

# માઇક્રોપ્રોસેસર અને માઇક્રોકન્ટ્રોલર (4341101) - સમર 2023 સોલ્યુશન

Milav Dabgar

15 જૂન 2023

## પ્રશ્ન 1 [a ગુણ]

3 માઇક્રોપ્રોસેસર અને માઇક્રોકન્ટ્રોલરની સરખામણી કરો.

જવાબ

જવાબ:

કોષ્ટક 1. માઇક્રોપ્રોસેસર વિરુદ્ધ માઇક્રોકન્ટ્રોલર

| ફીચર     | માઇક્રોપ્રોસેસર         | માઇક્રોકન્ટ્રોલર             |
|----------|-------------------------|------------------------------|
| વ્યાખ્યા | એકલ ચિપ પર CPU          | એકલ ચિપ પર સંપૂર્ણ કમ્પ્યુટર |
| મેમરી    | બાહ્ય RAM/ROM જરૂરી     | અંદર જ RAM/ROM               |
| ઉપયોગો   | સામાન્ય કમ્પ્યુટિંગ, PC | એમ્બેડેડ સિસ્ટમ, IoT         |
| ઉદાહરણો  | Intel 8085, 8086        | 8051, Arduino, PIC           |
| કિંમત    | વધારે                   | ઓછી                          |

મેમરી ટ્રીક

“પ્રોસેસર રામ માંગે, કન્ટ્રોલર રામ રાખે” (પ્રોસેસરને બહારથી રામ જોઈએ, કન્ટ્રોલરમાં રામ અંદર જ હોય છે)

## પ્રશ્ન 1 [b ગુણ]

4 RISC અને CISC ની સરખામણી કરો.

જવાબ

જવાબ:

કોષ્ટક 2. RISC વિરુદ્ધ CISC

| ફીચર             | RISC (રિડ્યુસ્ડ ઇન્સ્ટ્રક્શન સેટ કમ્પ્યુટર) | CISC (કોમ્પ્લેક્સ ઇન્સ્ટ્રક્શન સેટ કમ્પ્યુટર) |
|------------------|---|---|
| ઇન્સ્ટ્રક્શન     | થોડી, સરળ ઇન્સ્ટ્રક્શન                      | ઘણી, જટિલ ઇન્સ્ટ્રક્શન                        |
| એક્ઝિક્યુશન ટાઈમ | ફિક્સ્ડ (1 કલોક સાયકલ)                      | વેરિએબલ (ઘણી સાયકલ)                           |
| મેમરી એક્સેસ     | માત્ર લોડ/સ્ટોર દ્વારા                      | ઘણા મેમરી એક્સેસ મોડ                          |
| પાઇપલાઇનિંગ      | સરળ અમલીકરણ                                 | મુશ્કેલ અમલીકરણ                               |
| ઉદાહરણો          | ARM, MIPS                                   | Intel x86, 8085                               |
| હાર્ડવેર         | સરળ, ઓછા ટ્રાન્ઝિસ્ટર                       | જટિલ, વધુ ટ્રાન્ઝિસ્ટર                        |
| રજિસ્ટર સેટ      | વધુ રજિસ્ટર                                 | ઓછા રજિસ્ટર                                   |

## મેમરી ટ્રીક

“RISC ઝડપી, CISC બહોળું” (RISC ઝડપી હોય છે, CISC માં ઘણી ઇન્સ્ટ્રક્શન હોય છે)

## પ્રશ્ન 1 [c ગુણ]

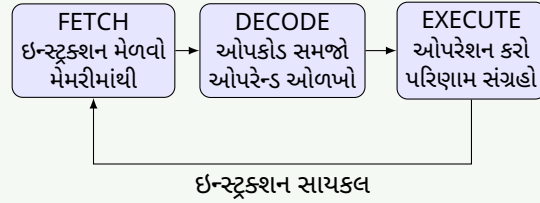
7 વ્યાખ્યાયિત કરો: માઇક્રોપ્રોસેસર, ઓપરેન્ડ, ઇન્સ્ટ્રક્શન સાયકલ, ઓપકોડ, ALU, મશીન સાયકલ, ટી-સ્ટેટ

જવાબ

જવાબ:

કોષ્ટક 3. વ્યાખ્યાઓ

| શબ્દ               | વ્યાખ્યા   |
|--------------------|--|
| માઇક્રોપ્રોસેસર    | એક ઇન્ટિગ્રેટેડ સર્કિટ પર CPU જે ઇન્સ્ટ્રક્શન પ્રોસેસ કરે છે   |
| ઓપરેન્ડ            | ઇન્સ્ટ્રક્શનમાં વપરાતી ડેટા વેલ્યુ                             |
| ઇન્સ્ટ્રક્શન સાયકલ | ઇન્સ્ટ્રક્શન ફેચ, ડિકોડ અને એક્ઝિક્યુટની સંપૂર્ણ પ્રક્રિયા     |
| ઓપકોડ              | ઓપરેશન કોડ જે CPU ને કહે છે કે કયું ઓપરેશન કરવાનું છે          |
| ALU                | અર્થમેટિક લોજિક યુનિટ જે ગણિત ઓપરેશન કરે છે                    |
| મશીન સાયકલ         | મૂળભૂત ઓપરેશન જેમ કે મેમરી રીડ/રાઈટ (ઇન્સ્ટ્રક્શન સાયકલનો ભાગ) |
| ટી-સ્ટેટ           | ટાઈમ સ્ટેટ - પ્રોસેસરમાં સમયનો સૌથી નાનો એકમ (ક્લોક પીરિયડ)    |



## મેમરી ટ્રીક

“મારો ઓલ્ડ Intel ચિપ ઓનલી મેક્સ ટ્રબલ” (માઇક્રોપ્રોસેસર, ઓપરેન્ડ, ઇન્સ્ટ્રક્શન, ઓપકોડ, ALU, મશીન, ટી-સ્ટેટ)

OR

## પ્રશ્ન 1 [c ગુણ]

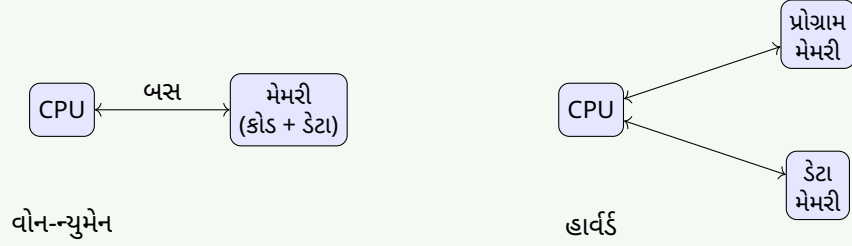
7 વોન-ન્યુમેન અને હાર્વર્ડ આર્કિટેક્ચરની તુલના કરો.

