Microprocessor and Microcontroller (4341101) - Summer 2024 Solution

Milav Dabgar

June 15, 2024

Question 1 [a marks]

3 Describe any one Port Configuration of 8051 Microcontroller.

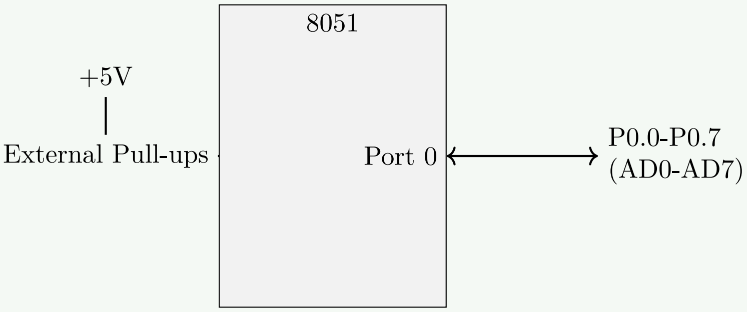
Solution

Answer:

Table 1. Port 0 Configuration

Configuration	Description
Port 0	Dual-purpose port - 8-bit open drain bidirectional I/O port and multiplexed low address/data bus. External pull-up resistors required for I/O functions.

Diagram:



Mnemonic

"PORT 0-PLAD" (Port 0 needs Pull-ups, works as Latch/Address/Data)

Question 1 [b marks]

4 Illustrate Microprocessor Architecture.

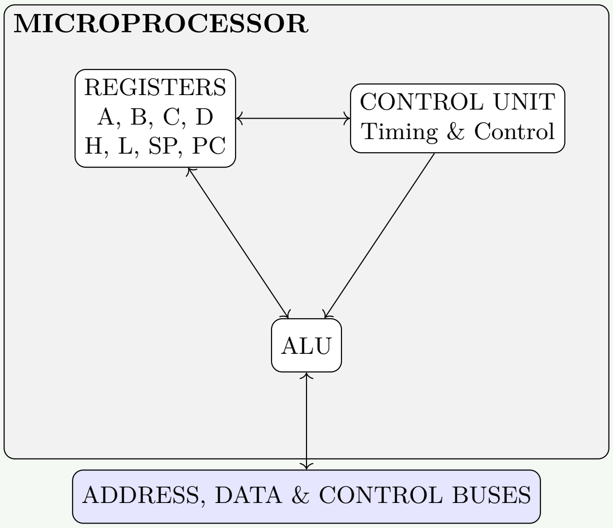
Solution

Answer:

Table 2. Microprocessor Components

Component	Function
ALU	Performs arithmetic and logical operations
Registers	Temporary storage for data and addresses
Control Unit	Directs operation of processor and data flow
Buses	Pathways for data transfer (address, data, control)

Diagram:



Mnemonic

"RABC" - "Registers, ALU, Buses, Control"

Question 1 [c marks]

7 Compare Von Neumann & Harvard architecture.

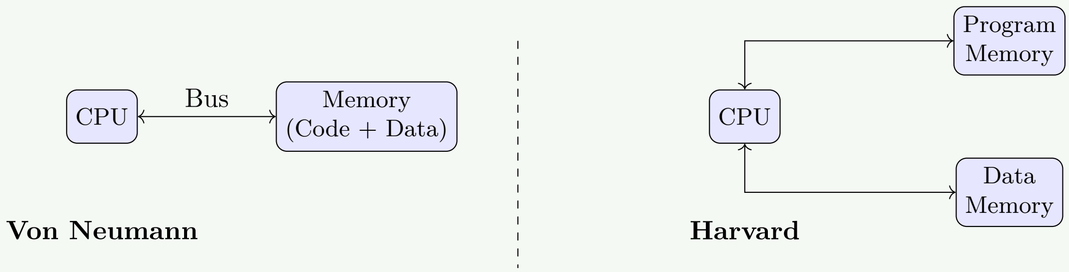
Solution

Answer:

Table 3. Von Neumann vs Harvard Architecture

Feature	Von Neumann Architecture	Harvard Architecture
Memory Buses	Single memory bus for instructions and data	Separate buses for program and data memory
Execution	Sequential execution	Parallel fetch and execute possible
Speed	Slower due to bus bottleneck	Faster due to simultaneous access
Memory Access	Single memory space	Separate memory spaces
Complexity	Simpler design	More complex design
Applications	General-purpose computing	DSP, microcontrollers, embedded systems
Examples	Most PCs, 8085, 8086	8051, PIC, ARM Cortex-M

Diagram:



Mnemonic

"Harvard Has Separate Streets" (Harvard Has Separate memory paths)

OR

Question 1 [c marks]

