

Subject Name (Gujarati)

4341101 -- Winter 2024

Semester 1 Study Material

Detailed Solutions and Explanations

પ્રશ્ન 1(અ) [3 ગુણ]

માઇક્રોકંટ્રોલર્સનાં સામાન્ય ફીચર્સની સૂચિ બનાવો.

જવાબ

ફીચર	હેતુ
CPU કોર	સૂચનાઓ પ્રોસેસ કરવા
મેમરી (RAM/ROM)	પ્રોગ્રામ અને ડેટા સ્ટોર કરવા
I/O પોર્ટ્સ	બાહ્ય ડિવાઇસ સાથે ઇન્ટરફેસ
ટાઇમર/કાઉન્ટર	સમય અંતરાલ માપવા
ઇન્ટરપ્ટ	અસિંક્રોનસ ઘટનાઓ સંભાળવા
સીરિયલ કમ્યુનિકેશન	અન્ય ડિવાઇસ સાથે ડેટા ટ્રાન્સફર

યાદ રાખવા માટે: "CPU-TIS: CPU-RAM-I/O-ટાઇમર-ઇન્ટરપ્ટ-સીરિયલ"

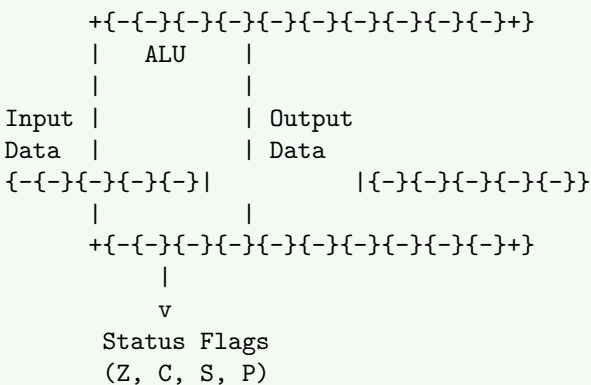
પ્રશ્ન 1(બ) [4 ગુણ]

ALU ના કાર્યો સમજાવો.

જવાબ

કાર્ય	વર્ણન
ગણિત ઓપરેશન્સ	સરવાળો, બાદબાકી, ઇન્ક્રિમેન્ટ, ડિક્રિમેન્ટ
લોજિકલ ઓપરેશન્સ	AND, OR, XOR, NOT, તુલના
ડેટા મૂવમેન્ટ	રજિસ્ટર અને મેમરી વચ્ચે ટ્રાન્સફર
ફ્લેગ સેટિંગ	ઓપરેશન પરિણામ પર આધારિત સ્ટેટસ ફ્લેગ અપડેટ

ડાયાગ્રામ:



યાદ રાખવા માટે: "ALFS: અરિથમેટિક લોજિક ફ્લેગ સ્ટેટસ"

પ્રશ્ન 1(ક) [7 ગુણ]

વ્યાખ્યાયિત કરો: મેમરી, ઓપરેન્ડ, ઇન્સ્ટ્રક્શન સાયકલ, ઓપકોડ, CU, મશીન સાયકલ, CISC

- મેમરી: યુનિક એડ્રેસ સાથે સ્ટોરેજ સેલનો વ્યવસ્થિત એરે
- ઓપરેન્ડ: સૂચનાઓ જેના પર ક્રિયા કરે છે તે ડેટા એલિમેન્ટ
- ઈન્સ્ટ્રક્શન સાયકલ: દરેક સૂચના માટે ફેચ-ડિકોડ-એક્ઝિક્યુટ સિક્વન્સ
- ઓપકોડ: પ્રોસેસરને કયું ઓપરેશન કરવાનું છે તે જણાવતો બાઇનરી કોડ

Instruction Cycle:

યાદ રાખવા માટે: "MO-ICO-MC: મેમરી-ઓપરેન્ડ-ઈન્સ્ટ્રક્શન-કંટ્રોલ-ઓપરેશન-મશીન-કોમ્પલેક્સ"

i) વ્યાખ્યાયિત કરો: માઇક્રોપ્રોસેસર ii) વોન-ન્યુમેન અને હાર્વર્ડ આર્કિટેક્ચરની તુલના કરો.

i) માઇક્રોપ્રોસેસર વ્યાખ્યા:

ii) વોન-ન્યુમેન VS હાર્વર્ડ આર્કિટેક્ચર:

2

સાચાગ્રામ:

```

Von{-Neumann:}
+{-{-{-{-{-{-{-{-{-{-{-}+          +{-{-{-{-{-{-{-{-{-{-{-}+}
| CPU      |{=====| Memory|}
+{-{-{-{-{-{-{-{-{-{-{-}+          +{-{-{-{-{-{-{-{-{-{-{-}+}

```

Harvard:

```

+{-{-{-{-{-{-{-{-{-{-{-}+          +{-{-{-{-{-{-{-{-{-{-{-}+}
| CPU      |====={| Program  |}
|          |          | Memory  |
|          |          +{-{-{-{-{-{-{-{-{-{-{-}+}
|          |          +{-{-{-{-{-{-{-{-{-{-{-}+}
|          |{=====| Data    |}
+{-{-{-{-{-{-{-{-{-{-{-}+          | Memory  |}
|          |          +{-{-{-{-{-{-{-{-{-{-{-}+}

```

યાદ રાખવા માટે: ``હાર્વર્ડ પાસે અલગ જગ્યાઓ છે"

પ્રશ્ન 2(અ) [3 ગુણ]

8085 માઇક્રોપ્રોસેસરના વિવિધ રજીસ્ટરો સમજાવો.