જવાબ

જવાબ:

કોષ્ટક 4. વોન-ન્યુમેન વિરુદ્ધ હાર્વર્ડ આર્કિટેક્ચર

| ફીચર        | વોન-ન્યુમેન આર્કિટેક્ચર                  | હાર્વર્ડ આર્કિટેક્ચર                   |
|-------------|--|--|
| મેમરી બસ    | ઇન્સ્ટ્રક્શન અને ડેટા માટે એક જ મેમરી બસ | પ્રોગ્રામ અને ડેટા મેમરી માટે અલગ બસ   |
| એક્ઝિક્યુશન | સિક્વેન્શિયલ એક્ઝિક્યુશન                 | પેરેલલ ફ્રેચ અને એક્ઝિક્યુટ શક્ય       |
| સ્પીડ       | બસ બોટલનેક ને કારણે ધીમું                | સમાંતર એક્સેસને કારણે ઝડપી             |
| જટિલતા      | સરળ ડિઝાઇન                               | વધુ જટિલ ડિઝાઇન                        |
| ઉપયોગો      | સામાન્ય કમ્પ્યુટિંગ                      | DSP, માઇક્રોકન્ટ્રોલર, એમ્બેડેડ સિસ્ટમ |
| સિક્યોરિટી  | ઓછી સુરક્ષિત (કોડ ડેટા તરીકે બદલી શકાય)  | વધુ સુરક્ષિત (કોડ ડેટાથી અલગ)          |
| ઉદાહરણ      | મોટાભાગના PC, 8085, 8086                 | 8051, PIC, ARM Cortex-M                |



### મેમરી ટ્રીક

“હાર્વર્ડ હંમેશા અલગ રસ્તા રાખે” (હાર્વર્ડમાં મેમરી પાથ અલગ હોય છે)

## પ્રશ્ન 2 [a ગુણ]

3 8085 માઇક્રોપ્રોસેસરનું ફ્લેગ રજિસ્ટર દોરો અને તેને સમજાવો.

### જવાબ

જવાબ:

|   |   |   |    |   |   |   |    |
|---|---|---|----|---|---|---|----|
| 7 | 6 | 5 | 4  | 3 | 2 | 1 | 0  |
| S | Z | - | AC | - | P | - | CY |

FLAG REGISTER (F)

### કોષ્ટક 5. ફ્લેગ રજિસ્ટર બિટ્સ

| ફ્લેગ | નામ             | કાર્ય                                      |
|-------|-----------------|--|
| S     | સાઈન            | જો પરિણામ નેગેટિવ હોય તો સેટ થાય (બિટ 7=1) |
| Z     | ઝીરો            | જો પરિણામ ઝીરો હોય તો સેટ થાય              |
| AC    | ઓકિઝિલિયરી કેરી | જો બિટ 3 થી બિટ 4 માં કેરી થાય તો સેટ થાય  |
| P     | પેરિટી          | જો પરિણામમાં ઇવન પેરિટી હોય તો સેટ થાય     |
| CY    | કેરી            | જો બિટ 7 થી કેરી કે બોરો થાય તો સેટ થાય    |

### મેમરી ટ્રીક

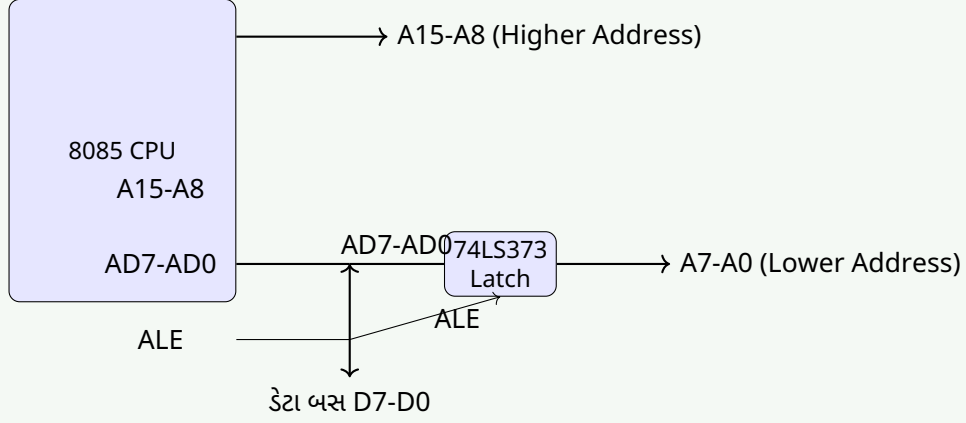
“સરસ ઝોમ્બી આજે પણ ચાલે” (સાઈન, ઝીરો, ઓકિઝિલિયરી, પેરિટી, કેરી)

## પ્રશ્ન 2 [b ગુણ]

4 8085 માઇક્રોપ્રોસેસર માટે એડ્રેસ અને ડેટાબસોનું ડી-મલ્ટીપ્લેક્સીંગ સમજાવો.

જવાબ

જવાબ:



- જરૂરિયાત: 8085 માં પિન બચાવવા માટે મલ્ટીપ્લેક્સર પિન (AD0-AD7) હોય છે
- પ્રક્રિયા:
  1. CPU એડ્રેસ AD0-AD7 પિન પર મૂકે છે
  2. ALE (એડ્રેસ લેય એનેબલ) સિગ્નલ HIGH થાય છે
  3. એડ્રેસ લેય (74LS373) લોઅર એડ્રેસ બિટ્સ પકડે છે
  4. ALE LOW થાય છે, એડ્રેસ લેય થઈ જાય છે
  5. AD0-AD7 પિન હવે ડેટા ટ્રાન્સફર માટે ફ્રી થઈ જાય છે

મેમરી ટ્રીક

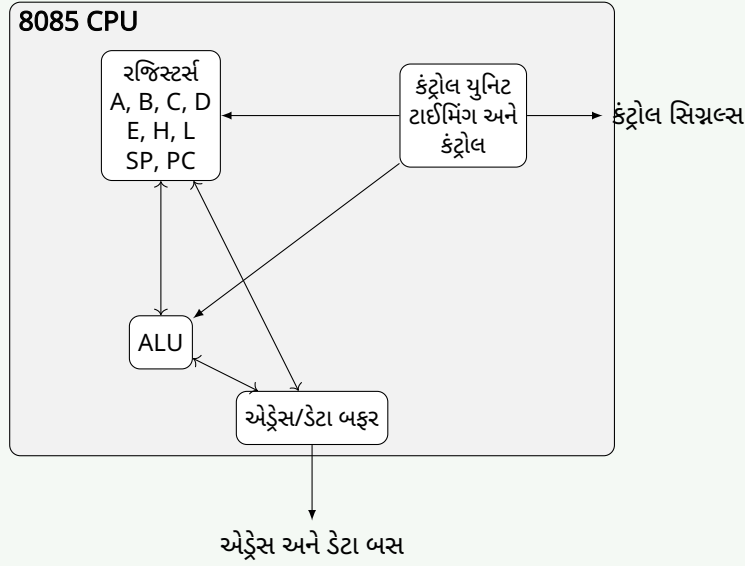
“ALE પહેલા, ડેટા પછી” (એડ્રેસ લેય એનેબલ પહેલા એડ્રેસ પકડે, પછી ડેટા આવે)

## પ્રશ્ન 2 [c ગુણ]

7 આકૃતિની મદદથી 8085 માઇક્રોપ્રોસેસરના આર્કિટેક્ચરનું વર્ણન કરો.

