

## BASIC INSTRUCTIONS (4 digits)

General Form 

a	b	c	d
---	---	---	---

a b is the 2-digit function code

### REGISTER INSTRUCTIONS

Digit	2 Registers																																	
Number	c and d are register numbers																																	
Digit	CODES																																	
	<table border="1"> <thead> <tr> <th>a</th> <th>b</th> <th></th> </tr> </thead> <tbody> <tr> <td>1</td> <td>0</td> <td><math>(R_c) \leftarrow (R_c) + (R_d)</math></td> </tr> <tr> <td>1</td> <td>1</td> <td><math>(R_c) \leftarrow (R_c) - (R_d)</math></td> </tr> <tr> <td>1</td> <td>2</td> <td><math>(R_c) \leftarrow (R_d) - (R_c)</math></td> </tr> <tr> <td>1</td> <td>3</td> <td><math>(R_c) \leftarrow (R_d)</math></td> </tr> <tr> <td>1</td> <td>4</td> <td><math>(R_c) \leftarrow</math> R H 6 dig <math>(R_d)</math></td> </tr> <tr> <td>1</td> <td>5</td> <td>INVALID CODE</td> </tr> <tr> <td>1</td> <td>6</td> <td>Left shift <math>(R_c)</math> by <math> (R_d) </math></td> </tr> <tr> <td>1</td> <td>7</td> <td>Right shift <math>(R_c)</math> by <math> (R_d) </math></td> </tr> <tr> <td>1</td> <td>8</td> <td>INVALID CODE</td> </tr> <tr> <td>1</td> <td>9</td> <td>Display <math>(R_c)</math> &amp; <math>(R_d)</math></td> </tr> </tbody> </table>	a	b		1	0	$(R_c) \leftarrow (R_c) + (R_d)$	1	1	$(R_c) \leftarrow (R_c) - (R_d)$	1	2	$(R_c) \leftarrow (R_d) - (R_c)$	1	3	$(R_c) \leftarrow (R_d)$	1	4	$(R_c) \leftarrow$ R H 6 dig $(R_d)$	1	5	INVALID CODE	1	6	Left shift $(R_c)$ by $ (R_d) $	1	7	Right shift $(R_c)$ by $ (R_d) $	1	8	INVALID CODE	1	9	Display $(R_c)$ & $(R_d)$
a	b																																	
1	0	$(R_c) \leftarrow (R_c) + (R_d)$																																
1	1	$(R_c) \leftarrow (R_c) - (R_d)$																																
1	2	$(R_c) \leftarrow (R_d) - (R_c)$																																
1	3	$(R_c) \leftarrow (R_d)$																																
1	4	$(R_c) \leftarrow$ R H 6 dig $(R_d)$																																
1	5	INVALID CODE																																
1	6	Left shift $(R_c)$ by $ (R_d) $																																
1	7	Right shift $(R_c)$ by $ (R_d) $																																
1	8	INVALID CODE																																
1	9	Display $(R_c)$ & $(R_d)$																																

#### Conditions for code 05

When condition is fulfilled, then the CONTROL LATCH (CL) is set to 1, otherwise 0

Value of d	Condition
0	$(R_c) = 0$
1	$(R_c) > 0$
2	$(R_c) < 0$
3	L H dig $(R_c) = 0$
4	R H dig $(R_c) = 0$

### SINGLE-ADDRESS INSTRUCTIONS

a determines the type of addressing

b determines the function

cd gives the 2-digit base address

CODES (For direct addressing)

a	b	
2	0	$(R_0, R_1) \leftarrow (cd)$
2	1	$(cd) \leftarrow (R_0, R_1)$ as data
2	2	$(cd) \leftarrow (R_0, R_1)$ as instruction
2	3	INVALID CODE
2	4	Branch & link to cd unconditionally
2	5	Branch & link to cd if $(CL) = 1$
2	6	Branch & link to cd if $(CL) = 0$
2	7	$(cd) \leftarrow (R_{LINK})$
2	8	Input to cd
2	9	Display (cd)

Digit a	Type of addressing	True address
2	Direct	cd
3	Modified by	$(R_3)$
4		$(R_4)$
5		$(R_5)$
6	Indirect	(cd)