જાણી

રજિસ્ટર	સાઇઝ	કાર્ય
એક્ઝ્યુમુલેટર (A)	8-બિટ	ગાણિતિક અને લોજિક માટે મુખ્ય રજિસ્ટર
જનરલ પર્પઝ (B,C,D,E,H,L)	8-બિટ	અસ્થાયી ડેટા સ્ટોરેજ
પ્રોગ્રામ કાઉન્ટર (PC)	16-બિટ	આગલી સૂચનાનું એડ્રેસ
સ્ટેક પોઇન્ટર (SP)	16-બિટ	સ્ટેકના ટોપને પોઇન્ટ કરે
ફ્લેગ રજિસ્ટર	8-બિટ	સ્ટેટસ ફ્લેગ્સ (Z,S,P,CY,AC)

યાદ રાખવા માટે: "AGSF: એક્ઝ્યુમુલેટર-જનરલ-સ્ટેક-ફ્લેગ્સ"

યાદ રાખવા માટે: "AGSF: એક્ઝ્યુમુલેટર-જનરલ-સ્ટેક-ફ્રેમવર્ક"

ପ୍ରଶ୍ନ 2(ଏ) [4 ମାର୍କ]

ઈન્સ્ટ્રક્શનનું ફેચિંગ, ડીકોડિંગ અને એક્ઝેક્યુશન સમજાવો.

정답

ફેઝ	પ્રવૃત્તિ	સંબંધિત હાર્ડવેર
ફેઝિંગ	PC માંના એડ્રેસથી મેમરીમાંથી સૂચના મેળવવી	PC, એડ્રેસ બસ, મેમરી
ડીકોડિંગ	ઓપરેશન પ્રકાર અને ઓપરેન્ડ ઓળખવા	ઈન્સ્ટ્રક્શન રજિસ્ટર, કંટ્રોલ યુનિટ
એક્ઝેક્યુશન	નિર્દિષ્ટ ઓપરેશન કરવું	ALU, રજિસ્ટર્સ, ડેટા બસ

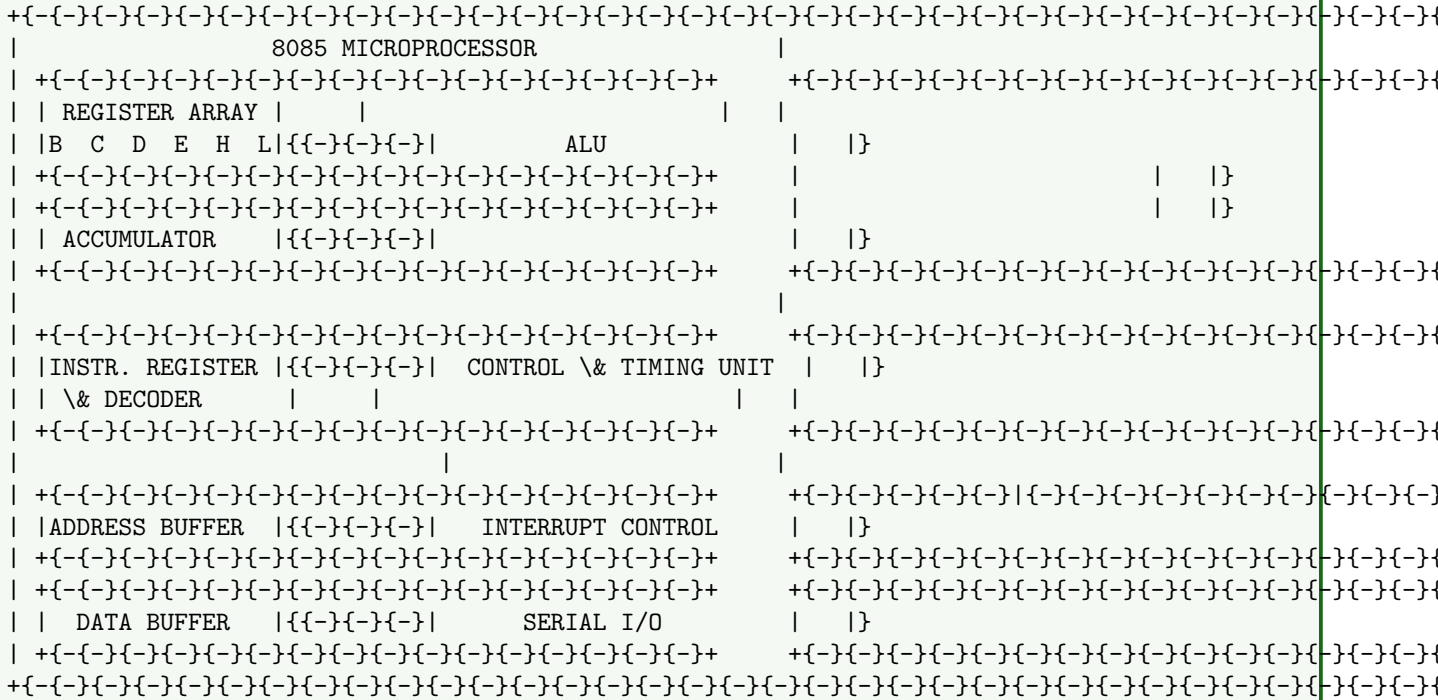
[illegible]

- ફેચિંગ: PC મેમરીને એડ્રેસ મોકલે, સૂચના IR માં લોડ થાય
 - ડીકોડિંગ: કંટ્રોલ યુનિટ સૂચના ઓપકોડ અને એડ્રેસિંગ મોડ સમજે
 - એક્ઝેક્યુશન: ALU ગાણિતિક/લોજિક કાર્ય કરે, રજિસ્ટર/મેમરી વચ્ચે ડેટા ફેરફાર થાય
- રાખવા માટે: "FDE: પહેલા લે, પછી સમજે, અંતે કરે"

આકૃતિની મદદથી 8085 માઇક્રોપ્રોસેસરના બ્લોક ડાયાગ્રામનું વર્ણન કરો.