જવાબ

જવાબ:



- મુખ્ય ઘટકો:
  - રજિસ્ટર્સ: સ્ટોરેજ લોકેશન (A, B-L, SP, PC, Flags)
  - ALU: ગાણિતિક અને લોજિકલ ઓપરેશન કરે છે
  - કંટ્રોલ યુનિટ: ટાઈમિંગ અને કંટ્રોલ સિગ્નલ જનરેટ કરે છે
  - બસ: એડ્રેસ બસ (16-bit), ડેટા બસ (8-bit), કંટ્રોલ બસ
- મુખ્ય ફીચર્સ:
  - 8-બિટ ડેટા બસ, 16-બિટ એડ્રેસ બસ (64KB એડ્રેસેબલ મેમરી)
  - 6 જનરલ-પર્પઝ રજિસ્ટર (B,C,D,E,H,L) અને એક્યુમ્યુલેટર
  - 5 ફ્લેગ્સ સ્ટેટસ માહિતી માટે

### મેમરી ટ્રીક

“RABC” - “રજિસ્ટર, ALU, બસ, કંટ્રોલ” (મુખ્ય ઘટકો)

OR

## પ્રશ્ન 2 [a ગુણ]

3 8085 માઈક્રોપ્રોસેસરનું બસ ઓર્ગનાઈઝેશન સમજાવો.

જવાબ

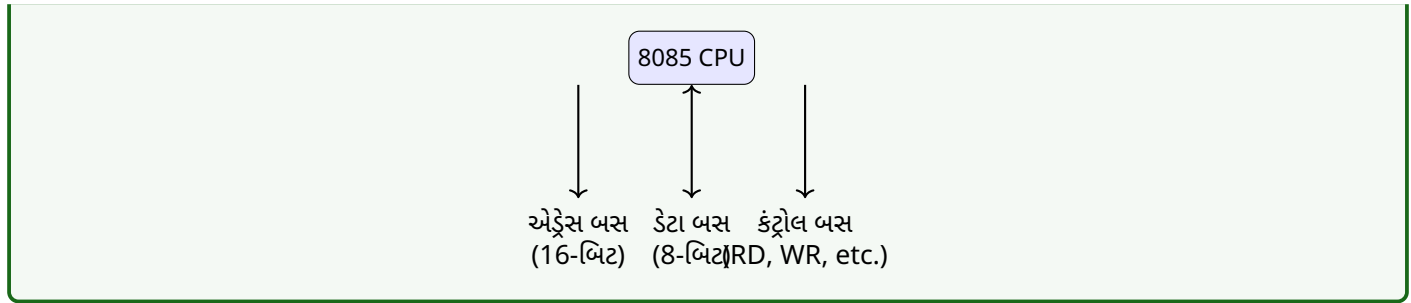
જવાબ:

કોષ્ટક 6. 8085 બસ ઓર્ગનાઈઝેશન

| બસ પ્રકાર  | વિસ્થ           | કાર્ય   |
|------------|-----------------|---|
| એડ્રેસ બસ  | 16-બિટ (A0-A15) | મેમરી/I/O ડિવાઈસ એડ્રેસ લઈ જાય છે             |
| ડેટા બસ    | 8-બિટ (D0-D7)   | CPU અને મેમરી/I/O વચ્ચે ડેટા ટ્રાન્સફર કરે છે |
| કંટ્રોલ બસ | વિવિધ સિગ્નલ્સ  | સિસ્ટમ ઓપરેશન કોઓર્ડિનેટ કરે છે               |

મુખ્ય કંટ્રોલ સિગ્નલ્સ:

- $\overline{RD}$ : રીડ સિગ્નલ (એક્ટિવ લો)
- $\overline{WR}$ : રાઈટ સિગ્નલ (એક્ટિવ લો)
- $\overline{ALE}$ : એડ્રેસ લેય અનેબલ
- $IO/\overline{M}$ : I/O (હાઈ) અને મેમરી (લો) ઓપરેશન વચ્ચે ભેદ પાડે છે



### મેમરી ટ્રીક

“ADC” - “અડ્રેસ શોધે, ડેટા ફરે, કંટ્રોલ ચલાવે”

OR

## પ્રશ્ન 2 [b ગુણ]

### 4 સમજાવો: પ્રોગ્રામ કાઉન્ટર અને સ્ટેક પોઈન્ટર

જવાબ

જવાબ:

કોષ્ટક 7. PC વિરુદ્ધ SP

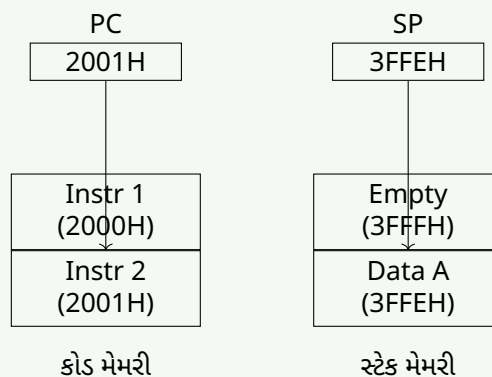
| રજિસ્ટર                | સાઈઝ   | કાર્ય   |
|------------------------|--------|---|
| પ્રોગ્રામ કાઉન્ટર (PC) | 16-બિટ | આગલા એક્ઝિક્યુટ થનાર ઇન્સ્ટ્રક્શનનું અડ્રેસ રાખે છે |
| સ્ટેક પોઈન્ટર (SP)     | 16-બિટ | મેમરીમાં સ્ટેકના ટોપને પોઈન્ટ કરે છે                |

**પ્રોગ્રામ કાઉન્ટર (PC):**

- ઇન્સ્ટ્રક્શન ફ્રેમ પછી ઓટોમેટિક વધે છે
- જમ્પ/કોલ ઇન્સ્ટ્રક્શન દ્વારા બદલાય છે
- પ્રોગ્રામ એક્ઝિક્યુશન સિક્વેન્સ કંટ્રોલ કરે છે
- રીસેટ પર 0000H પર સેટ થાય છે

**સ્ટેક પોઈન્ટર (SP):**

- સ્ટેક પર છેલ્લે પુશ કરેલ ડેટા આઈટમને પોઈન્ટ કરે છે
- સ્ટેક LIFO (લાસ્ટ ઇન ફર્સ્ટ આઉટ) પ્રમાણે કામ કરે છે
- સબરૂટિન કોલ અને ઇન્ટરપ્ટ દરમિયાન વપરાય છે
- સ્ટેક મેમરીમાં નીચે તરફ વધે છે (ઘટાડાય છે)



## મેમરી ટ્રીક

“PC આગળ જુઓ, SP સ્ટેક સંભાળો” (PC આગળું ઇન્સ્ટ્રક્શન જુઓ છે, SP સ્ટેક મેનેજ કરે છે)