7 Define RISC, CISC, Opcode, Operand, Instruction Cycle, Machine Cycle, and T State.

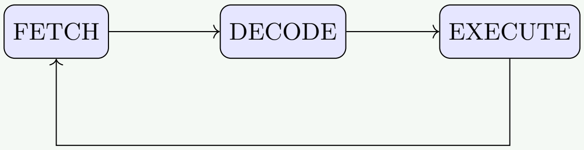
Solution

Answer:

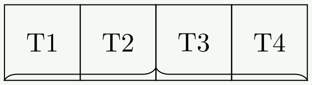
Table 4. Definitions

Term	**Definition**
RISC	Reduced Instruction Set Computer - architecture with simple instructions optimized for speed
CISC	Complex Instruction Set Computer - architecture with complex, powerful instructions
Opcode	Operation Code - part of instruction that specifies operation to be performed
Operand	Data value or address used in operation
Instruction Cycle	Complete process to fetch, decode and execute an instruction
Machine Cycle	Basic operation like memory read/write (subset of instruction cycle)
T-State	Time state - smallest unit of time in processor operation (clock period)

Diagram:



Instruction Cycle



Machine Cycle

Mnemonic

"RICO ITEM" (RISC, CISC, Opcode, Instruction cycle, T-state, Execute, Machine cycle)

Question 2 [a marks]

3 Define Data bus, Address bus and Control bus.

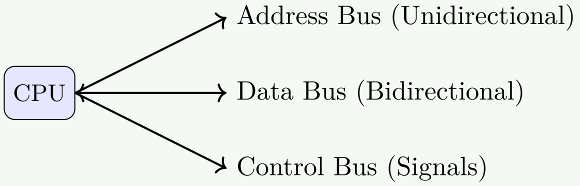
Solution

Answer:

Table 5. Bus Definitions

Bus Type	Definition
Data Bus	Bidirectional pathway that transfers actual data between microprocessor and peripheral devices
Address Bus	Unidirectional pathway that carries memory/IO device locations to be accessed
Control Bus	Group of signal lines that coordinate and synchronize all system operations

Diagram:



Mnemonic

“ADC” - “Address finds location, Data carries information, Control coordinates operations”

Question 2 [b marks]

4 Compare Microprocessor and Microcontroller.

Solution

Answer:

Table 6. Microprocessor vs Microcontroller

Feature	**Microprocessor**	**Microcontroller**
Definition	CPU on a single chip	Complete computer system on a chip
Memory	External RAM/ROM needed	Built-in RAM/ROM
I/O Ports	Limited or none on-chip	Multiple I/O ports on-chip
Peripherals	External peripherals needed	Built-in peripherals (timers, ADC, etc.)
Applications	General computing, PCs	Embedded systems, IoT devices
Cost	Higher for complete system	Lower (all-in-one solution)
Power	Higher	Lower

Mnemonic

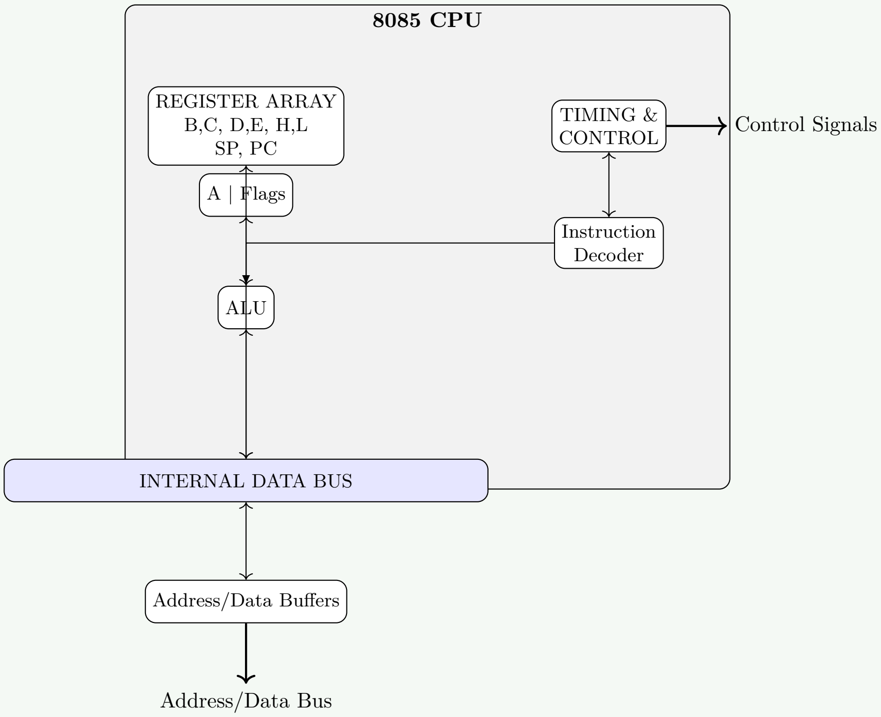
“MEMI-CAP” (Memory external/internal, Cost, Applications, Peripherals)

