

Detailed Solutions and Explanations

8051 માઇક્રોકન્ટ્રોલરના કોઈપણ એક પોર્ટ કન્ફિગરેશનનું વર્ણન કરો.

માઇક્રોપ્રોસેસર આર્કિટેક્ચરનું વર્ણન કરો.

ઘટક	કાર્ય
ALU	ગાણિતિક અને લોજિકલ ઓપરેશન કરે છે
રજિસ્ટર્સ	ડેટા અને એડ્રેસ માટે કામચલાઉ સ્ટોરેજ
કંટ્રોલ યુનિટ	પ્રોસેસર ઓપરેશન અને ડેટા ફ્લો નિર્દેશિત કરે છે
બસ	ડેટા ટ્રાન્સફર માટે પાથવે (એડ્રેસ, ડેટા, કંટ્રોલ)

[illegible]

“RABC” - “રજિસ્ટર, ALU, બસ, કંટ્રોલ”

વોન ન્યુમેન અને હાર્વર્ડ આર્કિટેક્ચરની તુલના કરો.

ફીચર	વોન ન્યુમેન આર્કિટેક્ચર	હાર્વર્ડ આર્કિટેક્ચર
મેમરી બસ	ઈન્સ્ટ્રક્શન અને ડેટા માટે એક જ મેમરી બસ	પ્રોગ્રામ અને ડેટા મેમરી માટે અલગ બસ
એક્ઝિક્યુશન સ્પીડ	સિક્વેન્શિયલ એક્ઝિક્યુશન બસ બોટલનેક ને કારણે ધીમું	પેરેલલ ફ્લો અને એક્ઝિક્યુટ શક્ય સમાંતર એક્સેસને કારણે ઝડપી
મેમરી એક્સેસ	એક જ મેમરી સ્પેસ	અલગ મેમરી સ્પેસ
જટિલતા	સરળ ડિઝાઇન	વધુ જટિલ ડિઝાઇન
ઉપયોગો	સામાન્ય કમ્પ્યુટિંગ	DSP, માઇક્રોકન્ટ્રોલર, એમ્બેડેડ સિસ્ટમ
ઉદાહરણો	મોટાભાગના PC, 8085, 8086	8051, PIC, ARM Cortex-M

Von Neumann:		Harvard:
+{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}+		+{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}+
CPU {-}{-}{-}{-}{-} Memory	CPU	{-}{-}{-}{-}{-} Program Memory }
+{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}+		+{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}+
	\^{ }	
	v	
	Data Memory	

મેમરી ટ્રીક

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

પ્રશ્ન 1(ક OR) [7 ગુણ]

RISC, CISC, Opcode, Operand, Instruction Cycle, Machine Cycle, અને T State ને વ્યાખ્યાયિત કરો.

정답

શબ્દ	વ્યાખ્યા
RISC	રિડ્યુસ્ડ ઇન્સ્ટ્રક્શન સેટ કમ્પ્યુટર - સરળ ઇન્સ્ટ્રક્શન સાથે સ્પીડ માટે ઓપ્ટિમાઇઝ્ડ આર્કિટેક્ચર
CISC	કોમ્પ્લેક્સ ઇન્સ્ટ્રક્શન સેટ કમ્પ્યુટર - જટિલ, શક્તિશાળી ઇન્સ્ટ્રક્શન સાથેનું આર્કિટેક્ચર
Opcode	ઓપરેશન કોડ - ઇન્સ્ટ્રક્શનનો ભાગ જે કયા ઓપરેશન કરવાના છે તે સ્પષ્ટ કરે છે
Operand	ઓપરેશનમાં વપરાતો ડેટા વેલ્યુ અથવા એડ્રેસ
Instruction Cycle	ઇન્સ્ટ્રક્શન ફેચ, ડિકોડ અને એક્ઝિક્યુટની સંપૂર્ણ પ્રક્રિયા
Machine Cycle	મૂળભૂત ઓપરેશન જેમ કે મેમરી રીડ/રાઈટ (ઇન્સ્ટ્રક્શન સાચકલનો ભાગ)
T-State	ટાઈમ સ્ટેટ - પ્રોસેસરમાં સમયનો સૌથી નાનો એકમ (ક્લોક પીરિયડ)

ડાયાગ્રામ:

[illegible][illegible]

મેમરી ટ્રીક

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

પ્રશ્ન 2(અ) [3 ગુણ]

ડેટા બસ, એડ્રેસ બસ અને કંટ્રોલ બસ વ્યાખ્યાયિત કરો.

