Program Control Inst

DOMS | Page No. | Date / /

- Program control instructions specify conditions for altering the content of the program counter.

	Name	Mnemonic	
1.	Branch	BR	one address inst
2.	Jump	JMP	Conditional / unconditional ↓ True - PC - new address False - PC - Next "
3.	Skip	SKP	Skip next inst.
4.	Call	CALL	Used with Subroutined / Procedures
5	Return	RET	
6	Compare (by Subtraction)	CMP	Used to set conditions for branch inst. by updating Status bits.
7.	Test (by Adding)	TST	

These insts. are used to transfer the program Control:

- To Jump from one memory location to any other memory location within a program.
- from one program to another program called as Subroutine.

Register Transfer Language

Register – Very fast computer memory; used to store data/instruction in execution.

- It is a group of flip-flops with one flip-flop capable of storing one bit of information

- Registers
  - Accumulator
  - General Purpose Registers
  - Special purpose Registers
    - MAR
    - MDR
    - PC
    - IR

Micro Operations –

The operations executed on data stored in registers are called micro-operation
eg: Shift, Count, Clear, Load.

- Micro Operations
  - Register Transfer Micro op
  - Arithmetic Micro op
  - Logic Micro-op
  - Shift Micro-op

* Register Transfer Language (RTL)

The symbolic notation used to describe the micro-op. transfers among register is called a register transfer language.

- It is not executed by computer.
- Here the term "register transfer" implies the availability of hardware logic circuits that can perfor a stated microoperation & transfer the result of the operation to the same of another register.
- A register transfer language is a syster for expressing in symbolic form the micro-op sequences among the registers of a digital module.

eg: ADD X - The data value stored at address X is added to AC

MAR ← X
MBR ← M[MAR]
AC ← AC+MBR

eg: ADD #3, DD
D2 ← D2+3 [RTL]

* Register transfer with Control fun
if (k=1) then R2 ← R1
k: R2 ← R1

Basic Symbols for Register Transfer:

	Symbols	Description	Example
1.	Capital letters & numerals	Denotes a register	MAR, R2
2.	Paranthesis	Denotes a part of register	R2(0-7), R2(2)

3. Arrow ← Denotes transfer of info. $R_2 \leftarrow R_1$

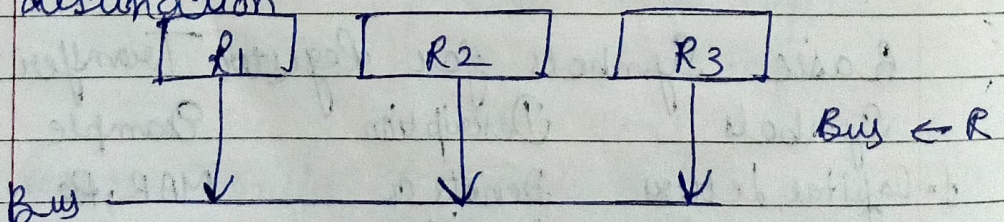
4. Colon : Denotes termination of control fun $P:$

5. Comma , Separate two micro op $A \leftarrow B, C \leftarrow D$

BUS Transfer

In a digital System with many registers. It is impractical to have data & control lines to directly allow each register to be loaded with the contents of every possible other registers.

- So, there is One Centralized set of circuits for data transfer - the bus
- It have control circuits to select which register is the source & which is the destination.
- Here we use bus because, bus is a path (of a group of wires) over which info. is transfered from any of several sources to any of several destination



Transfer from register to Bus

Bus transfer in RTL

$R2 \leftarrow R1$ or $Bus \leftarrow R1$, $R2 \leftarrow Bus$.

DOMS | Page No. | Date

Memory Transfer

The trasfer of info. from a memory word to the outside environment is called read operation.

memory word ⟶ Register

Write op: The transfer of new info. to be stored into the memory is called a write op.

Read op: Let a memory unit that receives the address from a register called the address register (MAR/AR). The data are transferred to another register called the data register.

Read op can be stated as

$DR \leftarrow M[AR]$

Write op: The write op. transfer the content of data register to a memory word M selected by the address

Write op can be stated

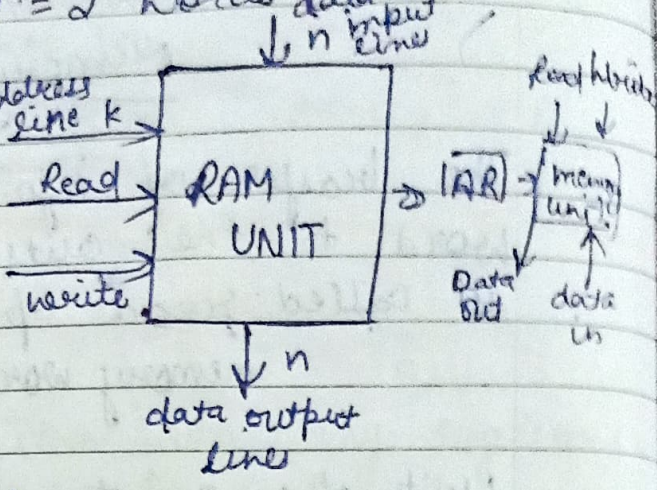
$M[AR] \leftarrow R_1$

It transfer info from R_1 in the memory word M selected by the address in AR.

