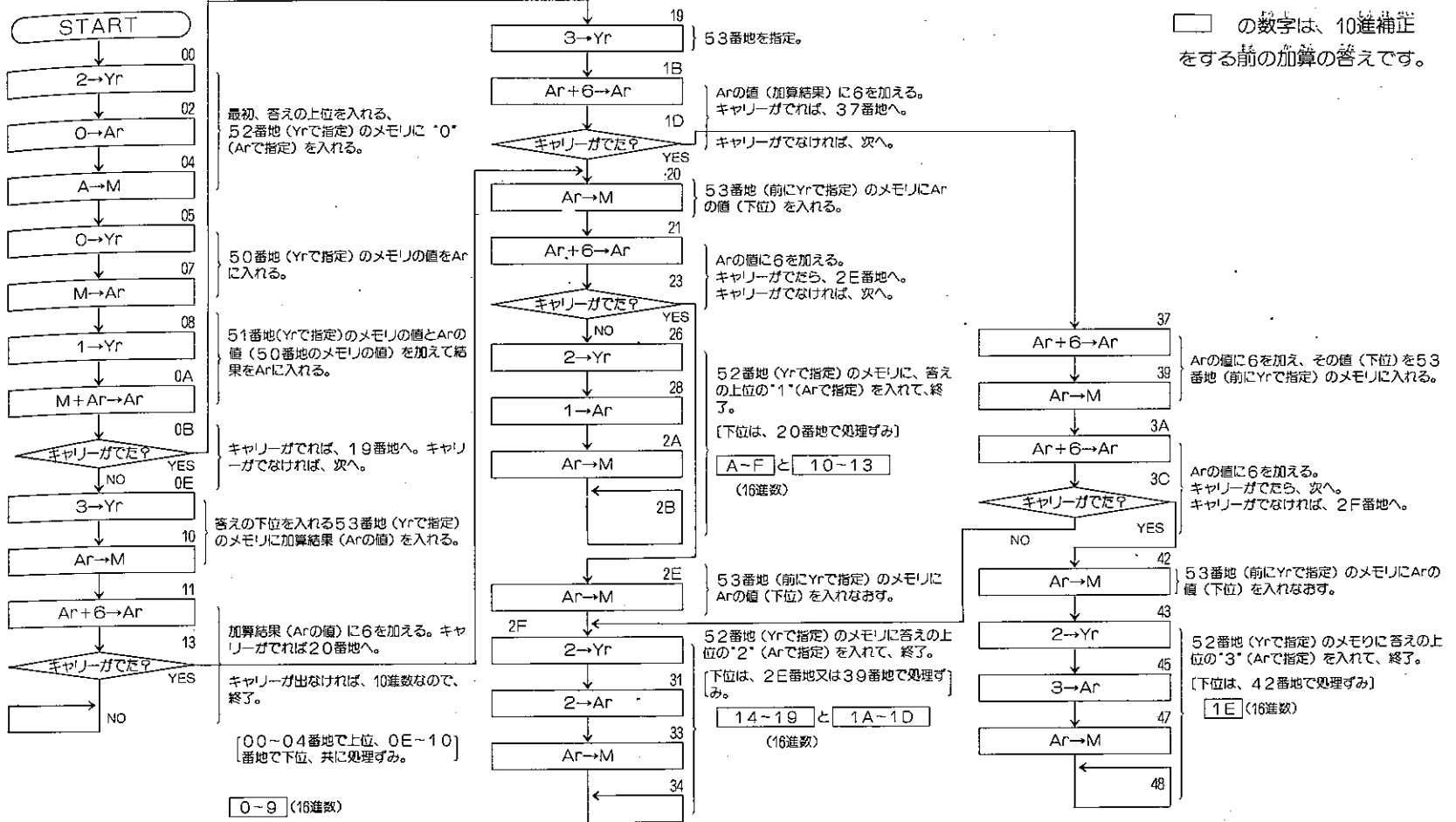
アドレス 50 51 52 53

$\bigcirc + \Delta = \square \square$

16進数 答えは10進数

◇ のこと

(フローチャート)



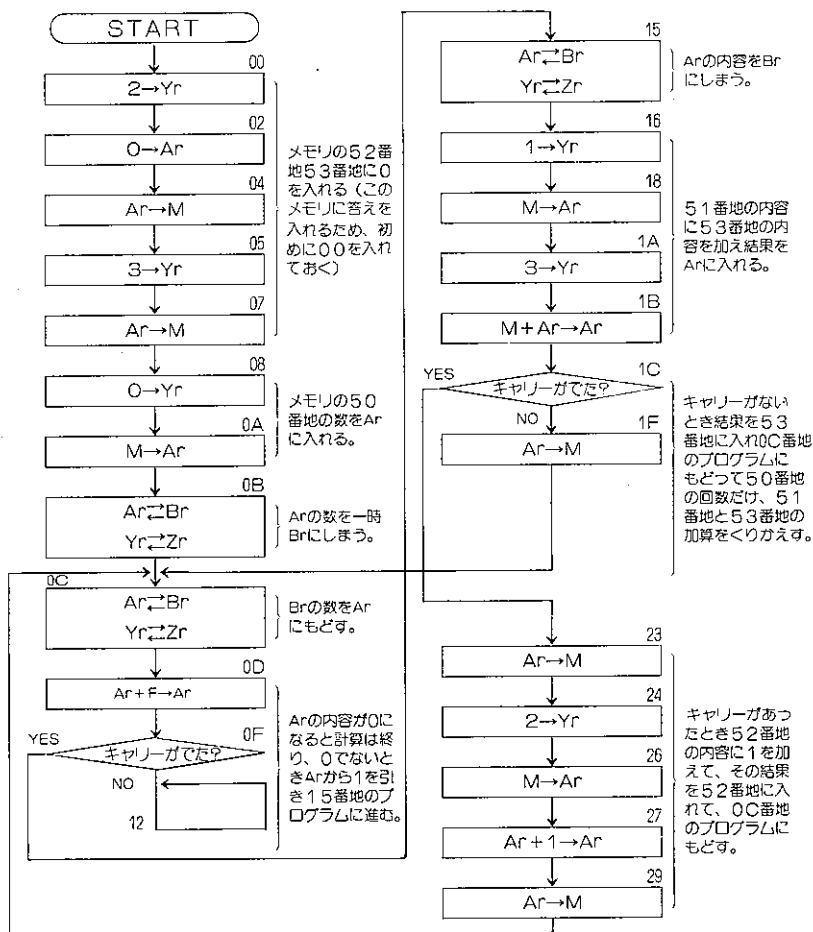
No. 39 1桁×1桁の16進乗算の実験

●プログラム

アドレス	命令記号	命令コード
00	T I Y	A
01	<2>	2
02	T I A	8
03	<0>	0
04	AM	4
05	T I Y	A
06	<3>	3
07	AM	4
08	T I Y	A
09	<0>	0
0A	MA	5
0B	CH	2
0C	CH	2
0D	A I A	9
0E	<F>	F
0F	JUMP	F
10	<1>	1
11	<5>	5
12	JUMP	F
13	<1>	1
14	<2>	2
15	CH	2
16	T I Y	A
17	<1>	1
18	MA	5
19	T I Y	A
1A	<3>	3
1B	M+	6
1C	JUMP	F
1D	<2>	2
1E	<3>	3
1F	AM	4
20	JUMP	F
21	<0>	0
22	<C>	C
23	AM	4
24	T I Y	A
25	<2>	2

アドレス	命令記号	命令コード
26	MA	5
27	A I A	9
28	<1>	1
29	AM	4
2A	JUMP	F
2B	<0>	0
2C	<C>	C

(フローチャート)



● 次は、「かけ算」に挑戦してみよう。
 かけ算は、たし算のくり返しです。
 たとえば、 5×6 は、6を5回たすことによつて答えが出てきます。
 このプログラムも、たし算を利用して作ってみよう。

◆このプログラムは、初めメモリ52、53を0にします。メモリ51のデータを、メモリ52、53に加えると同時にメモリ50のデータを-1します。メモリ50の減算で引けなくなったら、加算を中止します。

アドレス 50 51 52 53

○ × △ =
 ↓2ケタ

- (A) プログラムの書き込みと、確認をしてください。
 (B) メモリ50とメモリ51にデータを書き込み 答えは、52、53番地に入ります。

〈例〉 $3 \times 8 = \square$

RESET 10 0 ADDRSET 10 INCR 10 INCR

- (C) RESET 10 RUN キーを順に押してください。

- (D) 答えは、18 (16進数で表示)

RESET	キーを押す	● ● ● ● ● ● ● ●	<input type="text" value="A"/> <input type="text" value="5"/> <input type="text" value="2"/> <input type="text" value="!"/> <input type="text" value="8"/>
15	"	● ● ● ● ● ● ● ●	
12	"	● ● ● ● ● ● ● ●	
ADDRSET	"	⊗ ● ⊗ ● ● ⊗ ● ●	
INCR	"	⊗ ● ⊗ ● ● ⊗ ⊗	

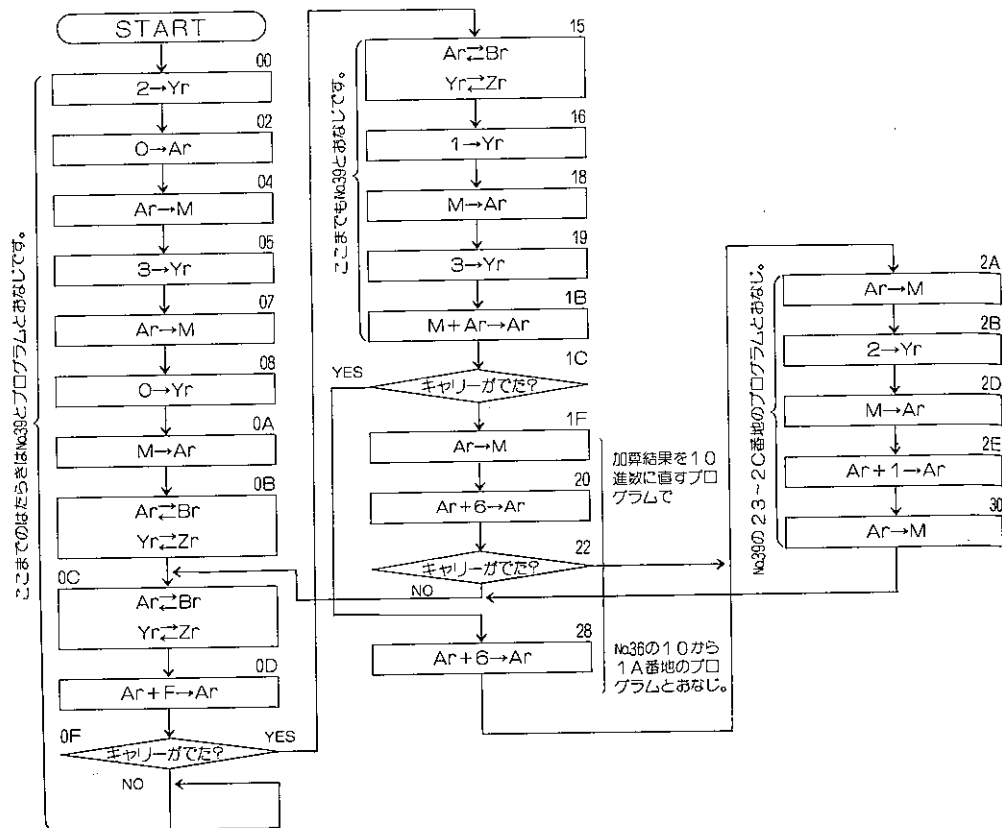
答 =

No. 40 1桁×1桁の10進乗算の実験

●プログラム

アドレス	命令記号	データ	アドレス	命令記号	データ
00	TIY	A	1C	JUMP	F
01	<2>	2	1D	<2>	2
02	TIA	8	1E	<8>	8
03	<0>	0	1F	AM	4
04	AM	4	20	AIA	9
05	TIY	A	21	<6>	6
06	<3>	3	22	JUMP	F
07	AM	4	23	<2>	2
08	TIY	A	24	<A>	A
09	<0>	0	25	JUMP	F
0A	MA	5	26	<0>	0
0B	CH	2	27	<C>	C
0C	CH	2	28	AIA	9
0D	AIA	9	29	<6>	6
0E	<F>	F	2A	AM	4
0F	JUMP	F	2B	TIY	A
10	<1>	1	2C	<2>	2
11	<5>	5	2D	MA	5
12	JUMP	F	2E	AIA	9
13	<1>	1	2F	<1>	1
14	<2>	2	30	AM	4
15	CH	2	31	JUMP	F
16	TIY	A	32	<0>	0
17	<1>	1	33	<C>	C
18	MA	5			
19	TIY	A			
1A	<3>	3			
1B	M+	6			

(フローチャート)



- 10進補正のある乗算（かけ算）を作ります。

答えは、52、53番地に入ります。

Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ メモリ50とメモリ51に10進数データを書き込みます。

〈例〉メモリ50〈4〉メモリ〈6〉

RESET 20 70 ADRESET LA INCR 770 INCR

Ⓒ RESET 22 RUN 順にキーを押してください。

Ⓓ RESET 25 22 ADRESET 2 INCR 4 答えは、RESETキー及び、INCRキーを押

した時、数字LEDに1ケタずつ表示されます。

●No.39と、ほとんど同じですがM+Ar→Arのところでは10進加算をしています。その分プログラムは長くなっています。

50番地 51番地 52番地 53番地

□ × □ = □ □
 (上位) (下位)

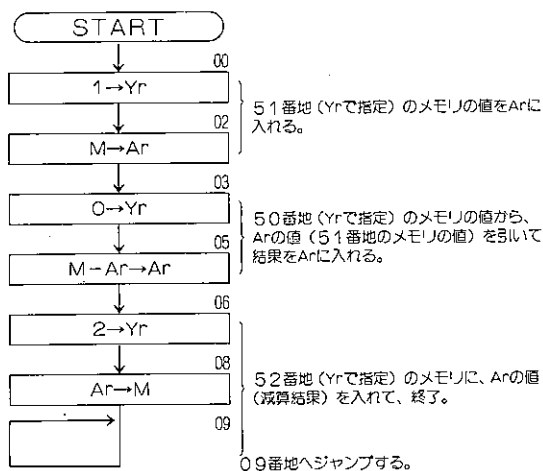
※No.39のプログラムの1Eと1F番地の間に加算結果を10進数に直すプログラムが入った形になります。

No. 41 1けた けた 1桁-1桁の10進減算の実験

●プログラム

アドレス	命令記号	命令コード
00	TIY	A
01	<1>	1
02	MA	5
03	TIY	A
04	<0>	0
05	M-	7
06	TIY	A
07	<2>	2
08	AM	4
09	JUMP	F
0A	<0>	0
0B	<9>	9

(フローチャート)



● 減算 (ひきざん) をしてみましょう。答えは1ケタです。

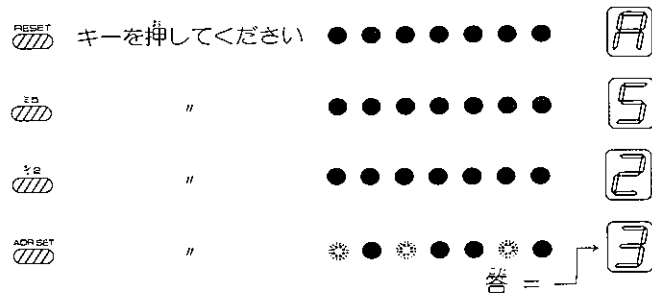
- (A) プログラムの書き込みと、確認をしてください。
- (B) メモリ50と51に10進数を書き込みます。(必ずメモリ50を大きな数にします)

〈例〉 9-6=□

RESET 50 51 ADDR SET 50 INCR 51 INCR

- (C) RESET 50 RUN キーを順に押してください。

(D) 答えは、52番地に入ります。



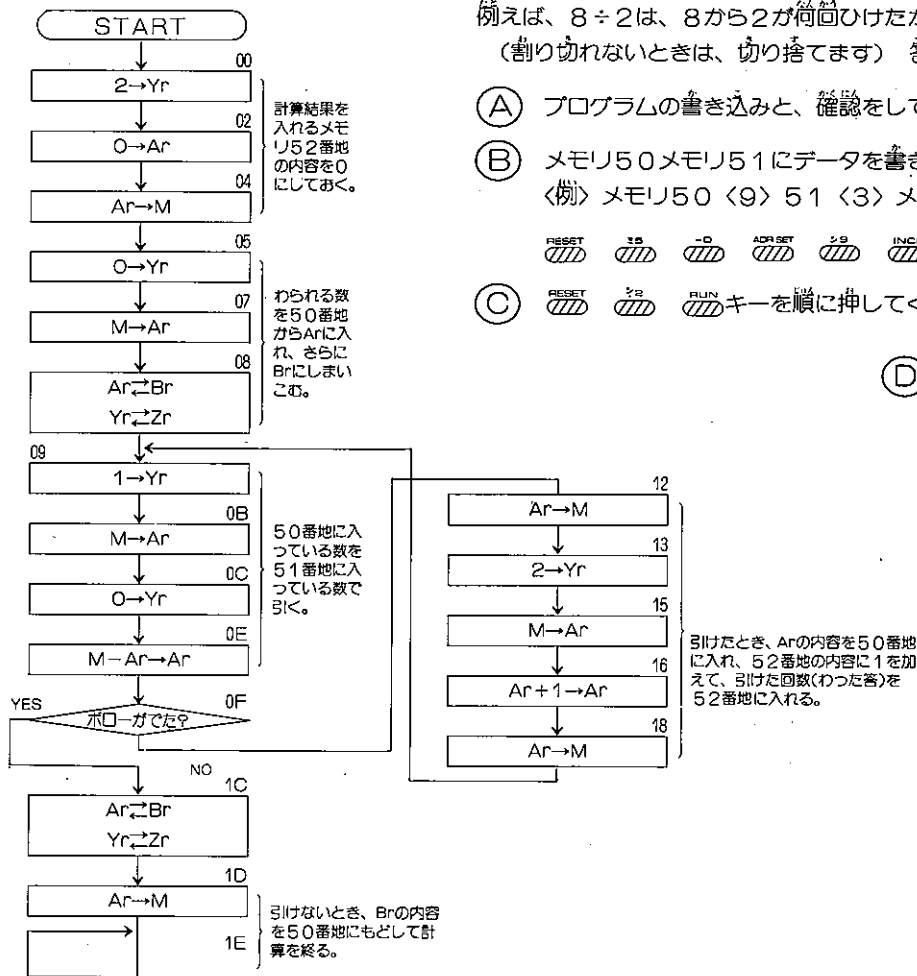
アドレス 50 51 52
○ - △ = □

No. 42 1桁÷1桁の除算の実験

●プログラム

アドレス	命令記号	命令コード
00	TIY	A
01	<2>	2
02	TIA	8
03	<0>	0
04	AM	4
05	TIY	A
06	<0>	0
07	MA	5
08	CH	2
09	TIY	A
0A	<1>	1
0B	MA	5
0C	TIY	A
0D	<0>	0
0E	M-	7
0F	JUMP	F
10	<1>	1
11	<C>	C
12	AM	4
13	TIY	A
14	<2>	2
15	MA	5
16	AIA	9
17	<1>	1
18	AM	4
19	JUMP	F
1A	<0>	0
1B	<9>	9
1C	CH	2
1D	AM	4
1E	JUMP	F
1F	<1>	1
20	<E>	E

(フローチャート)

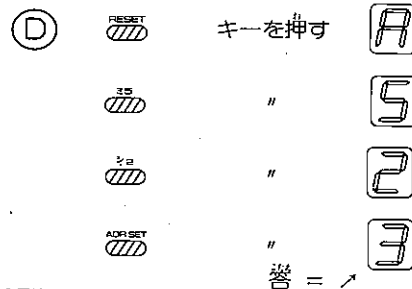


● 除算（わり算）は、減算（ひき算）を利用します。
 例えば、8÷2は、8から2が何回ひけたかを求めればよいのです。
 （割り切れないときは、切り捨てます） 答えは、52番地に入ります。

- Ⓐ プログラムの書き込みと、確認をしてください。
- Ⓑ メモリ50メモリ51にデータを書き込みます。
 〈例〉メモリ50〈9〉 51〈3〉メモリ51は0にしないこと。

RESET 20 2D ADPSET 29 INCR 23 INCR 9÷3=□

- Ⓒ RESET 20 RUN キーを順に押してください。



アドレス 50 51 52
 ○ ÷ △ = □

(3) 第3郡の命令について

第3群の命令は、CY、CIA、CIY、CAL RSTO、CAL SETR、CAL RSTR、CAL CMPL、CAL DSPR、CAL SHTS、の9つの命令です。実験を通じて順に説明していきます。

※CIY命令について

この命令は、Yレジスタの値と次の行に書き込まれているデータを比較します。Yレジスタの値とデータが等しければ、実行フラグは0となります。等しくない場合は実行フラグが1となります。Yレジスタの値は変化ありません。フローチャートでは、 $\langle Yr + \text{データ} \rangle$ (○→データ) と書きます。

※CAL SETR命令について

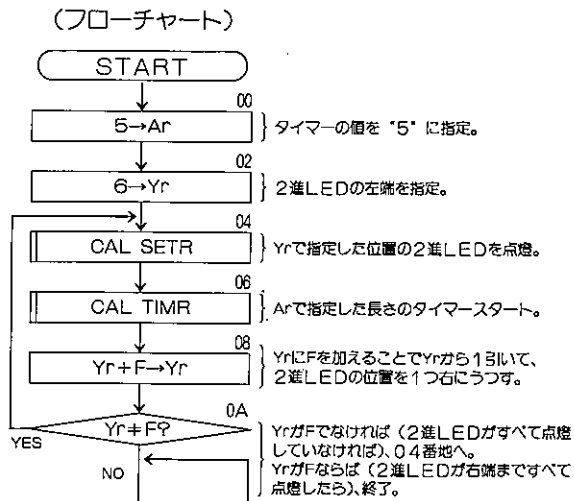
この命令は、Yレジスタで指定するRポート（例えばYレジスタが0のときR0ポートが指定される）に1を出力（2進LEDが点灯）します。Yレジスタの設定の後、この命令をあたえることによって、希望する2進LEDを点灯させることができます。

No. 43 CIY, CAL SETR命令を使った実験




左から右へ逐次点燈 (結果は全部点燈)

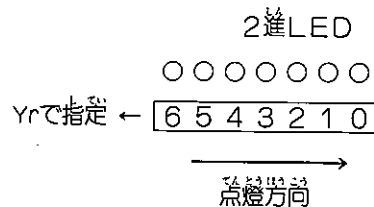
●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<5>	5
02	TIY	A
03	<6>	6
04	CAL	E
05	SETR	1
06	CAL	E
07	TIMR	C
08	AIY	B
09	<F>	F
0A	CIY	D
0B	<F>	F
0C	JUMP	F
0D	<0>	0
0E	<4>	4
0F	JUMP	F
10	<0>	0
11	<F>	F



Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ    キーを順に押してください。



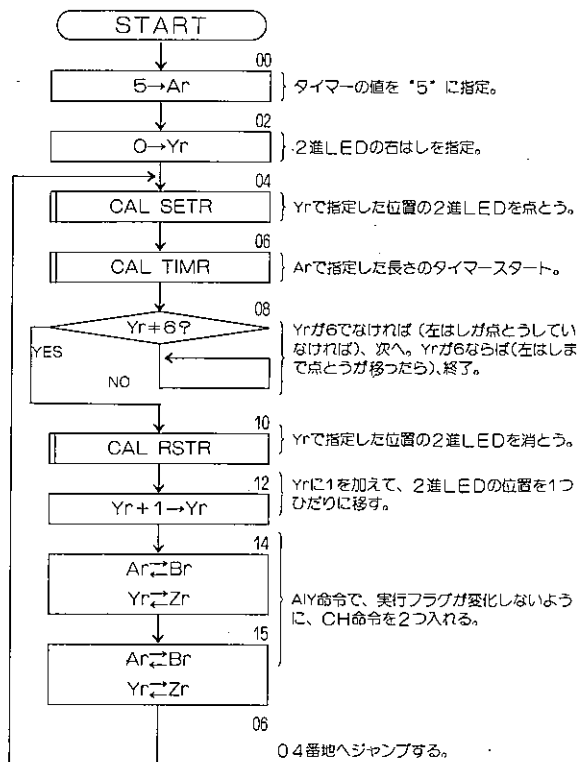
No. 44 CAL RSTR命令を使った実験

右から左へ点燈が逐次移動 (結果1ヶ点燈)

●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<5>	5
02	TIY	A
03	<0>	0
04	CAL	E
05	SETR	1
06	CAL	E
07	TIMR	C
08	CIY	D
09	<6>	6
0A	JUMP	F
0B	<1>	1
0C	<0>	0
0D	JUMP	F
0E	<0>	0
0F	<D>	D
10	CAL	E
11	RSTR	2
12	AIY	B
13	<1>	1
14	CH	2
15	CH	2
16	JUMP	F
17	<0>	0
18	<4>	4

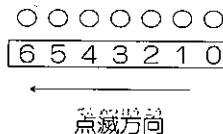
(フローチャート)



● 光 (LED) が移動するプログラムです。

Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ RESET RUN キーを順に押し実行します。



※ CAL RSTR 命令について

この命令もサーブプログラムです。No43でCAL SETRの説明をしましたが、CAL RSTRは、CAL SETRの反対の動きをします。Yレジスタで指定する2進LED出力を0(消す)にします。

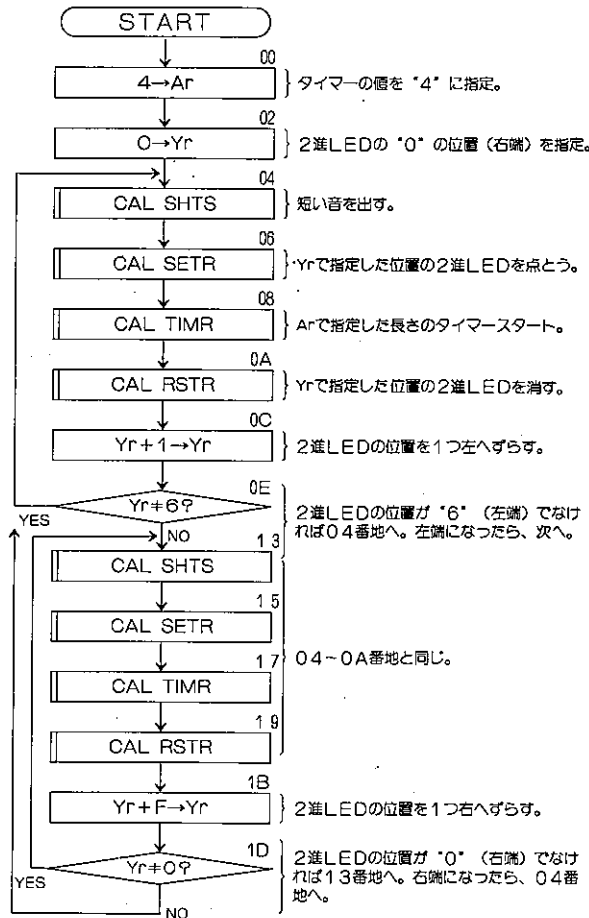
No. 45 CAL SHTS命令を使った実験

左右へ往復点灯をくり返す(移動音を出す)

●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<4>	4
02	TIY	A
03	<0>	0
04	[CAL	E
05	[SHTS	9
06	[CAL	E
07	[SETR	1
08	[CAL	E
09	[TIMR	C
0A	[CAL	E
0B	[RSTR	2
0C	AIY	B
0D	<1>	1
0E	CIY	D
0F	<6>	6
10	JUMP	F
11	<0>	0
12	<4>	4
13	[CAL	E
14	[SHTS	9
15	[CAL	E
16	[SETR	1
17	[CAL	E
18	[TIMR	C
19	[CAL	E
1A	[RSTR	2
1B	AIY	B
1C	<F>	F
1D	CIY	D
1E	<0>	0
1F	JUMP	F
20	<1>	1
21	<3>	3
22	JUMP	F
23	<0>	0
24	<4>	4

(フローチャート)

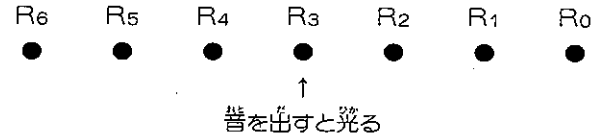


(A) プログラムの書き込みと、確認をしてください。

(B) キーを順に押し実行します。



●2進LEDに1を出力する前にピツという短い音を出してみました。移動の前に音が出ます。マイコンユニットから、本体ケース内アンプに音を出すばあい、R3を通して音を出すので、音がスピーカから出る時、LEDのR3が光ります。



※ CAL SHTS命令について

この命令を実行するとR3ポートを通して短い音が出ます。これはショートサウンドと呼び、後で説明すると共にプログラムでモールス音を組む場合に使います。モールス短音は短音と短い休みでできあがっています。(短音 ■■■ 休み)

No. 46 CAL RSTO^{めいれい}命令^{つか}を使った実験^{じっけん}

(数字LEDに“E”の点滅^{てんめつ})

●プログラム

アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	TIY	A	1C	JUMP	F
01	<1>	1	1D	<2>	2
02	MA	5	1E	<2>	2
03	TIY	A	1F	JUMP	F
04	<0>	0	20	<1>	1
05	M-	7	21	<F>	F
06	JUMP	F	22	[CAL	E
07	<0>	0	23	[TIMR	C
08	<D>	D	24	JUMP	F
09	AO	1	25	<0>	0
0A	JUMP	F	26	<F>	F
0B	<0>	0			
0C	<A>	A			
0D	TIY	A			
0E	<0>	0			
0F	TIA	8			
10	<E>	E			
11	AO	1			
12	AIY	B			
13	<1>	1			
14	TIA	8			
15	<2>	2			
16	[CAL	E			
17	[TIMR	C			
18	[CAL	E			
19	[RSTO	0			
1A	CIY	D			
1B	<A>	A			

- (A) プログラムの書き込みと、確認をしてください。
- (B) メモリ50、メモリ51にデータを書き込みます。(データ及び計算は16進数)
(エラー表示をさせるためにはメモリ51を大きく)

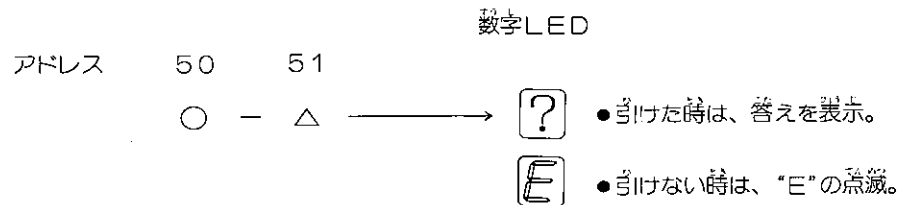
〈例〉8-9=?