Block	Function
ALU	Arithmetic & logical operations
Register Array	Temporary data storage
Instruction Register & Decoder	Hold & interpret instructions
Control & Timing Unit	Generate control signals
Address Buffer	Interface with address bus
Data Buffer	Interface with data bus
Serial I/O	Communication with SID/SOD
Interrupt Control	Handle interrupt requests

ડાયાગ્રામ:



- મુખ્ય કમ્પોનન્ટ્સ: ALU અને રજિસ્ટર્સ પ્રોસેસિંગ કોર બનાવે છે
- કંટ્રોલ પાથ: સૂચનાઓ રજિસ્ટર, ડિકોડર, કંટ્રોલ યુનિટ મારફતે વહે છે
- ડેટા પાથ: ડેટા બફર્સથી બાહ્ય બસ સુધી/થી ફેરફાર થાય છે
- ટાઇમિંગ: આંતરિક કલોક દ્વારા બધા ઓપરેશન્સનું સિંક્રોનાઇઝેશન

યાદ રાખવા માટે: "RAID: રજિસ્ટર્સ-ALU-ઈન્ટરફેસ-ડિકોડર્સ"

પ્રશ્ન 2(અ OR) [3 ગુણ]

માઇક્રોપ્રોસેસર અને માઇક્રોકંટ્રોલરની સરખામણી કરો.

જવાબ

લક્ષણ	માઇક્રોપ્રોસેસર	માઇક્રોકંટ્રોલર
ડિઝાઇન	માત્ર CPU	CPU + પેરિફેરલ્સ
મેમરી	બાહ્ય	આંતરિક (RAM/ROM)
I/O પોર્ટ્સ	મર્યાદિત	બિલ્ટ-ઇન ઘણા
ઉપયોગ	જનરલ કમ્પ્યુટિંગ	એમ્બેડેડ સિસ્ટમ
કિંમત	વધારે	ઓછી
ઉદાહરણ	Intel 8085/8086	Intel 8051

યાદ રાખવા માટે: "માઇક્રો-P પ્રોસેસ કરે, માઇક્રો-C કંટ્રોલ કરે"

પ્રશ્ન 2(બ OR) [4 ગુણ]

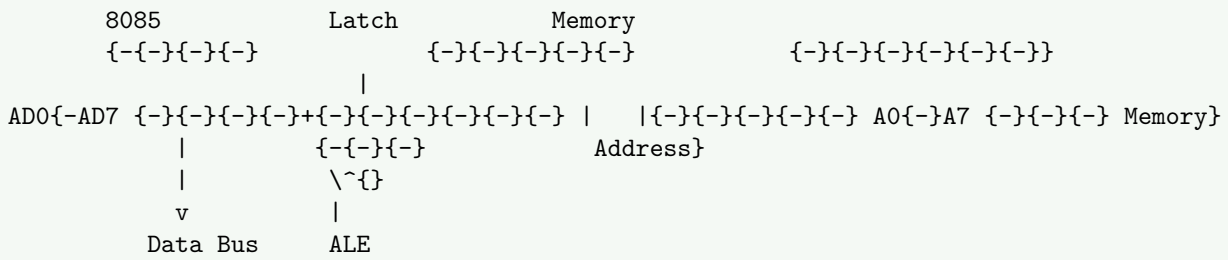
8085 માઇક્રોપ્રોસેસર માટે એડ્રેસ અને ડેટા બસોનું ડી-મલ્ટીપ્લેક્સીંગ સમજાવો.

જવાબ

સ્ટેપ	ક્રિયા
1	ALE સિગ્નલ હાઈ થાય
2	AD0-AD7 પર લોઅર એડ્રેસ (A0-A7) દેખાય
3	લેય ALE નો ઉપયોગ કરી એડ્રેસ પકડે

4 ALE લો થાય, AD0-AD7 હવે ડેટા ટ્રાન્સફર કરે

ડાયાગ્રામ:



- **મલ્ટિપ્લેક્સિંગ:** AD0-AD7 પિન્સ અલગ-અલગ સમયે એડ્રેસ અને ડેટા ટ્રાન્સફર કરે છે
- **ALE સિગ્નલ:** એડ્રેસ લેય અનેબલ એડ્રેસ ક્યારે પકડવું તે નિયંત્રિત કરે છે
- **8-બિટ લેય:** આખા મશીન સાયકલ દરમિયાન લોઅર એડ્રેસ બિટ્સ રાખે છે
- **ટાઇમિંગ:** ALE પલ્સના હાઈ સ્ટેટ દરમિયાન જ એડ્રેસ માન્ય રહે છે

યાદ રાખવા માટે: "ALAD: ALE ડેટા પહેલાં એડ્રેસ લેવું કરે"

પ્રશ્ન 2(ક OR) [7 ગુણ]

આકૃતિની મદદથી 8085 માઇક્રોપ્રોસેસરના પિન ડાયાગ્રામનું વર્ણન કરો.

જાદાબ

પિન ગ્રુપ	કાર્ય
એડ્રેસ/ડેટા	મલ્ટિપ્લેક્સ AD0-AD7, A8-A15
કંટ્રોલ	RD, WR, IO/M, S0, S1, ALE, CLK
ઇન્ટરપ્ટ	INTR, RST 5.5-7.5, TRAP
DMA	HOLD, HLDA
પાવર	Vcc, Vss
સીરિયલ I/O	SID, SOD
રીસેટ	RESET IN, RESET OUT

```
+{-{-}{-}{-}{-}{-}{-}{-}{-}+}
X1 {-{-}|1      40|{-}{-} Vcc}
X2 {-{-}|2      39|{-}{-} HOLD}
RESET OUT{-{-}|3 38|{-}{-} HLDA}
RESET IN {-{-}|4 37|{-}{-} CLK}
IO/M {-{-}|5     36|{-}{-} RESET IN}
S1 {-{-}|6       35|{-}{-} READY}
RD {-{-}|7       34|{-}{-} IO/M}
WR {-{-}|8       33|{-}{-} S1}
ALE {-{-}|9      32|{-}{-} RD}
S0 {-{-}|10     31|{-}{-} WR}
A15 {-{-}|11    30|{-}{-} ALE}
A14 {-{-}|12    29|{-}{-} S0}
A13 {-{-}|13    28|{-}{-} A15}
A12 {-{-}|14    27|{-}{-} A14}
A11 {-{-}|15    26|{-}{-} A13}
A10 {-{-}|16    25|{-}{-} A12}
A9 {-{-}|17     24|{-}{-} A11}
A8 {-{-}|18     23|{-}{-} A10}
AD7 {-{-}|19    22|{-}{-} A9}
AD6 {-{-}|20    21|{-}{-} A8}
+{-{-}{-}{-}{-}{-}{-}{-}{-}+
```

- યાદ રાખવા માટે: "ACID-PS: એડુસ-કંટ્રોલ-ઇન્ટરપ્ટ-DMA-પાવર-સીરિયલ"

8051 માઇક્રોકંટ્રોલરનાં ઇન્ટરફેસ સમજાવો.

