# Toggle in Programs

## Group 1 Microinstructions

```
200/7200  CLA   0 0000
201/7100  CLL   0 0000
202/7040  CMA   0 7777
203/7020  CML   1 7777
204/7020  CML   0 7777
205/7010  RAR   1 3777
206/7004  RAL   0 7777
207/7012  RTR   1 5777
210/7006  RTL   0 7777
211/7001  IAC   1 0000
212/7001  IAC   1 0001
213/7002  BSW   1 0100
214/7402  HLT   1 0100
```

Final HLT the AC=0100
LINK=1

## Operate Instructions

```
0200/7240  CLA CMA
201/7001   IAC by 0202
202/7640   SZA, CLA
203/7402   ERROR HLT - should not halt
204/7120   Set lnk to 1
205/7010   RAR  AC=4000
206/7510   SKip if AC bit 0=0
207/7410   SKP
210/7402   HLT - should not halt
211/7001   IAC +1
212/7002   BSW  AC=0140
213/1202   Add 7640 to 0140
214/7420   SKp if lnk=1
215/7402   HLT on error
216/7402   Good HLT  AC=0000
```

This program should halt at
loc. 00216 (addr. should read 0217)
AC=0000

## Group 2 Microinstructions

```
200/7300  CLA CLL
201/7440  SZA
202/7402  HLT
203/7430  SZL
204/7402  HLT
205/7020  CML 10000
206/7420  SNL
207/7402  HLT
210/7001  IAC 10001
211/7450  SNA
212/7402  HLT
213/7510  SPA
214/7402  HLT
215/7410  SKP
216/7402  HLT
217/7012  RTR 06000
220/7500  SMA
221/7402  HLT
222/7404  OSR 06001
223/7402  HLT
```

LA 200  SR=0001

## ISZ Instructions

```
200/7500
201/3300
202/7001
203/2300
204/5202
205/7440
206/7402
207/7402
```

This program halts at
loc 00307 (addr read 00210)
AC=0000

## JMS Instruction =ω

```
200/ 7300  Clear AC Clear link
201/ 3300  Zero pass counter
202/ 3204  Zero entry
203/ 4204  JMS
204/ 0000  Return addr. written here
205/ 1204  Get return addr.
206/ 7041  Complement and index AC
207/ 1215  Add to known good addr.
210/ 7440  SKP on AC=0
211/ 7402  Error halt
212/ 2300  Inc pass counter
213/ 5202  do again
214/ 7402  Good HLT
215/ 0204  constant
```

This program halts at 00214 (addr read 00215) AC=0000

## JMP Instruction

```
200/ 5210  JMP 210
201/ 7402  ERROR HLT
202/ 5206  JMP 206
203/ 7402  ERROR HLT
204/ 5212  JMP 212
205/ 7402  ERROR HLT
206/ 5204  JMP 204
207/ 7402  ERROR HLT
210/ 5204  JMP 202
211/ 7402  ERROR HLT
212/ 2300  LOOP TO DO 4096
213/ 5200  START PROG. AGA
214/ 7402  GOOD HLT AFTE
           4096 TIMES
```

This program tests jump, it halts at 00215

* Run this test twice

## Increment AC

```
200/ 7300        200/ 7001 =ω
201/ 7001        201/ 2300
202/ 2207  OR:   202/ 5001
203/ 5202        203/ 5200
204/ 2210
205/ 5204
206/ 5201
```

Visibly see AC increment

## WRITE ZERO'S - CLEAR MEM

```
0004/1007
0005/3410
0006/5004
0007/0000
0010/0011
```

7300
3410
5200

CLEARS 1 FIELD @ A TIME

* Change loc 0007 to any desired loc of contents

## Checker board =ω

```
7777/0000        0020/7300
0000/7300        0021/7020
0001/1007        0022/7420
0002/7040        0023/5025
0003/3007        0024/5027
0004/1007        0025/1032
0005/3410  OR    0026/7410
0006/5000        0027/1033
0007/0000        0030/3410
0010/0011        0031/5021
                 0032/5252
                 0033/2525
                 0010/0035
```

The MD reg alternates between 0000 and 7777

The MD reg. alternates between 5252 and 2525

## Print Character in Switch Reg (bit 04-11)

```
0000/ 7604
      6046
      6041
      5002
      5000
```

## Deposit SR into Corresponding Address

```
0000/ 7604
      3005
      1005
      3405
      5000
```

* Deposit contents of switch register into corresponding address.

## 4K Core Transfer (8K or more)

```
7600/ 6201   Change data field to 0 (specifies source field)
      1670   TAD I 7670
      6211   Change data field to 1 (specifies destination field)
      3670   DCA I 7670
      2270   Inc loc 7670
      5300   JMP -5
      7402   Halt
      0000
```

## Console Print Test =w

0000/ 7001
6046
6041
5002
5000

## Echo Test for : =w

| 1 terminal | 1-4 terminals (KL8A-M8319) |
|---|---|
| 0000/6032 | a00/ 7300 |
| 1/6031 | a01/ 1205 |
| 2/ 5001 | a02/ 6412 |
| 3/ 6036 | a03/ 6401 |
| 4/ 6046 | a04/ 5203 |
| 5/ 6041 | a05/ 0210 |
| 6/ 5005 | a06/ 6406 |
| 7/.5001 | a07/ 5203 |
| | a10/ 7000 |
| | a11/ 7000 |
| | a12/ 7000 |
| | a13/ 5206 |
| | a14/ 7000 |
| | a15/ 7000 |
| | a16/ 7000 |
| | a17/ 6405 |
| | a20/ 6404 |
| | a21/ 5203 |

## LP05 Printer

| ECHO | PRINT |
|---|---|
| 200/6031 | 200/7001 |
| a01/5200 | a01/6666 |
| a02/6036 | a02/6661 |
| a03/6666 | a03/5202 |
| a04/5200 | a04/5200 |

## Paper Tape - PC04

| | |
|---|---|
| a00/ 7001 | a00/ 7300 |
| 6026 | a01/ 6016 |
| a02/ 6021 | a02/ 6011 |
| a03/ 5202 | a03/ 5202 |
| a04/ 5200 | a04/ 5200 |
| Punches alternating 1's and 0's | Reads the tape |

## Dectape - TC01/TC08 Bootstrap

| | |
|---|---|
| 7613/ 6774 | a00/ 7606 |
| 7614/ 1aa2 | 6766 |
| 7615/ 6766 | 6771 |
| 7616/ 6771 | 5202 |
| 7617/ 5216 | 5200 |
| 7620/ 1223 | 7754/ 7577 |
| 7621/ 5215 | 7577 |
| 7622/ 0600 | LA a00 SR=0600 |
| 3/ 0220 | CONT SR=0220 |
| 7754/ 7577 | LA 7600 - REBOOT |
| 7577 | |

## TD8E SR Control Routine

0000/ 7300 CLA CLL
0001/ 7604 LAS
0002/ 6774 SDLC (Load TD Comm.
0003/ 5201
SR 0 = unit
SR 1 = fwd/rev.
SR 2 = stop/go
SR 3 = read/write.