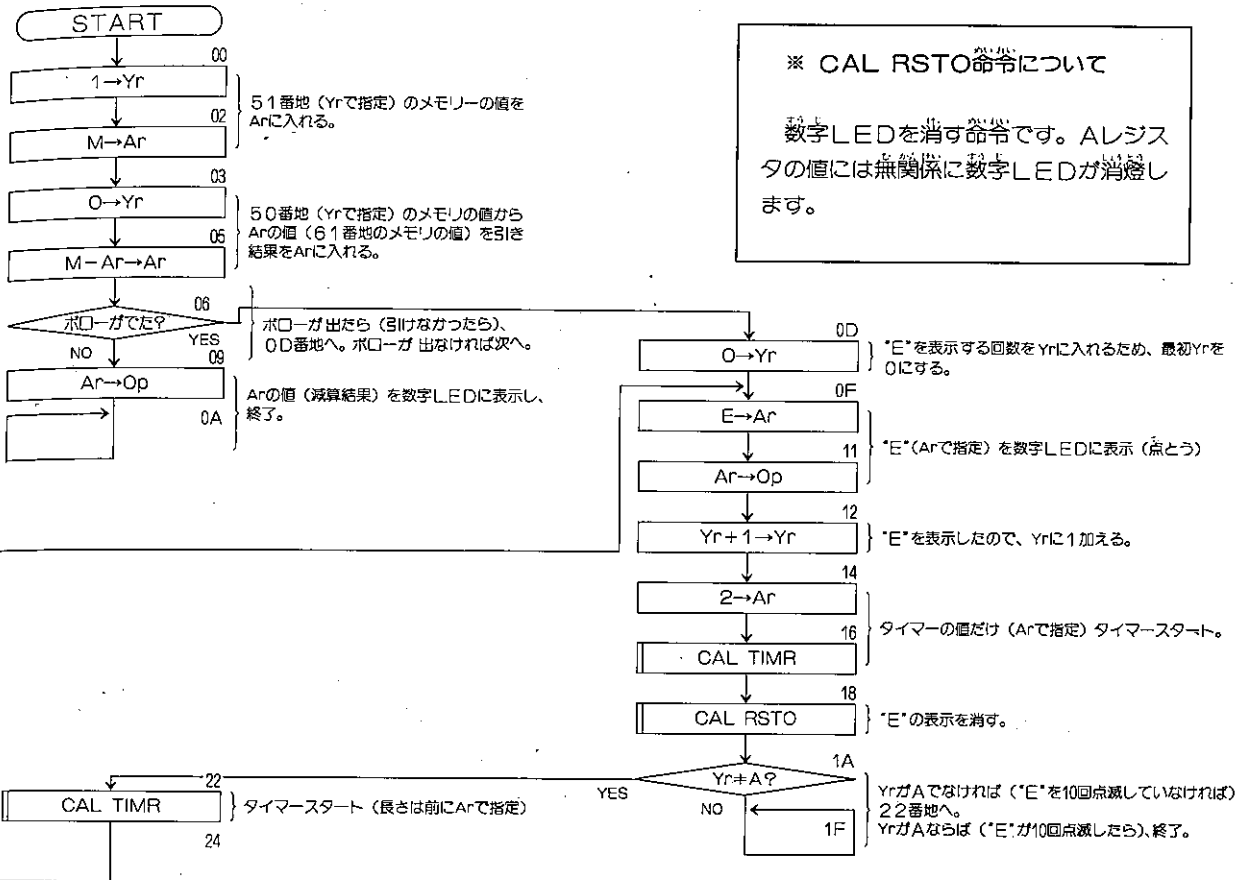
- (C) キーを順に押し実行します。 の点滅。

●数字LEDを消す命令を使ってエラーの表示を試みましょう。

引けない場合は、 と をタイマをはさんで交互に実行し、エラー表示Eの文字を10回点滅して消えます。引けた場合は、答えを出力して終了します。



(フローチャート)



※ CAL RSTO命令について

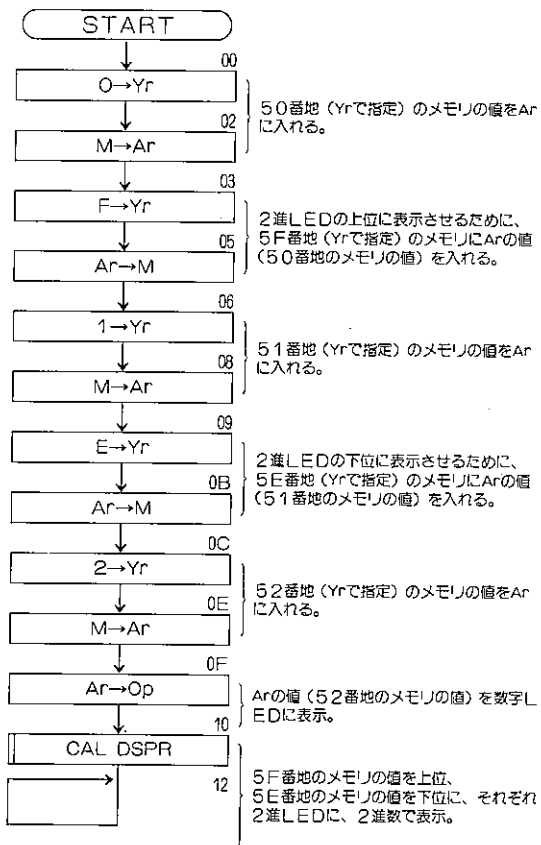
数字LEDを消す命令です。レジスタの値には無関係に数字LEDが消燈し ます。

No. 47 CAL DSPR命令を使った実験

●プログラム

アドレス	命令記号	命令コード
00	TIY	A
01	<0>	0
02	MA	5
03	TIY	A
04	<F>	F
05	AM	4
06	TIY	A
07	<1>	1
08	MA	5
09	TIY	A
0A	<E>	E
0B	AM	4
0C	TIY	A
0D	<2>	2
0E	MA	5
0F	AO	1
10	↑CAL	E
11	↓DSPR	D
12	JUMP	F
13	<1>	1
14	<2>	2

(フローチャート)



2進LED

数字LED

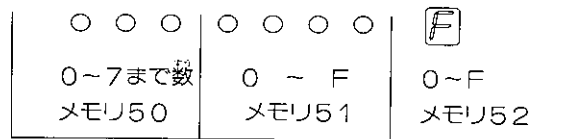
4 2 1 8 4 2 1
○ ○ ○ ○ ○ ○ ○



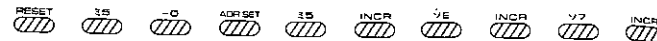
↑ ↑ ↑
アドレス 50 51 52

● 便利なプログラムを使って実験してみましょう。

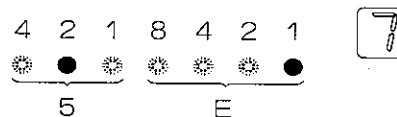
- Ⓐ プログラムの書き込みと、確認をしてください。
- Ⓑ メモリ50、51、52に表示させたいデータを書き込みます。



〈例〉メモリ50 <5>、51 <E>、52 <7>



- Ⓒ RESET, 71, RUN キーを順に押し実行します。



●簡単に3ケタの表示ができました。プログラムもCAL SETR、CAL RSTRを使って表示させる方法にくらべて大変短くなりますので、おおいに活用してください。

※ CAL DSPR命令について

CAL DSPRは2進LEDの表示をさせる命令です。メモリ5F、5Eに書き込まれているデータを、2進LEDに表示します。メモリ5Fは上位、メモリ5Eは下位に表示します。

○ ○ ○ ○ ○ ○ ○ ○

↑ ↑
上位 下位

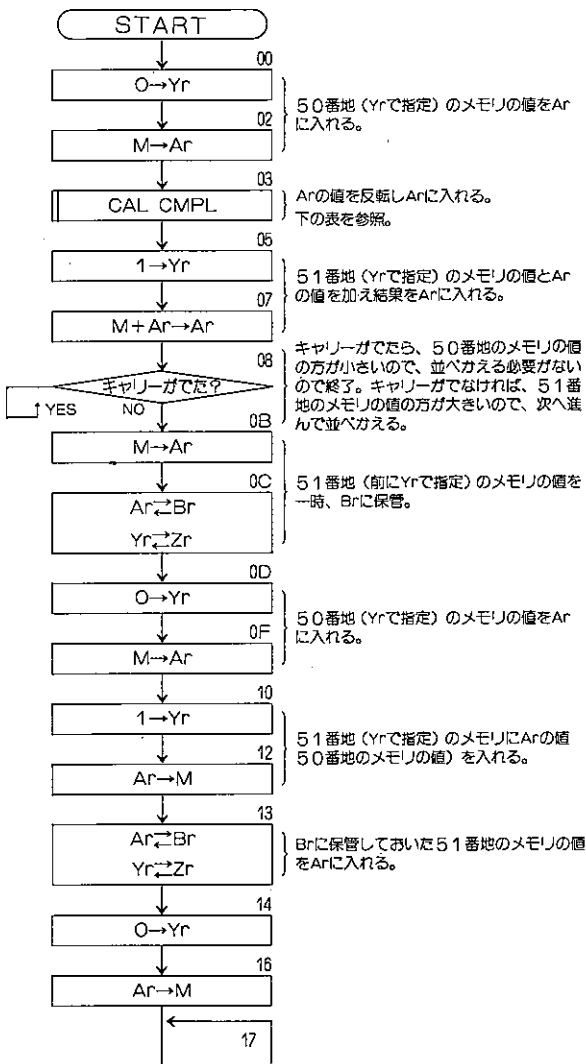
表示できる値 → (0~7) (0~F)

No. 48 CAL CMPL 命令を使った実験 (2つの数を小さい順に並べかえる)

●プログラム

アドレス	命令記号	命令コード
00	TIY	A
01	<0>	0
02	MA	5
03	CAL	E
04	CMPL	4
05	TIY	A
06	<1>	1
07	M+	6
08	JUMP	F
09	<0>	0
0A	<8>	8
0B	MA	5
0C	CH	2
0D	TIY	A
0E	<0>	0
0F	MA	5
10	TIY	A
11	<1>	1
12	AM	4
13	CH	2
14	TIY	A
15	<0>	0
16	AM	4
17	JUMP	F
18	<1>	1
19	<7>	7

(フローチャート)



● メモリの矢印の比較を [CAL CMPL] と [M+Ar→Ar] でしています。

Ⓐ プログラムの書き込みと、確認をしてください。

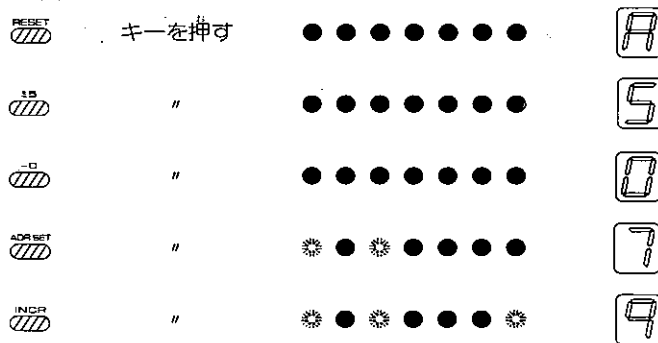
Ⓑ メモリ50、51にデータを書き込みます。

〈例〉メモリ50 <9> メモリ51 <7>



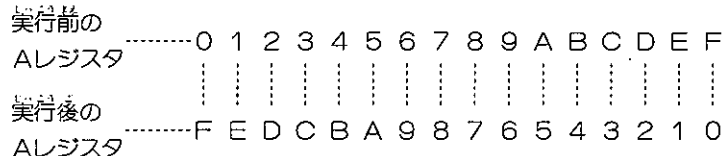
Ⓒ RESET, 50, RUN キーを順に押し実行します。

Ⓓ メモリ50、51を読み出します。小さい数の順に並べかえられます。



※ CAL CMPL 命令について

この命令は、Aレジスタを反転する命令で、反転する前の値と反転後の値をたし算するとすべてFになり、この関係を補数と言います。



No. 49 CIA めいれい命令を使った実験 つか じっけん

●プログラム

アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	TIY	A	25	M-	7
01	<0>	0	26	JUMP	F
02	MA	5	27	<3>	3
03	AIA	9	28	<0>	0
04	<6>	6	29	TIY	A
05	JUMP	F	2A	<4>	4
06	<3>	3	2B	AM	4
07	<0>	0	2C	AO	1
08	TIY	A	2D	JUMP	F
09	<1>	1	2E	<2>	2
0A	MA	5	2F	<D>	D
0B	CIA	C	30	TIY	A
0C		B	31	<0>	0
0D	JUMP	F	32	TIA	8
0E	<3>	3	33	<E>	E
0F	<0>	0	34	AO	1
10	TIY	A	35	TIA	8
11	<2>	2	36	<0>	0
12	MA	5	37	[CAL	E
13	AIA	9	38	[TIMR	C
14	<6>	6	39	[CAL	E
15	JUMP	F	3A	[RSTO	0
16	<3>	3	3B	A IY	B
17	<0>	0	3C	<1>	1
18	TIY	A	3D	C IY	D
19	<3>	3	3E	<5>	5
1A	MA	5	3F	JUMP	F
1B	CIA	C	40	<4>	4
1C	<E>	E	41	<5>	5
1D	JUMP	F	42	JUMP	F
1E	<3>	3	43	<4>	4
1F	<0>	0	44	<2>	2
20	TIY	A	45	[CAL	E
21	<2>	2	46	[TIMR	C
22	MA	5	47	JUMP	F
23	TIY	A	48	<3>	3
24	<0>	0	49	<2>	2

- メモリを次のように分けて演算をしてみましょう。

メモリ番地					
50	→	数字データ	52	→	数字データ
51	→	B (-記号)	53	→	E (=記号)

プログラムでこの演算記号の判断で演算を行い、結果は数字LEDに表示します。数字が10進数でない場合、演算記号に誤りのあった場合はEを表示します。引けない場合にもEを表示します。

- (A) プログラムの書き込みと、確認をしてください。
- (B) メモリ50、51、52、53番地にデータや記号を書き込みます。

(例) 50<9> 51 52<5> 53<E>



[9] [B(-)] [5] [E(=)] [答]

- (C) キーを順に押し実行します。
- (D) 50、52番地の値をいろいろ変えて実験してみましょう。

4

※ CIA命令について

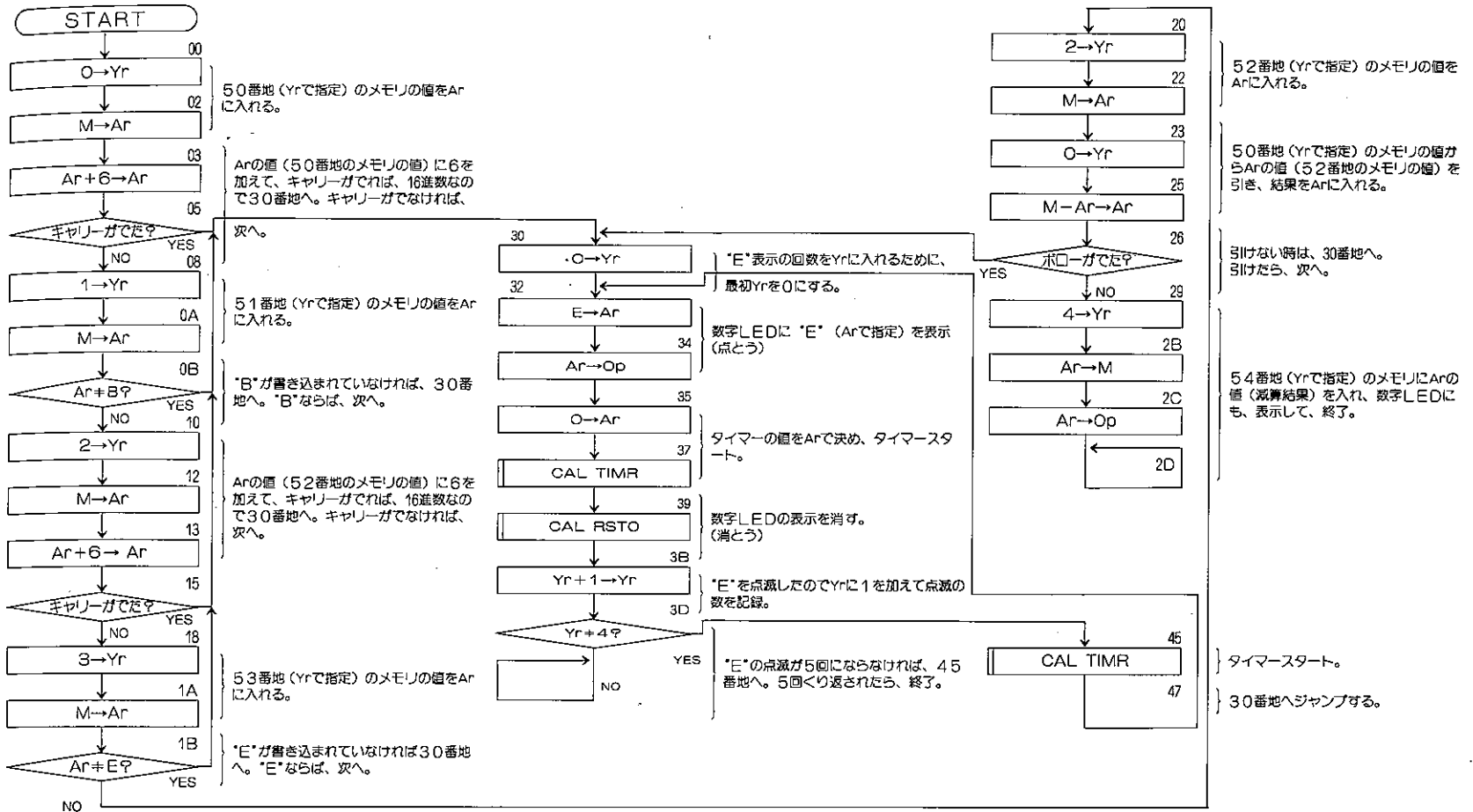
フローチャートでは、 $\langle Ar+? \rangle$ ($\circ \rightarrow 0 \sim F$) と書きます。レジスタから2行目に書いてあるデータを引きます。ひき算をしますがその答えはレジスタに書き込まれないのでAレジスタの値は変わりません。

ひき算の結果で実行フラグが変化します。 $\langle Ar-\circ=0 \rangle$ のとき、すなわちAレジスタと2行目のデータが同じ場合に実行フラグは0となります。 $\langle Ar-\circ \neq 0 \rangle$ のときすなわちAレジスタと2行目のデータが異なっているとき実行フラグは1となります。

この命令の次にJUMP命令やCAL命令を使って分岐させています。

命令語	マシンコード
命令.....→ CIA	C
データ.....→ <E>	E

(フローチャート)



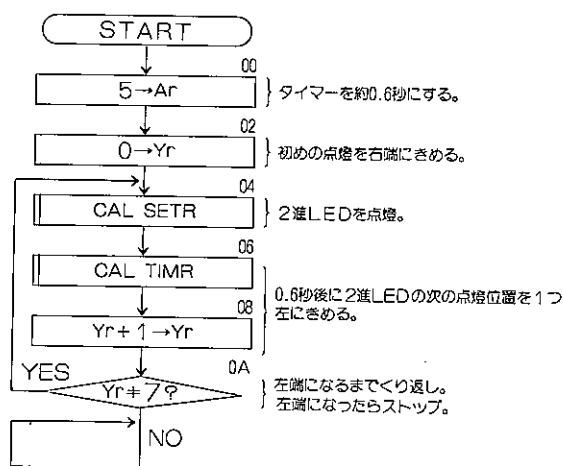
No. 51 2進LEDを右から左へ逐次点燈

しん みぎ ひたひ ちくじ てんとう
 右から左へ逐次点燈(結果は全部点燈)

●プログラム

アドレス	命令記号	命令コード
00	T I A	8
01	<5>	5
02	T I Y	A
03	<0>	0
04	CAL	E
05	SETR	1
06	CAL	E
07	TIMR	C
08	A I Y	B
09	<1>	1
0A	C I Y	D
0B	<7>	7
0C	JUMP	F
0D	<0>	0
0E	<4>	4
0F	JUMP	F
10	<0>	0
11	<F>	F

(フローチャート)



●2進LED表示をさせましょう。2進LED表示命令はサービスプログラムで、CAL SETRを使います。

- Ⓐ プログラムの書き込みと、確認をしてください。
- Ⓑ キーを順に押し実行すると2進LEDが全部点燈します。

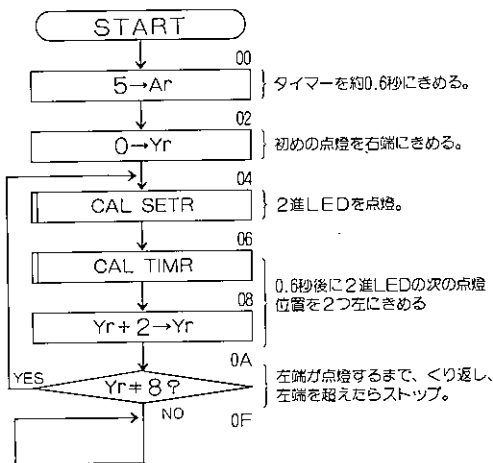
No. 52 2進LEDを右から左へ一つおきに^{しん}遂次点燈^{みぎ ひだり}

右から左へ一つ置きに^{みぎ ひだり}遂次点燈^{ちくじてんとう}(結果4ヶ点燈^{けっか てんとう})

●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<5>	5
02	TIY	A
03	<0>	0
04	CAL	E
05	SETR	1
06	CAL	E
07	TIMR	C
08	AIY	B
09	<2>	2
0A	CIY	D
0B	<8>	8
0C	JUMP	F
0D	<0>	0
0E	<4>	4
0F	JUMP	F
10	<0>	0
11	<F>	F

(フローチャート)



(A) プログラムの書き込みと、確認をしてください。

(B) キーを順に押し実行します。

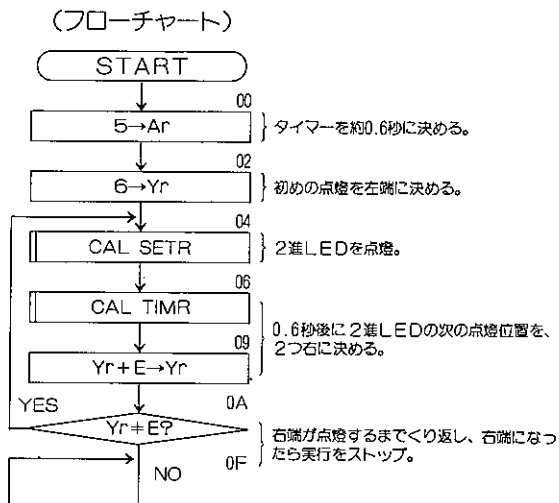
●一つ置きに点燈させるために、No51の $Y+1 \rightarrow Y$ を、 $Y+2 \rightarrow Y$ に変えてあります。

No. 53 2進LEDを左から右へ一つおきに逐次点燈

左から右へ1つ置きに逐次点燈(結果4つ点燈)

●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<5>	5
02	TIY	A
03	<6>	6
04	CAL	E
05	SETR	1
06	CAL	E
07	TIMR	C
08	AIY	B
09	<E>	E
0A	CIY	D
0B	<E>	E
0C	JUMP	F
0D	<0>	0
0E	<4>	4
0F	JUMP	F
10	<0>	0
11	<F>	F



Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ RESET 71 72 キーを順に押し実行します。

●左から右に一つ置きに点燈させるために、Yr + E → Yr を使っています。

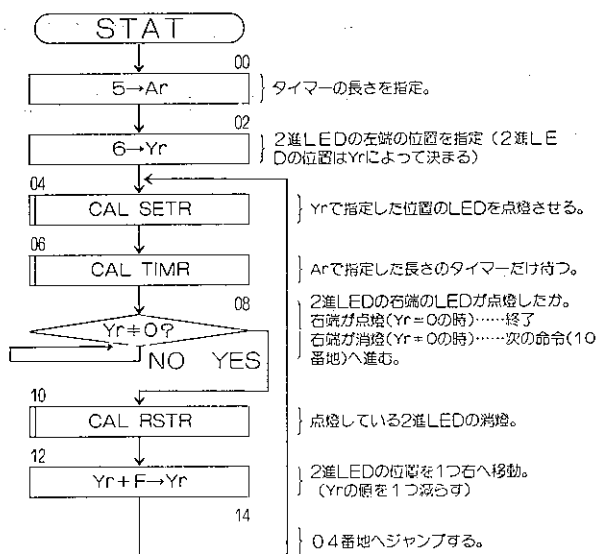
No. 54 2進LEDを左から右へ点滅しながら移動

左から右へ点燈が逐次移動(結果1ヶ点燈)

●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<5>	5
02	TIY	A
03	<6>	6
04	[CAL	E
05	[SETR	1
06	[CAL	E
07	[TIMR	C
08	CIY	D
09	<0>	0
0A	JUMP	F
0B	<1>	1
0C	<0>	0
0D	JUMP	F
0E	<0>	0
0F	<D>	D
10	[CAL	E
11	[RSTR	2
12	AIY	B
13	<F>	F
14	JUMP	F
15	<0>	0
16	<4>	4

(フローチャート)



(A) プログラムの書き込みと、確認をしてください。

(B) キーを順に押し実行します。



◆ No. 44と移動の方向を変えてみました。プログラムを少し変えることによってこんなに変化します。

No. 55 2進LEDを右左に往復点灯移動(その1)

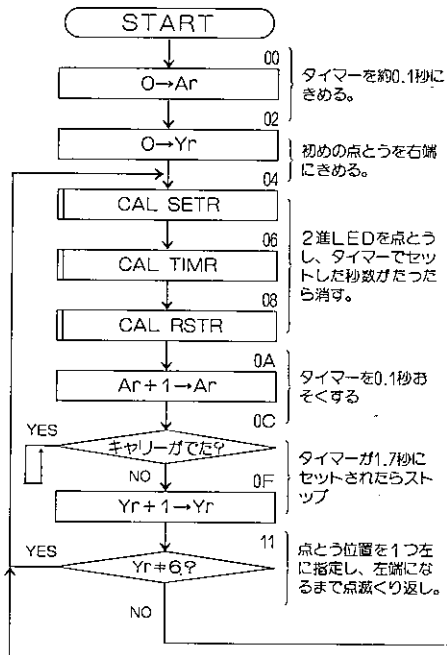
(高速→低速→停止)

●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<0>	0
02	TIY	A
03	<0>	0
04	[CAL	E
05	[SETR	1
06	[CAL	E
07	[TIMR	O
08	[CAL	E
09	[RSTR	E
0A	AIA	2
0B	<1>	9
0C	JUMP	F
0D	<0>	0
0E	<C>	C
0F	A IY	B
10	<1>	1
11	C IY	D
12	<6>	6
13	JUMP	F
14	<0>	0
15	<4>	4
16	[CAL	E
17	[SETR	1
18	[CAL	E
19	[TIMR	C
1A	[CAL	E
1B	[RSTR	2
1C	AIA	9
1D	<1>	1
1E	JUMP	F
1F	<1>	1
20	<E>	E
21	A IY	B
22	<F>	F
23	C IY	D
24	<0>	0
25	JUMP	F


アドレス	命令記号	命令コード
26	<1>	1
27	<6>	6
28	JUMP	F
29	<0>	0
2A	<4>	4

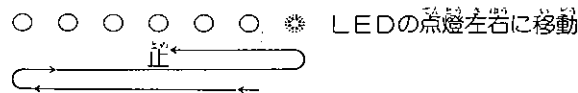
(フローチャート)



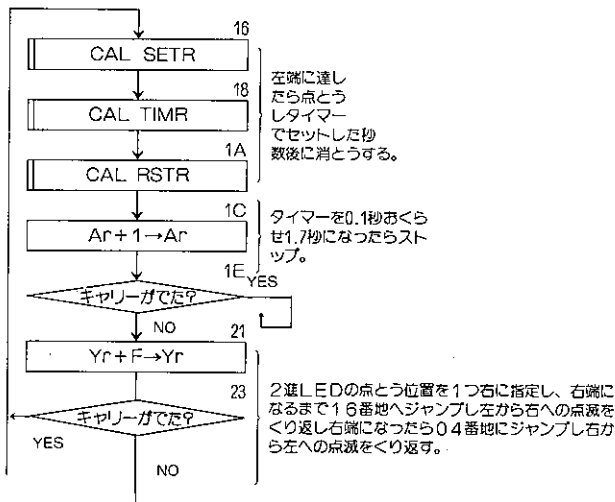
● タイマー定数 (レジスタ) をAIA命令で1をくり返し加えて行き表示スピードを変化させています。停止したときLEDは消えます。

Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ  キーを順に押し実行します。



● タイマー定数のレジスタが、Fから0になったときにこのプログラムは終了します。Ar+1→Arの次のJUMPでこの判断をしています。右の移動と左の移動の2ヶ所で判断をしています。



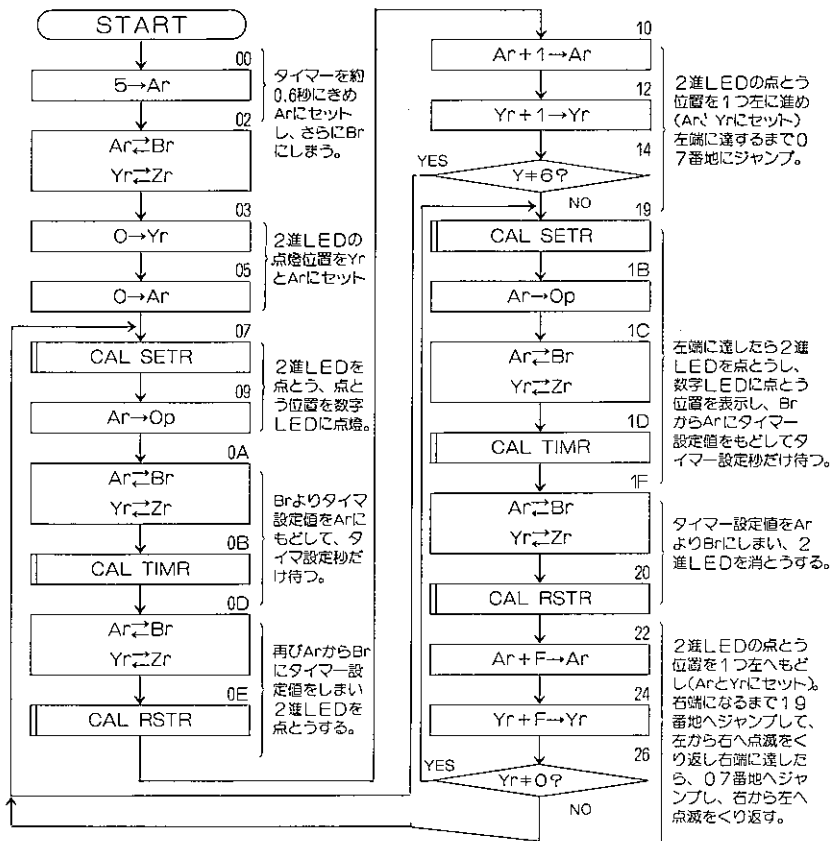
No. 56 2進LEDを右左に往復点燈移動(その2)

(数字LEDに点燈位置を表示)

●プログラム

アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	TIA	8	20	CAL	E
01	<5>	5	21	RSTR	2
02	CH	2	22	AIA	9
03	TIY	A	23	<F>	F
04	<0>	0	24	AIY	B
05	TIA	8	25	<F>	F
06	<0>	0	26	CIY	D
07	CAL	E	27	<0>	0
08	SETR	1	28	JUMP	F
09	AO	1	29	<1>	1
0A	CH	2	2A	<9>	9
0B	CAL	E	2B	JUMP	F
0C	TIMR	C	2C	<0>	0
0D	CH	2	2D	<7>	7
0E	CAL	E			
0F	RSTR	2			
10	AIA	9			
11	<1>	1			
12	AIY	B			
13	<1>	1			
14	CIY	D			
15	<6>	6			
16	JUMP	F			
17	<0>	0			
18	<7>	7			
19	CAL	E			
1A	SETR	1			
1B	AO	1			
1C	CH	2			
1D	CAL	E			
1E	TIMR	C			
1F	CH	2			

(フローチャート)



● 点燈している2進LEDの位置を、数字LEDに表示させるプログラムです。

(A) プログラムの書き込みと、確認をしてください。

(B) キーを順に押し実行します。

● Ar→Op で、2進LEDの位置の表示をしています。

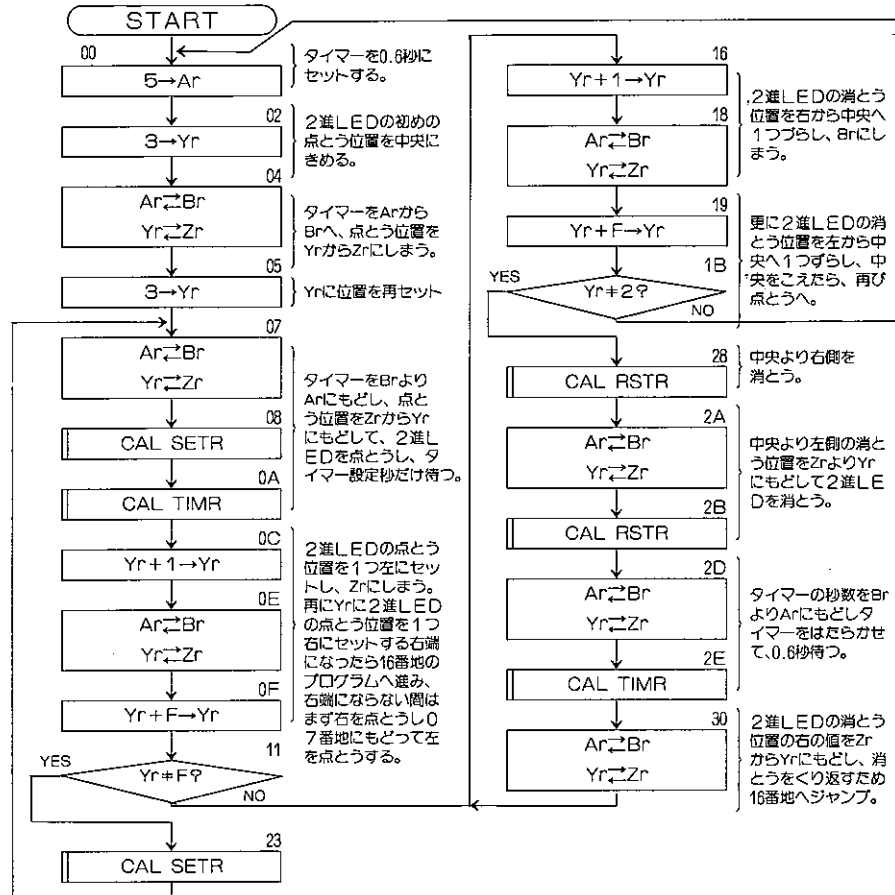
No. 57 2進LEDを中央より両端へ点燈移動

●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<5>	5
02	TIY	A
03	<3>	3
04	CH	2
05	TIY	A
06	<3>	3
07	CH	2
08	[CAL	E
09	[SETR	1
0A	[CAL	E
0B	[TIMR	C
0C	AIY	B
0D	<1>	1
0E	CH	2
0F	AIY	B
10	<F>	F
11	CIY	D
12	<F>	F
13	JUMP	F
14	<2>	2
15	<3>	3
16	AIY	B
17	<1>	1
18	CH	2
19	AIY	B
1A	<F>	F
1B	CIY	D
1C	<2>	2
1D	JUMP	F
1E	<2>	2
1F	<8>	8

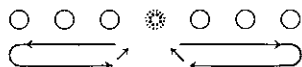
アドレス	命令記号	命令コード
20	JUMP	F
21	<0>	0
22	<0>	0
23	[CAL	E
24	[SETR	1
25	JUMP	F
26	<0>	0
27	<7>	7
28	[CAL	E
29	[RSTR	2
2A	CH	2
2B	[CAL	E
2C	[RSTR	2
2D	CH	2
2E	[CAL	E
2F	[TIMR	C
30	CH	2
31	JUMP	F
32	<1>	1
33	<6>	6

(フローチャート)



(A) プログラムの書き込みと、確認をしてください。

(B) キーを順に押し実行します。

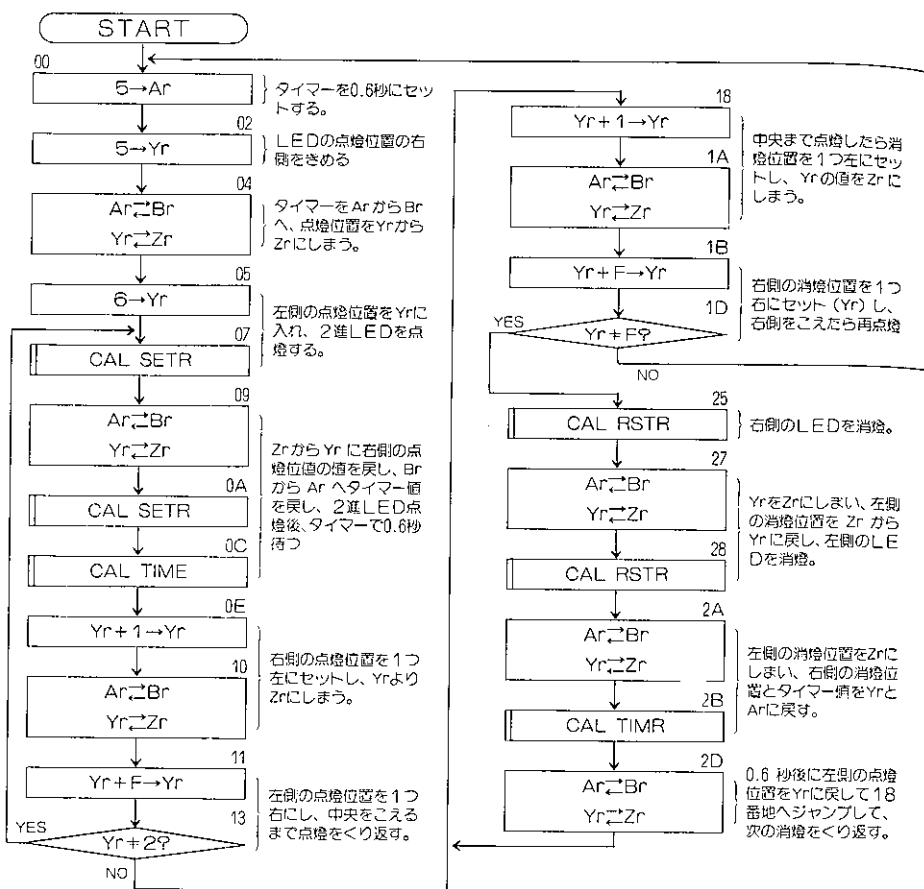


No. 58 2進LEDを両端より中央へ点灯移動

●プログラム

アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	TIA	8	19	<1>	1
01	<5>	5	1A	CH	2
02	TIY	A	1B	AIY	B
03	<0>	0	1C	<F>	F
04	CH	2	1D	CIY	D
05	TIY	A	1E	<F>	F
06	<6>	6	1F	JUMP	F
07	CAL	E	20	<2>	2
08	SETR	1	21	<5>	5
09	CH	2	22	JUMP	F
0A	CAL	E	23	<0>	0
0B	SETR	1	24	<0>	0
0C	CAL	E	25	CAL	E
0D	TIMR	C	26	RSTR	2
0E	AIY	B	27	CH	2
0F	<1>	1	28	CAL	E
10	CH	2	29	RSTR	2
11	AIY	B	2A	CH	2
12	<F>	F	2B	CAL	E
13	CIY	D	2C	TIMR	C
14	<2>	2	2D	CH	2
15	JUMP	F	2E	JUMP	F
16	<0>	0	2F	<1>	1
17	<7>	7	30	<8>	8
18	AIY	B			