## THREE-ADDRESS INSTRUCTIONS (8 digits)

General form :

a	b	c	d	e	f	g	h
---	---	---	---	---	---	---	---

a b is the 2-digit function code

cd, ef, and gh are the three 2-digit addresses

CODES (For direct addressing)

a	b	
7	0	$(cd) \leftarrow (ef) + (gh)$
7	1	$(cd) \leftarrow (ef) - (gh)$
7	2	$(cd) \leftarrow (ef) \times (gh)$
7	3	$(cd) \leftarrow (ef) \div (gh)$
7	4	Branch to cd if $(ef) = (gh)$
7	5	Branch to cd if $(ef) > (gh)$
7	6	Branch to cd if $ (ef)  >  (gh) $
7	7	If $(gh) \neq 0$ branch to cd & store link address in ef.
7	8	Input to cd; display (ef) & (gh)
7	9	Display (cd), (ef), (gh)

The type of addressing is determined by the digit a :

Digit a	Type of addressing	True addresses
7	Direct Modified by $(R_3)$ , $(R_4)$ , $(R_5)$ respectively	cd, ef, gh
8		cd + $(R_3)$ , ef + $(R_4)$ , gh + $(R_5)$
9	Indirect	(cd), (ef), (gh)

## BASIC INSTRUCTIONS (4 digits)

General Form 

a	b	c	d
---	---	---	---

a b is the 2-digit function code

### REGISTER INSTRUCTIONS

Register and digit	2 Registers	
c is a register number d is a decimal digit	c and d are register numbers	
<b>CODES</b>	<b>CODES</b>	
a b	a b	
0 0	$(R_c) \leftarrow (R_c) + d$	1 0 $(R_c) \leftarrow (R_c) + (R_d)$
0 1	$(R_c) \leftarrow (R_c) - d$	1 1 $(R_c) \leftarrow (R_c) - (R_d)$
0 2	$(R_c) \leftarrow d - (R_c)$	1 2 $(R_c) \leftarrow (R_d) - (R_c)$
0 3	$(R_c) \leftarrow d$	1 3 $(R_c) \leftarrow (R_d)$
0 4	INVALID CODE	1 4 $(R_c) \leftarrow R \text{ H } 6 \text{ dig } (R_d)$
0 5	Test $(R_c)$ for d	1 5 INVALID CODE
0 6	Left shift $(R_c)$ by d	1 6 Left shift $(R_c)$ by $\lfloor (R_d) \rfloor$
0 7	Right shift $(R_c)$ by d	1 7 Right shift $(R_c)$ by $\lfloor (R_d) \rfloor$
0 8	INVALID CODE	1 8 INVALID CODE
0 9	INVALID CODE	1 9 Display $(R_c)$ & $(R_d)$

#### Conditions for code 05

If the condition is fulfilled, then the CONTROL LATCH (CL) is set at 1, otherwise 0

Value of d	Condition
0	$(R_c) = 0$
1	$(R_c) > 0$
2	$(R_c) < 0$
3	L H dig $(R_c) = 0$
4	R H dig $(R_c) = 0$

### SINGLE-ADDRESS INSTRUCTIONS

a determines the type of addressing

b determines the function

cd gives the 2-digit base address

**CODES** (For direct addressing)

a b	
2 0	$(R_0, R_1) \leftarrow (cd)$
2 1	$(cd) \leftarrow (R_0, R_1)$ as data
2 2	$(cd) \leftarrow (R_0, R_1)$ as instruction
2 3	INVALID CODE
2 4	Branch & link to cd unconditionally
2 5	Branch & link to cd if $(CL) = 1$
2 6	Branch & link to cd if $(CL) = 0$
2 7	$(cd) \leftarrow (R_{LINK})$
2 8	Input to cd
2 9	Display (cd)

Digit a	Type of addressing	True address
2	Direct	cd
3	Modified by $\left\{ \begin{array}{l} (R_3) \\ (R_4) \\ (R_5) \end{array} \right.$	$cd + (R_3)$
4		$cd + (R_4)$
5		$cd + (R_5)$
6	Indirect	(cd)

## THREE-ADDRESS INSTRUCTIONS (8 digits)

General form :

a	b	c	d	e	f	g	h
---	---	---	---	---	---	---	---

a b is the 2-digit function code

cd, ef, and gh are the three 2-digit addresses

**CODES** (For direct addressing)

a b	
7 0	$(cd) \leftarrow (ef) + (gh)$
7 1	$(cd) \leftarrow (ef) - (gh)$
7 2	$(cd) \leftarrow (ef) \times (gh)$
7 3	$(cd) \leftarrow (ef) \div (gh)$
7 4	Branch to cd if $(ef) = (gh)$
7 5	Branch to cd if $(ef) > (gh)$
7 6	Branch to cd if $\lfloor (ef) \rfloor > \lfloor (gh) \rfloor$
7 7	If $(gh) \neq 0$ branch to cd & store link address
7 8	Input to cd; display (ef) & (gh)
7 9	Display (cd), (ef), (gh)