ઇન્ટરફેસ	વેક્ટર	પ્રાયોરિટી	સ્ત્રોત
External 0	0003H	1 (IP.0)	Pin INTO (P3.2)
Timer 0	000BH	2 (IP.1)	Timer 0 ઓવરફ્લો
External 1	0013H	3 (IP.2)	Pin INT1 (P3.3)
Timer 1	001BH	4 (IP.3)	Timer 1 ઓવરફ્લો
Serial	0023H	5 (IP.4)	સીરિયલ પોર્ટ ઇવેન્ટ્સ

```

+{-{-}{-}{-}{-}{-}{-}+}
| 8051 |
INT0 {-{-}{-}{-}{-}|      |}
|      |
INT1 {-{-}{-}{-}{-}|      | Interrupts are}
|      | prioritized and
TIMER0 {-{-}{-}{-}|      | can be enabled/disabled}
|      | individually
TIMER1 {-{-}{-}{-}|      |}
|      |
SERIAL {-{-}{-}{-}|      |}
+{-{-}{-}{-}{-}{-}{-}+}

```

याद राखवा माटे: "ETTES: External-Timer-Timer-External-Serial"

याद राजवा भाटे: "ETTES: External-Timer-Timer-External-Serial"

પ્રશ્ન ૩(બ) [4 ગુણ]

8051 માઇક્રોકંટ્રોલરનો પિન ડાયાગ્રામ દોરો.

정답

8051 Microcontroller		
+{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}+		
P1.0{--} 1	40 {--}{--}VCC}	
P1.1{--} 2	39 {--}{--}P0.0/AD0}	
P1.2{--} 3	38 {--}{--}P0.1/AD1}	
P1.3{--} 4	37 {--}{--}P0.2/AD2}	
P1.4{--} 5	36 {--}{--}P0.3/AD3}	
P1.5{--} 6	35 {--}{--}P0.4/AD4}	
P1.6{--} 7	34 {--}{--}P0.5/AD5}	
P1.7{--} 8	33 {--}{--}P0.6/AD6}	
RST {--} 9	32 {--}{--}P0.7/AD7}	
P3.0/RXD 10	31 {--}{--}EA/VPP}	
P3.1/TXD 11	30 {--}{--}ALE/PROG}	
P3.2/INT0 12	29 {--}{--}PSEN}	
P3.3/INT1 13	28 {--}{--}P2.7/A15}	
P3.4/T0{- 14	27 {--}{--}P2.6/A14}	
P3.5/T1{- 15	26 {--}{--}P2.5/A13}	
P3.6/WR{- 16	25 {--}{--}P2.4/A12}	
P3.7/RD{- 17	24 {--}{--}P2.3/A11}	
XTAL2 {- - 18	23 {--}{--}P2.2/A10}	
XTAL1 {- - 19	22 {--}{--}P2.1/A9}	
VSS {- - 20	21 {--}{--}P2.0/A8}	
+{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}+		

પિન ગ્રુપ	કાર્ય
P0	પોર્ટ 0, એડ્રેસ/ડેટા સાથે મલ્ટિપ્લેક્સડ
P1	પોર્ટ 1, જનરલ પર્પઝ I/O
P2	પોર્ટ 2, અપર એડ્રેસ અને I/O
P3	પોર્ટ 3, સ્પેશિયલ ફંક્શન્સ અને I/O
XTAL	ક્રિસ્ટલ ઓસિલેટર કનેક્શન્સ
કંટ્રોલ	RST, EA, ALE, PSEN

યાદ રાખવા માટે: "PORT 0123: ડેટા-જનરલ-એક્સ-સ્પેશિયલ"

પ્રશ્ન 3(ક) [7 ગુણ]

8051 માઇક્રોકંટ્રોલરનું આંતરિક રેમ ઓર્ગનાઇઝેશન સમજાવો.

જવાબ

RAM એરિયા	એડ્રેસ રેન્જ	ઉપયોગ
રજિસ્ટર બેન્ક્સ	00H-1FH	R0-R7 (4 બેન્ક્સ)
બિટ-એડ્રેસેબલ	20H-2FH	128 બિટ્સ (0-7FH)
સ્ક્રેચ પેડ	30H-7FH	જનરલ પર્પઝ
SFRs	80H-FFH	કંટ્રોલ રજિસ્ટર્સ

ડાયાગ્રામ:

8051 Internal RAM (128 bytes):

```

+[-][-][-][-][-][-][-][-][-][-][-][-][-][-][-][-]+ 00H}
| Register Bank 0|
+[-][-][-][-][-][-][-][-][-][-][-][-][-][-][-][-]+ 08H}
| Register Bank 1|
+[-][-][-][-][-][-][-][-][-][-][-][-][-][-][-][-]+ 10H}
| Register Bank 2|
+[-][-][-][-][-][-][-][-][-][-][-][-][-][-][-][-]+ 18H}
| Register Bank 3|
+[-][-][-][-][-][-][-][-][-][-][-][-][-][-][-][-]+ 20H}
| Bit-addressable|}
+[-][-][-][-][-][-][-][-][-][-][-][-][-][-][-][-]+ 30H}
|
| Scratch Pad |
|
+[-][-][-][-][-][-][-][-][-][-][-][-][-][-][-][-]+ 80H}
    
```

Special Function Registers:

```

+[-][-][-][-][-][-][-][-][-][-][-][-][-][-][-][-]+ 80H}
|
| SFRs |
| (not all bytes |
| are used) |
|
+[-][-][-][-][-][-][-][-][-][-][-][-][-][-][-][-]+ FFH}
    
```