Question 2 [c marks]

7 Sketch and explain 8085 block diagram.

Solution

Answer:
Diagram:



Main Components:

- **Register Array**: A (Accumulator), Flags, B-L, SP, PC, temp registers
- **ALU**: Performs arithmetic and logical operations
- **Timing & Control**: Generates control signals, handles interrupts
- **Bus Interface**: Connects CPU to external devices
- **Internal Data Bus**: Links internal components

Mnemonic

"RATBI" - "Registers, ALU, Timing, Buses, Interface"

OR

Question 2 [a marks]

3 Explain Accumulator, Program Counter and Stack Pointer.

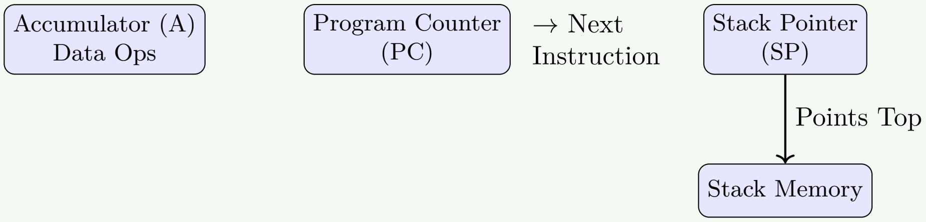
Solution

Answer:

Table 7. Register Functions

Register	Function
Accumulator (A)	8-bit register that stores results of arithmetic and logical operations
Program Counter (PC)	16-bit register that holds address of next instruction to be executed
Stack Pointer (SP)	16-bit register that points to current top of stack in memory

Diagram:



Mnemonic

“APS” - “Accumulator Processes, PC Predicts, SP Stacks”

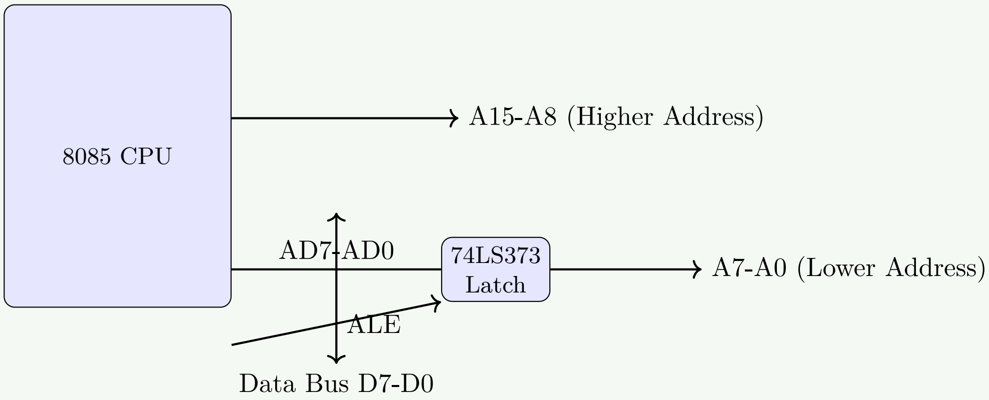
OR

Question 2 [b marks]

4 Sketch and explain Demultiplexing of Address bus and data bus.

Solution

Answer:
Diagram:



Process:

1. **Multiplexing**: AD0-AD7 pins share address and data signals to reduce pin count.
2. **Demultiplexing Steps**:
   - CPU places address on AD0-AD7 pins.
   - ALE (Address Latch Enable) signal goes HIGH.
   - External latch (74LS373) captures lower address bits.
   - ALE goes LOW, latching the address.
   - AD0-AD7 pins now carry data.

Mnemonic

“ALAD” - “ALE Active, Latch Address, After Data”

OR

Question 2 [c marks]