The type of addressing is determined by the digit a :

Digit a	Type of addressing	True address
7	Direct	cd, ef, gh
8	Modified by $(R_3)$ , $(R_4)$ , $(R_5)$ respectively	$cd + (R_3)$ , $ef + (R_4)$ , $gh + (R_5)$
9	Indirect	(cd), (ef), (gh)

**A PROGRAM FROM THE KEYBOARD**

'NORMAL RESET' key  
 keyboard register  
 the address of the first instruction  
 the 'LOAD ADDR' key  
 the first instruction  
 the 'LOAD STORE' key  
 the 'INCR ADDR' key  
 the next instruction  
 the 'LOAD STORE' key  
 more instructions, repeat from 7  
 data stores in the same way  
 (negative sign last)

**A PROGRAM**

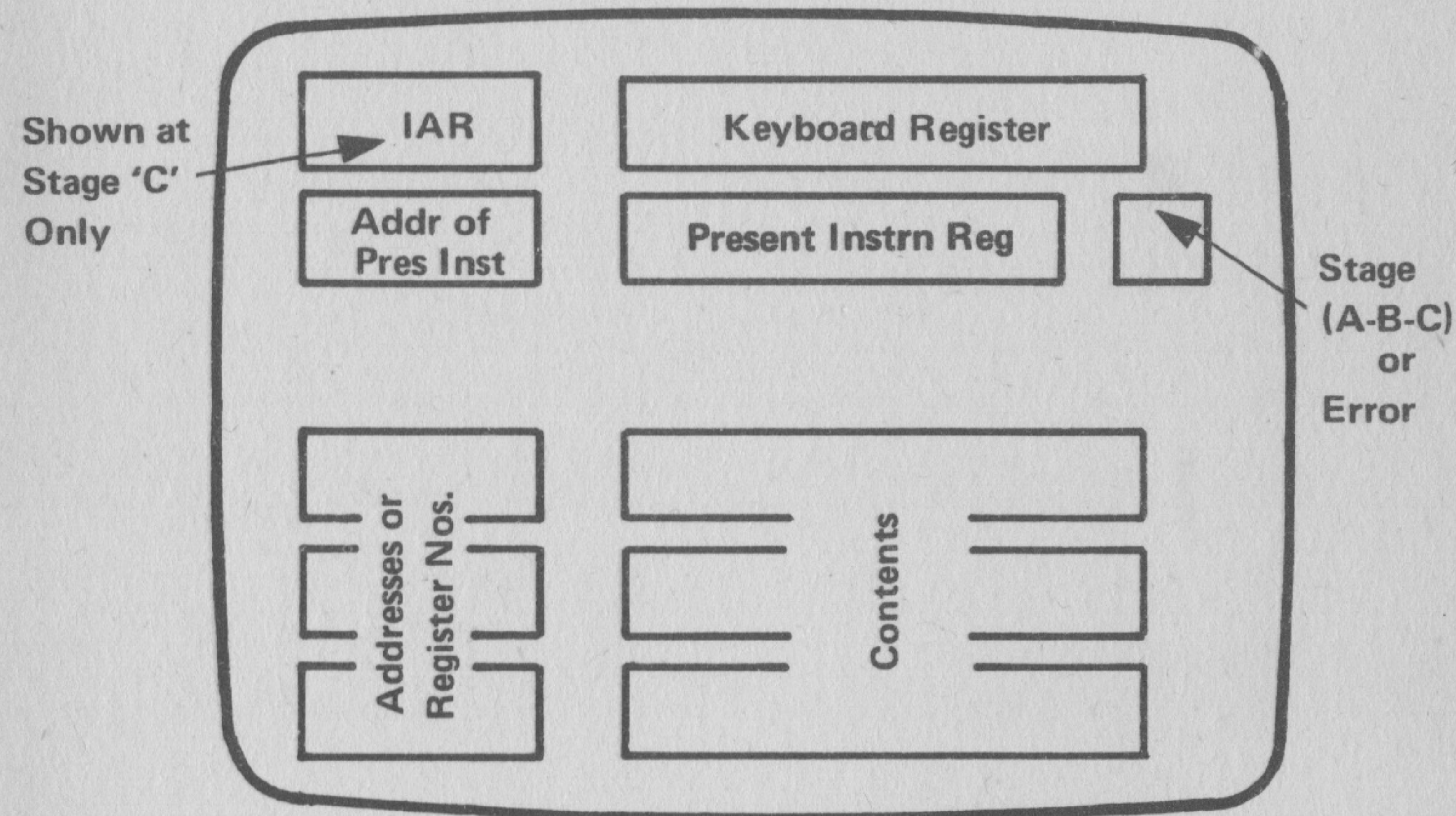
'NORMAL RESET' key  
 keyboard register  
 the address of the first instruction  
 the 'LOAD IAR' key  
 the stage keys 'A'-'B'-'C' or 'RUN'  
 e will stop if  
 'OP' key is touched  
 nput or display instruction is obeyed  
 overflow occurs or there is an invalid instruction