- **રજિસ્ટર બેન્ક્સ:** 8 રજિસ્ટર્સ (R0-R7)ની 4 બેન્ક્સ PSW દ્વારા સિલેક્ટેબલ
- **બિટ-એડ્રેસેબલ:** 16 બાઇટ્સ (128 બિટ્સ) વ્યક્તિગત રીતે બિટ તરીકે એડ્રેસેબલ
- **જનરલ પર્પઝ:** યુઝર વેરિએબલ્સ અને સ્ટેક સ્પેસ
- **SFRs:** ઉચ્ચ એડ્રેસ પર કંટ્રોલ અને સ્ટેટસ રજિસ્ટર્સ

યાદ રાખવા માટે: "RBBS: રજિસ્ટર્સ બિટ્સ બફર સ્પેશિયલ"

પ્રશ્ન 3(અ OR) [3 ગુણ]

SFRs ને તેમના એડ્રેસ સાથે સૂચિબદ્ધ કરો.

જવાબ

SFR	એડ્રેસ	કાર્ય
P0	80H	પોર્ટ 0
SP	81H	સ્ટેક પોઇન્ટર
DPL	82H	ડેટા પોઇન્ટર લો
DPH	83H	ડેટા પોઇન્ટર હાઇ
PCON	87H	પાવર કંટ્રોલ
TCON	88H	ટાઇમર કંટ્રોલ

TMOD	89H	ટાઇમર મોડ
P1	90H	પોર્ટ 1
SCON	98H	સીરિયલ કંટ્રોલ
P2	A0H	પોર્ટ 2
IE	A8H	ઇન્ટરપ્ટ એનેબલ
P3	B0H	પોર્ટ 3
IP	B8H	ઇન્ટરપ્ટ પ્રાયોરિટી
PSW	D0H	પ્રોગ્રામ સ્ટેટસ વર્ડ
ACC	E0H	એક્યુમ્યુલેટર
B	F0H	B રજિસ્ટર

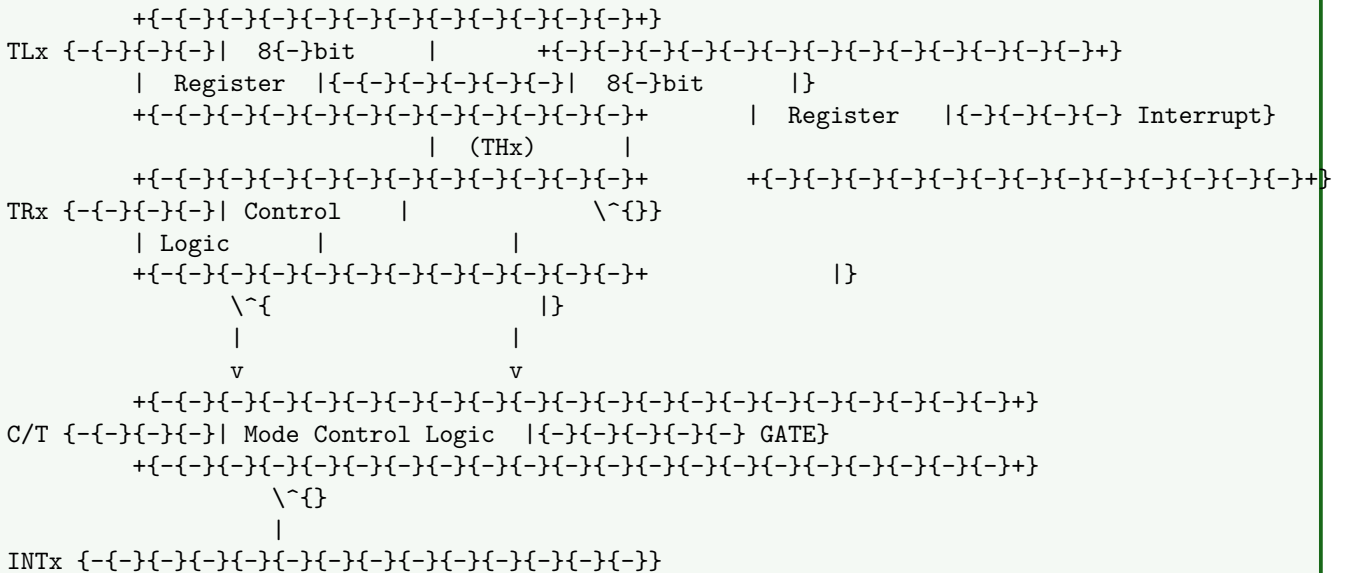
યાદ રાખવા માટે: ``PDPT-SP: પોર્ટ્સ-ડેટા-પ્રોગ્રામ-ટાઇમર્સ-સીરિયલ-પ્રાયોરિટીઝડ''

પ્રશ્ન 3(બ OR) [4 ગુણ]

8051 માઇક્રોકંટ્રોલરના ટાઇમર/કાઉન્ટર્સનો લોજિક ડાયાગ્રામ સમજાવો.

જવાબ

ટાઇમર/કાઉન્ટર ડાયાગ્રામ:



કમ્પોનન્ટ	કાર્ય
TLx, THx	ટાઇમર લો અને હાઈ બાઈટ રજિસ્ટર્સ
C/T	ટાઇમર (0) અથવા કાઉન્ટર (1) મોડ પસંદ કરે
GATE	બાહ્ય એનેબલ કંટ્રોલ
TRx	ટાઇમર રન કંટ્રોલ બિટ
મોડ કંટ્રોલ	ચાર ઓપરેશન મોડમાંથી એક પસંદ કરે

- ટાઇમર: આંતરિક કલોક વાપરે, મશીન સાયકલ ગણે
- કાઉન્ટર: T0/T1 પિન્સ પર બાહ્ય ઘટનાઓ ગણે
- કંટ્રોલ બિટ્સ: TMOD અને TCON રજિસ્ટર્સમાં સેટ થાય
- મોડ્સ: વિવિધ ટાઇમર કોન્ફિગરેશન (13/16/8-બિટ)

યાદ રાખવા માટે: ``TCG: ટાઇમર-કાઉન્ટર-ગેટ''

પ્રશ્ન 3(ક OR) [7 ગુણ]

8051 માઇક્રોકંટ્રોલરનો બ્લોક ડાયાગ્રામ સમજાવો.