7 List any seven features of 8085.

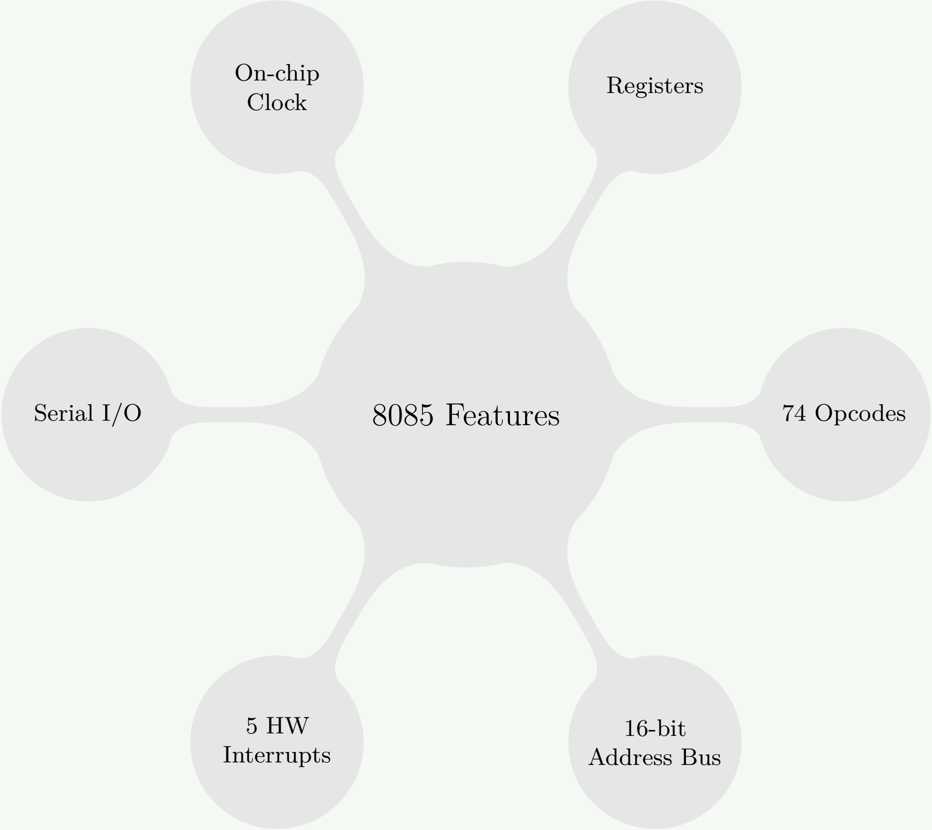
Solution

Answer:

Table 8. 8085 Features

Feature	Description
8-bit Data Bus	Transfers 8 bits of data in parallel
16-bit Address Bus	Can address up to 64KB of memory (2^{16})
Hardware Interrupts	5 hardware interrupts (TRAP, RST 7.5, 6.5, 5.5, INTR)
Serial I/O	SID and SOD pins for serial communication
Clock Generation	On-chip clock generator with crystal
Instruction Set	74 operation codes generating 246 instructions
Register Set	Six 8-bit registers (B,C,D,E,H,L), accumulator, flags, SP, PC

Diagram:



Mnemonic

"CHAIRS" - "Clock, Hardware interrupts, Address bus, Instruction set, Registers, Serial I/O"

Question 3 [a marks]

3 Illustrate any one Timer Mode of 8051.

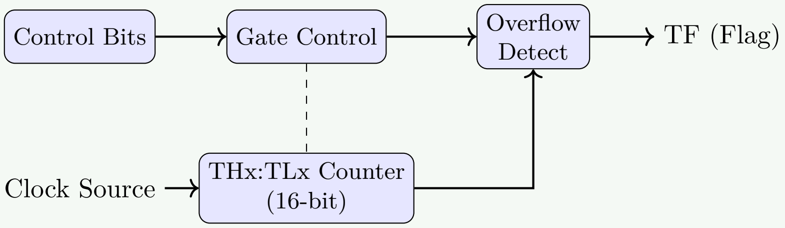
Solution

Answer:
Mode 1: 16-bit Timer/Counter

Table 9. Timer Mode 1

Feature	**Description**
Timer Structure	16-bit timer using THx and TLx registers
Operation	Counts from 0000H to FFFFH, then sets TF flag
Counter Size	Full 16-bit counter ($2^{16} = 65,536$ counts)
Registers	THx (high byte) and TLx (low byte)

Diagram:



Mnemonic

"MOGC" - "Mode 1 uses Overflow detection, Gate control, Complete 16-bits"

Question 3 [b marks]

4 State function of ALE, PSEN, RESET and TXD pin for 8051.

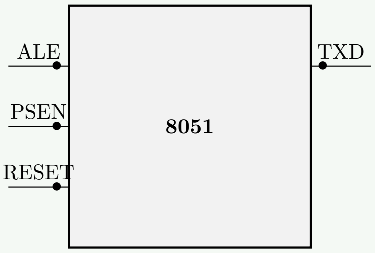
Solution

Answer:

Table 10. Pin Functions

Pin	**Function**
ALE	Address Latch Enable - Provides control signal to latch low byte of address from port 0
PSEN	Program Store Enable - Read strobe for external program memory access
RESET	Reset input - Forces CPU to initial state when held HIGH for 2 machine cycles
TXD	Transmit Data - Serial port output pin for serial data transmission

Diagram:



Mnemonic

"APTR" - "Address latch, Program store, Total reset, tRansmit data"

Question 3 [c marks]