FOR STOPPAGE	ACTION
stage keys touched	Touch 'A'-'B'-'C' or 'RUN'
unction	Key-in required data and touch 'RUN'
struction	To continue a program, touch 'RUN'
or invalid	Correct program and start again

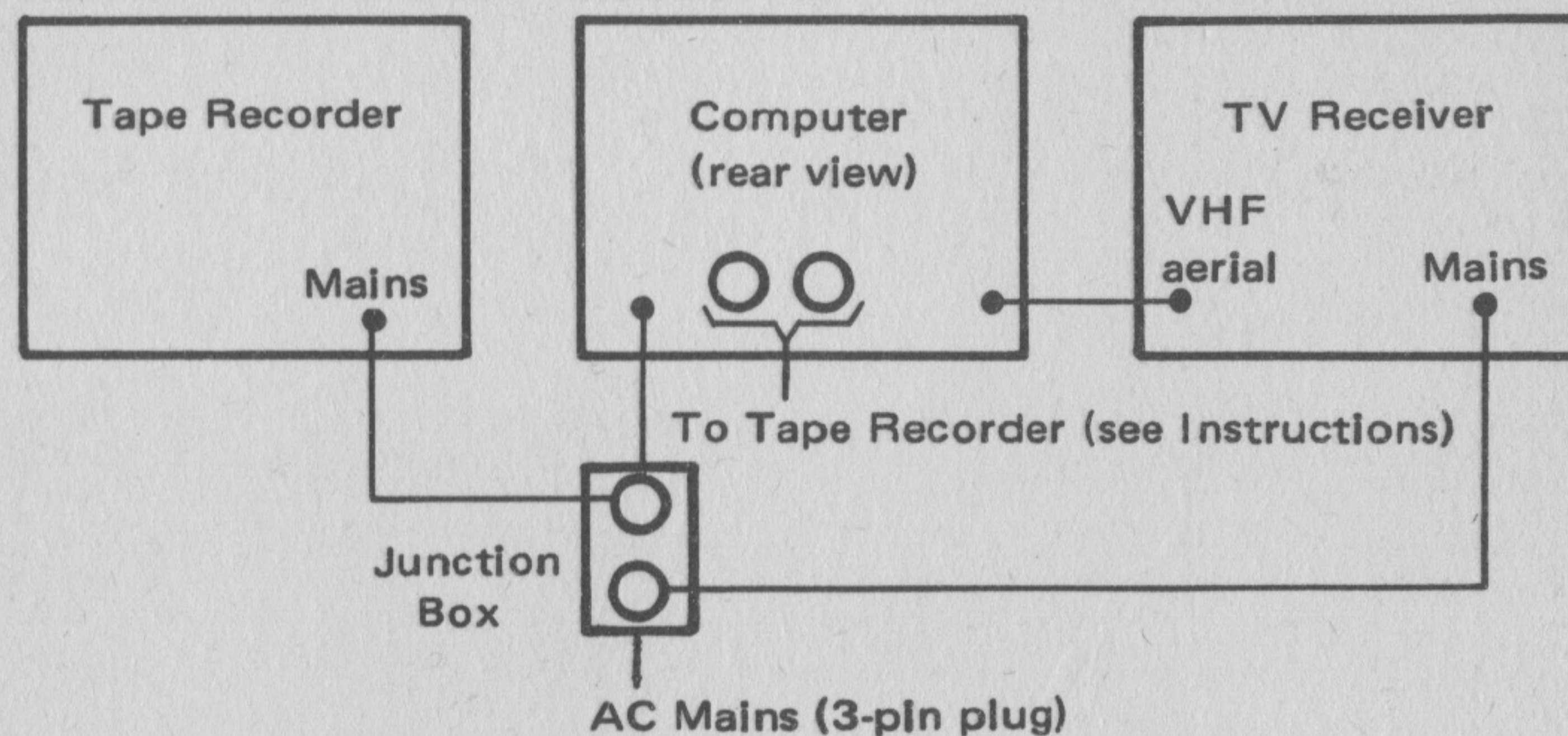
**OBEYING INSTRUCTIONS FROM THE KEYBOARD**

1. Touch 'KI RESET' key
  2. Clear keyboard register
  3. Key-in one 8-digit or two 4-digit instructions
  4. Touch 'A', 'B' or 'C' as required
- If two 4-digit instructions are entered, they will be obeyed in sequence.