장대씨

બસ પ્રકાર	વ્યાખ્યા
ડેટા બસ	બિડાયરેક્શનલ પાથવે જે માઇક્રોપ્રોસેસર અને પેરિફેરલ ડિવાઇસ વચ્ચે વાસ્તવિક ડેટા ટ્રાન્સફર કરે છે
એડ્રેસ બસ	યુનિડાયરેક્શનલ પાથવે જે એક્સેસ કરવાના મેમરી/IO ડિવાઇસ લોકેશન ધરાવે છે
કંટ્રોલ બસ	સિગ્નલ લાઈનોનો ગ્રુપ જે સિસ્ટમ ઓપરેશનને કોઓર્ડિનેટ અને સિન્ક્રોનાઇઝ કરે છે

[illegible]

મેમરી ટ્રીક

“ADC” - “એડ્રેસ લોકેશન શોધે, ડેટા માહિતી લઈ જાય, કંટ્રોલ ઓપરેશન કોઓર્ડિનેટ કરે”

મેમરી ટ્રીક

“ADC” - “એડ્રેસ લોકેશન શોધે, ડેટા માહિતી લઈ જાય, કંટ્રોલ ઓપરેશન કોઓર્ડિનેટ કરે”

પ્રશ્ન 2(બ) [4 ગુણ]

માઇક્રોપ્રોસેસર અને માઇક્રોકન્ટ્રોલરની સરખામણી કરો.

ફીચર	માઇક્રોપ્રોસેસર	માઇક્રોકન્ટ્રોલર
વ્યાપ્તિ	એકલ ચિપ પર CPU	એકલ ચિપ પર સંપૂર્ણ કમ્પ્યુટર સિસ્ટમ
મેમરી	બાહ્ય RAM/ROM જરૂરી	અંદર જ RAM/ROM
I/O પોર્ટ	મર્યાદિત અથવા ચિપ પર નથી	ચિપ પર ઘણા I/O પોર્ટ
પેરિફેરલ્સ	બાહ્ય પેરિફેરલ્સ જરૂરી	અંદર જ પેરિફેરલ્સ (ટાઇમર્સ, ADC, વગેરે)
ઉપયોગો	સામાન્ય કમ્પ્યુટિંગ, PC	એમ્બેડેડ સિસ્ટમ, IoT ડિવાઇસિસ
કિંમત	સંપૂર્ણ સિસ્ટમ માટે વધારે	ઓછી (ઓલ-ઇન-વન સોલ્યુશન)
પાવર કન્ઝમ્પશન	વધારે	ઓછું

ફીચર	માઇક્રોપ્રોસેસર	માઇક્રોકન્ટ્રોલર
વ્યાપ્તિ	એકલ ચિપ પર CPU	એકલ ચિપ પર સંપૂર્ણ કમ્પ્યુટર સિસ્ટમ
મેમરી	બાહ્ય RAM/ROM જરૂરી	અંદર જ RAM/ROM
I/O પોર્ટ	મર્યાદિત અથવા ચિપ પર નથી	ચિપ પર ઘણા I/O પોર્ટ
પેરિફેરલ્સ	બાહ્ય પેરિફેરલ્સ જરૂરી	અંદર જ પેરિફેરલ્સ (ટાઇમર્સ, ADC, વગેરે)
ઉપયોગો	સામાન્ય કમ્પ્યુટિંગ, PC	એમ્બેડેડ સિસ્ટમ, IoT ડિવાઇસિસ
કિંમત	સંપૂર્ણ સિસ્ટમ માટે વધારે	ઓછી (ઓલ-ઇન-વન સોલ્યુશન)
પાવર કન્ઝમ્પશન	વધારે	ઓછું

મેમરી ટ્રીક
“MEMI-CAP” (મેમરી બાહ્ય/આંતરિક, કિંમત, એપ્લિકેશન્સ, પેરિફેરલ્સ)