7 Explain functions of each block of 8051 Microcontroller.

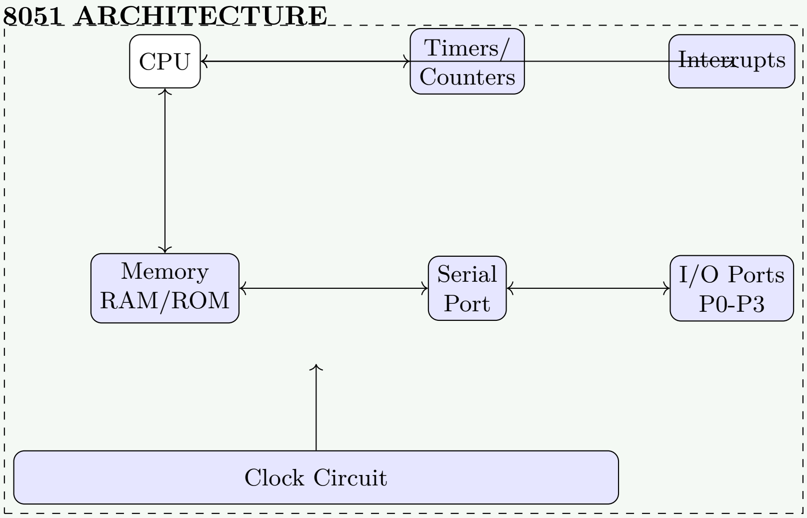
Solution

Answer:

Table 11. 8051 Blocks

Block	**Function**
CPU	8-bit processor that fetches and executes instructions
Memory	4KB internal ROM and 128 bytes of internal RAM
I/O Ports	Four 8-bit bidirectional I/O ports (P0-P3)
Timers/Counters	Two 16-bit timers/counters for timing and counting
Serial Port	Full-duplex UART for serial communication
Interrupts	Five interrupt sources with two priority levels
Clock Circuit	Provides timing for all operations

Diagram:



Mnemonic

"CRIMSON" - "CPU, RAM/ROM, I/O, Memory, Serial port, Oscillator, iNterrupts"

OR

Question 3 [a marks]

3 Illustrate any one Serial Communication Mode of 8051.

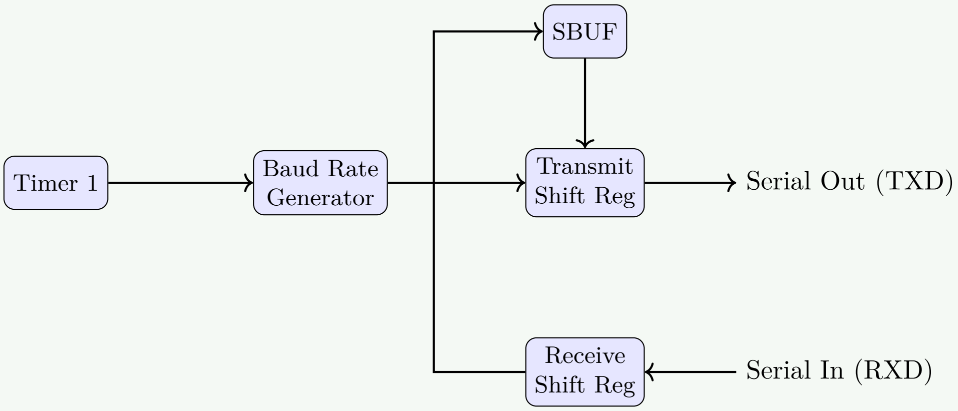
Solution

Answer:
Mode 1: 8-bit UART

Table 12. Serial Mode 1

Feature	**Description**
Format	10 bits (start bit, 8 data bits, stop bit)
Baud Rate	Variable, determined by Timer 1
Data Direction	Full-duplex (simultaneous transmit and receive)
Pins Used	TXD (P3.1) for transmit, RXD (P3.0) for receive

Diagram:



Mnemonic

“FADS” - “Format 10-bit, Auto baud from Timer 1, Duplex mode, Standard UART”

OR

Question 3 [b marks]

4 State function of RXD, INT0, T0 and PROG pin for 8051.

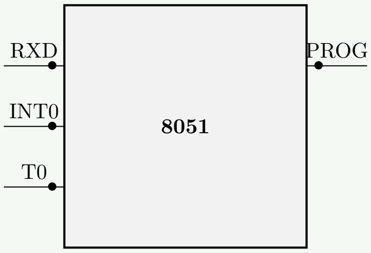
Solution

Answer:

Table 13. Pin Functions

Pin	**Function**
RXD (P3.0)	Receive Data - Serial port input pin for serial data reception
INT0 (P3.2)	External Interrupt 0 - Input that can trigger external interrupt
T0 (P3.4)	Timer 0 - External count input for Timer/Counter 0
PROG (EA)	Program Enable - When LOW, forces CPU to fetch code from external memory

Diagram:



Mnemonic

"RIPE" - "Receive data, Interrupt trigger, Pulse counting, External memory"

OR

Question 3 [c marks]