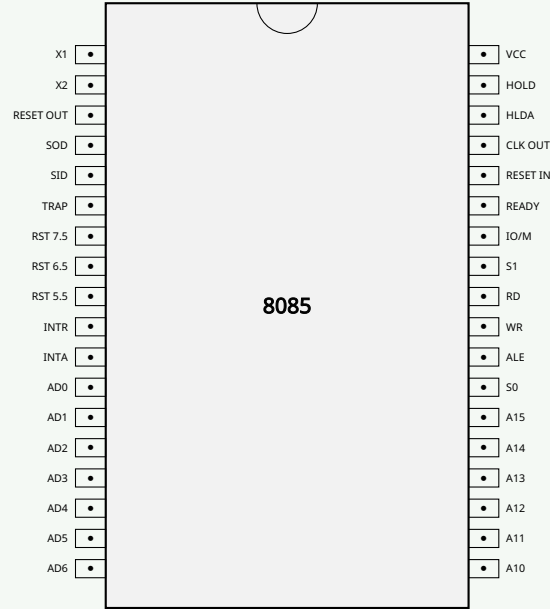
OR

## પ્રશ્ન ૨ [c ગુણ]

7 આકૃતિની મદદથી 8085 માઇક્રોપ્રોસેસરના પિન ડાયાગ્રામનું વર્ણન કરો.

## જવાબ

જવાબ:



પિન ગ્રુપ્સ:

1. પાવર & કલોક: Vcc, GND, X1, X2, CLK
2. એડ્રેસ/ડેટા: A8-A15, AD0-AD7 (મલ્ટીપ્લેક્સ)
3. કંટ્રોલ: ALE,  $\overline{RD}$ ,  $\overline{WR}$ ,  $IO/\overline{M}$
4. ઇન્ટરપ્ટ: INTR,  $\overline{INTA}$ , RST 5.5/6.5/7.5, TRAP
5. DMA: HOLD, HLDA
6. સિરિયલ I/O: SID, SOD
7. સ્ટેટસ: S0, S1

## મેમરી ટ્રીક

“PACI-DHS” (પાવર, એડ્રેસ, કંટ્રોલ, ઇન્ટરપ્ટ, DMA, હાર્ડવેર સ્ટેટસ, સિરિયલ)

## પ્રશ્ન ૩ [a ગુણ]

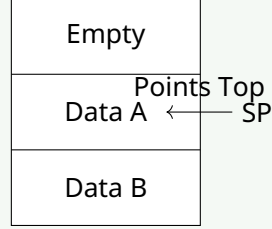
૩ સ્ટેક, સ્ટેક પોઈન્ટર અને સ્ટેક ઓપરેશન સમજાવો.

## જવાબ

જવાબ:

કોષ્ટક 8. સ્ટેક કન્સેપ્ટ

| શબ્દ          | વ્યાખ્યા   |
|---------------|--|
| સ્ટેક         | LIFO ક્રમમાં કામચલાઉ સ્ટોરેજ માટે વપરાતી મેમરી એરિયા |
| સ્ટેક પોઈન્ટર | 16-બિટ રજિસ્ટર જે સ્ટેકમાં ટોપ આઈટમને પોઈન્ટ કરે છે  |
| PUSH          | ડેટાને સ્ટેક પર સ્ટોર કરવાનું ઓપરેશન (SP ઘટે છે)     |
| POP           | સ્ટેક પરથી ડેટા મેળવવાનું ઓપરેશન (SP વધે છે)         |



## મેમરી ટ્રીક

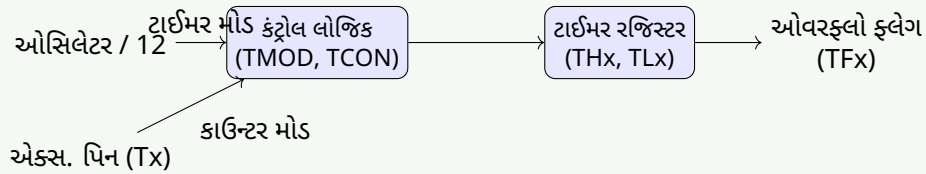
“LIFO પુશ-પોપ કરે” (છેલ્લો અંદર-પહેલો બહાર, પુશ અને પોપ ઓપરેશન સાથે)

## પ્રશ્ન 3 [b ગુણ]

4 8051 માઇક્રોકન્ટ્રોલરનો ટાઈમર્સ/કાઉન્ટર્સનો લોજિક ડાયાગ્રામ દોરો અને તેને સમજાવો.

## જવાબ

જવાબ:



- 8051 માં 2 16-બિટ ટાઈમર/કાઉન્ટર છે: ટાઈમર 0 અને ટાઈમર 1
- દરેક ટાઈમરમાં બે 8-બિટ રજિસ્ટર છે: THx (હાઈ બાઈટ) અને TLx (લો બાઈટ)
- 4 ઓપરેટિંગ મોડ્સ:
  - મોડ 0: 13-બિટ ટાઈમર
  - મોડ 1: 16-બિટ ટાઈમર
  - મોડ 2: 8-બિટ ઓટો-રિલોડ
  - મોડ 3: સ્પ્લિટ ટાઈમર મોડ
- બે ફંક્શન્સ:
  - ટાઈમર: આંતરિક કલોક પલ્સ ગણે છે
  - કાઉન્ટર: બાહ્ય ઘટનાઓની ગણતરી કરે છે

## મેમરી ટ્રીક

“TIME-C” (ટાઈમર ઈનપુટ, મોડ સિલેક્ટ, એક્સટર્નલ કાઉન્ટ)

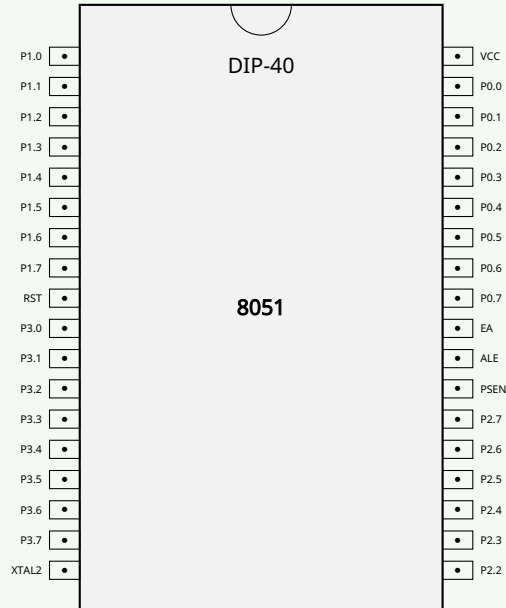


### પ્રશ્ન 3 [c ગુણ]

7 આકૃતિની મદદથી 8051 માઇક્રોકન્ટ્રોલરનો પિન ડાયાગ્રામ સમજાવો.