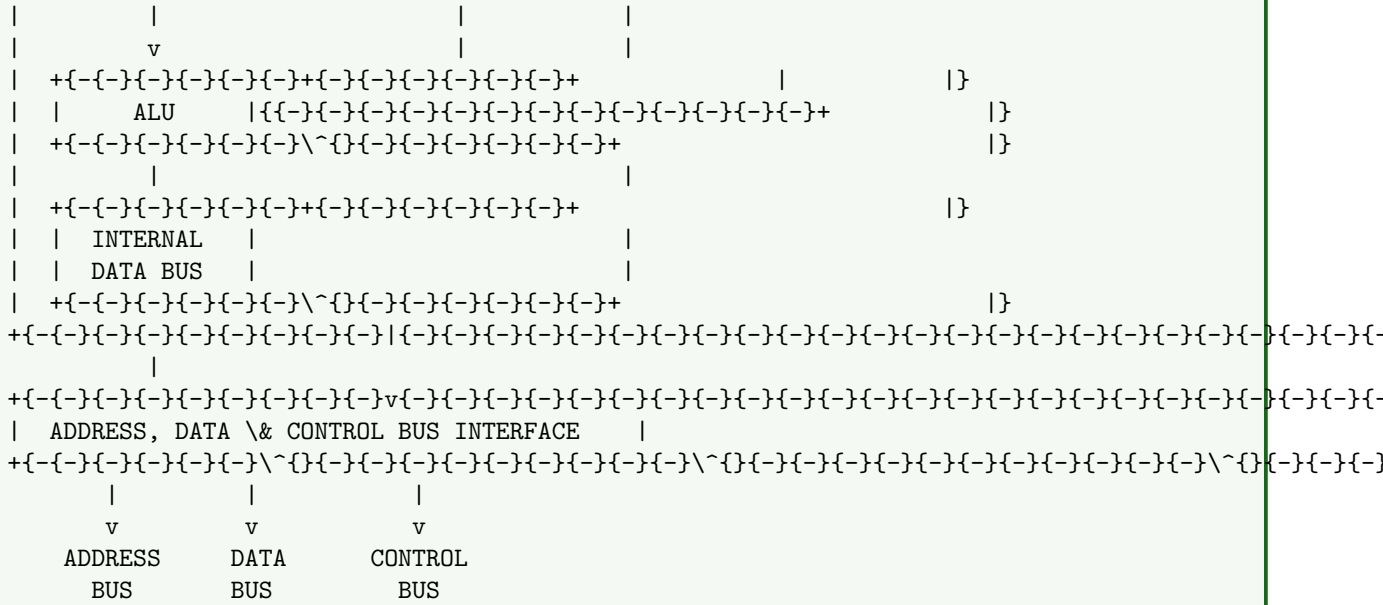
મેમરી ટ્રીક
“MEMI-CAP” (મેમરી બાહ્ય/આંતરિક, કિંમત, એપ્લિકેશન્સ, પેરિફેરલ્સ)

પ્રશ્ન 2(ક) [7 ગુણ]

8085 બ્લોક ડાયાગ્રામ સ્કેચ કરો અને સમજાવો.

```
+{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}
|
|          8085 CPU          |
|
| +{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}+
| | REGISTER      |           | TIMING \&       |   |
| | ARRAY         |           | CONTROL         |   |
| | {--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}|   | {--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}
| | A,Flags       |           | Instruction     |   |
| | B,C,D,E,H,L   | {{--}}{--}{--}{--}{--}| Decoder    |   | }
| | SP, PC        |           | Interrupt      |   |
| | W,Z(Temp)     |           | Control        |   |
| +{--}{--}{--}{--}{--}{--}\^{--}{--}{--}{--}{--}{--}{--}{--}{--}{--}+   +{--}{--}{--}{--}{--}{--}{--}\^{--}{--}{--}{--}{--}{--}{--}{--}{--}
```

[illegible][illegible]



મુખ્ય ઘટકો:

- રજિસ્ટર અરે: A (એક્યુમ્યુલેટર), ફ્લેગ્સ, B-L, SP, PC, ટેમ્પ રજિસ્ટર્સ
- ALU: ગાણિતિક અને લોજિકલ ઓપરેશન કરે છે
- ટાઈમિંગ & કંટ્રોલ: કંટ્રોલ સિગ્નલ્સ જનરેટ કરે છે, ઇન્ટરપ્ટ હેન્ડલ કરે છે
- બસ ઇન્ટરફેસ: CPU ને બાહ્ય ડિવાઇસ સાથે જોડે છે
- ઇન્ટરનલ ડેટા બસ: આંતરિક ઘટકોને જોડે છે

મેમરી ટ્રીક

“RATBI” - “રજિસ્ટર્સ, ALU, ટાઈમિંગ, બસ, ઇન્ટરફેસ”