((フローチャート))



●ネオンサインによく似た動きをする(光の点滅をくり返す)プログラムを作ってみました。

(A) プログラムの書き込みと、確認をしてください。

(B) キーを押し実行します。

●●●●●●●●●● “両端から中央へ点灯、中央から両端へ消灯”

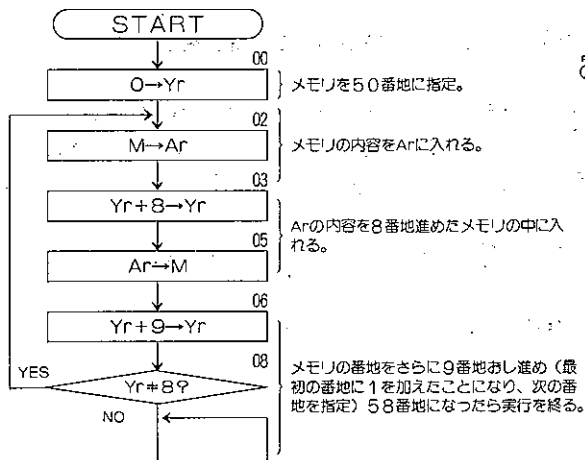
●点灯のスピードを変えるには、01番地のTIA、5のデータの値をいろんな値(0~F)に変えて実験してください。

No. 59 50~57番地より58~5F番地への移動

●プログラム

アドレス	命令記号	命令コード
00	TI Y	A
01	<0>	0
02	MA	.5
03	AI Y	B
04	<8>	8
05	AM	4
06	AI Y	B
07	<9>	9
08	CI Y	D
09	<8>	8
0A	JUMP	F
0B	<0>	0
0C	<2>	2
0D	JUMP	F
0E	<0>	0
0F	<D>	D

(フローチャート)



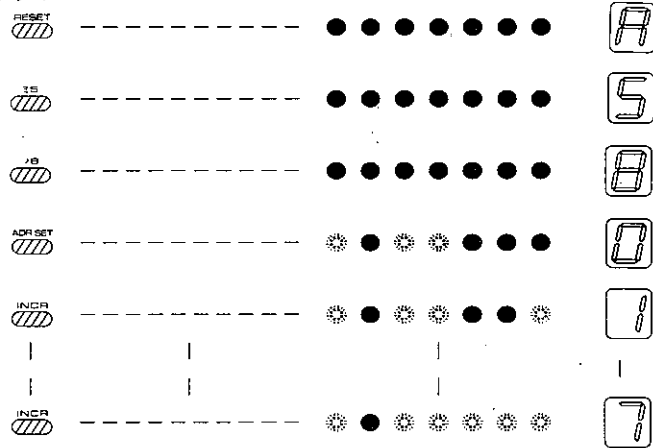
〈例〉メモリ50<0> 51<1> 52<2> 53<3> 54<4> 55<5>
56<6> 57<7>



Ⓒ RESET IS ADR SET O INCR 7 INCR キーを順に押し実行します。

Ⓓ 58~5Fへ転送されているか確認してみましょう。

押すキー



● メモリからメモリを移動させることを転送といいます。大型のコンピュータでは、たくさんのデータを短時間に転送させることがよくあります。FX-マイコンでは、データは少ないのですが転送のプログラムを作ってみました。

Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ メモリ50~57番地にデータを書き込みます。

● $Yr+8 \rightarrow Yr$ で50番地から転送先の58番地にメモリの指定を変えて、

$Yr+9 \rightarrow Yr$ でメモリの指定を1番地づつ変えています。これを8回くり返すことによって50~57のメモリは全て58~5Fへ転送されます。

実験の確認をしやすくするために58~5Fには0を書き込んでおきます。

No. 60 ないメモリ内の5以下の数いかがいくつあるか表示する実験ひょうじ

●プログラム

アドレス	命令記号	命令コード
00	T I A	8
01	<0>	0
02	CH	2
03	T I Y	A
04	<0>	0
05	MA	5
06	A I A	9
07	<A>	A
08	JUMP	F
09	<0>	0
0A	<F>	F
0B	CH	2
0C	A I A	9
0D	<1>	1
0E	CH	2
0F	A I Y	B
10	<1>	1
11	C I Y	D
12	<F>	F
13	JUMP	F
14	<0>	0
15	<5>	5
16	CH	2
17	AO	1
18	JUMP	F
19	<1>	1
1A	<8>	8

● 加算を使ったメモリチェックをしてみましょう。

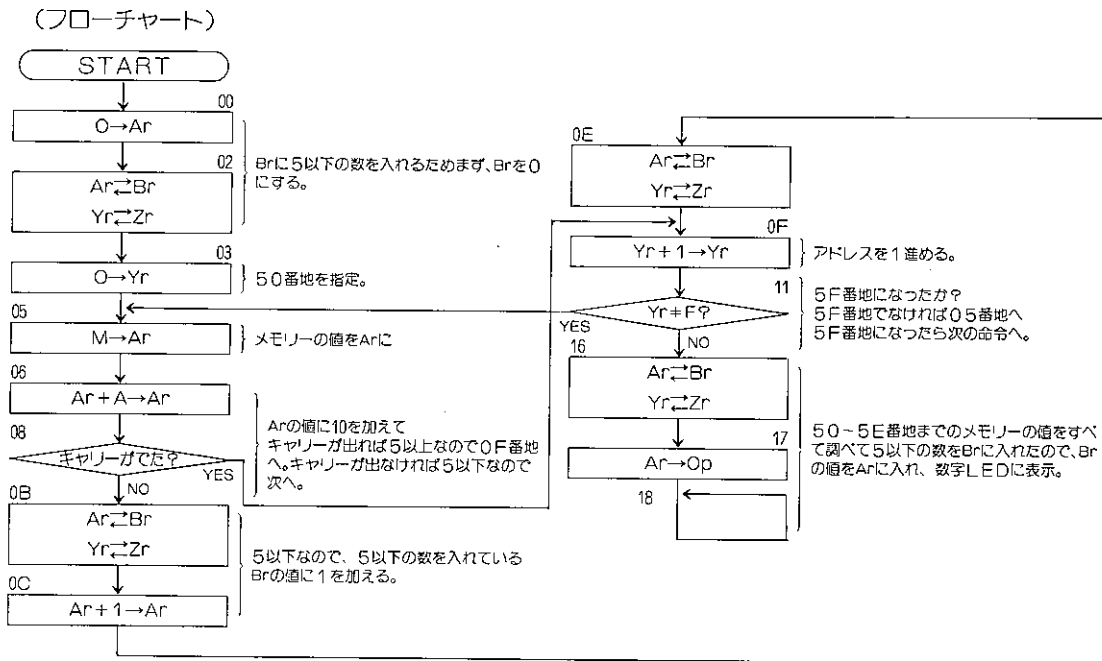
Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ メモリ50~5Eまでデータを書き込みます。
 (例) メモリ50~58<3>、59~5E<7>



Ⓒ キーを順に押し実行します--と

● 数字5以下のデータの数を、Bレジスタに書き込んでいます。

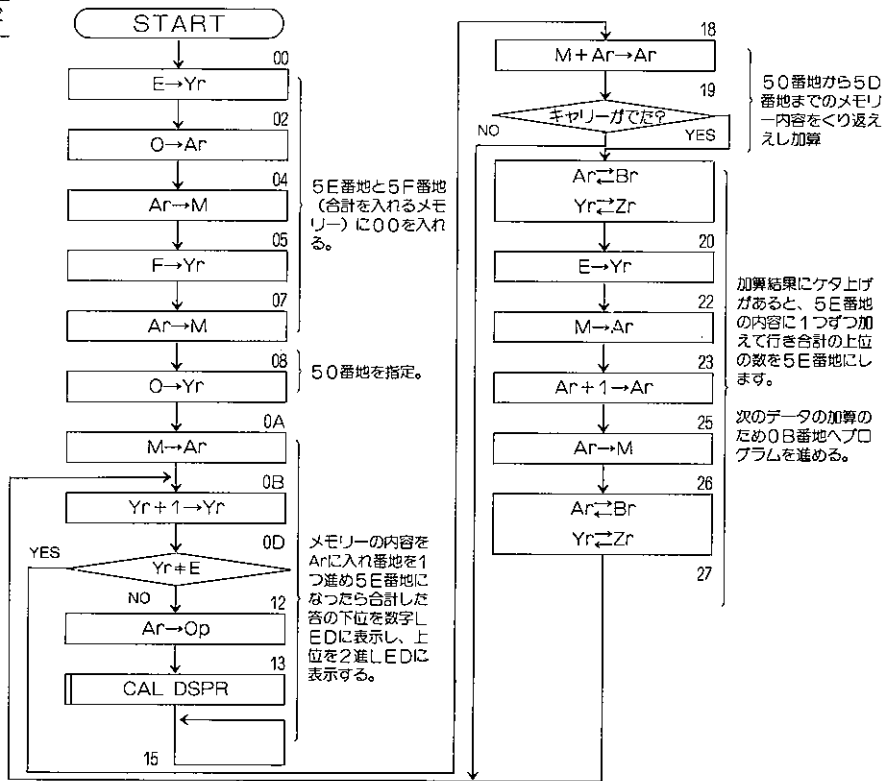


No. 61 メモリ内のデータ合計を表示する実験

●プログラム

アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	TIY	A	15	JUMP	F
01	<E>	E	16	<1>	1
02	TIA	8	17	<5>	5
03	<0>	0	18	M+	6
04	AM	4	19	JUMP	F
05	TIY	A	1A	<1>	1
06	<F>	F	1B	<F>	F
07	AM	4	1C	JUMP	F
08	TIY	A	1D	<0>	0
09	<0>	0	1E		B
0A	MA	5	1F	CH	2
0B	AIY	B	20	TIY	A
0C	<1>	1	21	<E>	E
0D	CIY	D	22	MA	5
0E	<E>	E	23	AIA	9
0F	JUMP	F	24	<1>	1
10	<1>	1	25	AM	4
11	<8>	8	26	CH	2
12	AO	1	27	JUMP	F
13	CAL	E	28	<0>	0
14	DSPR	D	29		B

(フローチャート)



● 14個のデータの加算です。

(A) プログラムの書き込みと、確認をしてください。

(B) メモリ50~5Dまで、データを書き込みます。

(例) 50~55<2> 56~59<1> 5A~5D<3>
2+2+2+2+2+2+1+1+1+1+1+3+3+3+3=10C(28)

RESET 15 0 ADDRESS 5B INCR 29 19 INCR

(C) RESET 71 RUN キーを順に押し実行します。

4 2 1 8 4 2 1

● ● ● ● ● ● ●



答えは16進数
で1C(28)
と表示します。

● 16進加算で、答えが2進LED、数字LEDに出できます。答えがあつ
ているか確かめるのに大変です。根気よく元張ってください。

データを変えて実験してみましょう。

No. 62 メモリ内のデータ平均値を表示する実験

50～5D番地内14個のデータの平均値(16進数)を表示する

※プログラムのメモリが、すくないので、割り切れない値の場合は、余りを切り捨ててあります。

●プログラム

アドレス	命令記号	命令コード
00	T I Y	A
01	<E>	E
02	T I A	8
03	<0>	0
04	AM	4
05	T I Y	A
06	<F>	F
07	AM	4
08	T I Y	A
09	<0>	0
0A	MA	5
0B	A I Y	B
0C	<1>	1
0D	C I Y	D
0E	<E>	E
0F	JUMP	F
10	<3>	3
11		B
12	CH	2
13	T I A	8
14	<0>	0
15	CH	2
16	T I Y	A
17	<F>	F
18	AM	4
19	T I A	8
1A	<E>	E
1B	M-	7
1C	JUMP	F
1D	<2>	2
1E	<7>	7

アドレス	命令記号	命令コード
1F	AM	4
20	CH	2
21	A I A	9
22	<1>	1
23	CH	2
24	JUMP	F
25	<1>	1
26	<9>	9
27	AM	4
28	T I Y	A
29	<E>	E
2A	MA	5
2B	A I A	9
2C	<F>	F
2D	JUMP	F
2E	<3>	3
2F	<5>	5
30	CH	2
31	AO	1
32	JUMP	F
33	<3>	3
34	<2>	2
35	AM	4
36	T I Y	A
37	<F>	F
38	JUMP	F
39	<2>	2
3A	<0>	0
3B	M+	6
3C	JUMP	F
3D	<4>	4

アドレス	命令記号	命令コード
3E	<2>	2
3F	JUMP	F
40	<0>	0
41		B
42	CH	2
43	T I Y	A
44	<E>	E
45	MA	5

アドレス	命令記号	命令コード
46	A I A	9
47	<1>	1
48	AM	4
49	CH	2
4A	JUMP	F
4B	<0>	0
4C		B

●平均値は全てのデータを加算して、データの数で割ればよいのです。このプログラムでは割り算で、余りが出ることがありますが、余りは切り捨てています。

(A) プログラムの書き込みと、確認をしてください。

(B) メモリ50～5Dにデータを書き込みます。

(例) 50～54<4> 55～59<6> 5A～5D<1>



(C) キーを順に押し実行します。

(D) 答えは データの加算 ÷ データの数で割 = 平均値

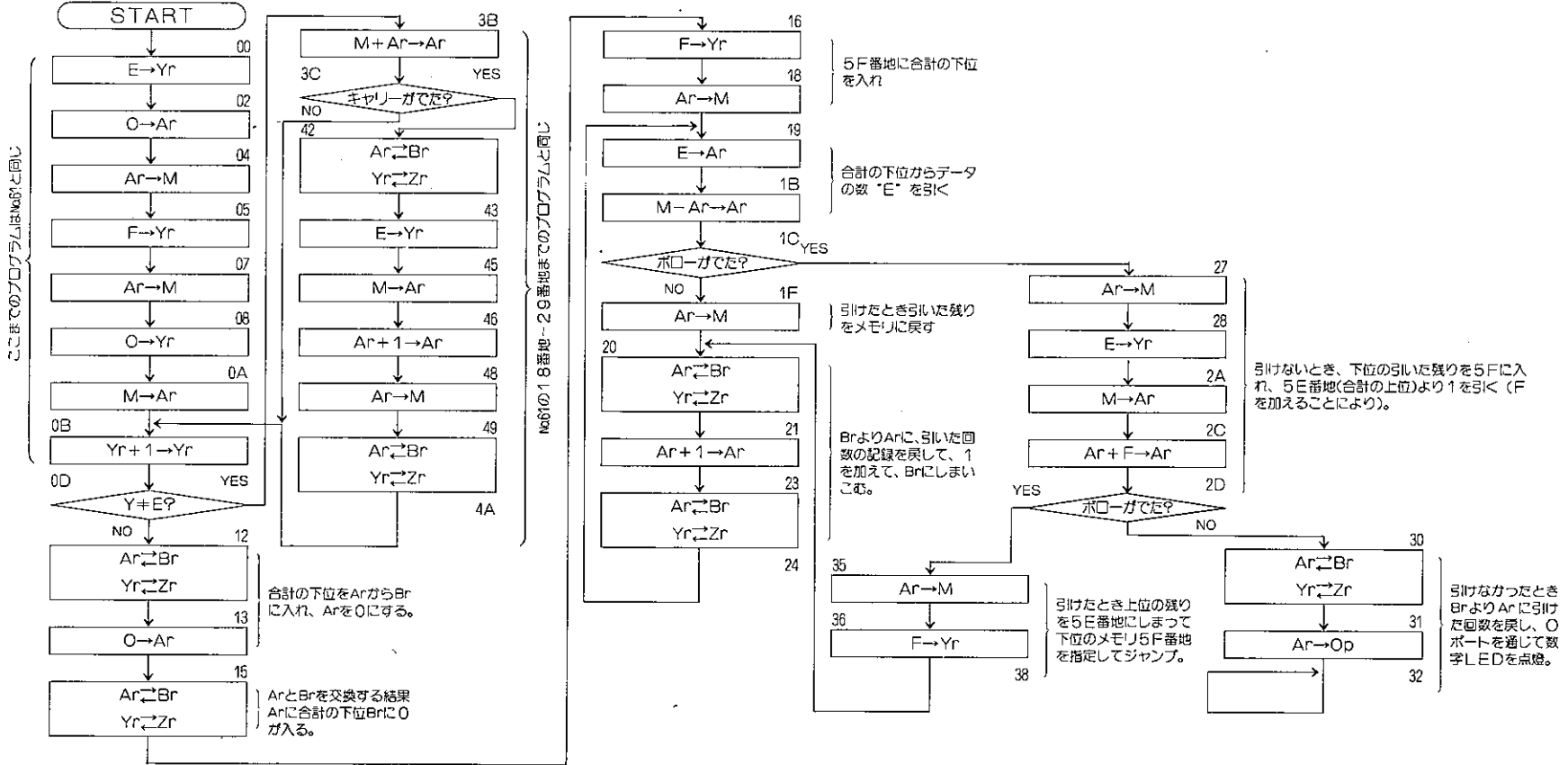
$$(4+4+4+4+4+6+6+6+6+6+1+1+1+1) \div 14 \div 3$$

(E) メモリ50～5Dのデータをいろいろ変えて、実験しましょう。割り切れない値が出る場合は、余りを切り捨てています。

●割り算はひき算のくり返して行ないます。引けた回数はBレジスタに入り、それを答えとして数字LEDに表示します。

これだけ長いプログラムになるとフローチャートも迷路/パズルのようになります。マイコンのプログラムも謎をとくつもりでじっくり考えましょう。

(フローチャート)



No. 63 ^{けた}2桁+^{けた}2桁の^{しん か さん}16進^{じっけん}加算の実験

●プログラム

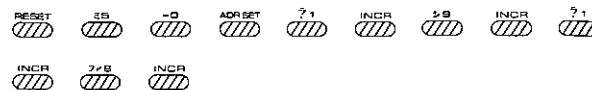
アドレス	命令記号	命令コード	アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	TIY	A	1C	<A>	A	38	<3>	3
01	<4>	4	1D	CIY	D	39	M+	6
02	TIA	8	1E	<F>	F	3A	AM	4
03	<0>	0	1F	JUMP	F	3B	AIY	B
04	AM	4	20	<0>	0	3C	<F>	F
05	TIY	A	21	<D>	D	3D	MA	5
06	<5>	5	22	TIY	A	3E	AIA	9
07	AM	4	23	<6>	6	3F	<1>	1
08	TIY	A	24	MA	5	40	AM	4
09	<6>	6	25	AO	1	41	AIY	B
0A	AM	4	26	TIY	A	42	<1>	1
0B	TIY	A	27	<5>	5	43	JUMP	F
0C	<1>	1	28	MA	5	44	<1>	1
0D	MA	5	29	TIY	A	45		B
0E	AIY	B	2A	<E>	E	46	JUMP	F
0F	<2>	2	2B	AM	4	47	<1>	1
10	M+	6	2C	TIY	A	48		B
11	JUMP	F	2D	<4>	4			
12	<3>	3	2E	MA	5			
13	<7>	7	2F	TIY	A			
14	AIY	B	30	<F>	F			
15	<3>	3	31	AM	4			
16	M+	6	32	[CAL	E			
17	JUMP	F	33	[DSPR	D			
18	<3>	3	34	JUMP	F			
19	<A>	A	35	<3>	3			
1A	AM	4	36	<4>	4			
1B	AIY	B	37	AIY	B			

Ⓐ プログラムの書き込みと、確認をしてください。

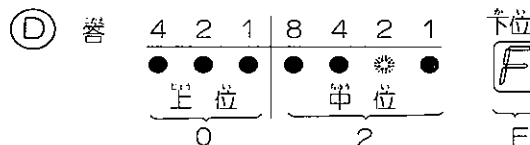
Ⓑ メモリ50~51、メモリ52~53にデータを
書き込みます。

(例) 50~51<19> 52~53<16>

50	51	52	53	54	55	56
1	9	1	6		2	F



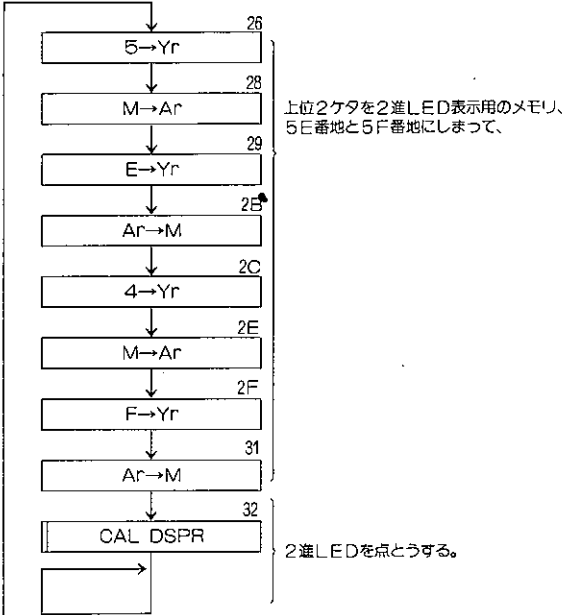
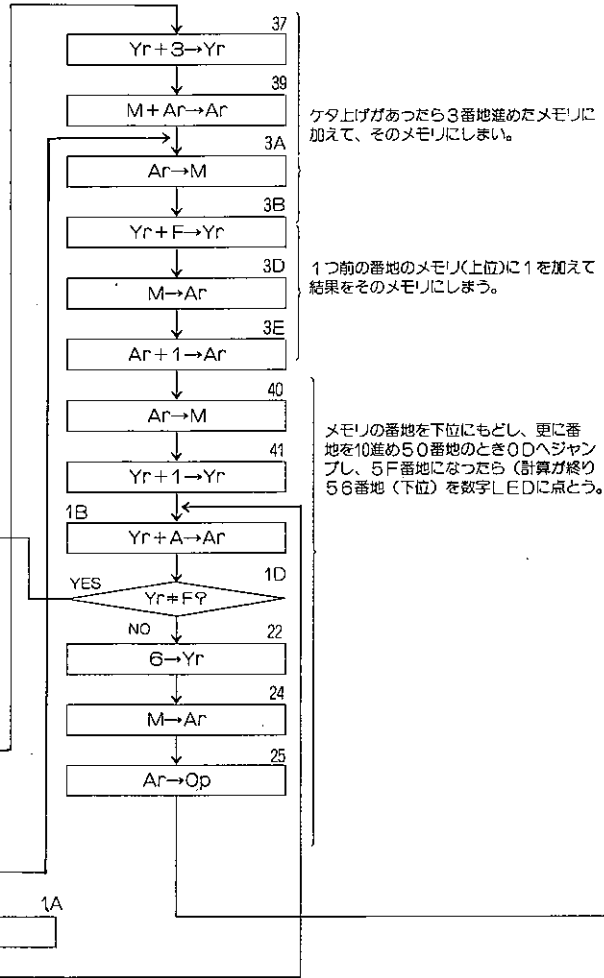
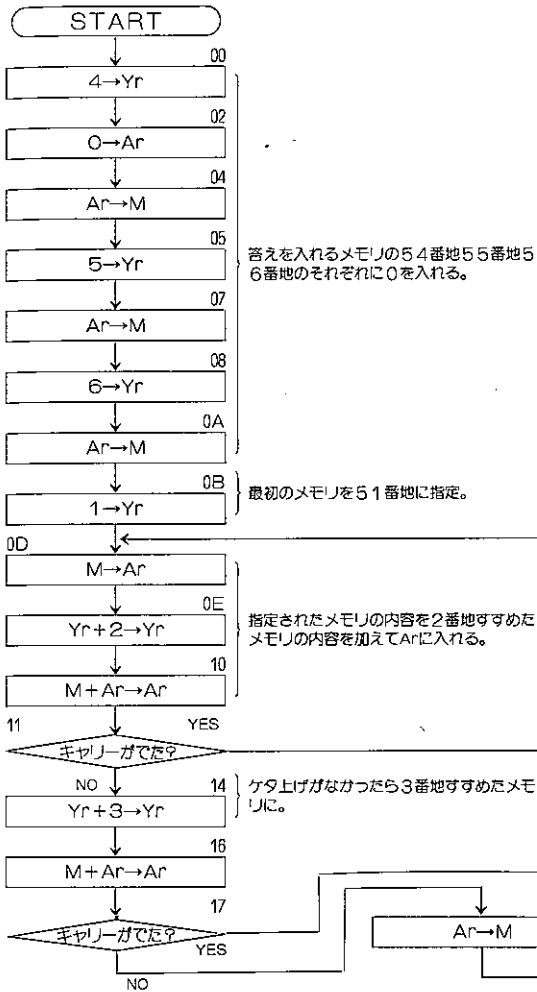
Ⓒ キーを順に押し実行します。



Ⓢ 16進数計算で
表示します。

●結果は、表示部全てを使って表示します。上位ケタと中位ケタは、2進数表示になります。

(フローチャート)



No. 64 2桁-2桁の16進減算の実験

2桁-2桁の16進数減算をし結果を表示する(引けないときはEを表示)

●プログラム

アドレス	命令記号	命令コード
00	T I Y	A
01	<0>	0
02	MA	5
03	T I Y	A
04	<4>	4
05	AM	4
06	T I Y	A
07	<1>	1
08	MA	5
09	T I Y	A
0A	<5>	5
0B	AM	4
0C	A I Y	B
0D	<E>	E
0E	MA	5
0F	A I Y	B
10	<2>	2
11	M-	7
12	JUMP	F
13	<2>	2
14	<E>	E
15	AM	4
16	A I Y	B
17	<D>	D
18	C I Y	D
19	<1>	1
1A	JUMP	F
1B	<0>	0
1C	<E>	E

アドレス	命令記号	命令コード
1D	T I Y	A
1E	<E>	E
1F	AM	4
20	T I Y	A
21	<F>	F
22	T I A	8
23	<0>	0
24	AM	4
25	[CAL	E
26	[DSPR	D
27	T I Y	A
28	<5>	5
29	MA	5
2A	AO	1
2B	JUMP	F
2C	<2>	2
2D		B
2E	AM	4
2F	A I Y	B
30	<F>	F
31	C I Y	D
32	<4>	4
33	JUMP	F
34	<3>	3
35	<C>	C
36	MA	5
37	A I A	9
38	<F>	F
39	JUMP	F

アドレス	命令記号	命令コード
3A	<0>	0
3B		B
3C	T I A	8
3D	<E>	E
3E	AO	1
3F	T I A	8
40	<2>	2
41	[CAL	E
42	[TMR	C

アドレス	命令記号	命令コード
43	[CAL	E
44	[RSTO	0
45	[CAL	E
46	[TMR	C
47	JUMP	F
48	<3>	3
49	<C>	C

Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ メモリ50~51、メモリ52~53にデータを書き込みます。
(例) 50~51 <32> 引かれる数、52~53 <19> 引く数

$$\begin{array}{|c|c|} \hline 50 & 51 \\ \hline 3 & 2 \\ \hline \end{array} - \begin{array}{|c|c|} \hline 52 & 53 \\ \hline 1 & 9 \\ \hline \end{array} = \begin{array}{|c|c|} \hline 54 & 55 \\ \hline 1 & 9 \\ \hline \end{array}$$

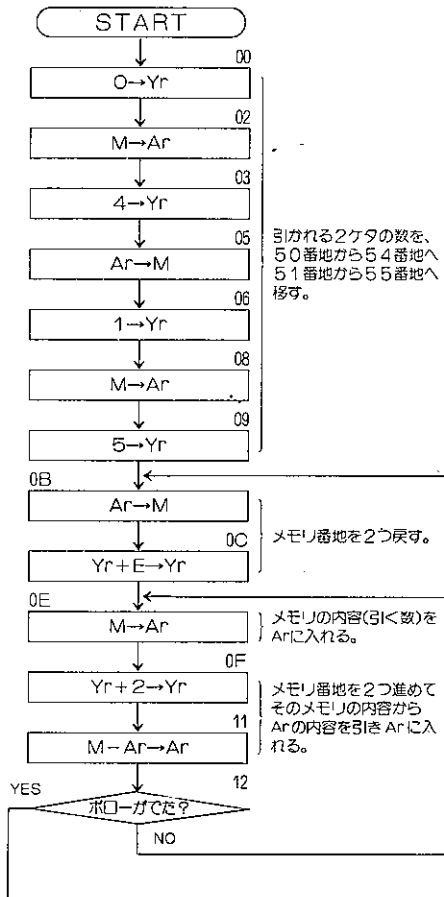
Ⓒ 16進数減算

Ⓒ RESET、0、11のキーを順に押し実行します。

Ⓓ $\begin{array}{|c|c|c|c|} \hline 4 & 2 & 1 & \\ \hline \bullet & \bullet & \bullet & \bullet \end{array} \mid \begin{array}{|c|c|c|c|} \hline 8 & 4 & 2 & 1 \\ \hline \bullet & \bullet & \bullet & \bullet \end{array} \begin{array}{|c|} \hline \text{7} \\ \hline \end{array}$ 16進数計算で表示します

◆結果は、表示部全てを使って表示します。上位ケタと中位ケタは、2進表示になります。

(フローチャート)

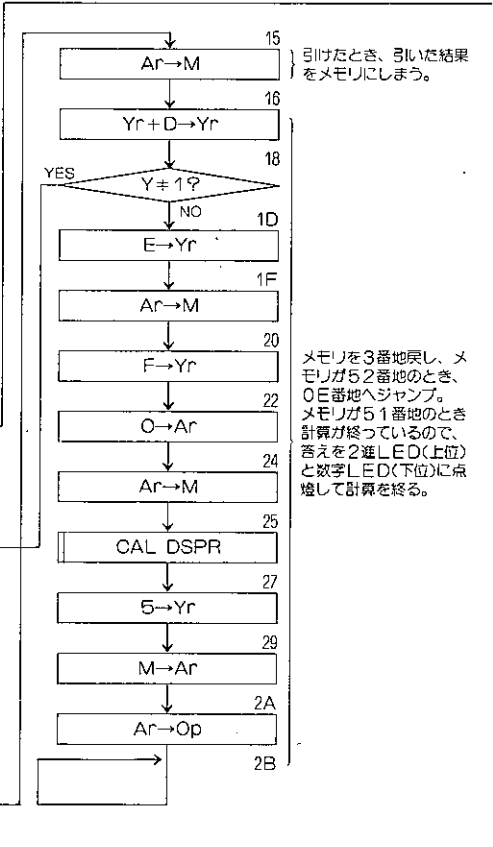


引かれる2ケタの数を、50番地から54番地へ51番地から55番地へ移す。

メモリ番地を2つ戻す。

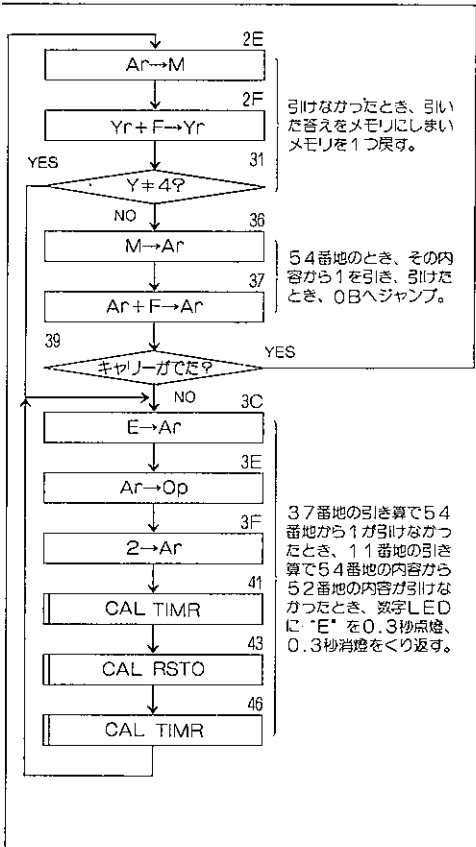
メモリの内容(引く数)をArに入れる。

メモリ番地を2つ進めてそのメモリの内容からArの内容を引きArに入れる。



引けたとき、引いた結果をメモリにしまう。

メモリを3番地戻し、メモリが52番地するとき、0E番地へジャンプ。メモリが51番地するとき計算が終っているので、答えを2進LED(上位)と数字LED(下位)に点燈して計算を終る。



引けなかったとき、引いた答えをメモリにしまいメモリを1つ戻す。

54番地するとき、その内容から1を引き、引けたとき、0Bへジャンプ。

37番地の引き算で54番地から1が引けなかったとき、11番地の引き算で54番地の内容から52番地の内容が引けなかったとき、数字LEDに"E"を0.3秒点燈、0.3秒消燈をくり返す。

No. 65 2桁×1桁の16進乗算の実験

2桁×1桁の16進数乗算(結果は53~55番地に3桁で入れる)

●プログラム

アドレス	命令記号	命令コード
00	T I Y	A
01	<5>	5
02	T I A	8
03	<0>	0
04	AM	4
05	A I Y	B
06	<F>	F
07	C I Y	D
08	<2>	2
09	JUMP	F
0A	<0>	0
0B	<4>	4
0C	MA	5
0D	C I A	C
0E	<0>	0
0F	JUMP	F
10	<1>	1
11	<5>	5
12	JUMP	F
13	<1>	1
14	<2>	2
15	A I A	9
16	<F>	F
17	CH	2
18	T I Y	A
19	<1>	1
1A	MA	5
1B	A I Y	B
1C	<4>	4