જવાબ

જવાબ:



પિન ગ્રુપ્સ:

1. પોર્ટ પિન્સ:

- P0 (પોર્ટ 0): 8-બિટ બિડાયરેક્શનલ, મલ્ટીપ્લેક્સડ એડ્રેસ/ડેટા
- P1 (પોર્ટ 1): 8-બિટ બિડાયરેક્શનલ I/O
- P2 (પોર્ટ 2): 8-બિટ બિડાયરેક્શનલ, હાયર એડ્રેસ બાઈટ
- P3 (પોર્ટ 3): 8-બિટ બિડાયરેક્શનલ ઓલ્ટરનેટ ફંક્શન સાથે

2. પાવર & ક્લોક: VCC, GND, XTAL1, XTAL2

3. કન્ટ્રોલ સિગ્નલ્સ:

- RST: રીસેટ ઈનપુટ
- ALE: એડ્રેસ લેય એનેબલ
- $\overline{PSEN}$ : પ્રોગ્રામ સ્ટોર એનેબલ
- $\overline{EA}$ : એક્સટર્નલ એક્સેસ

મેમરી ટ્રીક

“PORT-CAPS” (પોર્ટ્સ 0-3, ક્લોક, એડ્રેસ લેય, પ્રોગ્રામ સ્ટોર, સપ્લાય)

OR

### પ્રશ્ન 3 [a ગુણ]

3 8051 માઇક્રોકન્ટ્રોલર માટે સીરિયલ કોમ્યુનિકેશન મોડ્સ સમજાવો.

જવાબ

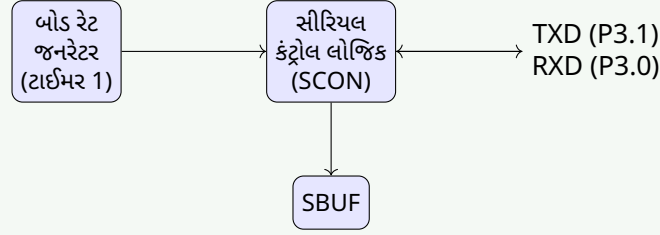
જવાબ:

કોષ્ટક 9. સીરિયલ મોડ્સ

| મોડ   | વર્ણન         | બોડ રેટ                                  | ડેટા બિટ્સ                 |
|-------|---------------|--|----------------------------|
| મોડ 0 | શિફ્ટ રજિસ્ટર | ફિક્સ્ડ ( $F_{OSC}/12$ )                 | 8 બિટ્સ                    |
| મોડ 1 | 8-બિટ UART    | વેરિએબલ                                  | 10 બિટ્સ (8+સ્ટાર્ટ+સ્ટોપ) |
| મોડ 2 | 9-બિટ UART    | ફિક્સ્ડ ( $F_{OSC}/32$ or $F_{OSC}/64$ ) | 11 બિટ્સ (9+સ્ટાર્ટ+સ્ટોપ) |
| મોડ 3 | 9-બિટ UART    | વેરિએબલ                                  | 11 બિટ્સ (9+સ્ટાર્ટ+સ્ટોપ) |

મુખ્ય ઘટકો:

- SBUF: સીરિયલ બફર રજિસ્ટર
- SCON: સીરિયલ કંટ્રોલ રજિસ્ટર
- P3.0 (RXD): રિસીવ પિન
- P3.1 (TXD): ટ્રાન્સમિટ પિન



મેમરી ટ્રીક

“SMART” (સીરિયલ મોડ્સ આર રેટ એન્ડ ટાઈમિંગ પર આધારીત)

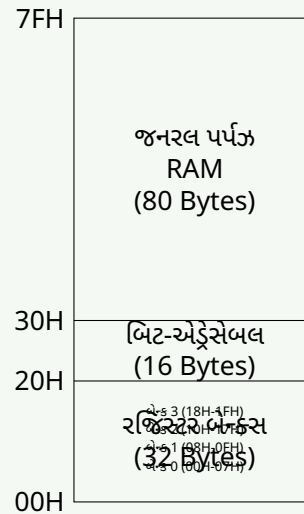
OR

### પ્રશ્ન 3 [b ગુણ]

4 8051 માઇક્રોકન્ટ્રોલરનું ઈન્ટર્નલ રેમ ઓર્ગનાઈઝેશન સમજાવો.

જવાબ

જવાબ:



કોષ્ટક 10. ઈન્ટર્નલ રેમ મેપ

| મેમરી રીજન      | એડ્રેસ રેન્જ | વર્ણન  |
|-----------------|--------------|--|
| રજિસ્ટર બેન્ક્સ | 00H-1FH      | 8 રજિસ્ટર (R0-R7) ની ચાર બેન્ક (0-3)                 |
| બિટ-એડ્રેસેબલ   | 20H-2FH      | 16 બાઈટ્સ (128 બિટ્સ) વ્યક્તિગત રીતે એડ્રેસ કરી શકાય |
| જનરલ પર્પઝ      | 30H-7FH      | વેરિએબલ્સ માટે સ્ક્રેચ પેડ RAM                       |
| SFR             | 80H-FFH      | સ્પેશિયલ ફંક્શન રજિસ્ટર્સ (RAM માં નથી)              |

**મુખ્ય લક્ષણો:**

- એક સમયે ફક્ત એક રજિસ્ટર બેન્ક એક્ટિવ હોય (PSW બિટ્સ દ્વારા પસંદ કરાય)
- બિટ-એડ્રેસેબલ એરિયામાં દરેક બિટ પોતાનું એડ્રેસ ધરાવે છે (20H.0-2FH.7)
- સ્ટેક આંતરિક RAM માં ક્યાંય પણ હોઈ શકે છે

**મેમરી ટ્રીક**

“RGB-S” (રજિસ્ટર્સ, જનરલ પર્પઝ, બિટ-એડ્રેસેબલ, SFRs)

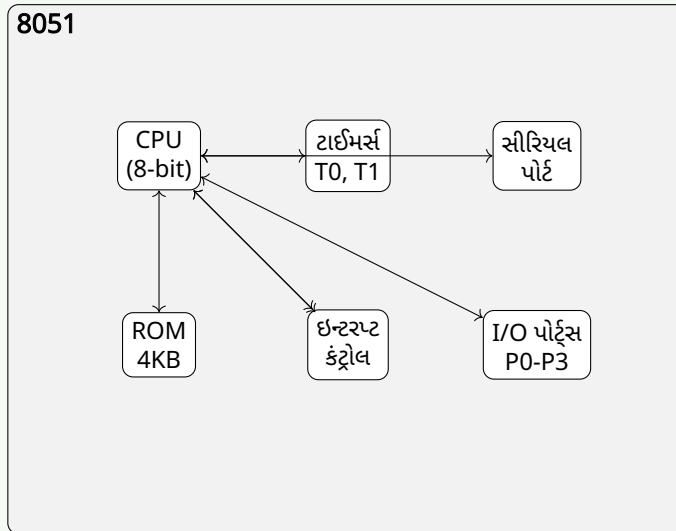
OR

**પ્રશ્ન 3 [c ગુણ]**

7 આકૃતિની મદદથી 8051 માઇક્રોકન્ટ્રોલરનું આર્કિટેક્ચર સમજાવો.