7 Describe ALU, PC, DPTR, RS0, RS1, Internal RAM and Internal ROM of 8051.

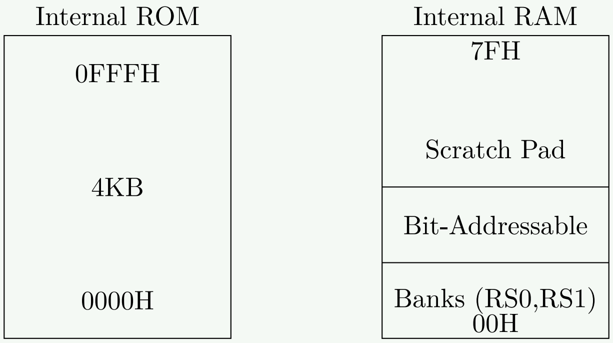
Solution

Answer:

Table 14. Register/Memory Description

Component	**Description**
ALU	Arithmetic Logic Unit - Performs math and logical operations
PC	Program Counter - 16-bit register that points to next instruction
DPTR	Data Pointer - 16-bit register (DPH+DPL) for external memory addressing
RS0, RS1	Register Bank Select bits in PSW - Select one of four register banks
Internal RAM	128 bytes on-chip RAM (00H-7FH) for variables and stack
Internal ROM	4KB on-chip ROM (0000H-0FFFH) for program storage

Diagram:



Mnemonic

"APRID" - "ALU Processes, PC Remembers, Register bank select, Internal memory, DPTR points"

Question 4 [a marks]

3 Develop an Assembly language program to divide 08H by 02H.

Mnemonic

"LDDS" - "Load dividend, Divisor in B, Divide, Store results"

Question 4 [b marks]

4 Develop an Assembly language program to add 76H and 32H.

Solution

Answer:

```
      MOV A, #76H    ; Load first number 76H into accumulator
      MOV R0, #32H   ; Load second number 32H into R0
      ADD A, R0      ; Add R0 to A (76H + 32H = A8H)
      MOV R1, A      ; Store result in R1 (A8H)
      JNC DONE       ; Jump if no carry
      MOV R2, #01H   ; If carry occurred, store 1 in R2
DONE: NOP            ; End program
```

Calculation:

$$\begin{array}{rcl} 76\text{H} & = & 0111\ 0110 \\ +\ 32\text{H} & = & 0011\ 0010 \\ \hline \text{A8H} & = & 1010\ 1000 \end{array}$$

Mnemonic

"LASER" - "Load A, Store second number, Execute addition, Result stored"

Question 4 [c marks]

7 What is Addressing mode? Classify it for 8051.

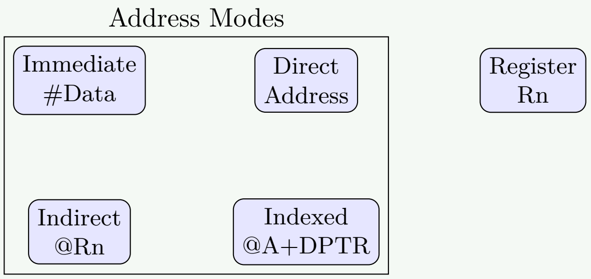
Solution

Answer:
Addressing Mode: Method to specify the location of operand/data for an instruction.

Table 15. Addressing Modes

Mode	Example	Description
Register	`MOV A, R0`	Operand in register
Direct	`MOV A, 30H`	Operand at specific memory location (30H)
Register Indirect	`MOV A, @R0`	Register (R0) contains address of operand
Immediate	`MOV A, #55H`	Operand (#55H) is part of instruction
Indexed	`MOVC A, @A+DPTR`	Base address (DPTR) + offset (A)
Bit	`SETB P1.0`	Individual bit addressable
Implied	`RRC A`	Operand implied by instruction

Diagram:



Mnemonic

"RIDDIB" - "Register, Immediate, Direct, Data indirect, Indexed, Bit"

OR

Question 4 [a marks]

3 Develop an Assembly language program to multiply 08H and 02H.

Mnemonic

"LMSR" - "Load numbers, Multiply, Store Result"

OR

Question 4 [b marks]

4 Develop an Assembly language program to subtract 76H from 32H.

Solution

Answer:

```
      MOV A, #32H     ; Load 32H into accumulator
      MOV R0, #76H    ; Load 76H into R0
      CLR C           ; Clear carry flag (borrow flag)
      SUBB A, R0      ; Subtract R0 from A with borrow (32H - 76H = BCH)
      MOV R1, A       ; Store result in R1 (BCH, which represents -44H)
```

Calculation:

$$\begin{array}{rcl} \text{32H} & = & \text{0011 0010} \\ \text{- 76H} & = & \text{0111 0110} \\ \hline \text{BCH} & = & \text{1011 1100 (Two's complement)} \end{array}$$