アドレス	命令記号	命令コード
1D	M+	6
1E	JUMP	F
1F	<2>	2
20	<D>	D
21	AM	4
22	A I Y	B
23		B
24	C I Y	D
25	<F>	F
26	JUMP	F
27	<1>	1
28	<A>	A
29	CH	2
2A	JUMP	F
2B	<0>	0
2C	<D>	D
2D	AM	4
2E	A I Y	B
2F	<F>	F
30	T I A	8
31	<1>	1
32	M+	6
33	JUMP	F
34	<3>	3
35	<F>	F
36	AM	4
37	A I Y	B
38	<1>	1
39	JUMP	F

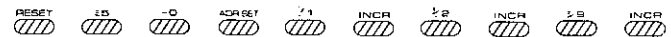
アドレス	命令記号	命令コード
3A	<2>	2
3B	<2>	2
3C	JUMP	F
3D	<2>	2
3E	<2>	2
3F	AM	4
40	A I Y	B
41	<F>	F
42	T I A	8

アドレス	命令記号	命令コード
43	<1>	1
44	M+	6
45	AM	4
46	T I Y	A
47	<5>	5
48	JUMP	F
49	<2>	2
4A	<2>	2

(A) プログラムの書き込みと、確認をしてください。

(B) メモリ50、51に2ケタ、メモリ52に1ケタを書き込みます。

(例)
$$\begin{array}{|c|c|} \hline 50 & 51 \\ \hline 1 & 2 \\ \hline \end{array} \times \begin{array}{|c|} \hline 52 \\ \hline 9 \\ \hline \end{array} = \begin{array}{|c|c|c|} \hline 53 & 54 & 55 \\ \hline 0 & A & 2 \\ \hline \end{array}$$

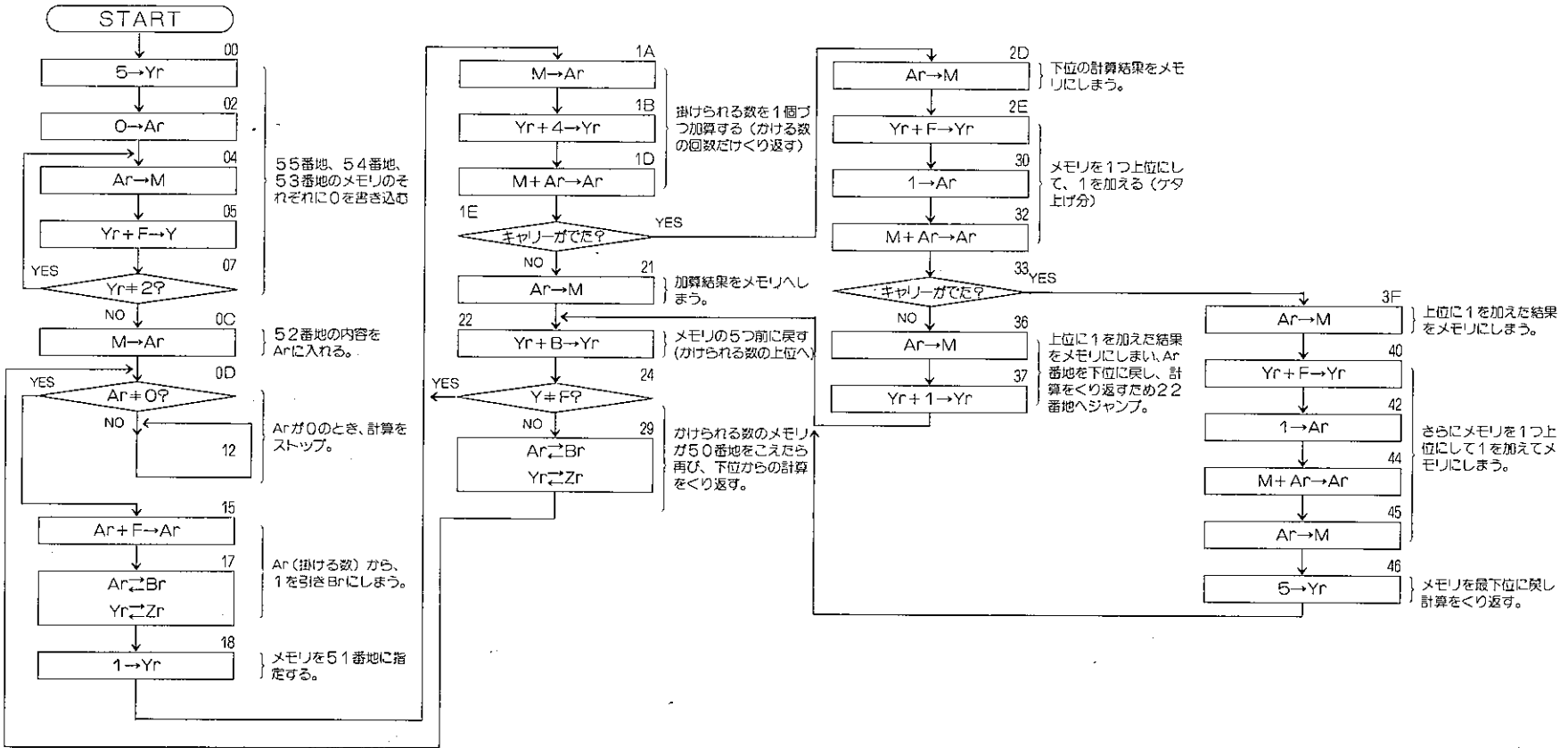


(C) RESET, 2, RUN キーを順に押し実行します。



④ 16進数計算で表示

(フローチャート)



No. 66 2桁÷1桁の16進除算の実験

(答えは2進LEDに2ケタで表示し、余りは数字LEDに表示)

●プログラム

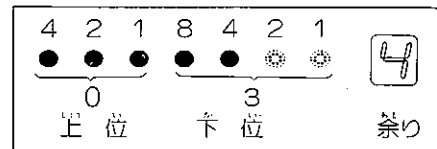
アドレス	命令記号	命令コード	アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	TIY	A	1C	CH	2	38	TIY	A
01	<8>	8	1D	AIA	9	39	<A>	A
02	MA	5	1E	<1>	1	3A	MA	5
03	TIY	A	1F	JUMP	F	3B	TIY	A
04	<C>	C	20	<2>	2	3C	<D>	D
05	AM	4	21	<5>	5	3D	M+	6
06	TIY	A	22	JUMP	F	3E	AM	4
07	<9>	9	23	<1>	1	3F	AO	1
08	MA	5	24	<1>	1	40	CH	2
09	TIY	A	25	CH	2	41	TIY	A
0A	<D>	D	26	TIY	A	42	<C>	C
0B	AM	4	27		B	43	AM	4
0C	TIA	8	28	MA	5	44	TIY	A
0D	<0>	0	29	AIA	9	45	<E>	E
0E	TIY	A	2A	<1>	1	46	AM	4
0F		B	2B	AM	4	47	TIY	A
10	AM	4	2C	JUMP	F	48		B
11	CH	2	2D	<1>	1	49	MA	5
12	TIY	A	2E	<2>	2	4A	TIY	A
13	<A>	A	2F	AM	4	4B	<F>	F
14	MA	5	30	TIY	A	4C	AM	4
15	TIY	A	31	<C>	C	4D	[CAL	E
16	<D>	D	32	MA	5	4E	[DSPR	D
17	M-	7	33	AIA	9	4F	JUMP	F
18	JUMP	F	34	<F>	F	50	<4>	4
19	<2>	2	35	JUMP	F	51	<F>	F
1A	<F>	F	36	<1>	1			
1B	AM	4	37		B			

(A) プログラムの書き込みと、確認をしてください。

(B) メモリ58~59に割られる数、5Aに割る数を書き込みます。(5Aには0は入れないこと)
 (例)メモリ58<1> 59<6> 5A<6>



(C) RESET, 7.1, RUN キーを順に押し実行しますと。

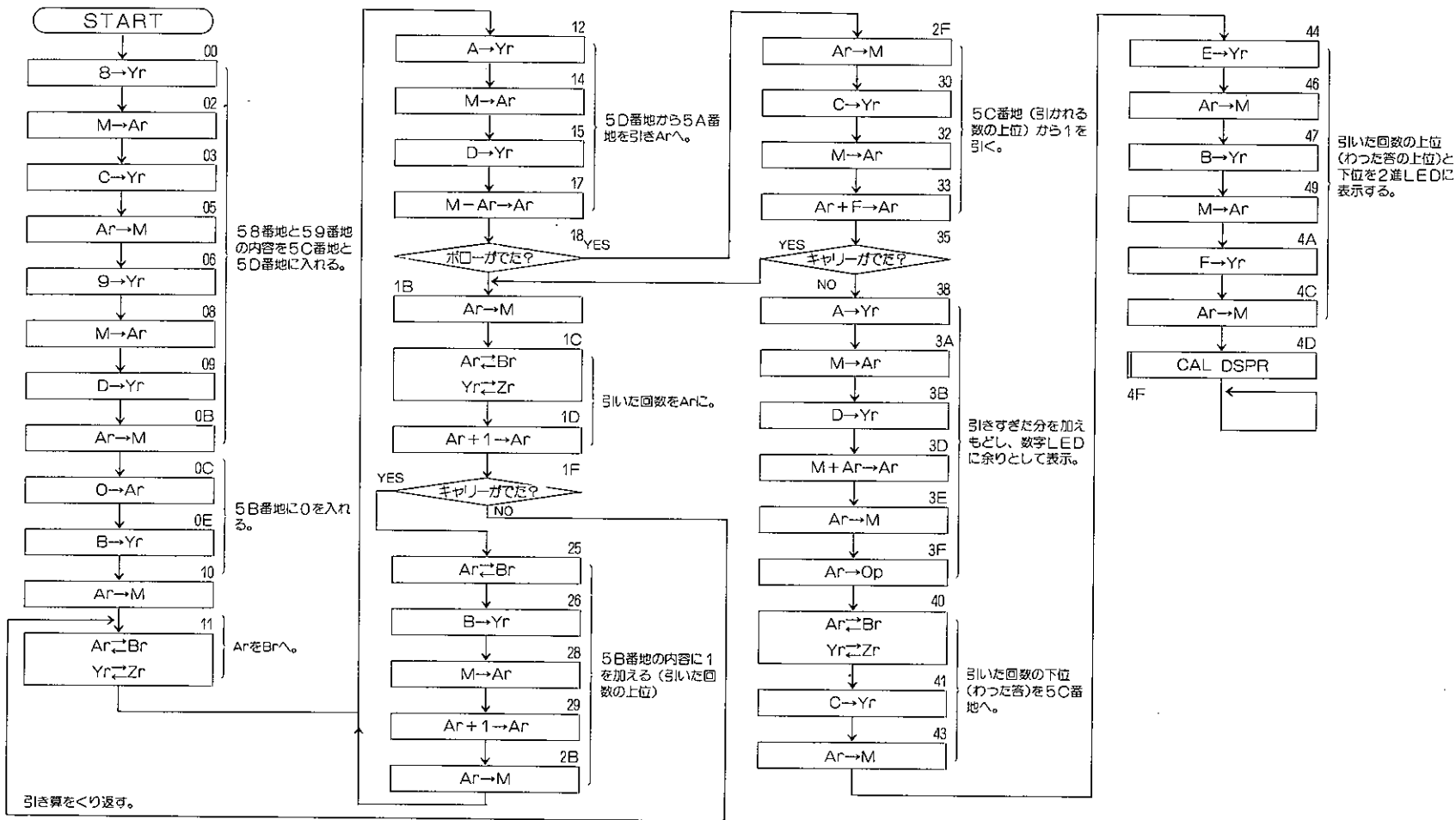


$$\begin{array}{|c|c|} \hline 58 & 59 \\ \hline 1 & 6 \\ \hline \end{array} \div \begin{array}{|c|} \hline 5A \\ \hline 6 \\ \hline \end{array} = \begin{array}{|c|c|} \hline 5B & 5C \\ \hline 0 & 3 \\ \hline \end{array} \oplus 16 \text{進数表示} \quad \text{余り} \begin{array}{|c|} \hline 4 \\ \hline \end{array}$$

$$30$$

●プログラムを注意してみてください。アドレスの最終番地4Fをこえています。実際のプログラムは、51番地まで書き込んであります。50~5Fは、プログラムで指定するメモリですが、ここにプログラムを書き込んであります。プログラムは、4Fをこえても続けて50以降に進みますから50~5Fまでもプログラムを書き込むことができます。しかしメモリ50~5Fを使う命令でプログラムが組んである場合は、命令によって、プログラムが書き換えられることがあるので注意しましょう。

(フローチャート)



(4) 第四群の命令について

第四群の命令は、CAL INPT、CAL CHNG、CAL SIFT、CAL ENDS、CAL ERRS、CAL LONS、CAL SUND、CAL DEM-、CAL DEM+の9つの命令です。命令を第一群から第四群に分けて説明してきましたが、この第四群の命令で、F X-マイコンの命令の説明はすべて終わります。

命令語が多いと覚えるのはたいへんですが、覚えている命令語が多いほど、プログラムが簡単にできることが、わかると思えます。

(ドレミファ音のデータ表)

レジスタの値	音階	レジスタの値	音階
0	無音	8	ラ
1	㇀	9	シ
2	シ	A	ド
3	ド	B	レ
4	レ	C	ミ
5	ミ	D	ファ
6	ファ	E	ソ
7	ソ	F	無音

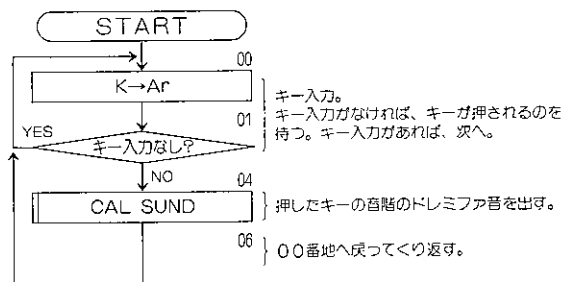
No. 67 CAL SUND命令を使った実験

CAL SUND命令を使ってドレミファ音を発生させるプログラムの基本

●プログラム

アドレス	命令記号	命令コード
00	KA	0
01	JUMP	F
02	<0>	0
03	<0>	0
04	CAL	E
05	SUND	B
06	JUMP	F
07	<0>	0
08	<0>	0

(フローチャート)



●簡単なオルガンを作ってみましょう。

- (A) プログラムの書き込みと、確認をしてください。
- (B) キーを順に押し実行します。
- (C) 迄のキーを押すと、ドレミファの音がでます。

※ CAL SUND命令について

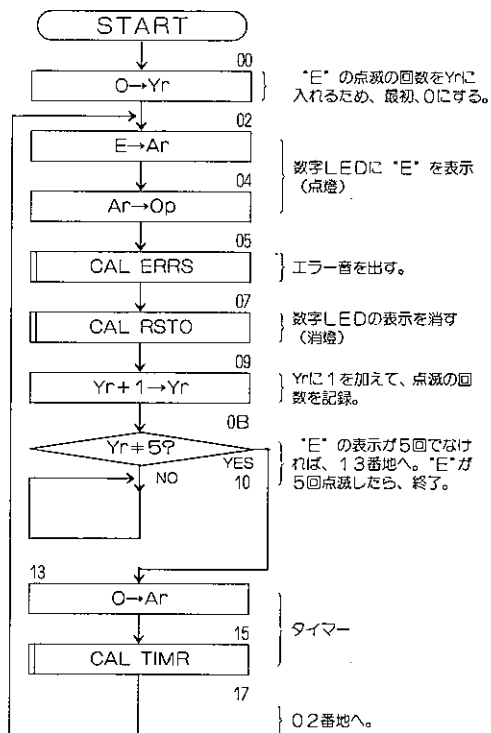
この命令は、レジスタの値を音階として出力する命令です。音階は、ゲームプログラムの電子オルガンや自動演奏と同じで、左の表のようになります。

No. 68 CAL ERRS^{めいれい}命令^{つか}を使った実験^{じっけん}

●プログラム

アドレス	命令記号	命令コード
00	T I Y	A
01	<0>	0
02	T I A	8
03	<E>	E
04	A O	1
05	[CAL	E
06	ERRS	8
07	[CAL	E
08	RSTO	0
09	A I Y	B
0A	<1>	1
0B	C I Y	D
0C	<5>	5
0D	JUMP	F
0E	<1>	1
0F	<3>	3
10	JUMP	F
11	<1>	1
12	<0>	0
13	T I A	8
14	<0>	0
15	[CAL	E
16	TIMR	C
17	JUMP	F
18	<0>	0
19	<2>	2

(フローチャート)



●エラー音とエラー表示 (E) をするプログラムです。

Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ キーを順に押し実行します。

※ CAL ERRS^{めいれい}命令^{つか}について

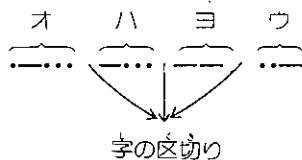
エラー音を出すプログラムです。マイコンゲームでエラー音がありました。この音はサービスプログラムにありますので、これを実行するのです。

No. 69 CAL LONS 命令を使った実験

CAL LONS 命令を使ってモールス符号を作る基本プログラム

ゲームプログラムでも、モールス符号を使ったものがありました。ここでは、モールス音の短音と長音を使って、モールス符号を作るプログラムを作ります。(モールス符号表は20Pを参照)

次のプログラムは「オハヨウ」という言葉を、モールスで発信させるプログラムです。

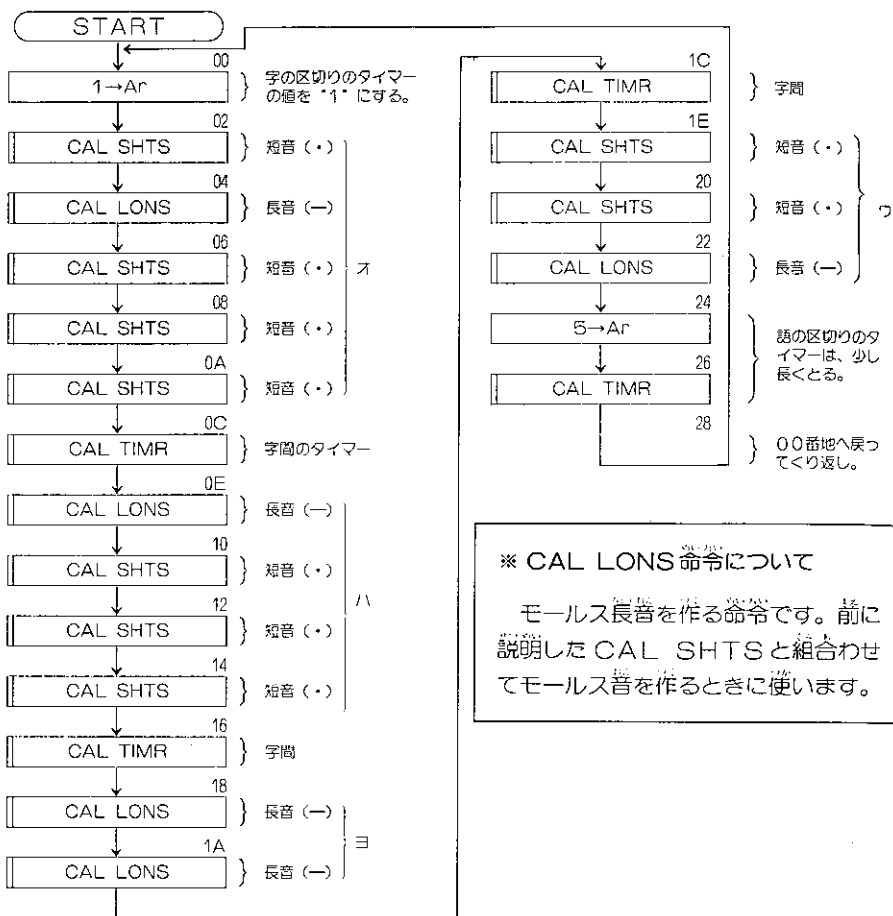


●プログラム

アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	T I A	8	16	[CAL	E
01	<1>	1	17	[TIMR	C
02	[CAL	E	18	[CAL	E
03	[SHTS	9	19	[LONS	A
04	[CAL	E	1A	[CAL	E
05	[LONS	A	1B	[LONS	A
06	[CAL	E	1C	[CAL	E
07	[SHTS	9	1D	[TIMR	C
08	[CAL	E	1E	[CAL	E
09	[SHTS	9	1F	[SHTS	9
0A	[CAL	E	20	[CAL	E
0B	[SHTS	9	21	[SHTS	9
0C	[CAL	E	22	[CAL	E
0D	[TIMR	C	23	[LONS	A
0E	[CAL	E	24	T I A	8
0F	[LONS	A	25	<5>	5
10	[CAL	E	26	[CAL	E
11	[SHTS	9	27	[TIMR	C
12	[CAL	E	28	JUMP	F
13	[SHTS	9	29	<0>	0
14	[CAL	E	2A	<0>	0
15	[SHTS	9			

- (A) プログラムの書き込みと、確認をしてください。
 (B) キーを順に押し実行します。

(フローチャート)



※ CAL LONS 命令について

モールス長音を作る命令です。前に説明した CAL SHTS と合わせてモールス音を作るときに使います。

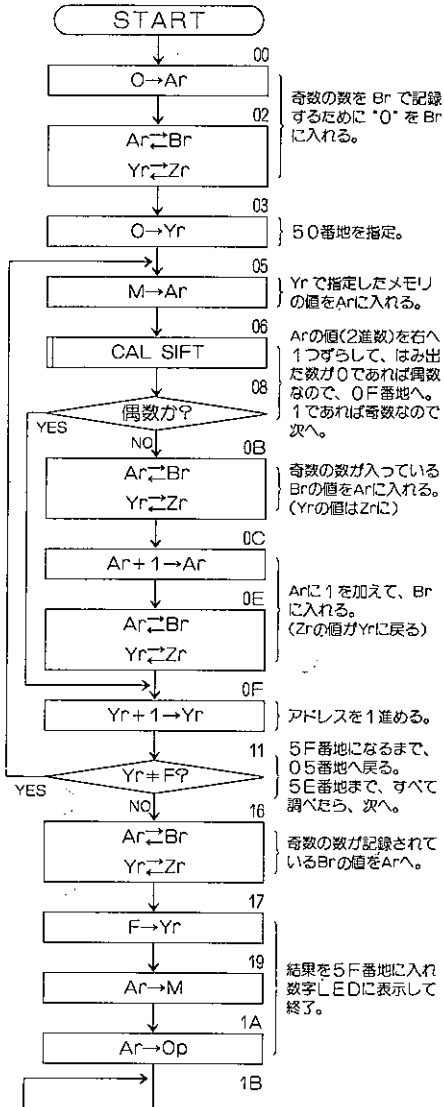
No.70 CAL SIFT命令を使った実験

めいれい つか じっけん

●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<0>	0
02	CH	2
03	TIY	A
04	<0>	0
05	MA	5
06	CAL SIFT	E
07	SIFT	6
08	JUMP	F
09	<0>	0
0A	<F>	F
0B	CH	2
0C	AIA	9
0D	<1>	1
0E	CH	2
0F	AIIY	B
10	<1>	1
11	CIY	D
12	<F>	F
13	JUMP	F
14	<0>	0
15	<5>	5
16	CH	2
17	TIY	A
18	<F>	F
19	AM	4
1A	AO	1
1B	JUMP	F
1C	<1>	1
1D		B

(フローチャート)



※ CAL SIFT 命令について

CAL SIFT 命令は、ALレジスタの2進数データを右へ1ケタシフトします。このとき実行フラグが下のように1又は0になります。

CAL SIFT 命令実行前		CAL SIFT 命令実行後	
16進数	ALレジスタ	ALレジスタ	実行フラグ
0	0000	0000	1
1	0001	0000	0
2	0010	0001	1
3	0011	0001	0
4	0100	0010	1
5	0101	0010	0
6	0110	0011	1
7	0111	0011	0
8	1000	0100	1
9	1001	0100	0
A	1010	0101	1
B	1011	0101	0
C	1100	0110	1
D	1101	0110	0
E	1110	0111	1
F	1111	0111	0

} 右へ1ケタシフト

●CAL SIFT 命令のはたらきを利用して、奇数の数をカウントするプログラムです。

(A) プログラムの書き込みと、確認をしてください。

(B) メモリ50~5Eにいろいろなデータを書き込みます。

(例)

50	51	52	53	54	55	59	5A	5B	5C	5D	5E
0	1	2	3	4	5	9	A	B	C	D	E

RESET 15 0 ADDRSET 0 INCR 71 - - - - VE INCR

(C) RESET 15 MULTI キーを順に押し実行します。 - - - - と、

●CAL SIFT 命令でALレジスタの内容を2進数で1ケタ右へシフトすると(※CAL SIFT 命令のはたらき参照)、偶数のとき実行フラグが1になり、奇数のとき実行フラグが0になります。

No.71 CAL DEM+とCAL ENDSの実験 じっけん

2ケタ+2ケタの10進加算の実験 しんかさん じっけん

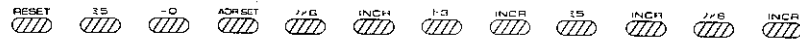
●プログラム

アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	T I Y	A	1D	C I Y	D
01	<1>	1	1E	<1>	1
02	MA	5	1F	JUMP	F
03	A I Y	B	20	<1>	1
04	<5>	5	21	<4>	4
05	AM	4	22	[CAL	E
06	A I Y	B	23]ENDS	7
07	<A>	A	24	T I Y	A
08	C I Y	D	25	<4>	4
09	<F>	F	26	MA	5
0A	JUMP	F	27	T I Y	A
0B	<0>	0	28	<F>	F
0C	<2>	2	29	AM	4
0D	T I A	8	2A	T I Y	A
0E	<0>	0	2B	<5>	5
0F	T I Y	A	2C	MA	5
10	<4>	4	2D	T I Y	A
11	AM	4	2E	<E>	E
12	T I Y	A	2F	AM	4
13	<3>	3	30	T I Y	A
14	MA	5	31	<6>	6
15	A I Y	B	32	MA	5
16	<3>	3	33	AO	1
17	CH	2	34	[CAL	E
18	CH	2	35]DSPR	D
19	[CAL	E	36	JUMP	F
1A]DEM+	F	37	<3>	3
1B	A I Y	B	38	<6>	6
1C	<D>	D			

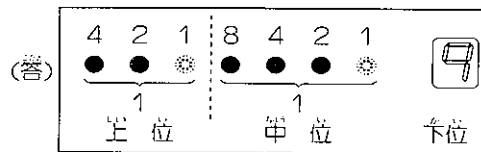
(A) プログラムの書き込みと、確認をしてください。

(B) メモリ50~53に加算する数を書き込みます。

〈例〉50<6> 51<3> 52<5> 53<6>
63+56=?



(C) キーを順に押し実行すると答えと終了音がでます。



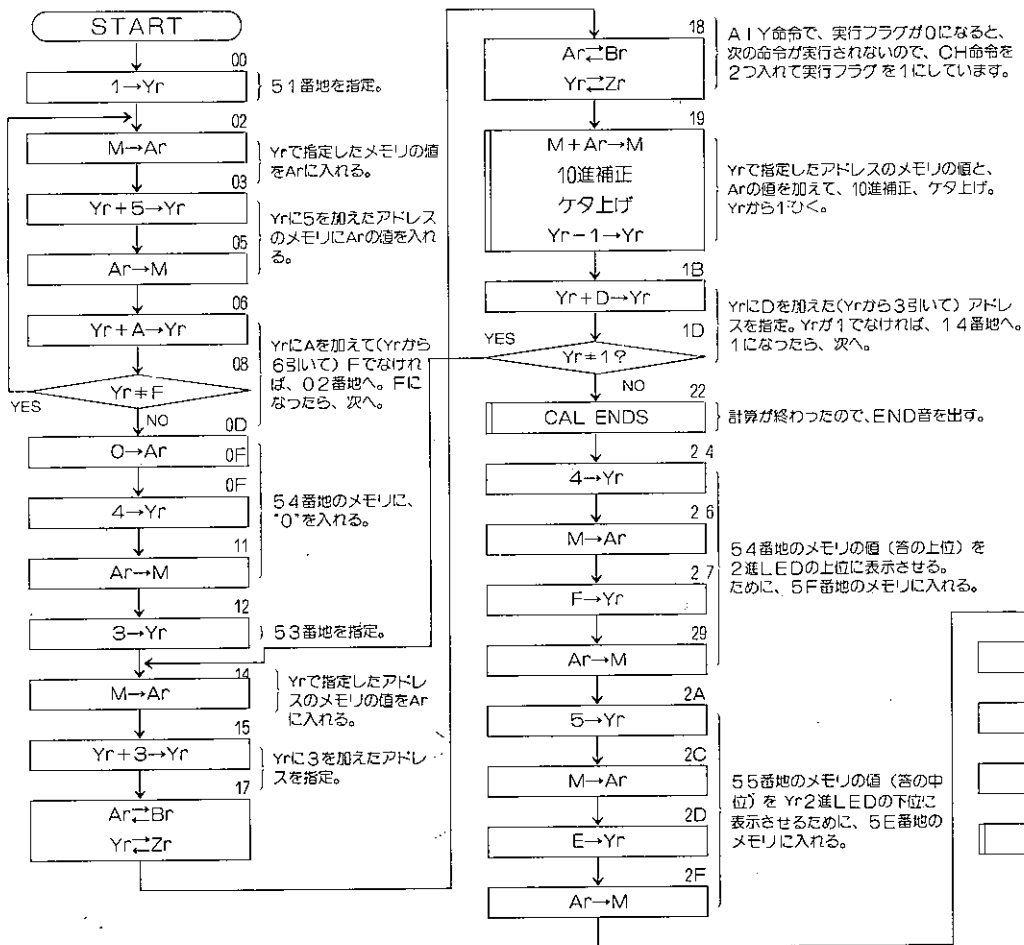
$$\begin{array}{|c|c|c|} \hline \text{メモリ} & 50 & 51 \\ \hline & 6 & 3 \\ \hline \end{array} + \begin{array}{|c|c|} \hline 52 & 53 \\ \hline 5 & 6 \\ \hline \end{array} = \begin{array}{|c|c|c|} \hline 54 & 55 & 56 \\ \hline 1 & 1 & 9 \\ \hline \end{array}$$

◆10進加算です。結果は10進数でできますから私達にもわかりやすくなりました。
10進演算は、プログラムが長くなりますから、サービスプログラムCAL DEM+を使ってプログラムを短くしています。

※ CAL DEM+命令について

10進補正加算をする命令です。Yレジスタで指定されたメモリとAレジスタの10進加算を行い、その結果はメモリに書き込まれます。ケタ上げがあれば上位の数に1を加えて行きます。この命令を実行後のYレジスタは実行前のYレジスタの値から1をひいた値となっています。

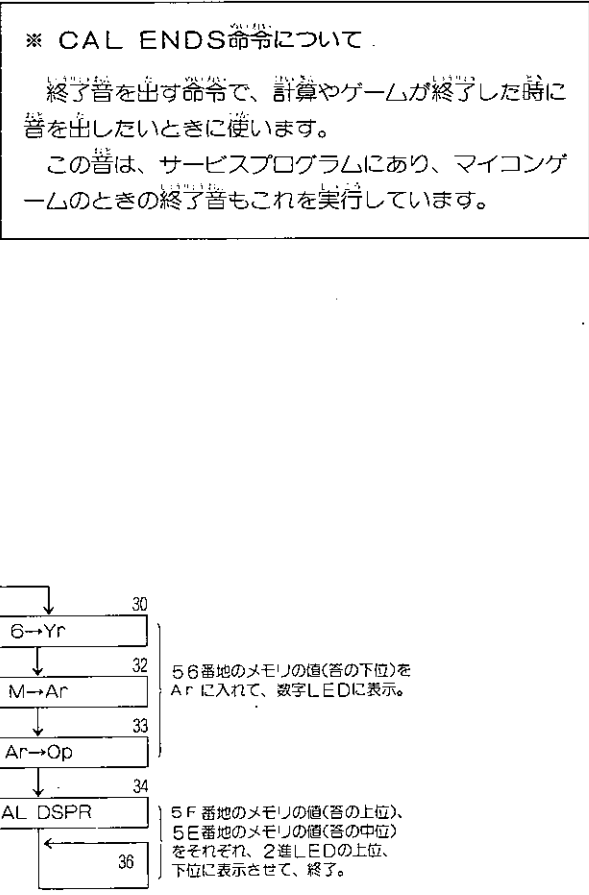
(フローチャート)



※ CAL ENDS命令について

終了音を出す命令で、計算やゲームが終了した時に音を出したいときに使います。

この音は、サービスプログラムにあり、マイコンゲームのときの終了音もこれを実行しています。



No.72 CAL DEM—命令を使った実験

2ケター2ケタの10進減算の実験

●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<0>	0
02	TIY	A
03	<A>	A
04	AM	4
05	AIY	B
06	<1>	1
07	CIY	D
08	<0>	0
09	JUMP	F
0A	<0>	0
0B	<4>	4
0C	TIY	A
0D	<6>	6
0E	MA	5
0F	TIY	A
10		B
11	CH	2
12	TIY	A
13	<7>	7
14	MA	5
15	TIY	A
16	<C>	C
17	AM	4
18	AIY	B
19	<D>	D
1A	MA	5
1B	AIY	B
1C	<3>	3

アドレス	命令記号	命令コード
1D	CH	2
1E	CH	2
1F	CH	2
20	CAL	E
21	DEM-	E
22	MA	5
23	CIY	D
24		B
25	JUMP	F
26	<3>	3
27	<2>	2
28	CIY	D
29	<0>	0
2A	JUMP	F
2B	<3>	3
2C	<F>	F
2D	CH	2
2E	AM	4
2F	CH	2
30	JUMP	F
31	<1>	1
32	<8>	8
33	CIY	D
34	<1>	1
35	JUMP	F
36	<4>	4
37	<6>	6
38	TIA	8
39	<E>	E
40	AO	1

アドレス	命令記号	命令コード
3A	CAL	E
3B	ERRS	8
3C	JUMP	F
3D	<3>	3
3E	<C>	C
3F	CH	2
40	AIA	9
41	<F>	F
42	AM	4

アドレス	命令記号	命令コード
43	JUMP	F
44	<2>	2
45	<E>	E
46	CAL	E
47	ENDS	7
48	JUMP	F
49	<4>	4
4A	<8>	8