**જવાબ**

જવાબ:



એક્સટર્નલ બસ ઇન્ટરફેસ

**મુખ્ય ઘટકો:**

- CPU: ALU, રજિસ્ટર્સ અને કન્ટ્રોલ લોજિક સાથે 8-બિટ પ્રોસેસર
- મેમરી:
  - 4KB આંતરિક ROM (પ્રોગ્રામ મેમરી)
  - 128 બાઈટ્સ આંતરિક RAM (ડેટા મેમરી)
- I/O: ચાર 8-બિટ I/O પોર્ટ્સ (P0-P3)
- ટાઈમર્સ: બે 16-બિટ ટાઈમર/કાઉન્ટર
- સીરિયલ પોર્ટ: ફુલ-ડુપ્લેક્સ UART
- ઇન્ટરપ્પ્ટ્સ: બે પ્રાયોરિટી લેવલ સાથે પાંચ ઇન્ટરપ્પ્ટ સોર્સ

## મેમરી ટ્રીક

“BASICS” (બસ, આર્કિટેક્ચર વિથ CPU, સીરિયલ પોર્ટ, I/O પોર્ટ્સ, કાઉન્ટર/ટાઈમર, સ્પેશિયલ ફંક્શન્સ)

## પ્રશ્ન 4 [a ગુણ]

3 રજિસ્ટર R5 અને R6 ના લોઅર નિબલને બદલવા માટે 8051 એસેમ્બલી લેંગ્વેજ પ્રોગ્રામ લખો: R5 ના લોઅર નિબલને R6 માં અને R6 ના લોઅર નિબલને R5 માં મૂકો.

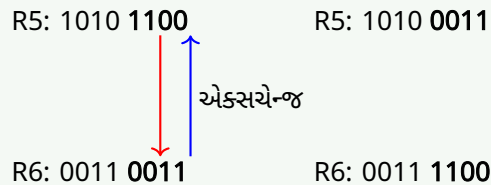
## જવાબ

## જવાબ:

```

1 ; Exchange lower nibbles of R5 and R6
2 MOV A, R5 ; Copy R5 to accumulator
3 ANL A, #0FH ; Mask upper nibble (keep only lower nibble)
4 MOV B, A ; Save R5's lower nibble in B
5
6 MOV A, R6 ; Copy R6 to accumulator
7 ANL A, #0FH ; Mask upper nibble (keep only lower nibble)
8 MOV C, A ; Save temporarily in a free register (R7)
9
10 MOV A, R5 ; Get R5 again
11 ANL A, #F0H ; Keep only upper nibble of R5
12 ORL A, C ; Combine with lower nibble from R6
13 MOV R5, A ; Store result back in R5
14
15 MOV A, R6 ; Get R6 again
16 ANL A, #F0H ; Keep only upper nibble of R6
17 ORL A, B ; Combine with lower nibble from R5
18 MOV R6, A ; Store result back in R6

```



## મેમરી ટ્રીક

“MAMS” (માસ્ક, એન્ડ, મુવ, સ્વેપ)

## પ્રશ્ન 4 [b ગુણ]

4 પોર્ટ P1.0 પર ઇન્ટરફેસ કરેલ LED ને 1ms ના સમય અંતરાલ પર બ્લિંક કરવા માટે 8051 એસેમ્બલી લેંગ્વેજ પ્રોગ્રામ લખો.

## જવાબ

## જવાબ:

```

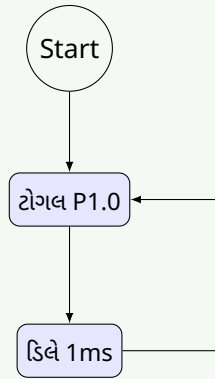
1 ORG 0000H ; Start at memory location 0000H
2 MAIN: CPL P1.0 ; Complement P1.0 (toggle LED)
3 ACALL DELAY ; Call delay subroutine
4 SJMP MAIN ; Loop forever

```

```

5 |
6 | DELAY: MOV R7, #2    ; Load R7 for outer loop (2)
7 | DELAY1: MOV R6, #250 ; Load R6 for inner loop (250)
8 | DELAY2: NOP         ; No operation (consume time)
9 |   NOP               ; Additional delay
10 |   DJNZ R6, DELAY2 ; Decrement R6 & loop until zero
11 |   DJNZ R7, DELAY1 ; Decrement R7 & loop until zero
12 |   RET               ; Return from subroutine

```



### મેમરી ટ્રીક

“TCDL” (સીરિયલ મોડ્સ આર રેટ એન્ડ ટાઈમિંગ પર આધારીત)

## પ્રશ્ન 4 [c ગુણ]

7 8051 માઈક્રોકન્ટ્રોલરના એડ્રેસિંગ મોડ્સની યાદી બનાવો અને ઓછામાં ઓછા એક ઉદાહરણ સાથે તેમને સમજાવો.

જવાબ

જવાબ:

કોષ્ટક 11. 8051 એડ્રેસિંગ મોડ્સ

| એડ્રેસિંગ મોડ       | વર્ણન                         | ઉદાહરણ   |
|---------------------|-------------------------------|--|
| રજિસ્ટર             | રજિસ્ટર્સ (R0-R7) વાપરે છે    | MOV A, R0 (R0 ને A માં મુવ કરે)                          |
| ડાયરેક્ટ            | ડાયરેક્ટ મેમરી એડ્રેસ વાપરે   | MOV A, 30H (30H પરથી ડેટા A માં મુવ કરે)                 |
| રજિસ્ટર ઇન્ડાયરેક્ટ | રજિસ્ટરને પોઇન્ટર તરીકે વાપરે | MOV A, @R0 (R0 માં રહેલા એડ્રેસ પરથી ડેટા A માં મુવ કરે) |
| ઇમીડિયેટ            | કોન્સ્ટન્ટ ડેટા વાપરે         | MOV A, #25H (A માં 25H લોડ કરે)                          |
| ઇન્ડેક્સ            | બેઝ એડ્રેસ + ઓફસેટ            | MOVC A, @A+DPTR (કોડ મેમરી એક્સેસ)                       |
| બિટ                 | વ્યક્તિગત બિટ્સ પર ઓપરેશન કરે | SETB P1.0 (પોર્ટ 1 ના બિટ 0 ને સેટ કરે)                  |
| ઇમ્પ્લોઈડ           | ઇમ્પ્લિસિટ ઓપરેન્ડ            | RRC A (A ને રાઈટ શ્રુ કેરી રોટેટ કરે)                    |

રજિસ્ટર  
MOV A, R5

R5 → A

ડાયરેક્ટ  
MOV A, 40H

Mem[40H] → A

ઇન્ડાયરેક્ટ  
MOV A, @R1

Mem[R1] → A

### મેમરી ટ્રીક

“RIDDIBM” (રજિસ્ટર, ઇમીડિયેટ, ડાયરેક્ટ, ડેટા, ઇન્ડાયરેક્ટ, બિટ, ઇમ્પ્લોઈડ)

OR

## પ્રશ્ન 4 [a ગુણ]

3 રજિસ્ટર R2 અને R3 નાં બાઈટ નો સરવાળો કરવા માટે 8051 એસેમ્બલી લેંગ્વેજ પ્રોગ્રામ લખો, પરિણામ બાહ્ય RAM માં 2040h (LSB) અને 2041h (MSB) મૂકો.