**LAYOUT OF TV DISPLAY**



**ELECTRICAL CONNECTIONS**



IBM UNITED KINGDOM LTD

EXPERIMENTAL

SCHOOLS COMPUTER

INSTRUCTION

CARD

INSTRUCTION CODES

BRIEF OPERATING INSTRUCTIONS

### LOADING A PROGRAM FROM THE KEYBOARD

1. Touch 'NORMAL RESET' key
2. Clear keyboard register
3. Key-in the address of the first instruction
4. Touch the 'LOAD ADDR' key
5. Key-in the first instruction
6. Touch the 'LOAD STORE' key
7. Touch the 'INCR ADDR' key
8. Key-in the next instruction
9. Touch the 'LOAD STORE' key
10. If any more instructions, repeat from 7
11. Load data stores in the same way (enter negative sign last)

### RUNNING A PROGRAM

1. Touch 'NORMAL RESET' key
2. Clear keyboard register
3. Key-in the address of the first instruction
4. Touch the 'LOAD IAR' key
5. Touch the stage keys 'A'-'B'-'C' or 'RUN'
6. Machine will stop if
  - a) 'STOP' key is touched
  - b) An input or display instruction is obeyed
  - c) An overflow occurs or there is an invalid instruction

### REASON FOR STOPPAGE

### ACTION

'STOP' or stage keys touched  
Input instruction

Touch 'A'-'B'-'C' or 'RUN'  
Key-in required data and  
touch 'RUN'

Output instruction

To continue a program,  
touch 'RUN'

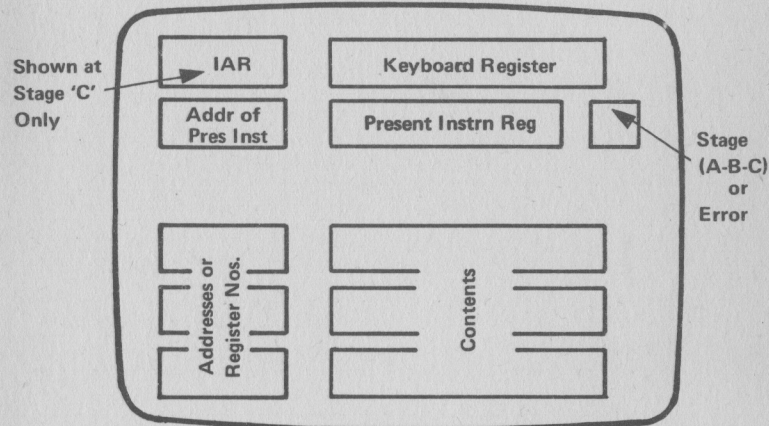
Overflow or invalid  
instruction

Correct program and start  
again

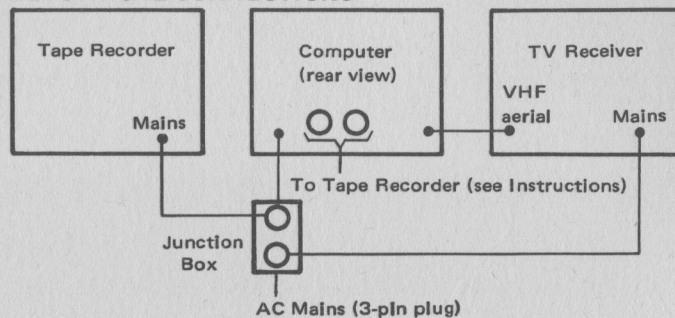
### OBEYING INSTRUCTIONS FROM THE KEYBOARD

1. Touch 'KI RESET' key
  2. Clear keyboard register
  3. Key-in one 8-digit or two 4-digit instructions
  4. Touch 'A', 'B' or 'C' as required
- If two 4-digit instructions are entered, they will be obeyed in sequence.

### LAYOUT OF TV DISPLAY



### ELECTRICAL CONNECTIONS



IBM UNITED KINGDOM LTD

EXPERIMENTAL

SCHOOLS COMPUTER

INSTRUCTION

CARD

INSTRUCTION CODES

BRIEF OPERATING INSTRUCTIONS