[illegible]

- યાદ રાખવા માટે: "CAPITALS: CPU આર્કિટેક્ચર પોર્ટ્સ I/O ટાઇમર ALU ઇન્ટરફેસ સીરિયલ"

ડેટાના બે બાઈટ ઉમેરીને પરિણામ R4 રજિસ્ટરમાં સંગ્રહિત કરવા માટે 8051 એસેમ્બલી લેંગ્વેજ પ્રોગ્રામ લખો.

- પ્રથમ ઓપરેન્ડ એક્યુમ્યુલેટરમાં લોડ કરો
- બીજો ઓપરેન્ડ રજિસ્ટર R3માં લોડ કરો
- ADD સૂચનાનો ઉપયોગ કરી સરવાળો કરો
- એક્યુમ્યુલેટરમાંથી પરિણામ R4માં સ્ટોર કરો

યાદ રાખવા માટે: ``LLAS: લોડ-લોડ-એડ-સ્ટોર``

પ્રશ્ન 4(બ) [4 ગુણ]

પોર્ટ-1 અને પોર્ટ-2ના કન્ટેન્ટને OR કરીને પછી પરિણામને બાહ્ય RAM સ્થાન 0200H માં મૂકવા માટે 8051 એસેમ્બલી લેંગ્વેજ પ્રોગ્રામ લખો.

જવાબ

```
MOV A, P1      ; {-1      }
ORL A, P2      ; {-2      OR }
MOV DPTR, \#0200H ; RAM    DPTR
MOVX @DPTR, A   ; RAM     0200H
```

મુખ્ય સ્ટેપ્સ:

- પોર્ટ-1 એક્યુમ્યુલેટરમાં વાંચો
- પોર્ટ-2 સાથે OR ઓપરેશન કરો
- બાહ્ય RAM માટે ડેટા પોઇન્ટર (DPTR) સેટ કરો
- પરિણામ બાહ્ય મેમરીમાં લખો

યાદ રાખવા માટે: ``PORT: પોર્ટ-OR-રજિસ્ટર-ટ્રાન્સફર``

પ્રશ્ન 4(ક) [7 ગુણ]

8051 માઇક્રોકંટ્રોલરના એડ્રેસિંગ મોડ્સની યાદી બનાવો અને ઓછામાં ઓછા એક ઉદાહરણ સાથે તેમને સમજાવો.

જવાબ

એડ્રેસિંગ મોડ	ઉદાહરણ	વર્ણન
ઇમીડિયેટ	MOV A, #25H	ડેટા સૂચનામાં છે
રજિસ્ટર	MOV A, R0	ડેટા રજિસ્ટરમાં છે
ડાયરેક્ટ	MOV A, 30H	ડેટા RAM એડ્રેસ પર છે
ઇન્ડાયરેક્ટ	MOV A, @R0	R0/R1 એડ્રેસ ધરાવે છે
ઇન્ડેક્સ	MOVC A, @A+DPTR	પ્રોગ્રામ મેમરી એક્સેસ
બિટ	SETB P1.3	વ્યક્તિગત બિટ્સ એક્સેસ
રિલેટિવ	SJMP LABEL	8-બિટ ઓફસેટ સાથે જમ્પ

ઉદાહરણો:

- ઇમીડિયેટ: MOV A, #55H (A માં 55H લોડ કરો)
- રજિસ્ટર: ADD A, R3 (A માં R3 ઉમેરો)
- ડાયરેક્ટ: MOV 40H, A (A ને એડ્રેસ 40H પર સ્ટોર કરો)
- ઇન્ડાયરેક્ટ: MOV @R0, #5 (R0 માં રહેલા એડ્રેસ પર 5 સ્ટોર કરો)
- ઇન્ડેક્સ: MOVC A, @A+DPTR (કોડ મેમરી વાંચો)
- બિટ: CLR C (કેરી ફ્લેગ સાફ કરો)
- રિલેટિવ: JZ LOOP (જો A ઝીરો હોય તો જમ્પ કરો)

યાદ રાખવા માટે: ``I'M DIRBI: ઇમીડિયેટ રજિસ્ટર ડાયરેક્ટ બિટ ઇન્ડેક્સ``

પ્રશ્ન 4(અ OR) [3 ગુણ]

નીચેની ઈન્સ્ટ્રક્શન્સ સમજાવો: (i) DJNZ (ii) POP (iii) CJNE.

જવાબ

ઈન્સ્ટ્રક્શન	સિન્ટેક્સ	ઓપરેશન
DJNZ	DJNZ Rn, rel	રજિસ્ટર ડેક્કિમેન્ટ, જમ્પ ઇફ નોટ ઝીરો
POP	POP direct	સ્ટેકમાંથી ડેટા ડાયરેક્ટ એડ્રેસ પર પોપ કરો
CJNE	CJNE A, #data, rel	કમ્પેર એન્ડ જમ્પ ઇફ નોટ ઇક્વલ

ઉદાહરણો અને સમજૂતી:

- **DJNZ R7, LOOP:** R7 ઘટાડે અને જો $R7 \neq 0$ LOOP
– લૂપ કાઉન્ટર અને ડિલે માટે વપરાય છે
 - **POP 30H:** સ્ટેકમાંથી ડેટા એડ્રેસ 30H પર કોપી કરે
– ડેટા રિટ્રીવલ પછી SP વધારે છે
 - **CJNE A, #25H, NOTEQUAL:** A ને 25H સાથે સરખાવે, સમાન ન હોય તો જમ્પ કરે
– જો $A < 25H$ હોય તો કેરી ફ્લેગ પણ સેટ કરે
- યાદ રાખવા માટે: "DPC: ડેક્રિમેન્ટ-પોપ-કમ્પેર"

પ્રશ્ન 4(બ OR) [4 ગુણ]