જવાબ

જવાબ:

```

1  MOV A, R2    ; Move R2 to accumulator
2  ADD A, R3    ; Add R3 to accumulator
3  MOV DPTR, #2040H ; Set DPTR to external RAM address 2040H
4  MOVX @DPTR, A ; Store the result (LSB) at 2040H
5
6  MOV A, #00H  ; Clear accumulator
7  ADDC A, #00H ; Add carry flag to accumulator
8  INC DPTR     ; Increment DPTR to 2041H
9  MOVX @DPTR, A ; Store the result (MSB) at 2041H

```

- બાહ્ય RAM એક્સેસ માટે MOVX વપરાય છે
- કેરી હેન્ડલ કરવા માટે ADDC વપરાય છે

મેમરી ટ્રીક

``MASIM" (મુવ, એડ, સ્ટોર, ઇન્ક્રિમેન્ટ, મુવ એગેન)

OR

## પ્રશ્ન 4 [b ગુણ]

4 12 MHz ની ક્રિસ્ટલ ફ્રિક્વન્સી સાથે 8051 માઇક્રોકન્ટ્રોલર માટે, 5ms નો ડિલે જનરેટ કરો.

જવાબ

જવાબ:

```

1  ; Delay of 5ms with 12MHz Crystal (1 machine cycle = 1 microsec)
2  DELAY: MOV R7, #5    ; 5 loops of 1ms each
3  LOOP1: MOV R6, #250  ; 250 x 4 microsec = 1000 microsec = 1ms
4  LOOP2: NOP           ; 1 microsec
5  NOP                 ; 1 microsec
6  DJNZ R6, LOOP2       ; 2 microsec (if jump taken)
7  DJNZ R7, LOOP1       ; Repeat 5 times for 5ms
8  RET                 ; Return from subroutine

```

ગણતરી:

- 12MHz ક્રિસ્ટલ =  $1\mu s$  મશીન સાયકલ
- ઇનર લૂપ: 2 NOPs ( $2\mu s$ ) + DJNZ ( $2\mu s$ ) =  $4\mu s$
- 250 પુનરાવર્તન  $\times 4\mu s$  =  $1000\mu s$  = 1ms
- આઉટર લૂપ: 5 પુનરાવર્તન  $\times 1ms$  = 5ms

મેમરી ટ્રીક

``LOON-5" (LOOp Nested for 5ms)

OR

## પ્રશ્ન 4 [c ગુણ]

7 8051 માઇક્રોકન્ટ્રોલર માટે કોઈ પણ સાત અર્થમેટિક સૂચનાઓ ઉદાહરણ સાથે સમજાવો.

જવાબ

જવાબ:

કોષ્ટક 12. અર્થમેટિક ઇન્સ્ટ્રક્શન્સ

| ઇન્સ્ટ્રક્શન | કાર્ય                           | ઉદાહરણ      | ફ્લેગ     |
|--------------|---------------------------------|-------------|-----------|
| ADD A,src    | સોર્સને A માં ઉમેરે છે          | ADD A,R0    | C, OV, AC |
| ADDC A,src   | સોર્સ + કેરીને A માં ઉમેરે છે   | ADDC A,#25H | C, OV, AC |
| SUBB A,src   | A માંથી સોર્સ + બોરો બાદ કરે છે | SUBB A,@R1  | C, OV, AC |
| INC          | 1 વધારો કરે છે                  | INC R3      | કોઈ નહીં  |
| DEC          | 1 ઘટાડો કરે છે                  | DEC A       | કોઈ નહીં  |
| MUL AB       | A અને B નો ગુણાકાર કરે છે       | MUL AB      | C, OV     |
| DIV AB       | A ને B વડે ભાગે છે              | DIV AB      | C, OV     |

મેમરી ટ્રીક

“ACID-IBM” (એડ, કેરી એડ, ઇન્ક્રિમેન્ટ, ડિક્રીમેન્ટ, મલ્ટીપ્લાય, બોરો સબટ્રેક્ટ, ડિવાઈડ)

## પ્રશ્ન 5 [a ગુણ]

3 વિવિધ ક્ષેત્રોમાં માઇક્રોકન્ટ્રોલરના ઉપયોગોની યાદી બનાવો.

જવાબ

જવાબ:

કોષ્ટક 13. માઇક્રોકન્ટ્રોલર એપ્લિકેશન્સ

| ક્ષેત્ર                  | ઉપયોગો   |
|--------------------------|--|
| કન્ઝ્યુમર ઇલેક્ટ્રોનિક્સ | ટીવી, વોશિંગ મશીન, માઇક્રોવેવ, રિમોટ કંટ્રોલ         |
| ઓટોમોટિવ                 | એન્જિન કંટ્રોલ, એન્ટિ-લોક બ્રેકિંગ, એરબેગ સિસ્ટમ્સ   |
| ઇન્ડસ્ટ્રીયલ             | ઓટોમેશન, રોબોટિક્સ, પ્રોસેસ કંટ્રોલ                  |
| મેડિકલ                   | પેશન્ટ મોનિટરિંગ, મેડિકલ સાધનો, ઇમ્પ્લાન્ટ્સ         |
| હોમ ઓટોમેશન              | સ્માર્ટ લાઈટિંગ, સિક્યુરિટી સિસ્ટમ્સ, HVAC કંટ્રોલ   |
| કોમ્યુનિકેશન             | મોબાઈલ ફોન, રાઉટર્સ, મોડેમ્સ                         |
| એરોસ્પેસ                 | નેવિગેશન સિસ્ટમ્સ, ફ્લાઈટ કંટ્રોલ, સેટેલાઈટ સિસ્ટમ્સ |

મેમરી ટ્રીક

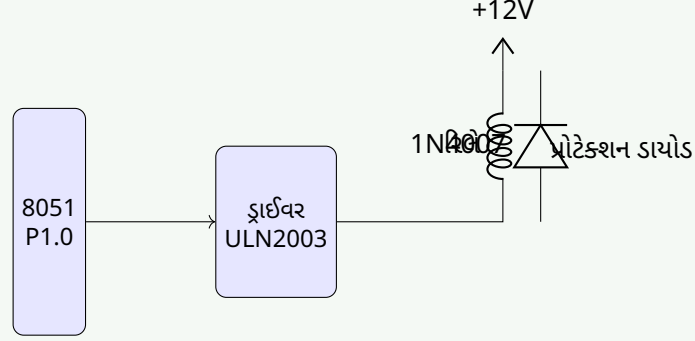
“CHAIM-MA” (કન્ઝ્યુમર, હોમ, ઓટોમોટિવ, ઇન્ડસ્ટ્રીયલ, મેડિકલ, મોબાઈલ, એરોસ્પેસ)

## પ્રશ્ન 5 [b ગુણ]

4 8051 માઇક્રોકન્ટ્રોલર સાથે રિલે ઇન્ટરફેસ કરો.