Mnemonic

"LESS" - "Load first number, Enable borrow (CLR C), Subtract, Store"

OR

Question 4 [c marks]

7 List types of instruction set. Explain any three with one example.

Solution

Answer:

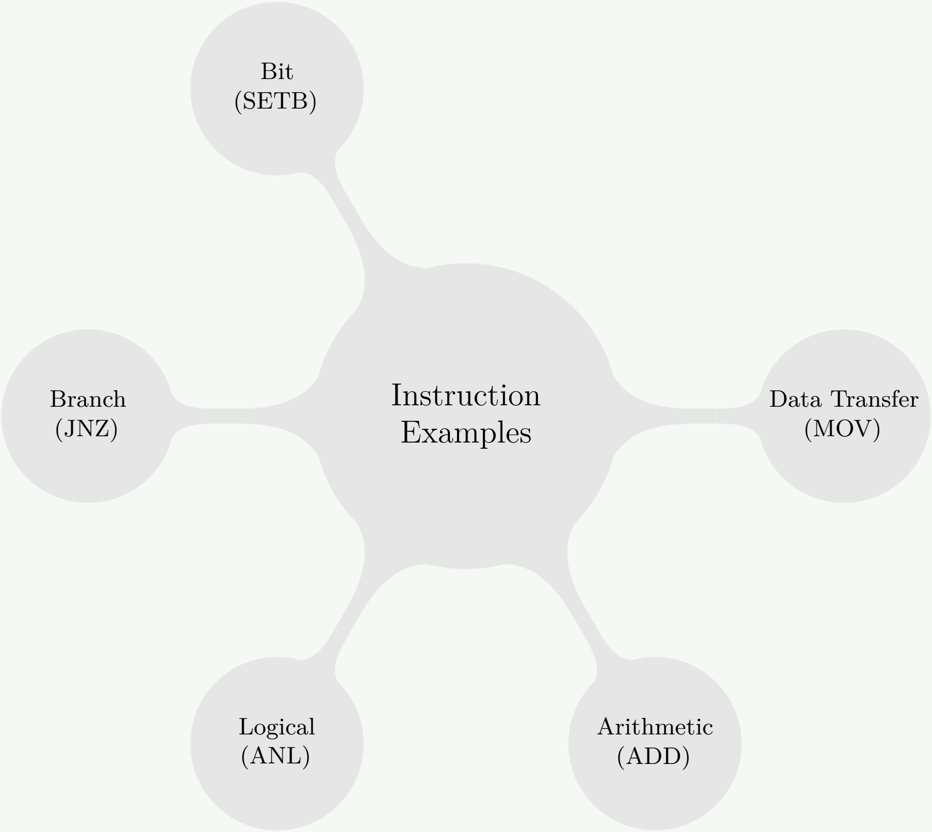
Table 16. Instruction Types

Group	Description	Example
Arithmetic	Mathematical operations	`ADD A, R0`
Logical	Logical operations	`ANL A, #0FH`
Data Transfer	Move data between locations	`MOV A, R7`
Branch	Change program flow	`JNZ LOOP`
Bit Manipulation	Operate on individual bits	`SETB P1.0`
Machine Control	Control processor operation	`NOP`

Explanation:

1. **Data Transfer Instructions**:
   - Move data between registers, memory, or I/O ports.
   - Example: `MOV A, 30H` (Move data from memory 30H to A).
2. **Arithmetic Instructions**:
   - Perform mathematical operations.
   - Example: `ADD A, R0` (Add content of R0 to A).
3. **Logical Instructions**:
   - Perform logical operations (AND, OR, XOR).
   - Example: `ANL A, #0FH` (AND A with 0FH).

Diagram:



Mnemonic

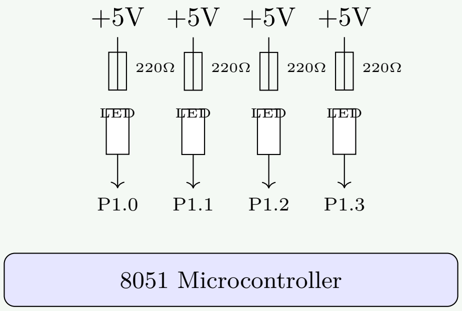
"BALDM" - "Branch, Arithmetic, Logical, Data transfer, Machine control"

Question 5 [a marks]

3 Sketch interfacing of four LEDs with 8051 Microcontroller.

Solution

Answer:
Diagram:



Details:

- LEDs connected to Port 1 (P1.0-P1.3).
- Common Anode configuration (Active Low) shown (Current flows into Port pin when LOW).
- Current limiting resistors (220Ω) protect LEDs.