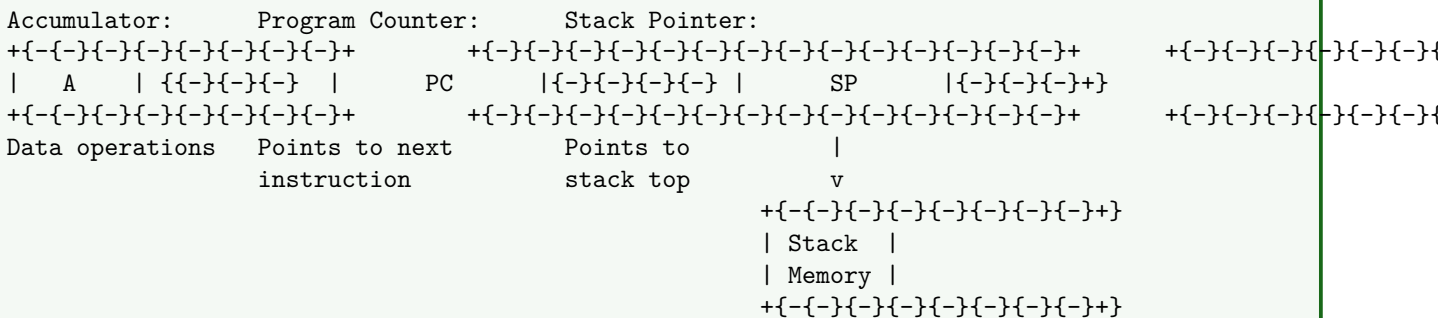
પ્રશ્ન 2(અ OR) [3 ગુણ]

એક્યુમ્યુલેટર, પ્રોગ્રામ કાઉન્ટર અને સ્ટેક પોઇન્ટર સમજાવો.

જવાબ

રજિસ્ટર	કાર્ય
એક્યુમ્યુલેટર (A)	8-બિટ રજિસ્ટર જે ગાણિતિક અને લોજિકલ ઓપરેશનના પરિણામો સ્ટોર કરે છે
પ્રોગ્રામ કાઉન્ટર (PC)	16-બિટ રજિસ્ટર જે આગલા એક્ઝિક્યુટ થનાર ઇન્સ્ટ્રક્શનનું એડ્રેસ રાખે છે
સ્ટેક પોઇન્ટર (SP)	16-બિટ રજિસ્ટર જે મેમરીમાં સ્ટેકના વર્તમાન ટોપને પોઇન્ટ કરે છે

ડાયાગ્રામ:



મેમરી ટ્રીક

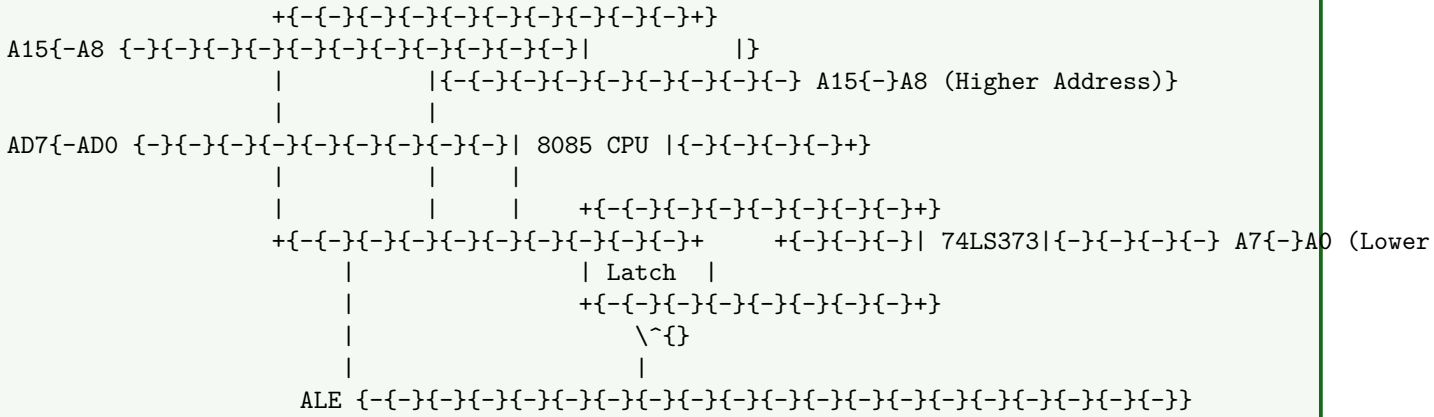
“APS” - “એક્યુમ્યુલેટર પ્રોસેસ કરે, PC આગલું ઇન્સ્ટ્રક્શન જુએ, SP સ્ટેક સંભાળે”

પ્રશ્ન 2(બ OR) [4 ગુણ]

એડ્રેસ બસ અને ડેટા બસનું ડિમલ્ટિપ્લેક્સિંગ સ્કેચ કરો અને સમજાવો.