જવાબ

જવાબ:



કાર્ય:

1. 8051 P1.0 પરથી કન્ટ્રોલ સિગ્નલ મોકલે છે
2. ડ્રાઇવર રિલે ચલાવવા માટે કરંટ એમ્પ્લીફાઇ કરે છે
3. પ્રોટેક્શન ડાયોડ બેક EMF થી થતું નુકસાન અટકાવે છે
4. રિલે જોડાયેલ ઉપકરણોને સ્વિચ કરે છે

મેમરી ટ્રીક

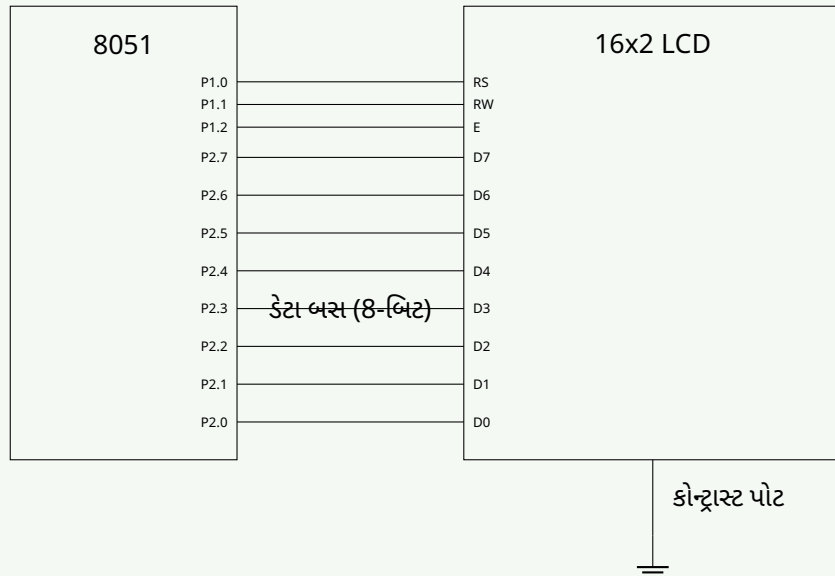
“DRIPS” (ડ્રાઇવર, રિલે, ઇનપુટ, પ્રોટેક્શન, સ્વિચિંગ)

## પ્રશ્ન 5 [c ગુણ]

7 8051 માઇક્રોકન્ટ્રોલર સાથે LCD ઇન્ટરફેસ કરો.

જવાબ

જવાબ:





**કનેક્શન:**

- કંટ્રોલ લાઈન્સ: RS (P1.0), RW (P1.1), E (P1.2)
- ડેટા લાઈન્સ: D0-D7 પોર્ટ 2 સાથે જોડાયેલ છે

**ઇનિશિયલાઇઝેશન કોડ:**

```

1  MOV A, #38H    ; 2 lines, 5x7 matrix
2  ACALL COMMAND
3  MOV A, #0EH    ; Display ON, cursor ON
4  ACALL COMMAND
5  MOV A, #01H    ; Clear LCD
6  ACALL COMMAND
7  MOV A, #06H    ; Increment cursor
8  ACALL COMMAND

```

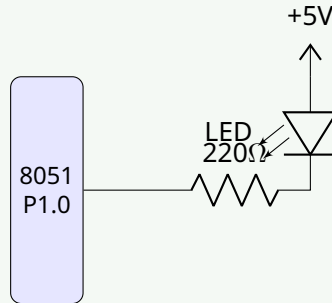
**મેમરી ટ્રીક**

“CIDER-8” (કંટ્રોલ લાઈન્સ, ઇનિશિયલાઇઝ, ડેટા બસ, એનેબલ, રજિસ્ટર સિલેક્ટ, 8-બિટ મોડ)

OR

**પ્રશ્ન 5 [a ગુણ]**

3 8051 માઇક્રોકન્ટ્રોલર સાથે LED નું ઇન્ટરફેસિંગ દોરો.

**જવાબ****જવાબ:****કાર્ય સિદ્ધાંત:**

- એક્ટિવ-લો કોન્ફિગરેશન: જ્યારે પિન = 0 ત્યારે LED ચાલુ થાય
- કરંટ લિમિટિંગ રેસિસ્ટર (220Ω) LED ને સુરક્ષિત કરે છે
- મહત્તમ કરંટ આશરે 10-20mA

**મેમરી ટ્રીક**

“CIRCLE” (કરંટ રેસિસ્ટર, IO પિન, કેથોડ થી LED, LED થી અર્થ/ગ્રાઉન્ડ)

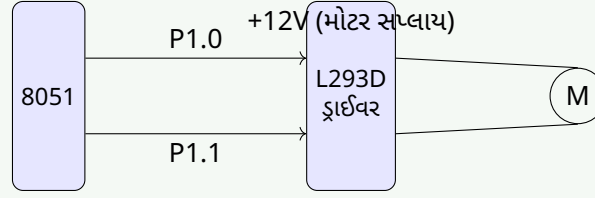
OR

**પ્રશ્ન 5 [b ગુણ]**

4 8051 માઇક્રોકન્ટ્રોલર સાથે DC મોટર ઇન્ટરફેસ કરો.

જવાબ

જવાબ:



કોષ્ટક 14. મોટર કંટ્રોલ લોજિક

| P1.0 | P1.1 | મોટર કાર્ય         |
|------|------|--------------------|
| 0    | 0    | સ્ટોપ (બ્રેક)      |
| 0    | 1    | ક્લોકવાઈઝ          |
| 1    | 0    | કાઉન્ટર-ક્લોકવાઈઝ  |
| 1    | 1    | સ્ટોપ (ફ્રી-રનિંગ) |

મેમરી ટ્રીક

“DICER” (ડ્રાઈવર ચિપ, ઇનપુટ, કંટ્રોલ લોજિક, એનેબલ મોટર, રોટેશન)

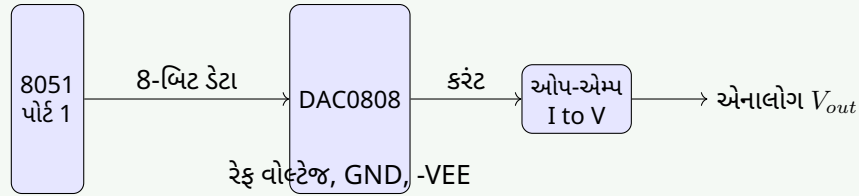
OR

## પ્રશ્ન 5 [c ગુણ]

7 8051 માઇક્રોકન્ટ્રોલર સાથે DAC0808 ઇન્ટરફેસ કરો.

જવાબ

જવાબ:



કનેક્શન્સ:

- P1.0-P1.7 → DAC ના D0-D7 ઇનપુટ્સ
- DAC આઉટપુટ કરે છે, જે ઓપ-એમ્પ દ્વારા વોલ્ટેજમાં ફેરવાય છે

ઉપયોગો: વેલ્ફોર્મ જનરેશન, મોટર સ્પીડ કંટ્રોલ

મેમરી ટ્રીક

“DACR” (ડિજિટલ ઇનપુટ, એનાલોગ આઉટપુટ, કન્વર્ઝન, રેફરન્સ વોલ્ટેજ)