●この実験は、引けた時はEND音が出て、5A~5C番地のメモリに答えが書き込まれますが、引けない時は“E”表示とエラー音が出ます。

(A) プログラムの書き込みと、確認をしてください。

(B) メモリ56、57とメモリ58、59にデータを書き込みます。

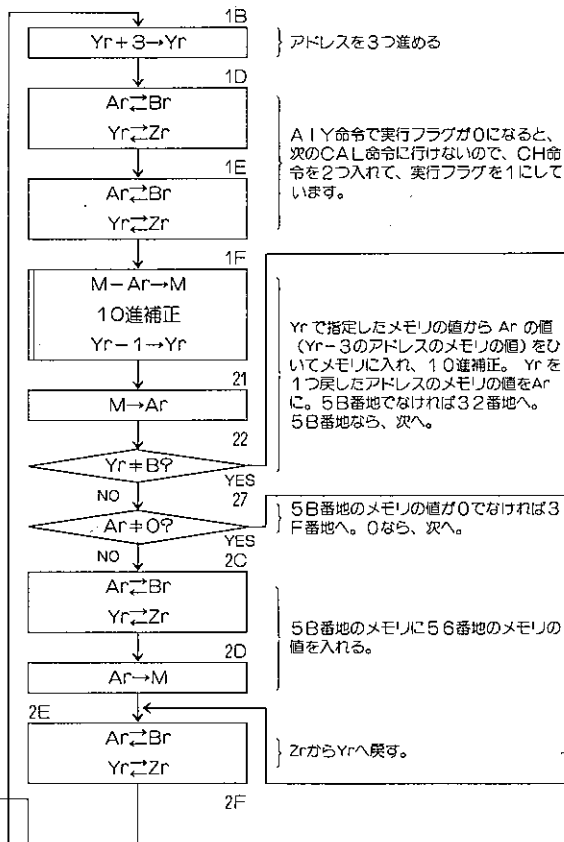
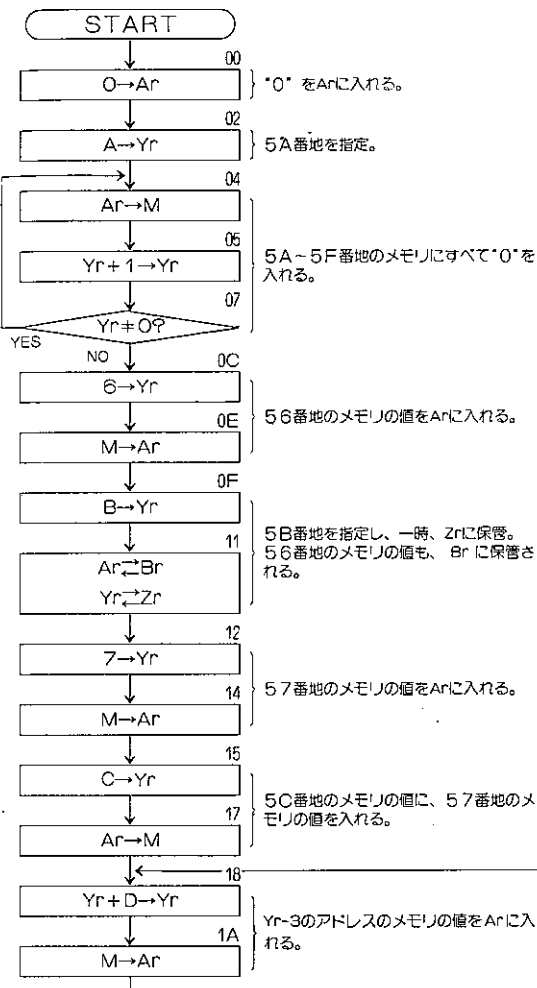
56	57	58	59	5A	5B	5C	メモリ番地
7	6	4	8	0	2	8	10進減算

RESET 15 16 ACSET 17 18 19 20 21 22 23 24

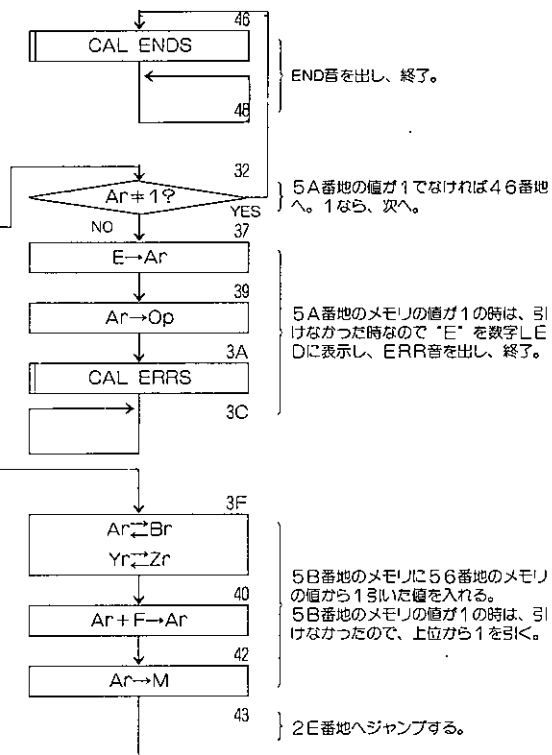
(C) RESET 1 2 3 キーを順に押し実行します。END音が出れば、終了。

(D) RESET 15 16 ACSET 0 1 2 3

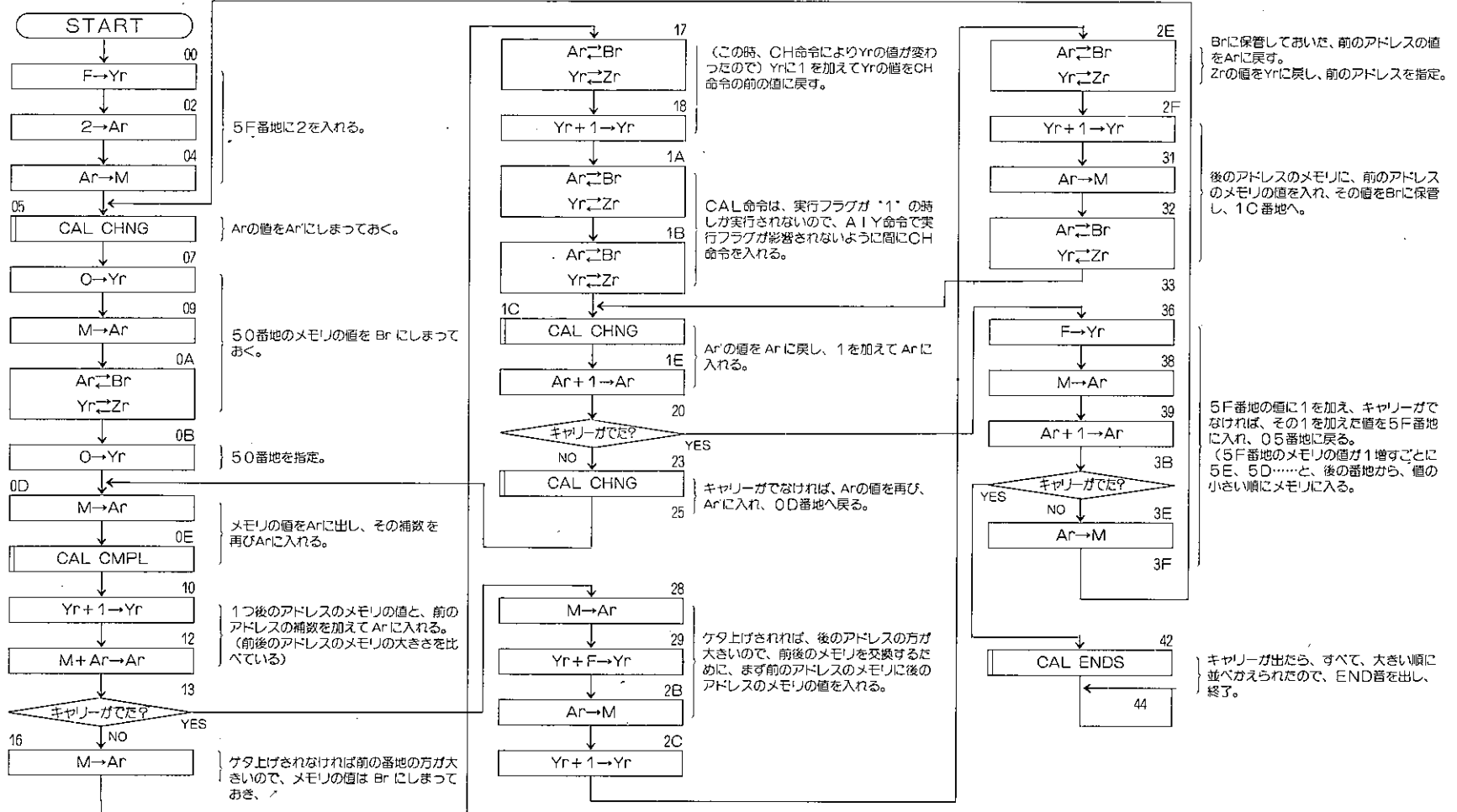
(フローチャート)



※CAL DEM命令について
Yレジスタで指定されるメモリからAレジスタの値を引きます。答えは10進補正されてメモリに書き込まれます。



(フローチャート)



No.74 ないメモリ内のデータをちい小さい順にじゅん並べならるじっけん実験

●プログラム

アドレス	命令記号	命令コード
00	TIY	A
01	<F>	F
02	TIA	8
03	<2>	2
04	AM	4
05	[CAL	E
06	[CHNG	5
07	TIY	A
08	<0>	0
09	MA	5
0A	CH	2
0B	TIY	A
0C	<0>	0
0D	MA	5
0E	[CAL	E
0F	[CMPL	4
10	AIY	B
11	<1>	1
12	M+	6
13	JUMP	F
14	<2>	2
15	<D>	D
16	MA	5
17	AIY	B
18	<F>	F
19	AM	4
1A	AIY	B
1B	<1>	1
1C	CH	2

アドレス	命令記号	命令コード
1D	AIY	B
1E	<1>	1
1F	AM	4
20	CH	2
21	[CAL	E
22	[CHNG	5
23	AIA	9
24	<1>	1
25	JUMP	F
26	<3>	3
27	<5>	5
28	[CAL	E
29	[CHNG	5
2A	JUMP	F
2B	<0>	0
2C	<D>	D
2D	MA	5
2E	CH	2
2F	AIY	B
30	<1>	1
31	CH	2
32	JUMP	F
33	<2>	2
34	<0>	0
35	TIY	A
36	<F>	F
37	MA	5
38	AIA	9
39	<1>	1

アドレス	命令記号	命令コード
3A	JUMP	F
3B	<4>	4
3C	<1>	1
3D	AM	4
3E	JUMP	F
3F	<0>	0

アドレス	命令記号	命令コード
40	<5>	5
41	[CAL	E
42	[ENDS	7
43	JUMP	F
44	<4>	4
45	<3>	3

(A) プログラムの書き込みと、確認をしてください。

(B) メモリ50~5E番地に、いろいろなデータを書き込みます。

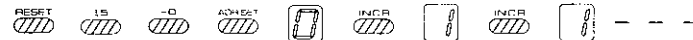
(例)

50	51	52	53	54	5B	5C	5D	5E
8	1	C	0	A	5	1	B	6



(C) キーを順に押し実行すると、END音が出てならべかわります。

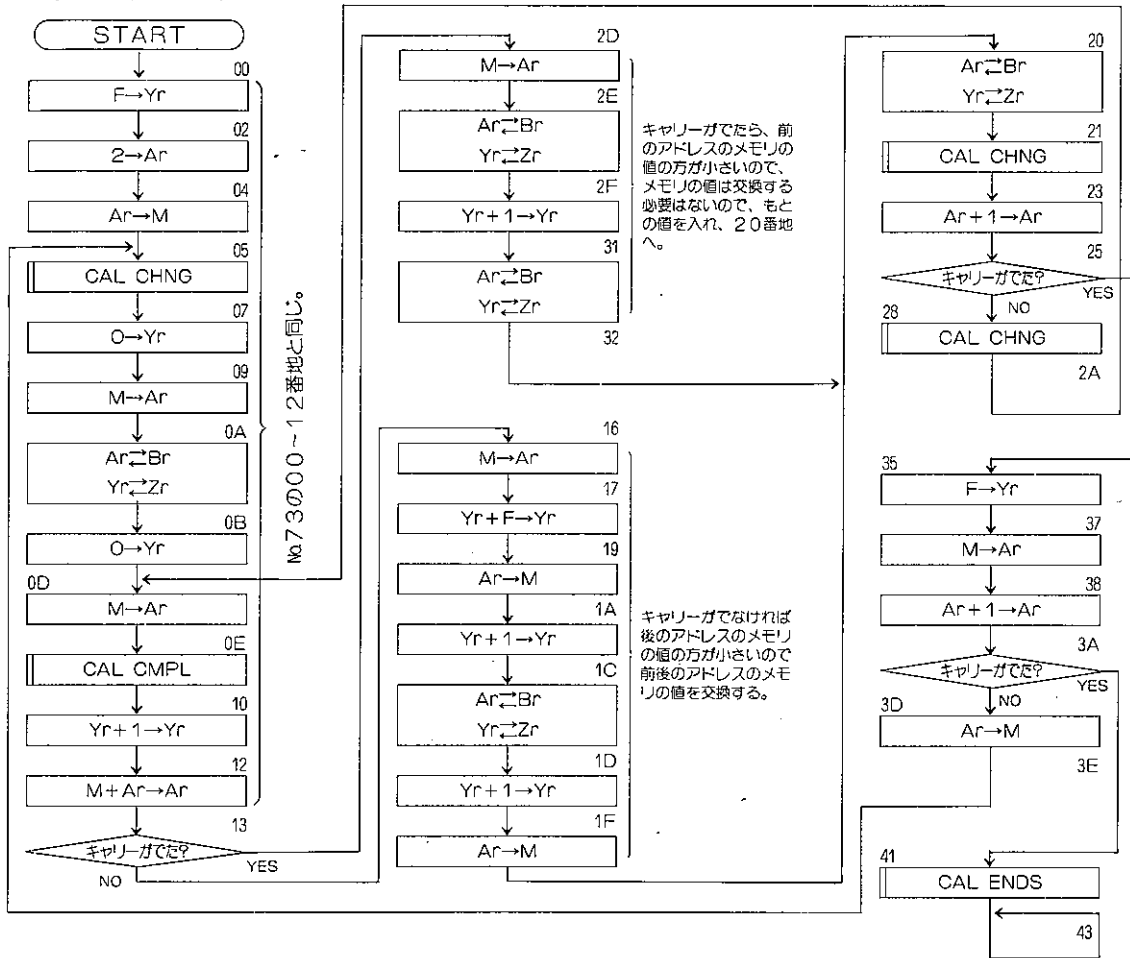
(D) END音を確認したあと、メモリ50~5Eを読んで、小さい順に並べ変えられているか確認してみましょう。



キーを順に押し、数字LEDで確認します。

◆並べ変えのプログラムです。 と をうまく組合せて作っております。

(フローチャート)



№73の00~12番地と同じ。

№73の1C~25番地と同じ。

№73の36~44番地と同じ。

ここでちょっと、№73のプログラム及びフローチャートと、この実験のプログラム、フローチャートと比べて下さい。

№73と、この実験との違いは、№73は、大きい順に並べるため、12番地で、キャリーが出た時は、後の番地の値が大きいのので、前後のメモリーを交換し、キャリーが出なければ、前の番地の値が大きいのので、そのまま。№74は、№73と逆で、小さい順に並べるため、12番地でキャリーが出た時は、そのまま、キャリーが出なければ前後のメモリーを交換する。

これを、プログラム(フローチャートも)で見ると、№73の16~1A番地と№74の2D~31番地及び、№73の28~32番地と№74の16~20番地がそれぞれ同じことがわかるでしょう。ここを入れ換えただけで、あとはアドレスがずれているだけです。

No. 75 ^{けた}2桁×^{けた}1桁の^{しんじょうざん}10進乗算

●プログラム


アドレス	命令記号	命令コード
00	TIA	8
01	<0>	0
02	TIY	A
03	<3>	3
04	AM	4
05	TIY	A
06	<4>	4
07	AM	4
08	TIY	A
09	<5>	5
0A	AM	4
0B	TIY	A
0C	<2>	2
0D	MA	5
0E	CH	2
0F	TIY	A
10	<1>	1
11	MA	5
12	AIY	B
13	<4>	4
14	CH	2
15	CH	2
16	[CAL	E
17	[DEM+	F
18	AIY	B
19	<C>	C
1A	CIY	D
1B	<F>	F
1C	JUMP	F

アドレス	命令記号	命令コード
1D	<1>	1
1E	<1>	1
1F	CH	2
20	AIA	9
21	<F>	F
22	CIA	C
23	<0>	0
24	JUMP	F
25	<0>	0
26	<E>	E
27	TIY	A
28	<3>	3
29	MA	5
2A	AIA	9
2B	<8>	8
2C	JUMP	F
2D	<4>	4
2E	<2>	2
2F	MA	5
30	TIY	A
31	<F>	F
32	AM	4
33	TIY	A
34	<4>	4
35	MA	5
36	TIY	A
37	<E>	E
38	AM	4
39	[CAL	E

アドレス	命令記号	命令コード
3A	[DSPR	D
3B	TIY	A
3C	<5>	5
3D	MA	5
3E	AO	1
3F	JUMP	F
40	<3>	3
41	<F>	F

アドレス	命令記号	命令コード
42	TIA	8
43	<C>	C
44	AO	1
45	[CAL	E
46	[ERRS	8
47	JUMP	F
48	<4>	4
49	<7>	7

答えは、53~55番地に、はいり、799までであれば、答えは2進数LEDと数字LEDで3ケタ表示されます。

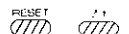
答えが800以上になった時は、メモリーには答えがはいりますが、答えの表示はされず、 (キャリー)の表示がされ、エラー音がでます。

(A) プログラムの書き込みと、確認をしてください。

(B) メモリ50、51、52にデータを書き込みます。


50	51	52	53	54	55	メモリー番地
7	2	8	5	7	6	10進乗算



(C)  キーを順に押し実行します。

(D) 答えは

4	2	1	8	4	2	1
○	●	○	●	○	○	○
	上位		中位			下位
	5		7			6



(フローチャート)

START

00 0→Ar

02 3→Yr

04 Ar→M

05 4→Yr

07 Ar→M

08 5→Ar

0A Ar→M

0B 2→Yr

0D M→Ar

0E Ar↔Br
Yr↔Zr

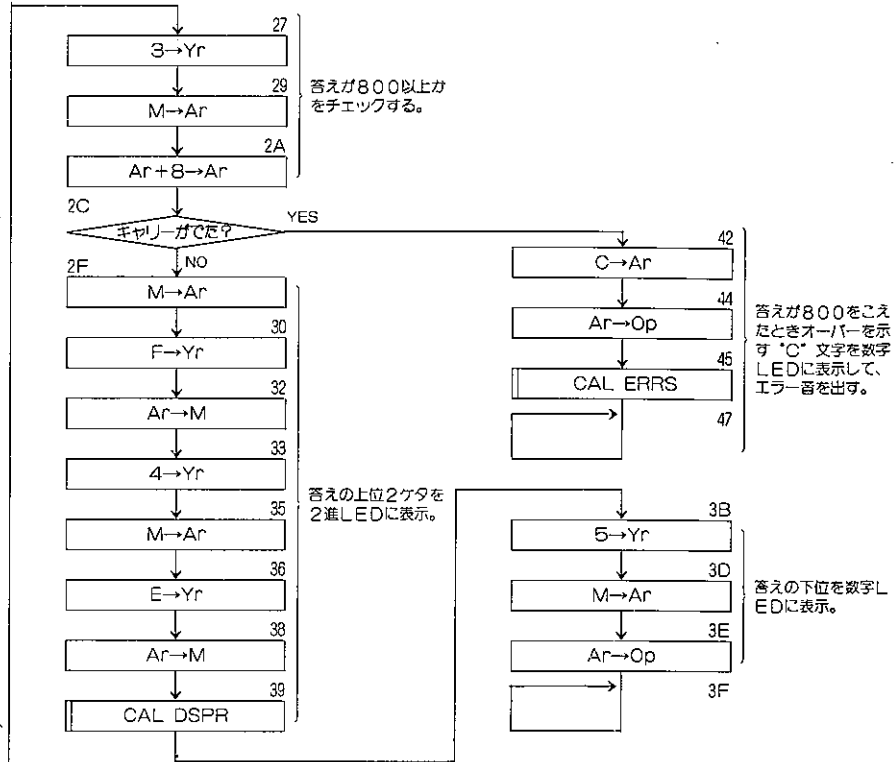
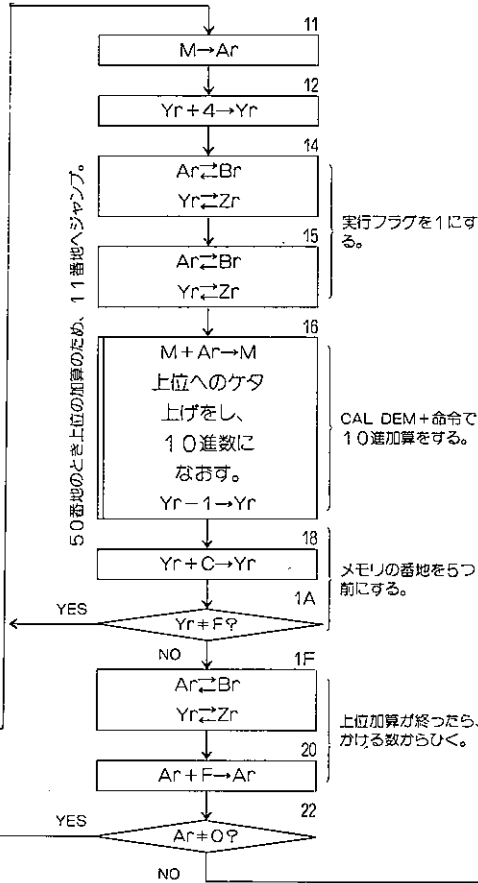
0F 1→Yr

53, 54, 55番地に0を入れる。

かける数をArに入れる。

かける数をBrへ。

かける数が0になるまで、加算をくりかえす。



No.76 2桁÷1桁の10進除算

●プログラム

アドレス	命令記号	命令コード
00	T I Y	A
01	<C>	C
02	T I A	8
03	<1>	1
04	[CAL	E
05	[CHNG	5
06	T I A	8
07	<0>	0
08	T I Y	A
09		B
0A	A I Y	B
0C	<1>	1
0D	C I Y	D
0E	<0>	0
0F	JUMP	F
10	<0>	0
11	<A>	A
12	T I Y	A
13	<8>	8
14	MA	5
15	CH	2
16	T I Y	A
17	<9>	9
18	MA	5
19	T I Y	A
1A	<E>	E
1B	AM	4
1C	T I Y	A

アドレス	命令記号	命令コード
1D	<A>	A
1E	MA	5
1F	T I Y	A
20	<E>	E
21	[CAL	E
22	[DEM-	E
23	MA	5
24	C I A	C
25	<0>	0
26	JUMP	F
27	<3>	3
28	<6>	6
29	[CAL	E
2A	[CHNG	5
2B	[CAL	E
2C	[DEM+	F
2D	A I Y	B
2E	<1>	1
2F	CH	2
30	CH	2
31	[CAL	E
32	[CHNG	5
33	JUMP	F
34	<1>	1
35	<C>	C
36	T I A	8
37	<0>	0
38	AM	4
39	CH	2

アドレス	命令記号	命令コード
3A	C I A	C
3B	<0>	0
3C	JUMP	F
3D	<4>	4
3E	<E>	E
3F	T I Y	A
40	<A>	A
41	MA	5
42	T I Y	A
43	<E>	E
44	[CAL	E
45	[DEM+	F
46	T I A	8

アドレス	命令記号	命令コード
47	<0>	0
48	AM	4
49	[CAL	E
4A	[ENDS	7
4B	JUMP	F
4C	<4>	4
4D		B
4E	A I A	9
4F	<F>	F
50	CH	2
51	JUMP	F
52	<2>	2
53	<9>	9

Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ メモリ58、59、5Aにデータを書き込みます。
(例)

58	59	5A	5B	5C	5D	5E	メモリ番地
7	5	8	0	9	0	3	10進除算

RESET INCR INCR INCR INCR INCR INCR INCR INCR INCR

Ⓒ RESET INCR INCR キーを順に押し実行します。END 後の、

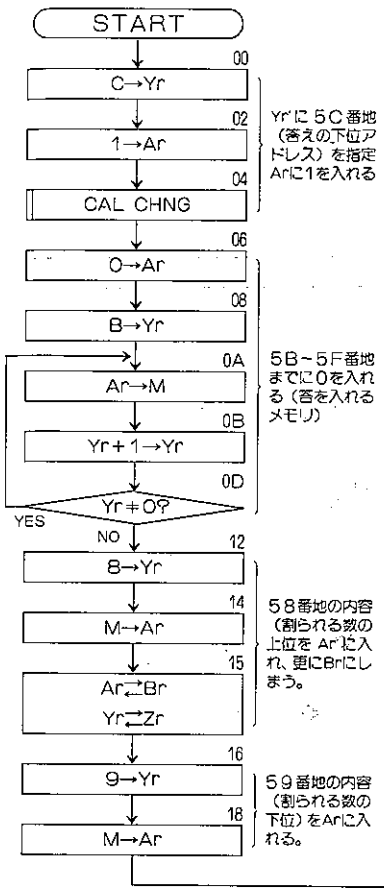
Ⓓ RESET INCR INCR INCR INCR INCR INCR INCR INCR INCR

答

余り

※ 0で割っては、答えは求められないのでさけてください。

(フローチャート)

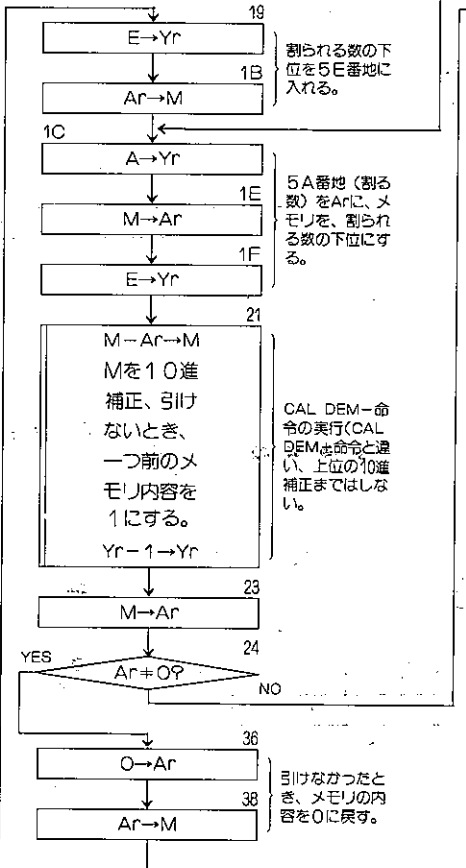


Yrに5C番地
(答えの下位ア
ドレス)を指定
Arに1を入れる

5B~5F番地
までに0を入れる
(答を入れる
メモリ)

5B番地の内容
(割られる数の
上位を Ar に入
れ、更に Br にし
まう。

59番地の内容
(割られる数の
下位)を Ar に入
れる。

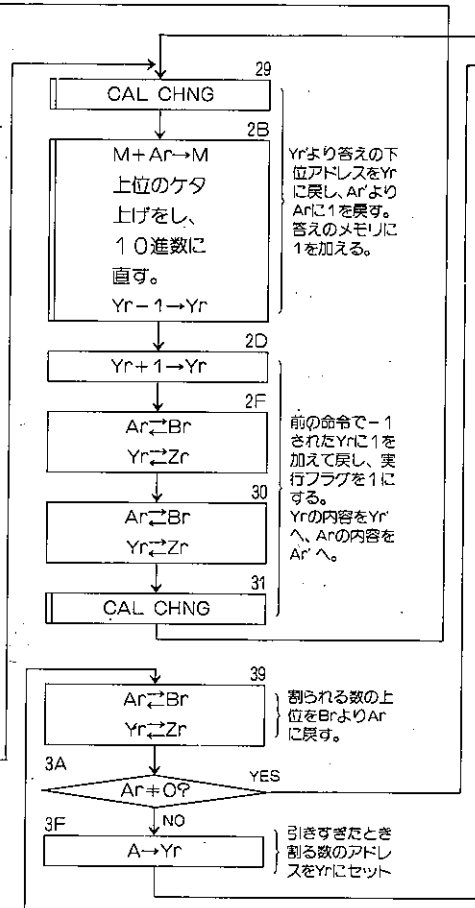


割られる数の下
位を5E番地に
入れる。

5A番地(割る
数)を Ar に、メ
モリを、割られ
る数の下位にす
る。

CAL DEM-命
令の実行(CAL
DEMは命令と違
い、上位の10進
補正まではしな
い。

引けなかったと
き、メモリの内
容を0に戻す。

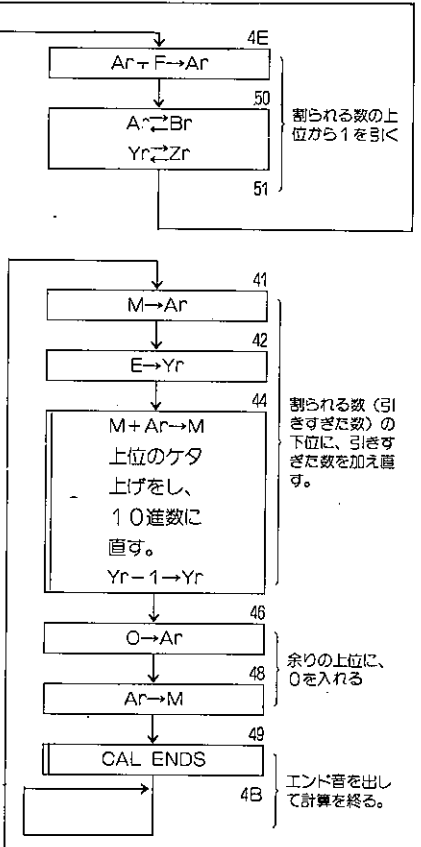


Yrより答えの下
位アドレスを Yr
に戻し、Arより
Arに1を戻す。
答えのメモリに
1を加える。

前の命令で-1
された Yr に 1 を
加えて戻し、実
行フラグを 1 に
する。
Yr の内容を Yr
へ、Ar の内容を
Ar へ。

割られる数の上
位を Br より Ar
に戻す。

引きすぎたとき
割る数のアドレ
スを Yr にセット



割られる数の上
位から 1 を引く

割られる数(引
きすぎた数)の
下位に、引きす
ぎた数を加え直
す。

余りの上位に、
0を入れる

エンド音を出し
て計算を終る。

No. 77 メモリ内の合計を求める

●プログラム

アドレス	命令記号	命令コード
00	TIA	8
01	<0>	0
02	TIY	A
03	<C>	C
04	AM	4
05	AIY	B
06	<1>	1
07	CIY	D
08	<0>	0
09	JUMP	F
0A	<0>	0
0B	<4>	4
0C	TIY	A
0D	<C>	C
0E	MA	5
0F	CY	3
10	MA	5
11	TIY	A
12	<F>	F
13	[CAL	E
14	[DEM+	F
15	TIY	A
16	<C>	C
17	MA	5
18	AIA	9
19	<1>	1
1A	AM	4
1B	CIA	C
1C	<C>	C

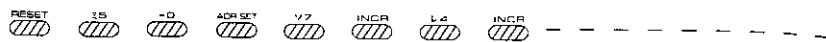
アドレス	命令記号	命令コード
1D	JUMP	F
1E	<0>	0
1F	<F>	F
20	TIY	A
21	<F>	F
22	MA	5
23	AO	1
24	CH	2
25	TIY	A
26	<D>	D
27	MA	5
28	TIY	A
29	<F>	F
2A	AM	4
2B	[CAL	E
2C	[DSPR	D
2D	CH	2
2E	AM	4
2F	JUMP	F
30	<2>	2
31	<F>	F

●たくさんのデータを10進加算してみましょう。

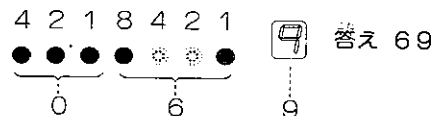
- (A) プログラムの書き込みと、確認をしてください。
 (B) メモリ50~5Bに10進数(0~9)のデータを書き込みます。

(例)

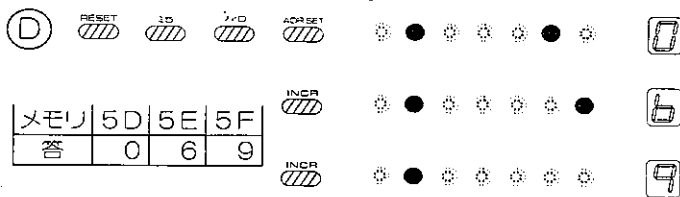
50	51	52	53	54	55	56	57	58	59	5A	5B	メモリ
7	4	3	9	5	8	3	4	8	9	7	2	合計