12 MHz ની ક્રિસ્ટલ ફ્રિક્વન્સી સાથે 8051 માઇક્રોકંટ્રોલર માટે, 4ms નો ડિલેય જનરેટ કરો.

જવાબ

```
;          = 12 MHz
;          = 1 s
;          = 4 ms = 4000

MOV R7, \#16 ;          (16 x 250 = 4000)
DELAY1: MOV R6, \#250 ;
DELAY2: NOP ; 1
NOP ; 1
DJNZ R6, DELAY2 ; 2          (250 x 4 = 1000 )
DJNZ R7, DELAY1 ; 16 x 250 = 4000
RET ;
```

ગણતરી:

- દરેક ઇનર લૂપ: $4 \text{ સાયકલ} \times 250 = 1000$
- આઉટર લૂપ: $16 \text{ ફેરા} \times 1000 = 16,000$
- 12MHz પર, 1 મશીન સાયકલ = $1 \mu\text{s}$
- કુલ ડિલે = 4ms (4000 સાયકલ)

યાદ રાખવા માટે: "LNDD: લોડ-NOP-ડેક્રિમેન્ટ-ડેક્રિમેન્ટ"

પ્રશ્ન 4(ક OR) [7 ગુણ]

8051 માઇક્રોકંટ્રોલર માટે કોઈપણ સાત લોજિકલ ઇન્સ્ટ્રક્શન ઉદાહરણ સાથે સમજાવો.

જવાબ

ઈન્સ્ટ્રક્શન	ઉદાહરણ	ઓપરેશન
ANL	ANL A, #3FH	લોજિકલ AND
ORL	ORL P1, #80H	લોજિકલ OR
XRL	XRL A, R0	લોજિકલ XOR
CLR	CLR A	ક્લિયર (0 સેટ)
CPL	CPL P1.0	કોમ્પ્લિમેન્ટ (ઇન્વર્ટ)
RL	RL A	રોટેટ લેફ્ટ
RR	RR A	રોટેટ રાઇટ

ઉદાહરણો સાથે સમજૂતી:

1. **ANL A, #0FH:** હાઈ નિબલ માર્સ્ક કરે ($A = A \text{ AND } 0FH$)
 - પહેલાં: $A = 95H$, પછી: $A = 05H$
2. **ORL 20H, A:** મેમરીમાં બિટ્સ સેટ કરે ($20H = 20H \text{ OR } A$)
 - પહેલાં: $20H = 55H$, $A = 0AH$, પછી: $20H = 5FH$
3. **XRL A, #55H:** ચોક્કસ બિટ્સ ટોગલ કરે ($A = A \text{ XOR } 55H$)
 - પહેલાં: $A = AAH$, પછી: $A = FFH$
4. **CLR C:** કેરી ફ્લેગ સાફ કરે ($C = 0$)
 - સબટ્રેક્શન ઓપરેશન પહેલા વપરાય છે
5. **CPL A:** બધા બિટ્સ ઇન્વર્ટ કરે ($A = \text{NOT } A$)
 - પહેલાં: $A = 55H$, પછી: $A = AAH$
6. **RL A:** એક્યુમ્યુલેટરને એક બિટ ડાબી તરફ રોટેટ કરે
 - પહેલાં: $A = 85H$ (10000101), પછી: $A = 0BH$ (00001011)
7. **RR A:** એક્યુમ્યુલેટરને એક બિટ જમણી તરફ રોટેટ કરે
 - પહેલાં: $A = 85H$ (10000101), પછી: $A = C2H$ (11000010)

યાદ રાખવા માટે: "A-OX-CCR: AND OR XOR ક્લિયર કોમ્પિલમેન્ટ રોટેટ"

પ્રશ્ન 5(અ) [3 ગુણ]

વિવિધ ક્ષેત્રોમાં માઇક્રોકંટ્રોલરની એપ્લિકેશનોની સૂચિ બનાવો.

જવાબ

ક્ષેત્ર	એપ્લિકેશન્સ
ઔદ્યોગિક	મોટર કંટ્રોલ, ઓટોમેશન, PLCs
મેડિકલ	પેશન્ટ મોનિટરિંગ, ડાયગ્નોસ્ટિક ઉપકરણો
કન્ઝ્યુમર	વોશિંગ મશીન, માઇક્રોવેવ, રમકડાં
ઓટોમોટિવ	એન્જિન કંટ્રોલ, ABS, એરબેગ સિસ્ટમ
કમ્યુનિકેશન	મોબાઇલ ફોન, મોડેમ, રાઉટર
સિક્યુરિટી	એક્સેસ કંટ્રોલ, અલાર્મ સિસ્ટમ

યાદ રાખવા માટે: "I-MACS: ઇન્ડસ્ટ્રિયલ-મેડિકલ-ઓટોમોટિવ-કન્ઝ્યુમર-સિક્યુરિટી"

પ્રશ્ન 5(બ) [4 ગુણ]

8051 માઇક્રોકંટ્રોલર સાથે પુશ બટન સ્વિચ અને LED ઇન્ટરફેસ કરો.

જવાબ

સર્કિટ ડાયાગ્રામ:

```

Vcc
|
R (10K)
|
P1.0{-}{-}{+}{-}{-}{-}{-}{-}{-}{-} Push Button {-}{-}{-}{-}{-}{-}{-} GND}

Vcc

```



```

MOV A, \#06H      ;
ACALL COMMAND     ;
MOV A, \#80H      ;
ACALL COMMAND     ;

; "HELLO"
MOV A, \#{H       ; H   }
ACALL DISPLAY     ;
MOV A, \#{E       ; E   }
ACALL DISPLAY     ;
MOV A, \#{L       ; L   }
ACALL DISPLAY     ;
MOV A, \#{L       ; L   }
ACALL DISPLAY     ;
MOV A, \#{O       ; O   }
ACALL DISPLAY     ;

SJMP $            ;

;
COMMAND:
MOV P2, A         ;
CLR P3.0          ; RS=0
CLR P3.1          ; R/W=0
SETB P3.2         ; E=1
ACALL DELAY       ;
CLR P3.2          ; E=0
RET               ;

;
DISPLAY:
MOV P2, A         ;
SETB P3.0         ; RS=1
CLR P3.1          ; R/W=0
SETB P3.2         ; E=1
ACALL DELAY       ;
CLR P3.2          ; E=0
RET               ;

;
DELAY:
MOV R7, \#50      ;
DELAY_LOOP:
DJNZ R7, DELAY_LOOP ; R7=0
RET               ;

END               ;