Mnemonic

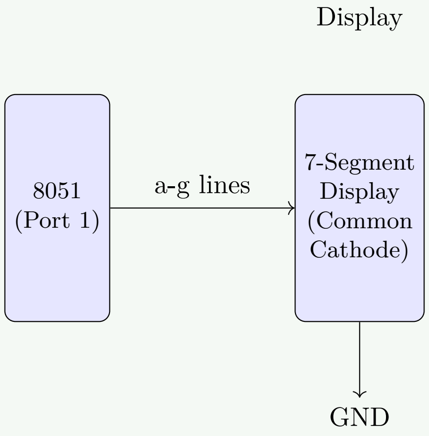
"PALS" - "Port pins, Active-low control, LEDs, Simple circuit"

Question 5 [b marks]

4 Sketch interfacing of 7 segment LED with 8051 Microcontroller.

Solution

Answer:
Diagram:



Code Example:

```
; Display digit 5 (Pattern: 6DH)
MOV A, #6DH      ; Segment pattern for 5 (a,c,d,f,g ON)
MOV P1, A        ; Send to port P1 to drive segments
```

Mnemonic

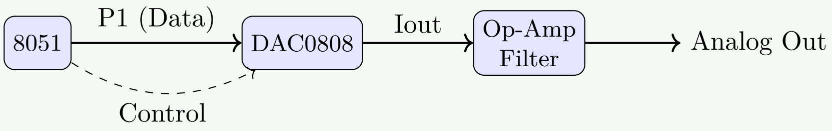
"SPACE-7" - "Seven Pins, Active segments, Common ground, Easy display"

Question 5 [c marks]

7 Explain interfacing of DAC with 8051 Microcontroller and write necessary program.

Solution

Answer:
Diagram:



Program (Sawtooth Wave):

```
START:  MOV R0, #00H     ; Initialize R0 to 0
LOOP:   MOV P1, R0       ; Output value to DAC
        CALL DELAY       ; Wait for some time
        INC R0           ; Increment value (00->FF->00)
        SJMP LOOP        ; Repeat to create sawtooth wave

DELAY:  MOV R7, #50      ; Load delay counter
        DJNZ R7, $       ; Simple delay loop
        RET
```

Mnemonic

"DICAF" - "Digital input, Increment, Convert to analog, Amplify, Filter"

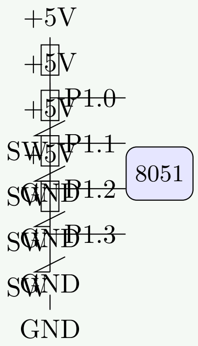
OR

Question 5 [a marks]

3 Sketch interfacing of four Switches with 8051 Microcontroller.

Solution

Answer:
Diagram:



Mnemonic

"PIPS" - "Pull-ups, Input pins, Press for zero, Switches"

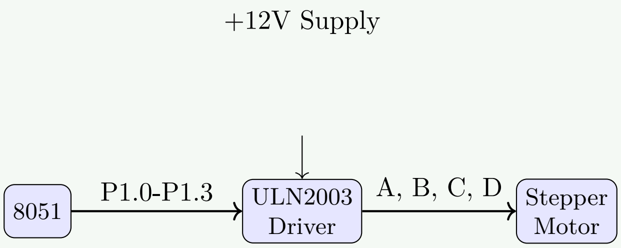
OR

Question 5 [b marks]

4 Sketch interfacing of Stepper motor with 8051 Microcontroller.

Solution

Answer:
Diagram:



Mnemonic

"CUPS" - "Controller outputs sequence, ULN2003 amplifies, Phases energized, Stepping motion"

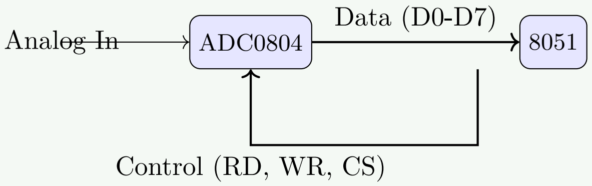
OR

Question 5 [c marks]

7 Explain interfacing of ADC with 8051 Microcontroller and write necessary program.

Solution

Answer:
Diagram:



Program:

```
START:  MOV P1, #0FFH    ; Configure P1 as input
READ:   CLR P3.0         ; CS = 0 (Select ADC)
        CLR P3.2         ; WR = 0 (Start Conversion)
        NOP
        SETB P3.2        ; WR = 1 (Latch Start)

WAIT:   JB P3.3, WAIT    ; Wait for INTR = 0 (EOC)

        CLR P3.1         ; RD = 0 (Enable Data Output)
        MOV A, P1        ; Read Data
        SETB P3.1        ; RD = 1
        SETB P3.0        ; CS = 1 (Deselect)

        MOV R0, A        ; Store Data
        SJMP READ        ; Repeat
```

Mnemonic

“CARSW” - “Convert Analog, Read Digital, Start conversion, Wait for completion”