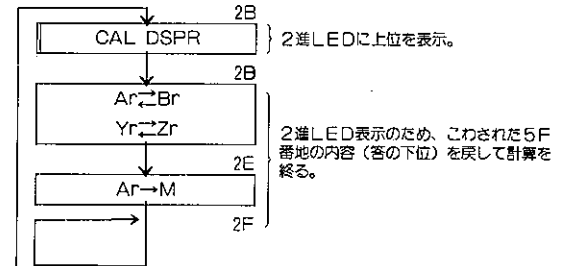
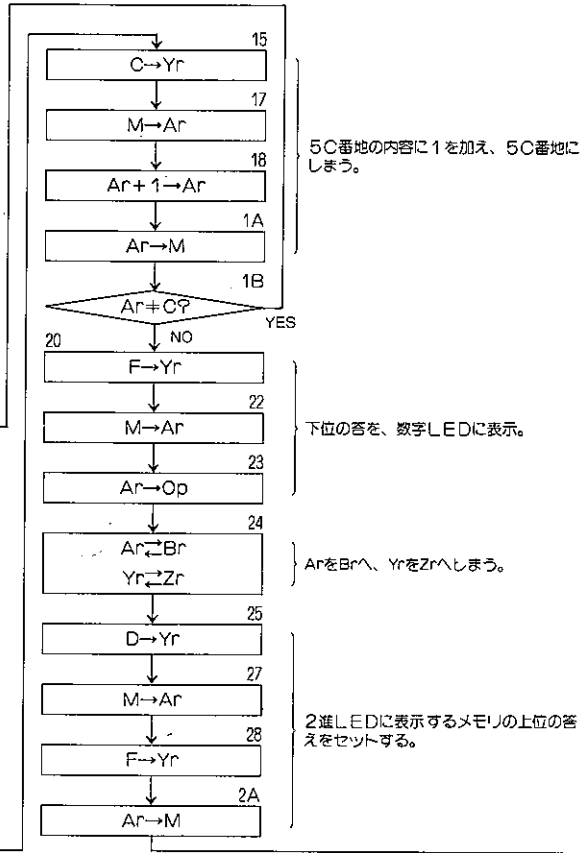
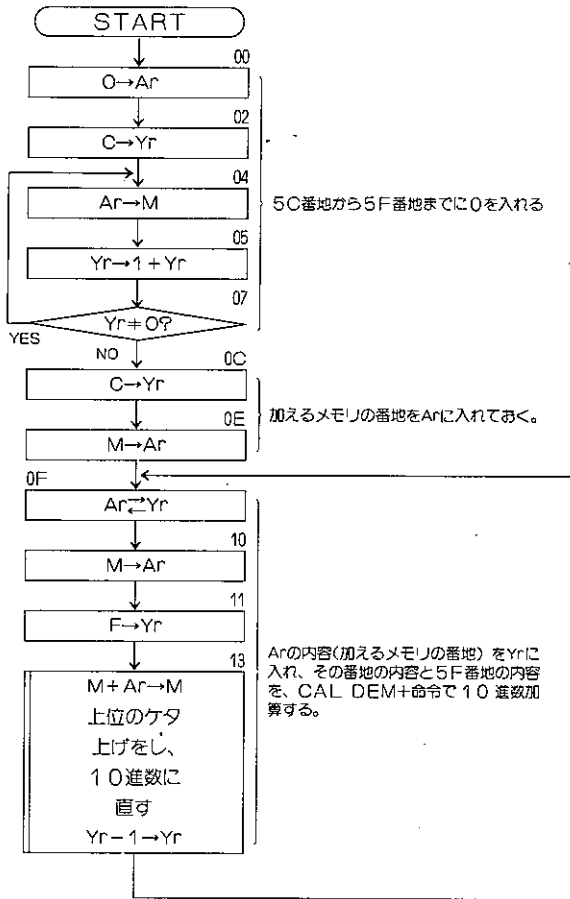
- (C) RESET, 15, 10 キーを順に押し実行しますと答えを表示します。



メモリを読みだしてみよう。



(フローチャート)



No.78 メモリ内の平均値を求める

ない へいきん ち もと

●プログラム

アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	TIA	8	1D	<C>	C
01	<0>	0	1E	MA	5
02	TIY	A	1F	AIA	9
03	<C>	C	20		B
04	AM	4	21	JUMP	F
05	TIY	A	22	<2>	2
06		B	23	<D>	D
07	AM	4	24	TIY	A
08	TIY	A	25		B
09	<A>	A	26	MA	5
0A	AM	4	27	AO	1
0B	CY	3	28	[CAL	E
0C	MA	5	29	[ENDS	7
0D	TIY	A	2A	JUMP	F
0E	<C>	C	2B	<2>	2
0F	[CAL	E	2C	<A>	A
10	[DEM+	F	2D	TIY	A
11	TIY	A	2E		B
12	<A>	A	2F	MA	5
13	MA	5	30	AIA	9
14	AIA	9	31	<1>	1
15	<1>	1	32	CH	2
16	AM	4	33	CH	2
17	CIA	C	34	JUMP	F
18	<A>	A	35	<2>	2
19	JUMP	F	36	<7>	7
1A	<0>	0			
1B		B			
1C	TIY	A			

(A) プログラムの書き込みと、確認をしてください。

(B) メモリ50~59番地に10進数のデータを書き込みます。

(例)

50	51	52	53	54	55	56	57	58	59
3	4	7	1	9	3	5	6	8	2

RESET 15 0 ADRESET 13 INCR 14 INCR -----

(C) RESET 15 RUN キーを順に押し実行します。

(D) 答えは、数字LEDに表示されEND音が鳴ります。
※平均値を四捨五入して表示されています。



◆このプログラムは、データの合計をデータの数で割って、小数点以下の数を四捨五入して、平均値を求めるプログラムです。しかし、データの合計、割算、四捨五入のすべてのプログラムを書くと、プログラムメモリが足りないので、50~59番地までの10個のメモリを10進数加算することにより、その平均値(5.6)は、合計の10のケタが1のケタに、1のケタが0.1のケタになることを利用し、割算を省略し、データの合計と、四捨五入のプログラムで平均値を出してあります。

No. 79 データ入力によるモールス自動発信

にゅうりよく

じ どうはっしん

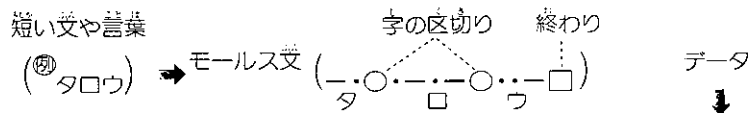
●モールス符号を50～5F番地までに書き込んで、モールス音を作るプログラムです。

●プログラム

アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	T I Y	A	21	CH	2
01	<0>	0	22	T I A	8
02	MA	5	23	<6>	6
03	C I A	C	24	[CAL	E
04	<F>	F	25	[TIMR	C
05	JUMP	F	26	CH	2
06	<0>	0	27	A I Y	B
07	<F>	F	28	<1>	1
08	T I A	8	29	MA	5
09	<9>	9	2A	JUMP	F
0A	[CAL	E	2B	<0>	0
0B	[TIMR	C	2C	<3>	3
0C	JUMP	F	2D	[CAL	E
0D	<0>	0	2E	[SIFT	6
0E	<0>	0	2F	JUMP	F
0F	C I A	C	30	<3>	3
10	<E>	E	31	<D>	D
11	JUMP	F	32	CH	2
12	<1>	1	33	T I A	8
13	<7>	7	34	<2>	2
14	JUMP	F	35	JUMP	F
15	<1>	1	36	<2>	2
16	<4>	4	37	<4>	4
17	[CAL	E	38	[CAL	E
18	[SIFT	6	39	[LONS	A
19	JUMP	F	3A	JUMP	F
1A	<2>	2	3B	<2>	2
1B	<D>	D	3C	<7>	7
1C	[CAL	E	3D	[CAL	E
1D	[SIFT	6	3E	[SHTS	9
1E	JUMP	F	3F	JUMP	F
1F	<3>	3	40	<2>	2
20	<8>	8	41	<7>	7

データ	モールス
0	→ 短音
1	→ 長音
2	→ 字の区切り
3	→ 語の区切り
E	→ 終わり
F	→ くり返し

このデータを、50～5F番地までに書き込む。



(例)

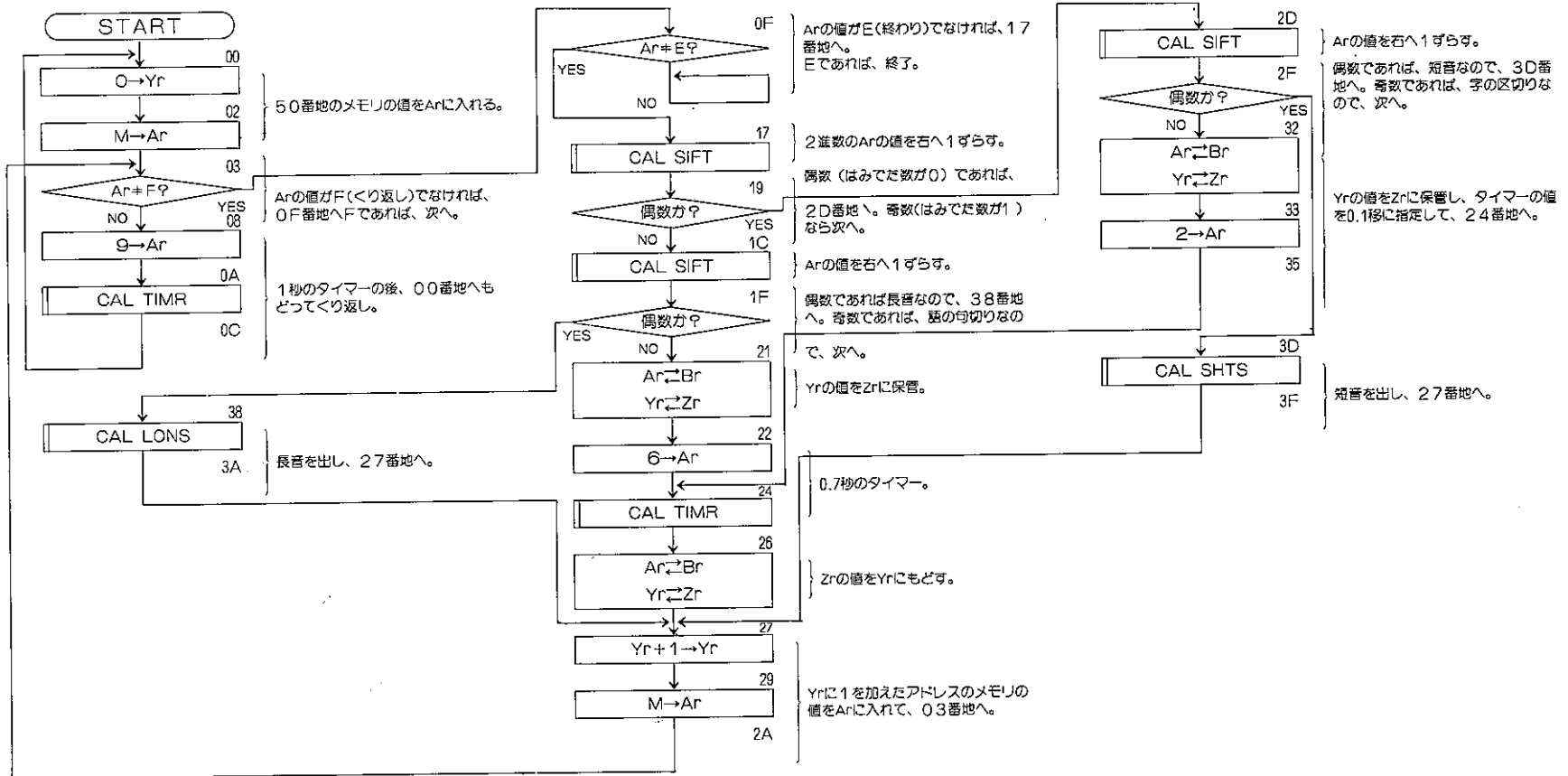
アドレス	50	51	52	53	54	55	56	57	58	59	5A	5B
データ	1	0	2	0	1	0	1	2	0	0	1	E

●50番地から5B番地に書き込んだ、タロウのモールス音が、一回スピーカーから発信されます。

(A) プログラムと、モールス、データの書き込みと、確認をしてください。

(B) キーを順に押し実行します。

(フローチャート)



No. 80 ^{ふん} ^{びょう} 7分59秒までのデクリメントタイマー

●7分59秒までのタイマーです。50番地に分を、51番地と52番地に秒を書き込んで、プログラムをスタートさせます。2進LEDと数字LEDの表赤が1秒ごとデクリメントされ（1つづつ小さくなり）0分00秒になったとき、エンド音を出します。

●プログラム

①			②		
アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	T I Y	A	19	[CAL	E
01	<0>	0	1A	[TIMR	C
02	MA	5	1B	CH	2
03	T I Y	A	1C	C I A	C
04	<F>	F	1D	<0>	0
05	AM	4	1E	JUMP	F
06	CH	2	1F	<3>	3
07	T I Y	A	20	<6>	6
08	<1>	1	21	[CAL	E
09	MA	5	22	[CHNG	5
0A	T I Y	A	23	C I A	C
0B	<E>	E	24	<0>	0
0C	AM	4	25	JUMP	F
0D	[CAL	E	26	<3>	3
0E	[CHNG	5	27	<C>	C
0F	T I A	8	28	CH	2
10	<9>	9	29	C I A	C
11	CH	2	2A	<0>	0
12	T I Y	A	2B	JUMP	F
13	<2>	2	2C	<4>	4
14	MA	5	2D	<6>	6
15	AO	1	2E	AO	1
16	[CAL	E	2F	[CAL	E
17	[DSPR	D	30	[DSPR	D
18	CH	2			

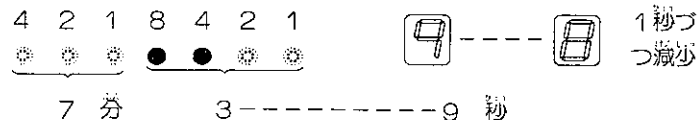
③			④		
アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
31	[CAL	E	41	T I A	8
32	[ENDS	7	42	<9>	9
33	JUMP	F	43	JUMP	F
34	<3>	3	44	<1>	1
35	<3>	3	45	<5>	5
36	A I A	9	46	A I A	9
37	<F>	F	47	<F>	F
38	AO	1	48	AM	4
39	JUMP	F	49	CH	2
3A	<1>	1	4A	T I A	8
3B	<6>	6	4B	<5>	5
3C	A I A	9	4C	JUMP	F
3D	<F>	F	4D	<3>	3
3E	AM	4	4E	<E>	E
3F	[CAL	E			
40	[CHNG	5			

Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ メモリ50、51、52にデータを書き込みます。

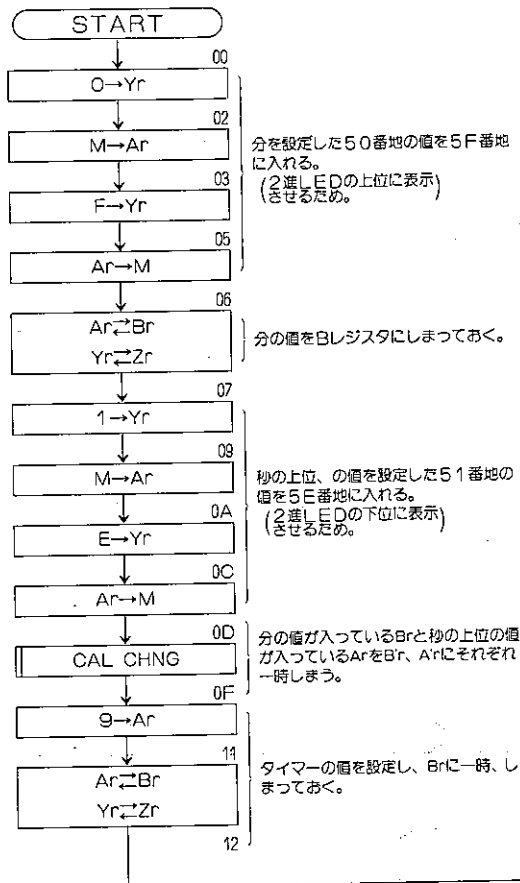
(例)

Ⓒ キーを順に押し実行します。



Ⓒで設定した分、秒が1秒ごと減少し、0分0秒になったとき、END音を出して止まります。

(フローチャート)



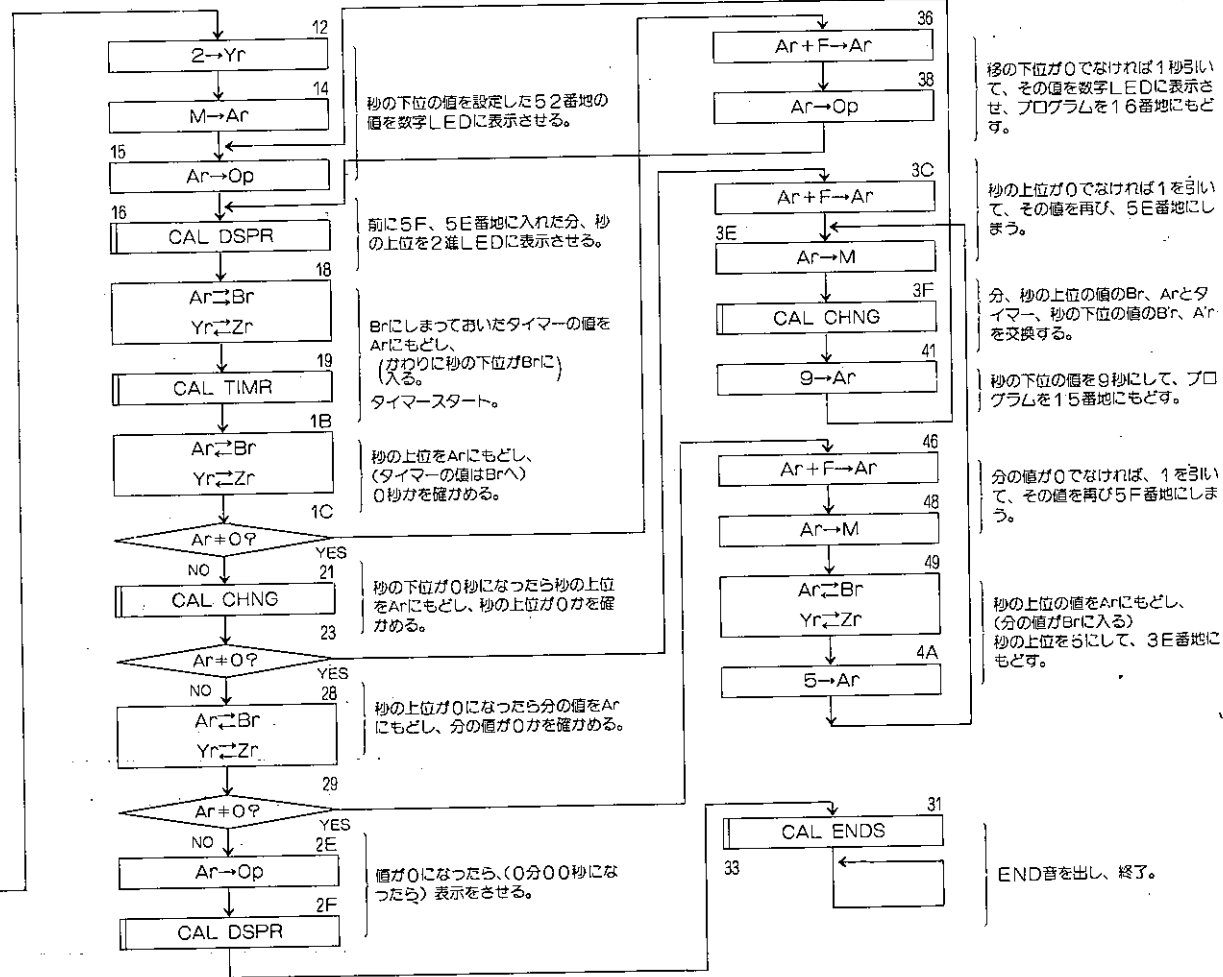
分を設定した50番地の値を5F番地に入れる。
(2進LEDの上位に表示)させるため。

分の値をBレジスタにしまっておく。

秒の上位、の値を設定した51番地の値を5E番地に入れる。
(2進LEDの下位に表示)させるため。

分の値が入っているBrと秒の上位の値が入っているArをBr、Arにそれぞれ一時しめよう。

タイマーの値を設定し、Brに一時、しめておく。



秒の下位の値を設定した52番地の値を数字LEDに表示させる。

前に5F、5E番地に入れた分、秒の上位を2進LEDに表示させる。

Brにしまっておいたタイマーの値をArにもとし、(かわりに秒の下位がBrに入る。)
タイマースタート。

秒の上位をArにもとし、(タイマーの値はBrへ) 0秒かを確かめる。

秒の下位が0秒になったら秒の上位をArにもとし、秒の上位が0かを確かめる。

秒の上位が0になったら分の値をArにもとし、分の値が0かを確かめる。

値が0になったら、(0分00秒になったら)表示をさせる。

移の下位が0でなければ1秒引いて、その値を数字LEDに表示させ、プログラムを16番地にもどす。

秒の上位が0でなければ1を引いて、その値を再び、5E番地にしよう。

分、秒の上位の値のBr、Arとタイマー、秒の下位の値のBr、Arを交換する。

秒の下位の値を9秒にして、プログラムを15番地にもどす。

分の値が0でなければ、1を引いて、その値を再び5F番地にしよう。

秒の上位の値をArにもとし、(分の値がBrに入る) 秒の上位を5にして、3E番地にもどす。

END音を出し、終了。

No. 82 モールス、エレキー

マイコンでモールスエレキーを作ってみました。

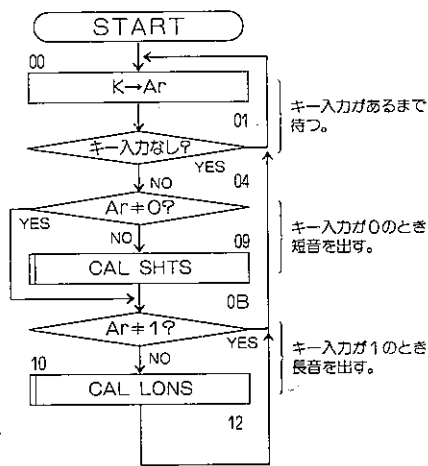
エレキーは、モールスの短点、長点をキーを押すだけで自動的に発生させるもので、だれでも正確なモールス発信ができるようになります。

短音キー  長音キー 



●プログラム



アドレス	命令記号	命令コード
00	KA	0
01	JUMP	F
02	<0>	0
03	<0>	0
04	CIA	C
05	<0>	0
06	JUMP	F
07	<0>	0
08		B
09	[CAL	E
0A	[SHTS	9
0B	CIA	C
0C	<1>	1
0D	JUMP	F
0E	<0>	0
0F	<0>	0
10	[CAL	E
11	[LONS	A
12	JUMP	F
13	<0>	0
14	<0>	0

(フローチャート)



Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ   キーを順に押し実行します。

Ⓒ  キー.....短音  キー.....長音
 モールス発信をしてください。きれいに打てます。

●モールス エレキーについて

エレクトロニック・キーヤーは、略してエレキーと呼ばれています。モールス符号の短点と長点が、簡隔のそろったきれいな符号で、電子的に発生できるもので、さかんに使用されるようになりました。

今迄の電鍵ですと、入のカンや手のくせで正確にモールス符号を打つことができないということがたびたびありました。又、受信している人も正確に解読できず、違った文章で解読するということがありました。しかし、モールス符号の短点と長点の正確な発信が、エレキー（左右の横振れ電鍵）を左右に動かすだけで可能になりました。

エレキーを使えば、だれでも、早くて正確なモールス符号を打つことができることは、大きな魅力です。アマチュア無線を楽しんでいる人達の間では愛好者が急速にふえてきています。

No. 83 メトロノーム

音楽の早さを決めるメトロノームを作ってみましょう。
メモリ50を拍子、メモリ51を早さの指定メモリとします。

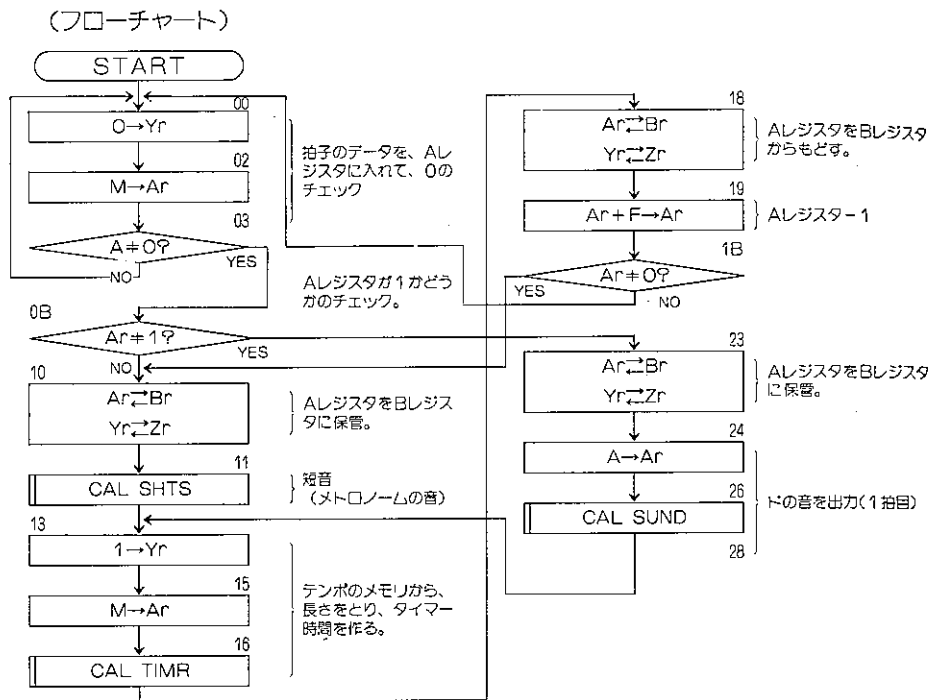
●プログラム

アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	TIY	A	16	CAL	E
01	<0>	0	17	TIMR	C
02	MA	5	18	CH	2
03	CIA	C	19	AIA	9
04	<0>	0	1A	<F>	F
05	JUMP	F	1B	CIA	C
06	<0>	0	1C	<0>	0
07		B	1D	JUMP	F
08	JUMP	F	1E	<1>	1
09	<0>	0	1F	<0>	0
0A	<0>	0	20	JUMP	F
0B	CIA	C	21	<0>	0
0C	<1>	1	22	<0>	0
0D	JUMP	F	23	CH	2
0E	<2>	2	24	TIA	8
0F	<3>	3	25	<A>	A
10	CH	2	26	CAL	E
11	CAL	E	27	SUND	B
12	SHTS	9	28	JUMP	F
13	TIY	A	29	<1>	1
14	<1>	1	2A	<3>	3
15	MA	5			

- (A) プログラムの書き込みと、確認をしてください。
- (B) メモリ50(拍子)、51(早さ)にデータを書き込みます。
(例) 50<2>、51<3> (大きい数字になるほどテンポが遅い)

RESET 50 51 ADDRESS 50 51 INCR 50 51

- (C) RESET 50 RUN キーを順に押し実行します。




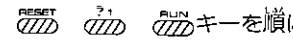
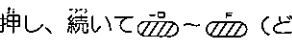
No. 86 おとあな 落とし穴ゲーム

メモリ50~5Fまでの1ヶ所を落とし穴として0を書き込みます。(メモリ、5Fには0を書いてはいけません。) ゲームは16進加算で行い、ゲームをスタートすると不特定なアドレスに止まります。

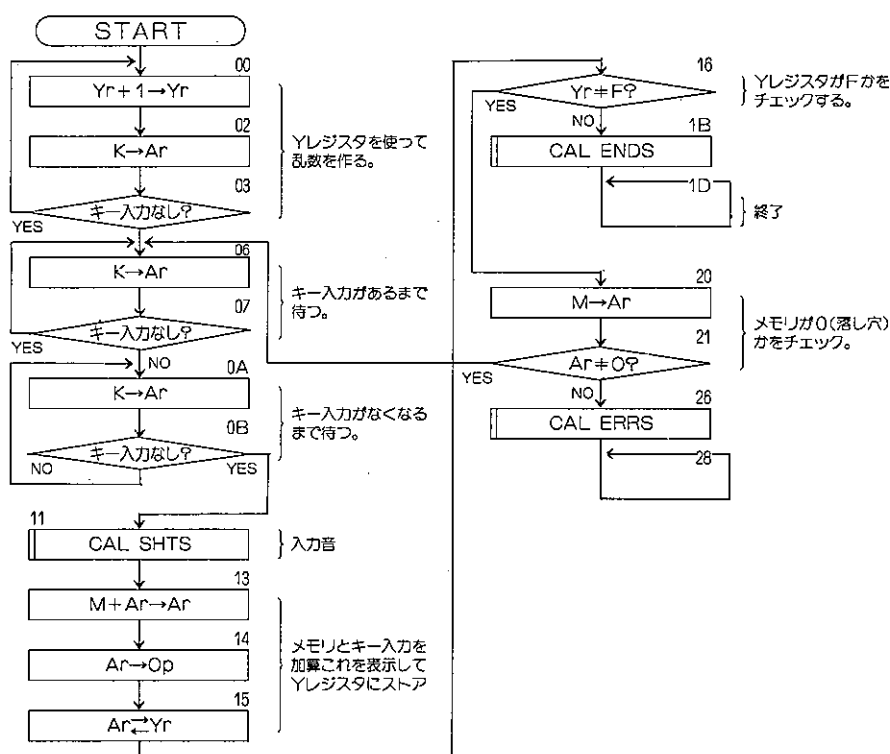
キー入力の値と表示される数字LEDの指定するメモリの値を加算した値がFになれば、ゲーム終了でエンド音が出ます。Fにならないばあい加算したアドレスへ移動しますから、くりかえしキー入力を行いFをさがします。加算した値が落とし穴のアドレスに止まるとエラー音が出ます。

●プログラム



アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	A I Y	B	17	<F>	F
01	<1>	1	18	JUMP	F
02	KA	0	19	<2>	2
03	JUMP	F	1A	<0>	0
04	<0>	0	1B	CAL	E
05	<0>	0	1C	ENDS	7
06	KA	0	1D	JUMP	F
07	JUMP	F	1E	<1>	1
08	<0>	0	1F	<D>	D
09	<6>	6	20	MA	5
0A	KA	0	21	C I A	C
0B	JUMP	F	22	<0>	0
0C	<1>	1	23	JUMP	F
0D	<1>	1	24	<0>	0
0E	JUMP	F	25	<6>	6
0F	<0>	0	26	CAL	E
10	<A>	A	27	ERRS	8
11	CAL	E	28	JUMP	F
12	SHTS	9	29	<2>	2
13	M+	6	2A	<8>	8
14	AO	1			
15	CY	3			
16	C I Y	D			

- Ⓐ プログラムの書き込みと、確認をしてください。
- Ⓑ メモリ50~5F迄の1ヶ所に落とし穴として0を書き込みます。
 (例) 50<1> 51<2> 52<0> 53<3> -----

- Ⓒ  キーを順に押し、続いて (どのキーでも良い) キーを押し実行します。最初は表示はしていませんが0~Fのキーを押すとアドレスが決まり数字LEDに表示されゲームスタートです。0~Fのキーを押すとアドレスが移動します。
- Ⓓ アドレスがFになれば終了(エンド音) アドレスが落とし穴の位置に止まると(エラー音)



(フローチャート)




No. 87 うんどうしんけい 運動神経ゲーム

2進LED、R3が点滅します。R3が消えているときに、  のキーを押し合い、単く押した側に点数が入り、2進LEDが点燈します。点数が7点になればゴールとなり、ゴール音を出します。R3が点燈中にキーを押すと、押した側の点数はフリヤされます。

4 2 1 8 4 2 1 満点
● ● ● ● ● ● ● 全部点燈7点

 キーの入の点数 $\frac{\text{点}}{\text{滅}}$  キーの入の点数

- Ⓐ プログラムの書き込みと、確認をしてください。
- Ⓑ    キーを順に押し実行します。
- Ⓒ もう一度行うには、   で行います。

●プログラム

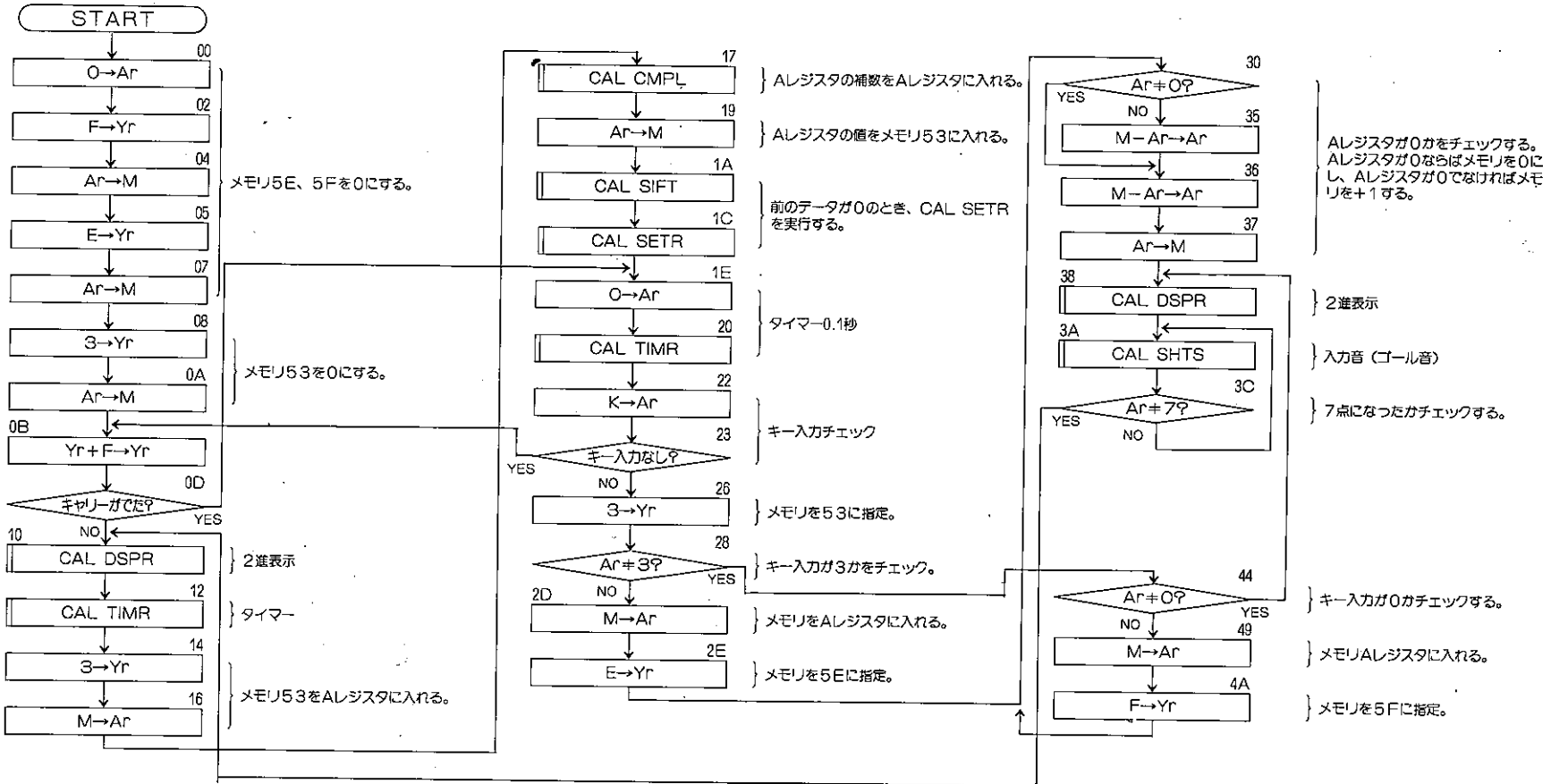
①		
アドレス	命令記号	命令コード
00	TIA	8
01	<0>	0
02	TIY	A
03	<F>	F
04	AM	4
05	TIY	A
06	<E>	E
07	AM	4
08	TIY	A
09	<3>	3
0A	AM	4
0B	AIY	B
0C	<F>	F
0D	JUMP	F
0E	<1>	1
0F	<E>	E
10	┌CAL	E
11	└DSPR	D
12	┌CAL	E
13	└TIMR	C