```

કમ્પોનન્ટ	કનેક્શન	હેતુ
ડેટા પિન	P2.0-P2.7	ડેટા/કમાન્ડ ટ્રાન્સફર
RS (રજિસ્ટર સિલેક્ટ)	P3.0	કમાન્ડ (0) અથવા ડેટા (1) સિલેક્ટ
R/W (રીડ/રાઇટ)	P3.1	રાઇટ (0) અથવા રીડ (1) સિલેક્ટ
E (એનેબલ)	P3.2	ફોલોિંગ એજ પર ડેટા લેવ

- **ઇનિશિયલાઇઝેશન:** 8-બિટ, 2 લાઇન, કર્સર ઓપ્શન માટે LCD કોન્ફિગર
- **ડેટા ટ્રાન્સફર:** RS=0 સાથે કમાન્ડ, RS=1 સાથે કેરેક્ટર મોકલાય
- **કેરેક્ટર્સ:** ASCII વેલ્યુ એક પછી એક મોકલી ટેક્સ્ટ દર્શાવાય
- **ટાઇમિંગ:** યોગ્ય સિગ્નલ ટાઇમિંગ માટે ડિલે રૂટીન

યાદ રાખવા માટે: "DICE: ડેટા-ઇન્ટરફેશન-કંટ્રોલ-એનેબલ"

પ્રશ્ન 5(અ OR) [3 ગુણ]

8051 માઇક્રોકંટ્રોલર સાથે LM35 નું ઇન્ટરફેસિંગ દોરો.

જવાબ

સર્કિટ ડાયાગ્રામ:

[illegible]

Component	Function
LM35	Temperature sensor (10mV/°C)
ADC0804	Analog-to-Digital Converter
8051	Microcontroller to read temperature

મુખ્ય પોઇન્ટ્સ:

- LM35 તાપમાનના પ્રમાણમાં એનાલોગ વોલ્ટેજ ઉત્પન્ન કરે છે
- ADC0804 એનાલોગ વોલ્ટેજને ડિજિટલ વેલ્યુમાં રૂપાંતરિત કરે છે
- 8051 ADC નું નિયંત્રણ કરે છે અને તાપમાન ડેટા વાંચે છે
- રિઝોલ્યુશન: $10\text{mV}/\rightarrow 8 - \text{ADC} \sim 0.2$

યાદ રાખવા માટે: "TAC: તાપમાન-એનાલોગ-કન્વર્ટ"

પ્રશ્ન 5(બ OR) [4 ગુણ]

8051 માઇક્રોકંટ્રોલર સાથે સ્ટેપર મોટર ઇન્ટરફેસ કરો.

ଉଦାହ

સર્કિટ ડાયાગ્રામ:

8051	ULN2003
+{--}{--}{--}{--}{--}{--}{--}+	{--}{--}{--}{--}{--}{--}{--}{--}
	+{--}{--}{--}{--} IN1 OUT1 {--}{--}{--}+
P1.0 {--}{--}{--}{--}{--}{--}{--}{--} {--}{--}{--}{--} IN2 OUT2 {--}{--}{--}+	
P1.1 {--}{--}{--}{--}{--}{--}{--}{--} {--}{--}{--}{--} IN3 OUT3 {--}{--}{--}+{--}{--}{--}{--} 4{--}win	
P1.2 {--}{--}{--}{--}{--}{--}{--}{--} {--}{--}{--}{--} IN4 OUT4 {--}{--}{--}+	Stepper}
P1.3 {--}{--}{--}{--}{--}{--}{--}{--}	Motor}
+{--}{--}{--}{--}{--}{--}{--}+	{--}{--}{--}{--}{--}{--}{--}{--}


```

Analog Input{-}{-}{-}| ADC0804 |}
0{-5V | | |}
      v | | |
      +{-}{-}{-}+{-}{-}{-}{-}+| | +{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}+}
      | | | | |
Vref/2{-| | | |}
      | | | | |
      |{-}{-}{-}{-}| P0.0{-}P0.7|}
CS{-}{-}{-}{-}{-}| | | |}
RD{-}{-}{-}{-}{-}| | | |}
WR{-}{-}{-}{-}{-}| | | |}
INTR{-}{-}{-}| | |{-}{-}{-}{-}| P3.2 |}
      | | | | |
      +{-}{-}{-}{-}{-}{-}{-}{-}{-}+ | | +{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}+}
      +{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}+ +{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}+}

```

```
ORG 0000H
```

```
WAIT:    JB P1.3, WAIT ;                (INTR = 0)
```

```
SETB P1.0      ; CS = 1 (    )
```

```
ACALL DELAY      ;
SJMP START      ;
```

END

ADC0804 ફીચર્સ:

- 8-બિટ રિઝોલ્યુશન (256 સ્ટેપ્સ)
- 0-5V ઇનપુટ રેન્જ
- સિંગલ-ચેનલ ઓપરેશન
- ~100 μ s કન્વર્ઝન ટાઇમ
- ઇન્ટરફેસ પ્રોટોકોલ:
 1. CS એક્ટિવ કરો, કન્વર્ઝન શરૂ કરવા WR પલ્સ કરો
 2. INTR લો થાય ત્યાં સુધી રાહ જુઓ (કન્વર્ઝન પૂર્ણ)
 3. ડેટા વાંચવા RD એક્ટિવ કરો
 4. કામ પૂરું થયા પછી CS ડિએક્ટિવ કરો

યાદ રાખવા માટે: "CRIW: કંટ્રોલ-રીડ-ઇન્ટરપ્ટ-રાઇટ"