જવાબ

ડાયાગ્રામ:



પ્રક્રિયા:

1. મલ્ટિપ્લેક્સિંગ: પિન કાઉન્ટ ઘટાડવા માટે AD0-AD7 પિન એડ્રેસ અને ડેટા સિગ્નલ શેર કરે છે
2. ડિમલ્ટિપ્લેક્સિંગના સ્ટેપ્સ:
 - CPU AD0-AD7 પિન પર એડ્રેસ મૂકે છે
 - ALE (એડ્રેસ લેય એનેબલ) સિગ્નલ HIGH થાય છે
 - બાહ્ય લેય (74LS373) લોઅર એડ્રેસ બિટ્સ પકડે છે
 - ALE LOW થાય છે, એડ્રેસ લેય થઈ જાય છે
 - AD0-AD7 પિન હવે ડેટા લઈ જાય છે

મેમરી ટ્રીક

“ALAD” - “ALE એક્ટિવ, લેય એડ્રેસ, આફ્ટર ડેટા”

પ્રશ્ન 2(ક OR) [7 ગુણ]

8085 ની કોઈપણ સાત વિશેષતાઓની યાદી આપો.

જવાબ

વિશેષતા	વર્ણન
8-બિટ ડેટા બસ	8 બિટ્સ ડેટા પેરેલલમાં ટ્રાન્સફર કરે છે
16-બિટ એડ્રેસ બસ	64KB સુધીની મેમરી એડ્રેસ કરી શકે છે (2^{16})
હાર્ડવેર ઇન્ટરપ્ટ	5 હાર્ડવેર ઇન્ટરપ્ટ (TRAP, RST 7.5, 6.5, 5.5, INTR)
સિરિયલ I/O	સિરિયલ કમ્યુનિકેશન માટે SID અને SOD પિન
ક્લોક જનરેશન	ક્લોક સાથે ઓન-ચિપ ક્લોક જનરેટર
ઇન્સ્ટ્રક્શન સેટ	74 ઓપરેશન કોડ્સ જે 246 ઇન્સ્ટ્રક્શન જનરેટ કરે છે
રજિસ્ટર સેટ	છ 8-બિટ રજિસ્ટર (B, C, D, E, H, L), એક્યુમ્યુલેટર, ફ્લેગ્સ, SP, PC

[illegible]

“CHAIRS” - “ક્લોક, હાર્ડવેર ઇન્ટરપ્રટ, એક્ઝેસ બસ, ઇન્સ્ટ્રક્શન સેટ, રજિસ્ટર્સ, સિરિયલ I/O”

8051 ના કોઈપણ એક ટાઈમર મોડને સમજાવો.

મોડ 1: 16-બિટ ટાઈમર/કાઉન્ટર

સાધાગ્રામ:

“MOGC” - “મોડ 1 ઓવરફ્લો ડિટેક્શન, ગેટ કંટ્રોલ, કમ્પ્લીટ 16-બિટ”

પ્રશ્ન 3(બ) [4 ગુણ]

8051 માટે ALE, PSEN, RESET અને TXD પિનનું ફંક્શન લખો.

જવાબ

પિન	ફંક્શન
ALE	એડ્રેસ લેય એનેબલ - પોર્ટ 0 માંથી એડ્રેસનો લો બાઈટ લેય કરવા માટે કંટ્રોલ સિગ્નલ પૂરું પાડે છે
PSEN RESET	પ્રોગ્રામ સ્ટોર એનેબલ - બાહ્ય પ્રોગ્રામ મેમરી એક્સેસ માટે રીડ સ્ટ્રોબ રીસેટ ઇનપુટ - 2 મશીન સાયકલ સુધી HIGH રાખવાથી CPU ને પ્રારંભિક સ્થિતિમાં ફોર્સ કરે છે
TXD	ટ્રાન્સમિટ ડેટા - સિરિયલ ડેટા ટ્રાન્સમિશન માટે સિરિયલ પોર્ટ આઉટપુટ પિન

સાચાગ્રામ:

```
8051 Pin Functions
+{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}+
|                                     |
|           +{-}{-}{-}{-}{-}{-}{-}+          |}
| ALE {-|             |- TXD |}
|       |               |         |
| PSEN{-| 8051 |         |}
|        |         |         |
| RESET{-|            |      |}
|           +{-}{-}{-}{-}{-}{-}{-}+          |}
|                                     |
+{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}+
```