②		
アドレス	命令記号	命令コード
14	TIY	A
15	<3>	3
16	MA	5
17	┌CAL	E
18	└CMPL	4
19	AM	4
1A	┌CAL	E
1B	└SIFT	6
1C	┌CAL	E
1D	└SETR	1
1E	TIA	8
1F	<0>	0
20	┌CAL	E
21	└TIMR	C
22	KA	0
23	JUMP	F
24	<0>	0
25		B
26	TIY	A
27	<3>	3




③		
アドレス	命令記号	命令コード
28	CIA	C
29	<3>	3
2A	JUMP	F
2B	<4>	4
2C	<4>	4
2D	MA	5
2E	TIY	A
2F	<E>	E
30	CIA	C
31	<0>	0
32	JUMP	F
33	<3>	3
34	<6>	6
35	M-	7
36	M-	7
37	AM	4
38	┌CAL	E
39	└DSPR	D
3A	┌CAL	E
3B	└SHTS	9

④		
アドレス	命令記号	命令コード
3C	CIA	C
3D	<7>	7
3E	JUMP	F
3F	<1>	1
40	<0>	0
41	JUMP	F
42	<3>	3
43	<A>	A
44	CIA	C
45	<0>	0
46	JUMP	F
47	<3>	3
48	<8>	8
49	MA	5
4A	TIY	A
4B	<F>	F
4C	JUMP	F
4D	<3>	3
4E	<0>	0


(フローチャート)



No. 88 すうじ 数字ゲーム

   キー入力後、1回自のキー入力（1～Fのどのキーでもよい）は、親のカードとなります。2回自のキー入力（1～Fのどのキーでもよい）は、子のカードとなりこの値だけを表示します。

表示カードが小さい場合は、何度でもカードを引く（キー入力をする）ことができます。引いたカードは加算されて表示され、値がFをこえるとエラー音を出します。

親カードよりも大きくなったと思うところで勝負キーを押します。親と同じか親より大きい値であれば勝となりエンド音を出します。親より小さい場合はエラー音を出します。

●プログラム




①		
アドレス	命令記号	命令コード
00	AIY	B
01	<1>	1
02	KA	0
03	JUMP	F
04	<0>	0
05	<0>	0
06	CY	3
07	TIY	A
08	<0>	0
09	AM	4
0A	[CAL	E
0B	[SHTS	9
0C	KA	0
0D	JUMP	F
0E	<1>	1
0F	<3>	3
10	JUMP	F
11	<0>	0
12	<C>	C
13	AIY	B



②		
アドレス	命令記号	命令コード
14	<1>	1
15	KA	0
16	JUMP	F
17	<1>	1
18	<3>	3
19	CY	3
1A	TIY	A
1B	<1>	1
1C	AM	4
1D	AO	1
1E	[CAL	E
1F	[SHTS	9
20	KA	0
21	JUMP	F
22	<2>	2
23	<7>	7
24	JUMP	F
25	<2>	2
26	<0>	0
27	AIY	B

③		
アドレス	命令記号	命令コード
28	<1>	1
29	KA	0
2A	JUMP	F
2B	<2>	2
2C	<7>	7
2D	CIA	C
2E	<0>	0
2F	JUMP	F
30	<3>	3
31	<F>	F
32	CY	3
33	MA	5
34	TIY	A
35	<1>	1
36	M-	7
37	JUMP	F
38	<4>	4
39	<A>	A
3A	[CAL	E
3B	[ENDS	7

④		
アドレス	命令記号	命令コード
3C	JUMP	F
3D	<3>	3
3E	<C>	C
3F	CY	3
40	TIY	A
41	<1>	1
42	M+	6
43	JUMP	F
44	<4>	4
45	<A>	A
46	AM	4
47	JUMP	F
48	<1>	1
49	<D>	D
4A	[CAL	E
4B	[ERRS	8
4C	JUMP	F
4D	<4>	4
4E	<C>	C

Ⓐ プログラムの書き込みと、確認をしてください。

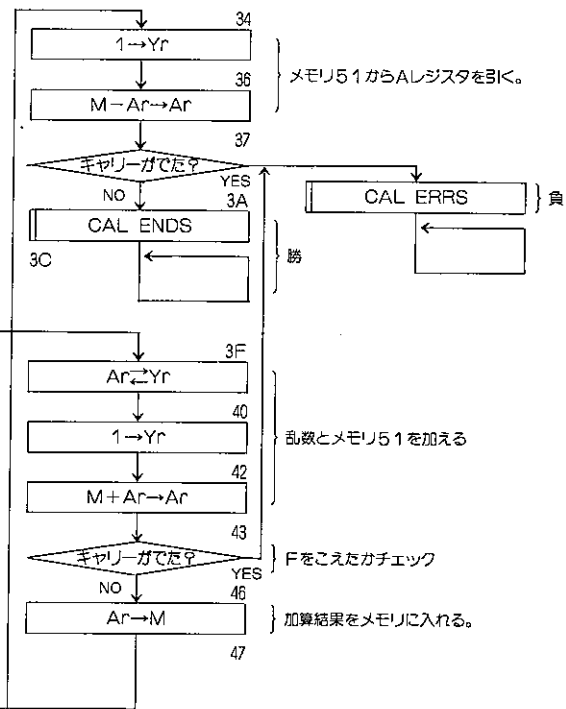
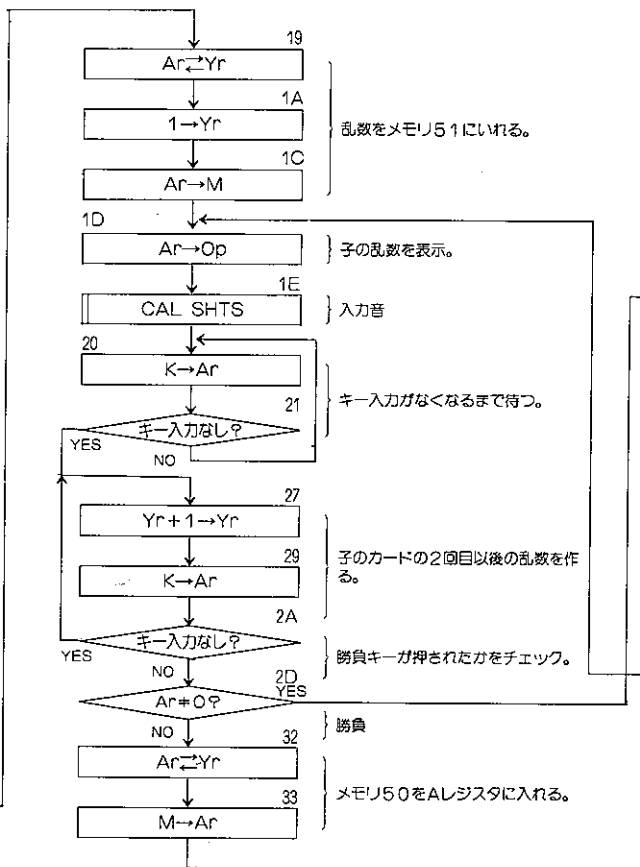
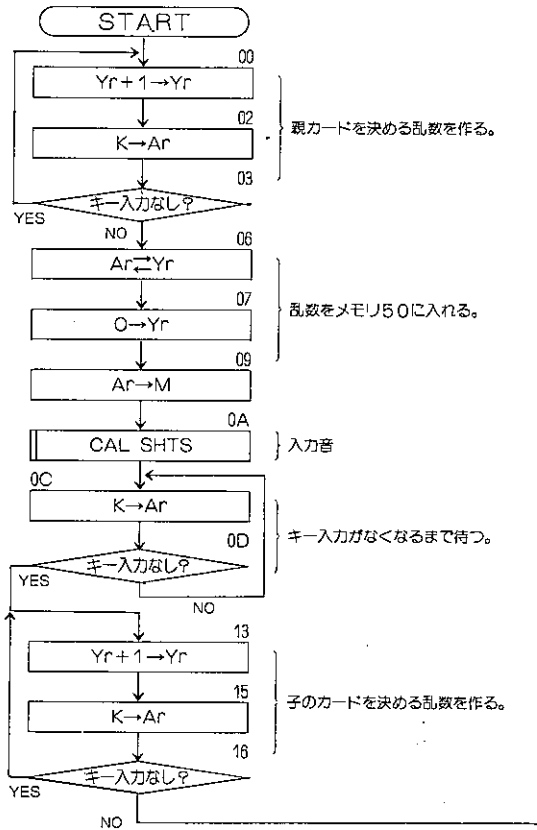
Ⓑ    キーを順に押し実行します。

Ⓒ 1回自のキー入力  表示ナシ。2回自のキー入力 で表示 
 （親カード） （子カード）

勝負結果の値は、親はメモリ50、子はメモリ51に入っています。

Ⓓ もう一度行うとき、   で行います。

(フローチャート)



No. 89 神経衰弱ゲーム

しんけいすいじやく

キー入力で指定したメモリを表示させます。2回目のキー入力で前回の表示と一致していれば正解音（エンド音）が出ます。違っていれば、エラー音を出します。トランプの神経衰弱と同様にゲームします。

●プログラム

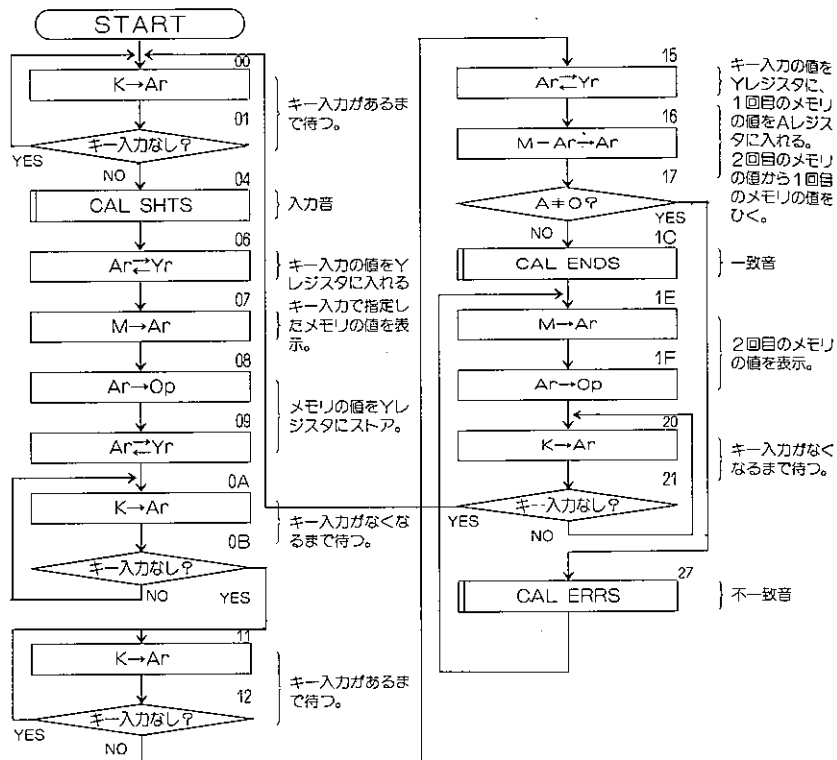
アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	KA	0	19	JUMP	F
01	JUMP	F	1A	<2>	2
02	<0>	0	1B	<7>	7
03	<0>	0	1C	[CAL	E
04	[CAL	E	1D	[ENDS	7
05	[SHTS	9	1E	MA	5
06	CY	3	1F	AO	1
07	MA	5	20	KA	0
08	AO	1	21	JUMP	F
09	CY	3	22	<0>	0
0A	KA	0	23	<0>	0
0B	JUMP	F	24	JUMP	F
0C	<1>	1	25	<2>	2
0D	<1>	1	26	<0>	0
0E	JUMP	F	27	[CAL	E
0F	<0>	0	28	[ERRS	8
10	<A>	A	29	JUMP	F
11	KA	0	2A	<1>	1
12	JUMP	F	2B	<E>	E
13	<1>	1			
14	<1>	1			
15	CY	3			
16	M-	7			
17	CIA	C			
18	<0>	0			

- (A) プログラムの書き込みと、確認をしてください。
- (B) メモリ50~5Fに、2つ、づつ同じデータを書き込みます。
〔例〕50<3> 51<1> 52<3> 53<2> -----

RESET 15 00 ADABET 13 INCR 21 INCR 13 INCR -----

- (C) RESET 21 RUN キーを順に押し実行します。
21 ~ 27のキーを押すとメモリ50~5Fまでの値が表示されます。
- (D) 1回目のキー入力で指定したメモリを表示させる。
2回目のキー入力で前回の表示と一致すれば
正解音（エンド音）、違っていればエラー音が出ます。

(フローチャート)






No. 90 加算ゲーム

かさん

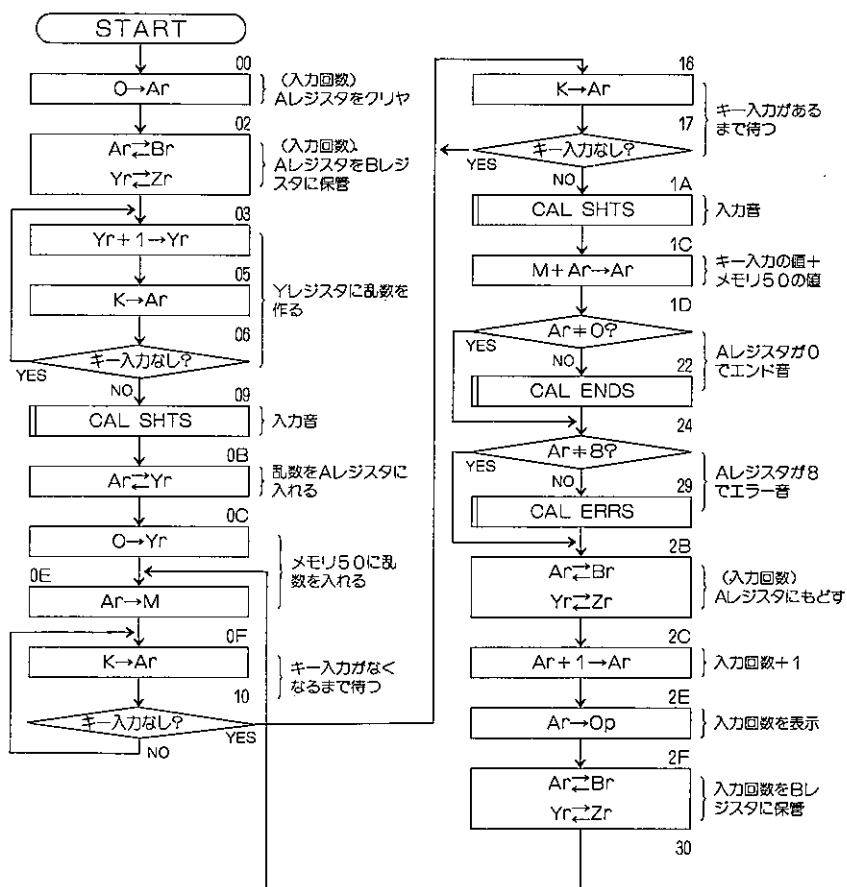
初めのキー入力で、不明の値の乱数が発生します。この乱数にキー入力の値を加算して0にして終了します。(エンド音出る)加算した値が8になると、エラー音が出ます。キー入力の回数を数字LEDに表示しますから何回で終了できるかためてみましょう。

●プログラム

アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	TIA	8	1A	CAL	E
01	<0>	0	1B	SHTS	9
02	CH	2	1C	M+	6
03	A I Y	B	1D	C I A	C
04	<1>	1	1E	<0>	0
05	KA	0	1F	JUMP	F
06	JUMP	F	20	<2>	2
07	<0>	0	21	<4>	4
08	<3>	3	22	CAL	E
09	CAL	E	23	ENDS	7
0A	SHTS	9	24	C I A	C
0B	CY	3	25	<8>	8
0C	T I Y	A	26	JUMP	F
0D	<0>	0	27	<2>	2
0E	AM	4	28		B
0F	KA	0	29	CAL	E
10	JUMP	F	2A	ERRS	8
1.1	<1>	1	2B	CH	2
12	<6>	6	2C	A I A	9
13	JUMP	F	2D	<1>	1
14	<0>	0	2E	AO	1
15	<F>	F	2F	CH	2
16	KA	0	30	JUMP	F
17	JUMP	F	31	<0>	0
18	<1>	1	32	<E>	E
19	<6>	6			

- (A) プログラムの書き込みと、確認をしてください。
- (B)    キーを順に押し実行します。
- (C) 初めのキー入力(1~Fのどのキーを押ししてもよい)で開始です。

(フローチャート)



No. 91 かずあて 数当てゲーム

メモリ50に書き込んである0~Fまでの数を当てるゲームです。書き込まれた数を想像してキー入力します。メモリ50とキー入力がかじであればエンド音を出します。又キー入力が±2以内であれば、ポツポツという音それ以外ではポツという音が出ます。数字LEDにはキー入力の回数が表示されます。

●プログラム

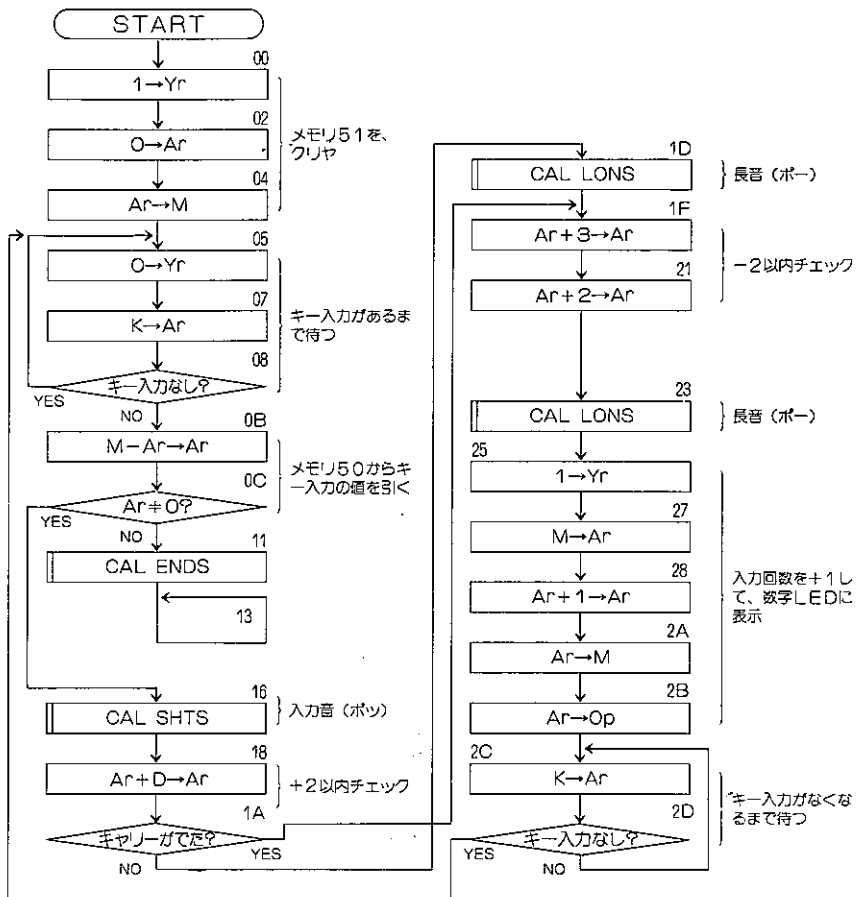
アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	TIY	A	1B	<1>	1
01	<1>	1	1C	<F>	F
02	TI A	8	1D	[CAL	E
03	<0>	0	1E	LONS	A
04	AM	4	1F	AIA	9
05	TIY	A	20	<3>	3
06	<0>	0	21	AIA	9
07	KA	0	22	<2>	2
08	JUMP	F	23	[CAL	E
09	<0>	0	24	LONS	A
0A	<5>	5	25	TIY	A
0B	M-	7	26	<1>	1
0C	CIA	C	27	MA	5
0D	<0>	0	28	AIA	9
0E	JUMP	F	29	<1>	1
0F	<1>	1	2A	AM	4
10	<6>	6	2B	AO	1
11	[CAL	E	2C	KA	0
12	LENS	7	2D	JUMP	F
13	JUMP	F	2E	<0>	0
14	<1>	1	2F	<5>	5
15	<3>	3	30	JUMP	F
16	[CAL	E	31	<2>	2
17	LSHTS	9	32	<C>	C
18	AIA	9			
19	<D>	D			
1A	JUMP	F			

- (A) プログラムの書き込みと、確認をしてください。
- (B) メモリ50に1~Fの任意のデータ(数)を書き込みます。
(例)メモリ50<5>を書き込む。



- (C) キーを順に押し実行します。
- (D) メモリ50に書き込まれた数を想像してキー入力します。

(フローチャート)



No. 92 ターゲットゲーム

標的の光が2進LEDの6から0に向かって順に移動します。点灯しているLEDの番号と同じ値のキーを押して光を当てます。うまく当たれば表示はストップし、エンド音を出します。

標的が右へ移動し終わったところでエラー音を出します。

●プログラム

①			②		
アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	TIA	8	18	<2>	2
01	<6>	6	19	<A>	A
02	TIY	A	1A	┌CAL	E
03	<0>	0	1B	└ENDS	7
04	AM	4	1C	MA	5
05	MA	5	1D	CY	3
06	CY	3	1E	┌CAL	E
07	┌CAL	E	1F	└SETR	1
08	└SETR	1	20	JUMP	F
09	KA	0 TIA 8	21	<2>	2
0A	KA	0 <0> 0	22	<0>	0
0B	KA	0 ┌CAL E	23	AIY	B
0C	KA	0 └TIMR C	24	<F>	F
0D	KA	0	25	JUMP	F
0E	KA	0	26	<0>	0
0F	JUMP	F	27	<9>	9
10	<2>	2	28	TIY	A
11	<3>	3	29	<0>	0
12	TIY	A	2A	MA	5
13	<0>	0	2B	CY	3
14	M-	7	2C	┌CAL	E
15	CIA	C	2D	└RSTR	2
16	<0>	0	2E	KA	0
17	JUMP	F	2F	JUMP	F




③

アドレス	命令記号	命令コード
30	<3>	3
31	<5>	5
32	JUMP	F
33	<2>	2
34	<E>	E
35	CY	3
36	AIA	9
37	<F>	F

④

アドレス	命令記号	命令コード
38	JUMP	F
39	<0>	0
3A	<2>	2
3B	┌CAL	E
3C	└ERRS	8
3D	JUMP	F
3E	<3>	3
3F	<D>	D

Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ    キーを順に押し実行すると、すぐにスタートします。

Ⓒ 点灯しながら移動するLEDの番号と同じ値のキーを押してください。

- 止った所の2進LEDの位置を点数にして、2~3人でゲームを試みるのもおもしろいでしょう。
光が走るターゲットです。



しんかさん No. 93 16進加算ゲーム

メモリ50～5Fまでの内容を一定時間、5Fから順に数字LEDに表示します。表示中にキー入力すると、表示内容とキー入力を16進加算し、ケタ上げで0になると得点されます。キー入力16回くり返すとエンド音を出し得点を表示します。
 $5+B=16 \rightarrow 0=1$ 点

●プログラム

①			②		
アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	TIA	8	18	<0>	0
01	<0>	0	19	JUMP	F
02	CH	2	1A	<3>	3
03	TIY	A	1B	<9>	9
04	<F>	F	1C	CH	2
05	KA	0	1D	AIA	9
06	JUMP	F	1E	<1>	1
07	<0>	0	1F	CH	2
08	<C>	C	20	[CAL	E
09	JUMP	F	21	LONS	A
0A	<3>	3	22	A IY	B
0B	<E>	E	23	<F>	F
0C	MA	5	24	JUMP	F
0D	AO	1	25	<0>	0
0E	TIY A	8	26	<5>	5
0F	<F>	F	27	CH	2
10	[CAL	E	28	AO	1
11	[TIMR	C	29	[CAL	E
12	KA	0	2A	[ENDS	7
13	JUMP	F	2B	JUMP	F
14	<2>	2	2C	<2>	2
15	<E>	E	2D		B
16	M+	6	2E	TIA	8
17	CIA	C	2F	<0>	0

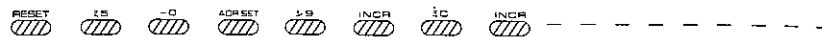
③			④		
アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
30	[CAL	E	3B	JUMP	F
31	[TIMR	C	3C	<2>	2
32	KA	0	3D	<2>	2
33	JUMP	F	3E	[CAL	E
34	<3>	3	3F	[RSTO	0
35	<9>	9	40	JUMP	F
36	JUMP	F	41	<0>	0
37	<1>	1	42	<5>	5
38	<6>	6			
39	[CAL	E			
3A	[SHTS	9			

Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ メモリ50～メモリ5Fまでにデータを書き込みます。

(例)

50	51	52	53	54	55	...	5B	5C	5D	5E	5F
9	C	4	A	F	3		E	1	8	B	6



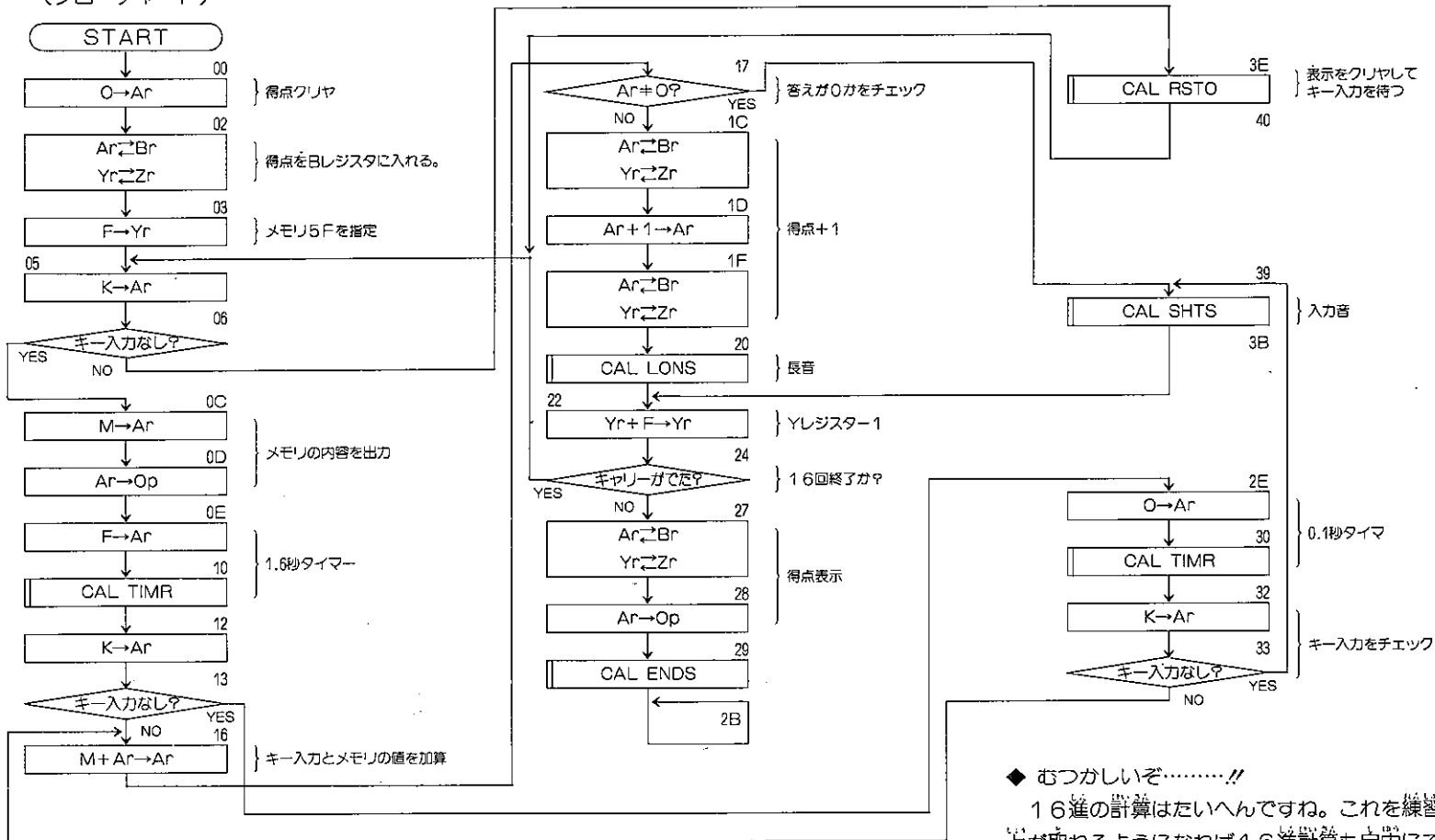
Ⓒ [RESET] [L/R] [RUN] キーを順に押し実行します。

Ⓓ メモリに書き込んであるデータを表示中にキー入力すると、表示内容とキー入力を16進加算します。ケタ上げで0になると得点となります。

Ⓔ 最後にエンド音を出し得点を表示します。

⑤ 短音が鳴るまでに加算させるキーを押し、鳴り終るまで押してください。キー入力がなくても16回短音を出しメモリの内容を表示して最後にエンド音を出します。

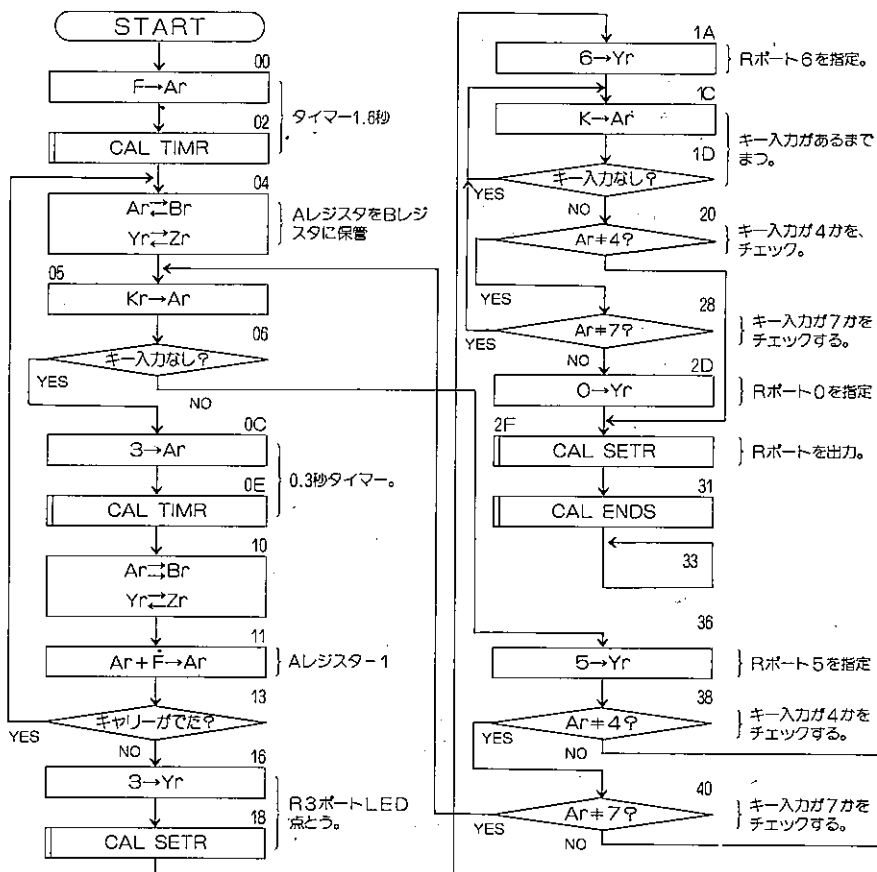
(フローチャート)



◆ おつかしいぞ……!!