Let each register (word) consists of n bits of data

& RAM Contains $r = 2^k$ words

- n data input lines
- n data output line
- K address lines
- A Read Control line
- A write Control line.



Memory is usually accessed in Computer by putting desired address in a special register, memory address register (MAR/AR).

When memory is accessed, the contents of the MAR get sent to the memory units address lines

DOMS | Page No.
Date / /

Instruction Code

↓

group of bits ⇒ to the computer to perform some specific tasks.

2 Parts — opcode
— Address

An inst code is a group of bits that instruct the Computer to perform a specific operation.

Most basic Part of inst. code is its operation code:

The operation code of an inst. is a group of bits that define such operation as add, sub, mul, shift & complement.

| opcode | Address |

opcode → Operation (Add, Mul, Sub)

Address → tells the address of the Operand on which the operation is going to perform

Operand → A, B, C, R, etc.

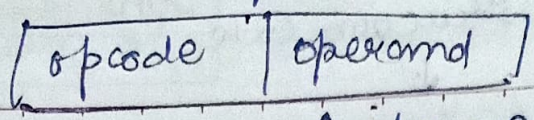
Word → 1024 bits (2^{10})

format → 16 bits → 12 bits = Address
→ 4 bits = Opcode

→ Immediate Operand
→ direct addressing
→ Indirect "

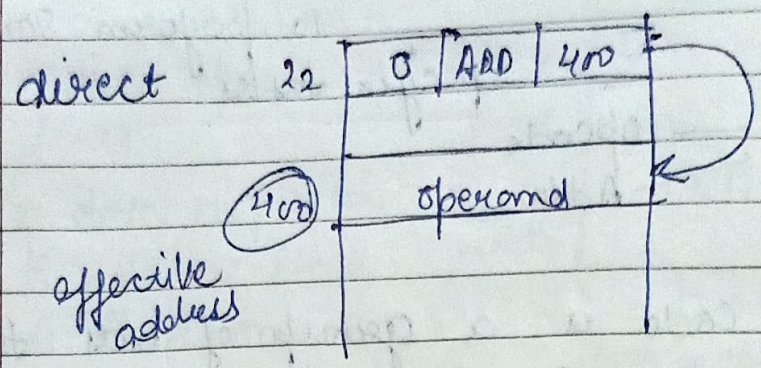
} methods to fetch operand

Immediate operand

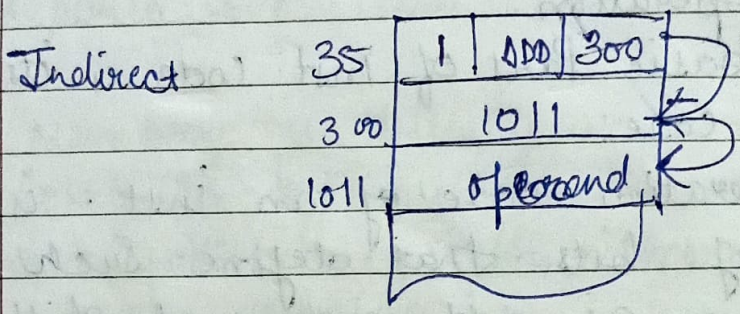


(address acts as a operand)

direct



Indirect



Inst Cycle

Basic fxn of a computer is execution of program.

A program is a set of inst.

To process an inst.

Program Stores in memory.

→ executed by a cycle of each inst.

Inst Cycle is subdivided into a sequence of sub cycles or phases

Fetch → Decode → operand Fetch → Execute

- Fetch → Fetch an inst from memory
- Decode the Inst (find meaning) of
- Read the effective address from memory if the inst has indirect address
- Execute inst.

The control goes back to Step 1. to This process continues indifinitely unless a Halt inst is encountered.

To perform these the processor has to perform set of micro op.

Flowchart

Start SC ← 0

Step 0 : Initialize.

- PC stores the add. of 1 inst
- Sequence Counter SC is initialize to 0 providing a decoded timing Signal To