16進の計算はたいへんですね。これを練習して、10歳以上が取れるようになれば16進計算も自由にできるようになりコンピュータを扱うときに大変便利になります。

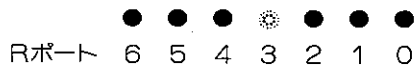
(フローチャート)



Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ キーを順に押し実行します。(再ゲームの時も同じ)

Ⓒ しばらく待ち(8秒程) LEDの点灯を確認して、 キーと キーを押し合います。



● 運動神経を競うゲームです。

タイミングを計って、キーの押し合いです。フライングはいけません。早く撃たなければなりません。スリルがありますね。

注意：Rポート○と書いてあるのは
2進LEDの位位を示す。

● ● ● ● ● ● ● ←Rポート
R6 R5 R4 R3 R2 R1 R0

No. 95 スロットルマシン

スロットルマシンを組んでみました。
 図の様に (イ) (ロ) (ハ) の部分に分けて表示します。



(イ) の表示部は、7までの表示しかしません。0~Fの値ですから8以上の数は想像します。

イ、ロ、ハの全てが一致した場合は、エンド音が続けて2回鳴ります。〈イとロ〉又は〈ロとハ〉が一致した場合には、エンド音が1回鳴ります。

スタートしますと、表示は早い速度で変わります。まず キーを押して、(イ) の表示をストップさせます。続いて キーを押して (ロ) の表示をストップさせます。次は キーを押して (ハ) の表示をストップさせます。これで一致した表示があればエンド音を鳴らします。

●プログラム

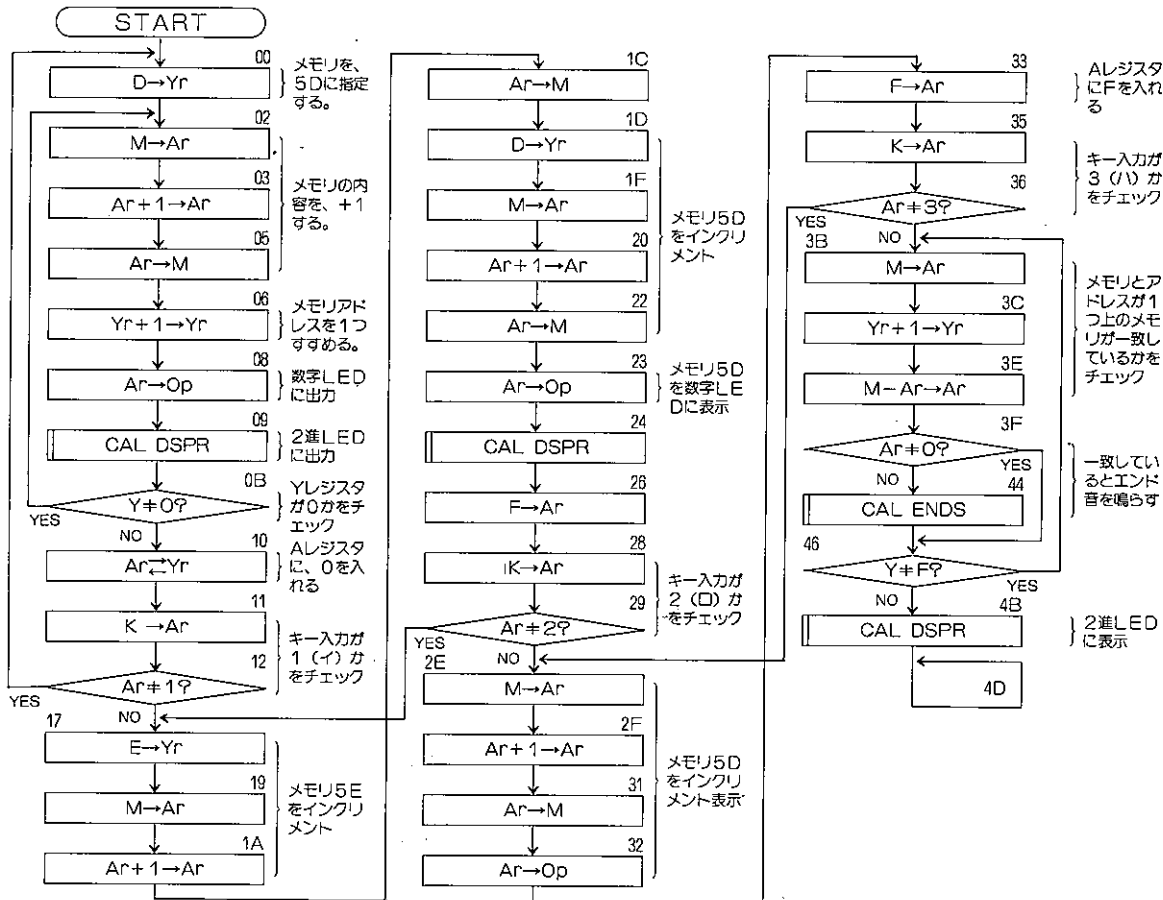
①		
アドレス	命令記号	命令コード
00	T I Y	A
01	<D>	D
02	MA	5
03	A I A	9
04	<1>	1
05	AM	4
06	A I Y	B
07	<1>	1
08	AO	1
09	┌CAL	E
0A	└DSPR	D
0B	C I Y	D
0C	<0>	0
0D	JUMP	F

②		
アドレス	命令記号	命令コード
0E	<0>	0
0F	<2>	2
10	CY	3
11	KA	0
12	C I A	C
13	<1>	1
14	JUMP	F
15	<0>	0
16	<0>	0
17	T I Y	A
18	<E>	E
19	MA	5
1A	A I A	9
1B	<1>	1

③		
アドレス	命令記号	命令コード
1C	AM	4
1D	T I Y	A
1E	<D>	D
1F	MA	5
20	A I A	9
21	<1>	1
22	AM	4
23	AO	1
24	┌CAL	E
25	└DSPR	D
26	T I A	8
27	<F>	F
28	KA	0
29	C I A	C
2A	<2>	2
2B	JUMP	F
2C	<1>	1
2D	<7>	7
2E	MA	5
2F	A I A	9
30	<1>	1
31	AM	4
32	AO	1
33	T I A	8
34	<F>	F
35	KA	0

④		
アドレス	命令記号	命令コード
36	C I A	C
37	<3>	3
38	JUMP	F
39	<2>	2
3A	<E>	E
3B	MA	5
3C	A I Y	B
3D	<1>	1
3E	M-	7
3F	C I A	C
40	<0>	0
41	JUMP	F
42	<4>	4
43	<6>	6
44	┌CAL	E
45	└ENDS	7
46	C I Y	D
47	<F>	F
48	JUMP	F
49	<3>	3
4A		B
4B	┌CAL	E
4C	└DSPR	D
4D	JUMP	F
4E	<4>	4
4F	<D>	D

(フローチャート)



Ⓐ プログラムの書き込みと、確認をしてください。

Ⓑ キーを順に押し実行します。

Ⓒ イ、ロ、ハのLEDの表示が一致しない場合、再度 キーを押し挑戦してください。

●ゲームセンターにあるスロットルマシンと同じですね。

ちがいは、コインが出てこないことでしょう。1人でも楽しめるので連だめに、やってみるのもおもしろいですよ。



7 (イ) 7 (ロ) 7 (ハ)

イ、ロ、ハ、全てが一致、エンド音が続けて2回鳴ります。

やった~~~~!!

No. 96 数字の記憶力テスト

あらかじめ、メモリ50～5Eの値を読み出して、おぼえておきましょう。
メモリ50の値から順に記憶をたどりながらキーを押して当てていきます。
エンド音はエラー音を出して点数を表示します。

●プログラム

アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	TIA	8	1A	[CAL	E
01	<0>	0	1B	[SHTS	9
02	CH	2	1C	CH	2
03	TIY	A	1D	AIA	9
04	<0>	0	1E	<1>	1
05	KA	0	1F	CH	2
06	JUMP	F	20	A IY	B
07	<0>	0	21	<1>	1
08	<5>	5	22	C IY	D
09	KA	0	23	<F>	F
0A	JUMP	F	24	JUMP	F
0B	<1>	1	25	<0>	0
0C	<0>	0	26	<5>	5
0D	JUMP	F	27	[CAL	E
0E	<0>	0	28	[ENDS	7
0F	<9>	9	29	CH	2
10	[CAL	E	2A	AO	1
11	[CMPL	4	2B	JUMP	F
12	M+	6	2C	<2>	2
13	C I A	C	2D		B
14	<F>	F	2E	[CAL	E
15	JUMP	F	2F	[ERRS	8
16	<2>	2	30	JUMP	F
17	<E>	E	31	<2>	2
18	MA	5	32	<9>	9
19	AO	1			

- (A) プログラムの書き込みと、確認をしてください。
(B) メモリ50～5Eに任意のデータ(値)を書き込みます。

(例)

50	51	52	53	54	55	56	57	
3	4	5	6	7	8	9	A	

----- キーを押す。

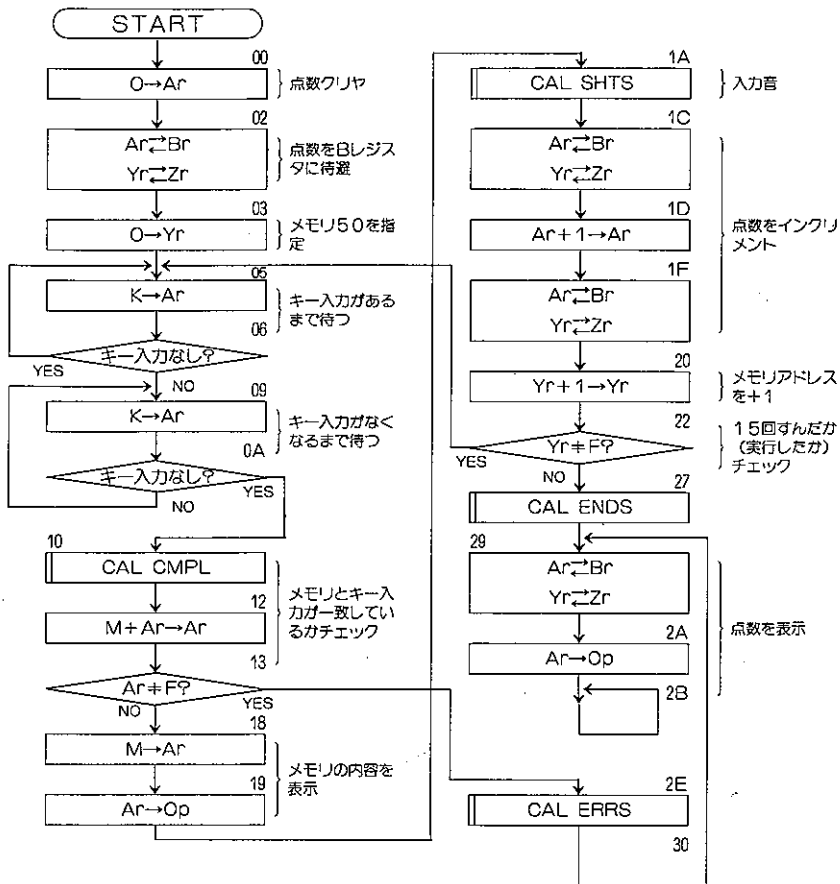
- (C) キーを順に押し実行します。

- (D) メモリ50から記憶している値を順にキーを押します。
正しい場合はその値と、入力音が出ます。
エラー音は、エンド音が出るまで入力します。

(例) ----- (Fで満点)
エラー音は、エンド音で点数を表示します。 (10点)

- ひき算、 $M - Ar \rightarrow Ar$ のかわりに、 $CAL CMPL$ と、たし算、 $M + Ar \rightarrow Ar$ を使ってみました。

(フローチャート)



No. 97 0~9までの^{らんすう}乱数をメモリへストア




メモリ50~5Fに乱数を書き込むプログラムです。


乱数とは、きまつた順序のない大きさの数字のことで、お互いの大きさに無関係な数字が並びます。

●プログラム

アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	T I Y	A	19	<0>	0
01	<0>	0	1A	<4>	4
02	T I A	8	1B	JUMP	F
03	<F>	F	1C	<0>	0
04	A I A	9	1D	<2>	2
05	<1>	1	1E	CH	2
06	CH	2	1F	AM	4
07	KA	0	20	A I Y	B
08	JUMP	F	21	<1>	1
09	<1>	1	22	C I Y	D
0A	<5>	5	23	<0>	0
0B	CH	2	24	JUMP	F
0C	AO	1	25	<1>	1
0D	CH	2	26	<6>	6
0E	KA	0	27	[CAL	E
0F	JUMP	F	28	[ENDS	7
10	<1>	1	29	JUMP	(E) F
11	<E>	E	2A	<2>	2
12	JUMP	F	2B	<9>	9
13	<0>	0			
14	<E>	E			
15	CH	2			
16	C I A	C			
17	<9>	9			
18	JUMP	F			

(A) プログラムの書き込みと、確認をしてください。

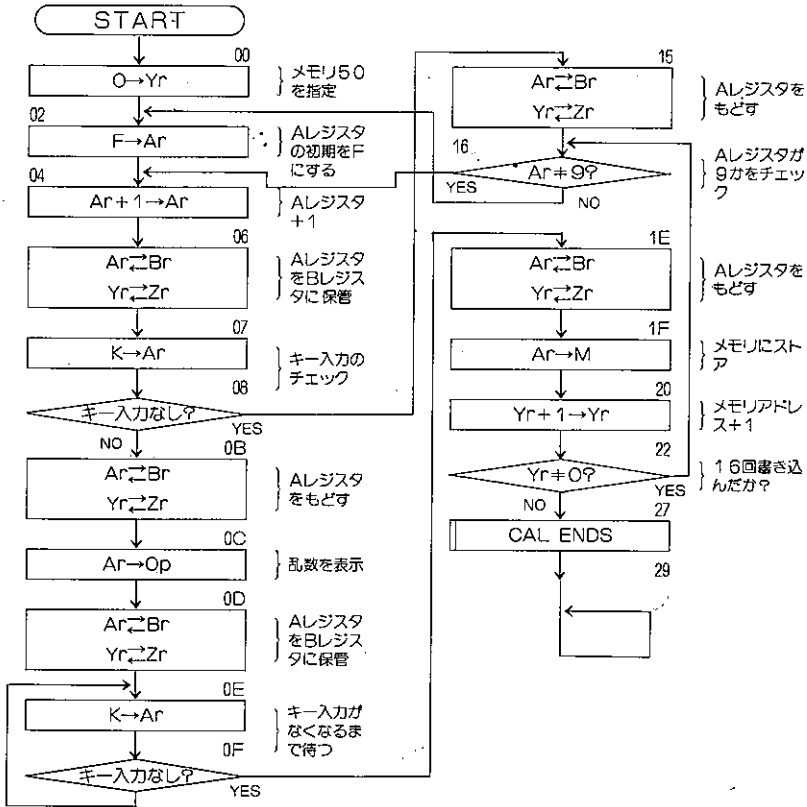
(B)    キーを順に押し実行します。

(C)  (数字LED) 表示を見ながら、エンド音が出るまでキー入力します。(0~F迄のどのキーを押してもよい)
乱数は、キー入力で作ります。キー入力があると、そのときの乱数を数字LEDに表示し、メモリにも書き込みます。

(D) 50~5F迄のメモリを読み出してみましょう。乱数が書き込まれています。

◆ $F \rightarrow Ar$ で、乱数の始めの値(0)を指定します。 $Ar \rightarrow 9$ で、乱数の終りの値(9)を指定しています。この二つの命令によって乱数の範囲を変えることができます。(このプログラムの乱数の範囲は0~9迄ですが、音楽用データの乱数は、1~E迄にします。)

(フローチャート)



No. 98 おんかい あ 音階当てゲーム

このゲームは、メモリーの50～5Eに書き込まれている音階データが、50から順に出力されるので、キー入力で、音を当てていきます。

メモリーの5Eまですべて正解すれば、END音を出し、途中でまちがえたら、ERR音を出して、その得点が表示されます。

※ 音階データの0とFは、無音ですから、音のデータとしてはいれません。

★ 勝気で、負けず嫌いなキミは……

キミのために、ちょっといい事を教えよう。

メモリーの50～5Eに、音のデータ(1～E)を書き込みますが入れた数字をすべて覚えておくのです。(ちょっと大変だけど、がんばって!) 友達と一緒にこのゲームをやれば、友達は、キミの音感の良さに感心するでしょう。(但し、友達には内緒にね!)

●プログラム

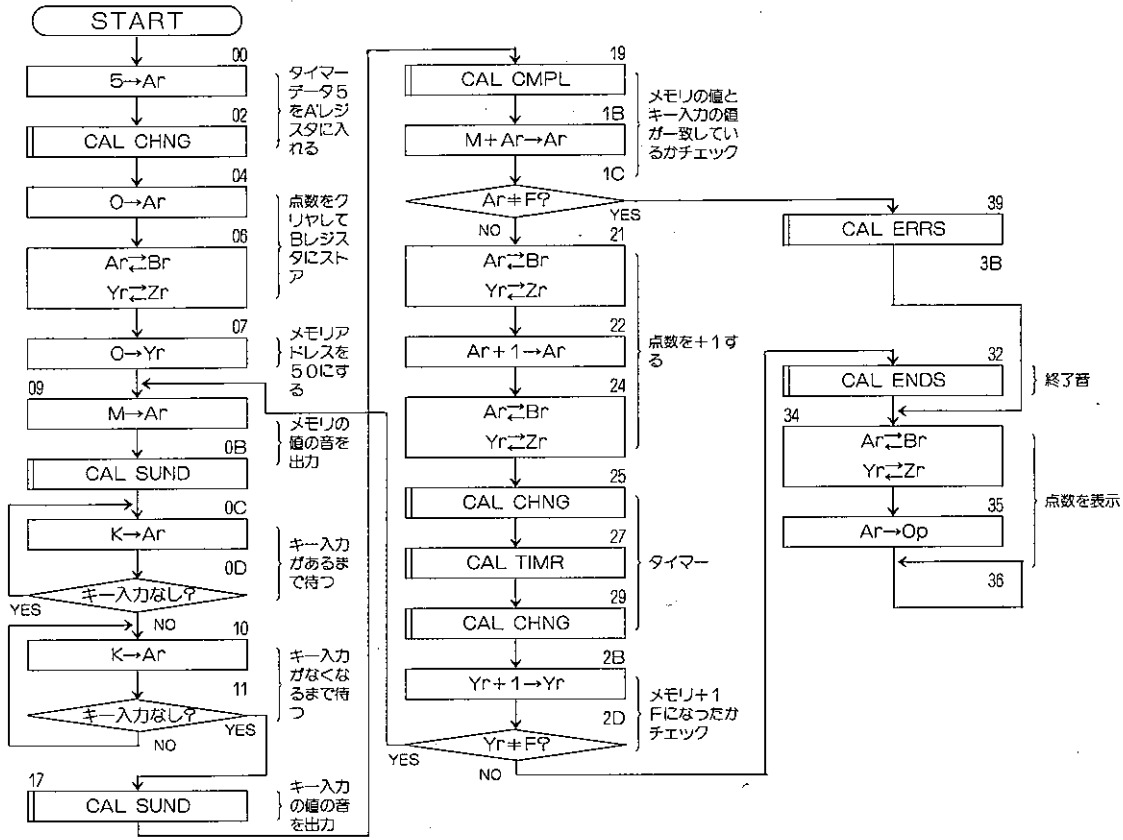
①		
アドレス	命令記号	命令コード
00	TIA	8
01	<5>	5
02	[CAL	E
03	[CHNG	5
04	TIA	8
05	<0>	0
06	CH	2
07	TIY	A
08	<0>	0
09	MA	5
0A	[CAL	E
0B	[SUND	B
0C	KA	0
0D	JUMP	F
0E	<0>	0
0F	<C>	C

②		
アドレス	命令記号	命令コード
10	KA	0
11	JUMP	F
12	<1>	1
13	<7>	7
14	JUMP	F
15	<1>	1
16	<0>	0
17	[CAL	E
18	[SUND	B
19	[CAL	E
1A	[CMPL	4
1B	M+	6
1C	CIA	C
1D	<F>	F
1E	JUMP	F
1F	<3>	3

③		
アドレス	命令記号	命令コード
20	<9>	9
21	CH	2
22	AIA	9
23	<1>	1
24	CH	2
25	[CAL	E
26	[CHNG	5
27	[CAL	E
28	[TIMR	C
29	[CAL	E
2A	[CHNG	5
2B	AIY	B
2C	<1>	1
2D	CIY	D
2E	<F>	F
2F	JUMP	F

④		
アドレス	命令記号	命令コード
30	<0>	0
31	<9>	9
32	[CAL	E
33	[ENDS	7
34	CH	2
35	AO	1
36	JUMP	F
37	<3>	3
38	<6>	6
39	[CAL	E
3A	[ERRS	8
3B	JUMP	F
3C	<3>	3
3D	<4>	4

(フローチャート)



Ⓐ メモリの50から順に、5Eまで、1~Eの範囲で音階データを書き込みます。

(例)



Ⓑ プログラムの書き込みと、確認をしてください。

Ⓒ RESET 71 ALPH キーを順に押すと、プログラムが実行され、ゲームがスタートします。




No. 99 ドレミファ音のおんらんすうの乱数をメモリへストア

メモリーに適当な数を入れる場合、キーを押して書き込むと、どうしても、荷を入れようか迷ったり、入れた数字がわかってしまっておもしろくなかったりします。でも、この実験を使えば、メモリーに荷が入るかは、誰もわからないので、フェアにゲームを楽しむことができます。

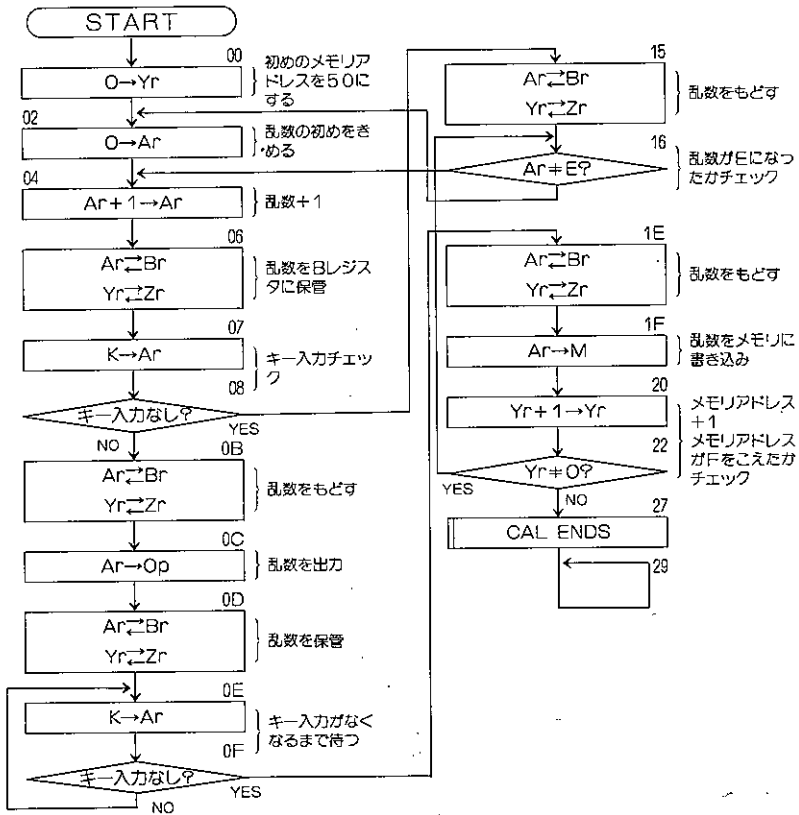
●プログラム

アドレス	命令記号	命令コード
00	T I Y	A
01	<0>	0
02	T I A	8
03	<0>	0
04	A I A	9
05	<1>	1
06	CH	2
07	KA	0
08	JUMP	F
09	<1>	1
0A	<5>	5
0B	CH	2
0C	AO	1
0D	CH	2
0E	KA	0
0F	JUMP	F
10	<1>	1
11	<E>	E
12	JUMP	F
13	<0>	0
14	<E>	E
15	CH	2

アドレス	命令記号	命令コード
16	C I A	C
17	<E>	E
18	JUMP	F
19	<0>	0
1A	<4>	4
1B	JUMP	F
1C	<0>	0
1D	<2>	2
1E	CH	2
1F	AM	4
20	A I Y	B
21	<1>	1
22	C I Y	D
23	<0>	0
24	JUMP	F
25	<1>	1
26	<6>	6
27	↑CAL	E
28	↓ENDS	7
29	JUMP	F
2A	<2>	2
2B	<9>	9

- ① プログラムの書き込みと、確認をしてください。
- ②    キーを順に押し実行します。
- ③ 数字キー（どれでもよい）を16回押してください。キーを押すごとに書き込まれる乱数が表示されます。
押し終ると、END音がでて、メモリに乱数が書き込まれます。
- ④ メモリ50～5Fまで1～Eの乱数が書き込まれました。読み出してみましょう。

(フローチャート)



No.100 もぐらたたき

メモリ50～5Eに、0～6の任意の値(数)を書き込みます。モグラは、メモリ5Eから順にメモリに書き込まれた位置に、罾を出すのですばやくたたいて得点します。メモリアドレスを-1して行き50番地のモグラが出終るとエンド音を出します。

●プログラム

①			②		
アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
00	TIA	8	18	CIA	C
01	<0>	0	19	<0>	0
02	┌CAL	E	1A	JUMP	F
03	└CHNG	5	1B	<2>	2
04	TIY	A	1C	<6>	6
05	<E>	E	1D	┌CAL	E
06	MA	5	1E	└SHTS	9
07	CY	3	1F	┌CAL	E
08	┌CAL	E	20	└CHNG	5
09	└SETR	1	21	AIA	9
0A	CY	3	22	<1>	1
0B	CH	2	23	AO	1
0C	TIY	A	24	┌CAL	E
0D	<F>	F	25	└CHNG	5
0E	KA	0	26	MA	5
0F	JUMP	F	27	CY	3
10	<3>	3	28	┌CAL	E
11	<9>	9	29	└RSTR	2
12	CH	2	2A	CY	3
13	KA	0	2B	A I Y	B
14	JUMP	F	2C	<F>	F
15	<3>	3	2D	JUMP	F
16	<5>	5	2E	<4>	4
17	M-	7	2F	<6>	6

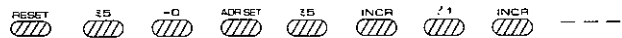
③			④		
アドレス	命令記号	命令コード	アドレス	命令記号	命令コード
30	┌CAL	E	3F	JUMP	F
31	└ENDS	7	40	<0>	0
32	JUMP	F	41	<E>	E
33	<3>	3	42	CH	2
34	<2>	2	43	JUMP	F
35	CH	2	44	<2>	2
36	JUMP	F	45	<6>	6
37	<0>	0	46	KA	0
38	<E>	E	47	JUMP	F
39	TIA	8	48	<0>	0
3A	<0>	0	49	<6>	6
3B	┌CAL	E	4B	JUMP	F
3C	└TIMR	C	4B	<4>	4
3D	A I Y	B	4C	<6>	6
3E	<F>	F			

① プログラムの書き込みと、確認をしてください。

② メモリ50～5Eに0～6までのデータを書き込みます。

(例)

50	51	52	53	54	55	56
5	1	6	2	5	0	4

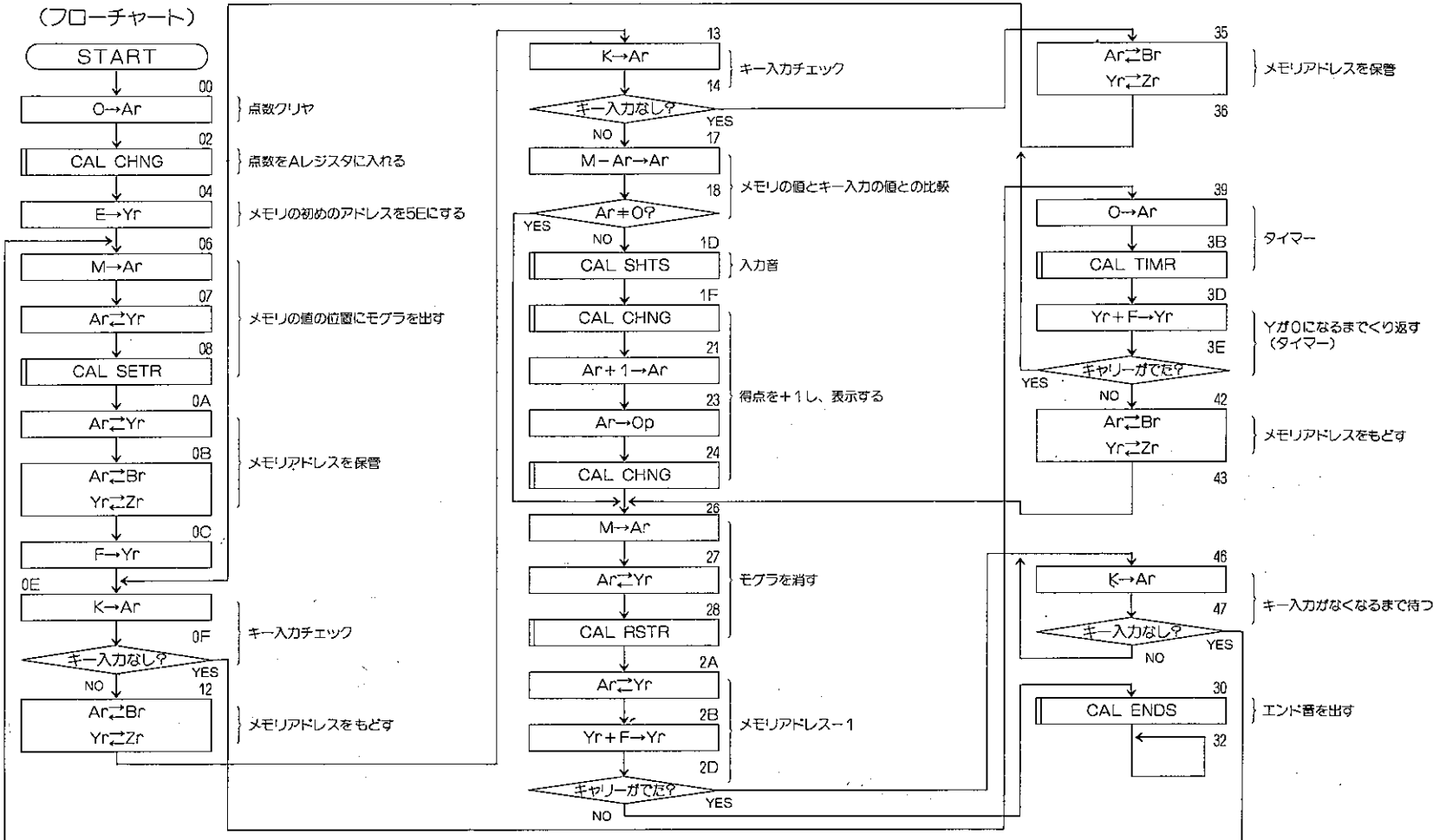


③ キーを順に押す。LED点燈ゲームスタート。



LED点燈中に、すばやく～のキーを押す。

(フローチャート)



● ^{あたら}新しいプログラムを^{つく}作ってみましょう。

アドレス	命令記号	命令コード
00		
01		
02		
03		
04		
05		
06		
07		
08		
09		
0A		
0B		
0C		
0D		
0E		
0F		

アドレス	命令記号	命令コード
10		
11		
12		
13		
14		
15		
16		
17		
18		
19		
1A		
1B		
1C		
1D		
1E		
1F		

アドレス	命令記号	命令コード
20		
21		
22		
23		
24		
25		
26		
27		
28		
29		
2A		
2B		
2C		
2D		
2E		
2F		

アドレス	命令記号	命令コード
30		
31		
32		
33		
34		
35		
36		
37		
38		
39		
3A		
3B		
3C		
3D		
3E		
3F		

アドレス	命令記号	命令コード
40		
41		
42		
43		
44		
45		
46		
47		
48		
49		
4A		
4B		
4C		
4D		
4E		
4F		



〈学習研究社〉

FXシリーズ説明書

- 発行人・馬淵・宏
- 回路考案・電子ブロック機器製造㈱
- 発行所・株式会社 学習研究社
- レイアウト・銜タツク
- 印刷所・ダイドー紙工㈱

昭和56年9月初版（無断複製・転載・翻訳を禁ず）
★本書および機器に関するお問合せは、
文書は ☎146 東京都大田区仲池上1-17-15
学研第2ビル 知育玩具事業部サービス部
電話は 東京(03)754-5344へお願い
いたします。

学研電子ブロック FX-マイコン R-165 命令記号早見表

命令コード	命令記号	フローチャート	実行フラグ	命令語のはたらき	説明ページ	命令コード	命令記号	フローチャート	実行フラグ	命令語のはたらき	説明ページ
0	ケー KA	K→Ar	0.1	Arに、押した数字キーの内容が入る。キー入力がないと実行フラグが1。	27	E0	コール CAL RSTO	CAL RSTO	1	数字LEDを消す。	73
1	エー AO	Ar→Op	1	Arの内容を数字LEDに点灯する。	23	E1	コール CAL SETR	CAL SETR	1	2進LEDを1個点灯する。	68
2	シー CH	Ar↔Br, Yr↔Zr	1	ArとBr, YrとZrの内容をそれぞれ、チェンジする。	25 42	E2	コール CAL RSTR	CAL RSTR	1	2進LEDを1個消す。	70
3	シー CY	Ar↔Yr	1	ArとYrの内容をチェンジする。	78	E3	コール CAL INPT	CAL INPT	1	マイコンユニット以外の入力をArに入れる。	157
4	イー AM	Ar→M	1	Arの内容をMに入れる。	44	E4	コール CAL CMPL	CAL CMPL	1	Arの内容を反転させる。	75
5	エー MA	M→Ar	1	Mの内容をArに入れる。	45	E5	コール CAL CHNG	CAL CHNG	1	Ar, Br, Yr, Zrと各補助レジスタをチェンジする。	108
6	エム M+	M+Ar→Ar	0.1	Mの内容とArの内容を加えてArに入れる。ケタ上げがあると実行フラグ1。	45	E6	コール CAL SIFT	CAL SIFT	0.1	Arの内容を1ビット右へ移動させる。偶数のとき1。	103
7	エム M-	M-Ar→Ar	0.1	Mの内容からArの内容を引いてArに入れる。引けないとき実行フラグ1。	46	E7	コール CAL ENDS	CAL ENDS	1	エンド音を出す。	105
8	ティー TIA	○→Ar	1	○をArに入れる(○は0～Fの数)	23	E8	コール CAL ERRS	CAL ERRS	1	エラー音を出す。	101
9	イー AIA	Ar+○→Ar	0.1	Arに○を加えた結果をArに入れる。ケタ上げがあると実行フラグ1。	29	E9	コール CAL SHTS	CAL SHTS	1	ビツという音を出す。	71
A	ティー TIY	○→Yr	1	○をYrに入れる。	44	EA	コール CAL LONS	CAL LONS	1	ビーという音を出す。	102
B	イー AIY	Yr+○→Yr	0.1	Yrに○を加えた結果をYrに入れる。ケタ上げがあると実行フラグ1。	44	EB	コール CAL SUND	CAL SUND	1	ドレミア音を出す。	100
C	シー CIA	Ar+○?	0.1	Arと○をくらべて、実行フラグを条件別に出す。Ar+○のとき実行フラグ1。	76	EC	コール CAL TIMR	CAL TIMR	1	命令の実行をおくらせる。	47
D	シー CIY	Yr+○?	0.1	Yrと○をくらべて、実行フラグを条件別に出す。Yr+○のとき実行フラグ1。	68	ED	コール CAL DSPR	CAL DSPR	1	Mの内容を2進LEDに点灯する。	74
F	ジャンプ JUMP		1	実行フラグの条件別に、指定したアドレスにジャンプする。	25 27	EE	コール CAL DEM-	M-Ar→M 10進補正	1	MからArを引き10進数に直してMに入れる。	107
						EF	コール CAL DEM+	M+Ar→M 10進補正	1	MにArを加え10進数に直してMに入れる。	104

アドレス (補助レジスタ) アドレス
 Ar……Aレジスタ (6F) Ar……Aレジスタ (69)
 Br……Bレジスタ (6C) Br……Bレジスタ (67)
 Yr……Yレジスタ (6E) Yr……Yレジスタ (68)
 Zr……Zレジスタ (6D) Zr……Zレジスタ (66)
 M……メモリ (50～5F) ○……0～Fまでの数が入る

FX-マイコン実験中に、書き込みや動作が不安定になったら、パイロットランプが点灯していても電池をとりかえてください。

実行フラグの状態とJUMP命令の分岐例