મેમરી ટ્રીક

“APTR” - “એડ્રેસ લેય, પ્રોગ્રામ સ્ટોર, ટોટલ રીસેટ, ટ્રાન્સમિટ ડેટા”

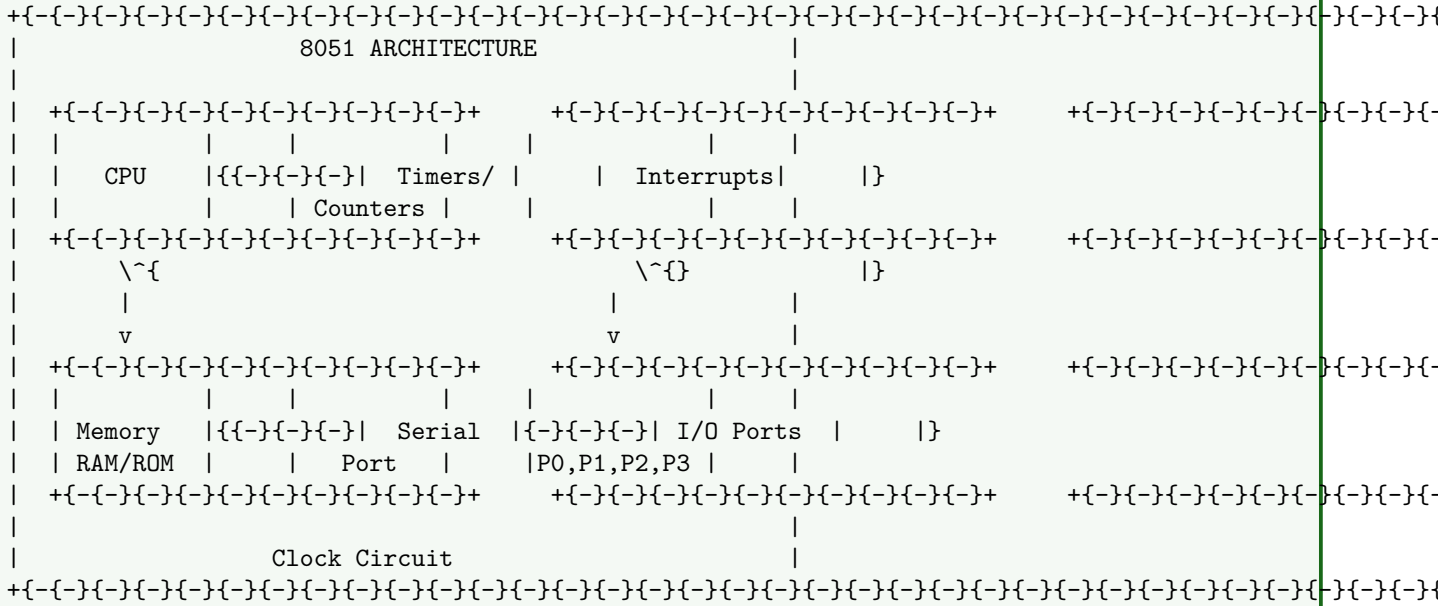
પ્રશ્ન 3(ક) [7 ગુણ]

8051 માઇક્રોકન્ટ્રોલરના દરેક બ્લોકના કાર્યો સમજાવો.

ଉଦାହରଣ

બ્લોક	કાર્ય
CPU	8-બિટ પ્રોસેસર જે ઇન્સ્ટ્રક્શન ફેચ અને એક્ઝિક્યુટ કરે છે
મેમરી	4KB ઇન્ટરનલ ROM અને 128 બાઈટ્સ ઇન્ટરનલ RAM
I/O પોર્ટ્સ	ચાર 8-બિટ બિડાયરેક્શનલ I/O પોર્ટ્સ (P0-P3)
ટાઈમર/કાઉન્ટર	ટાઈમિંગ અને કાઉન્ટિંગ માટે બે 16-બિટ ટાઈમર/કાઉન્ટર
સિરિયલ પોર્ટ	સિરિયલ કમ્યુનિકેશન માટે ફુલ-ડુપ્લેક્સ UART
ઇન્ટરપ્ટ	બે પ્રાથોરિટી લેવલ સાથે પાંચ ઇન્ટરપ્ટ સોર્સ
ક્લોક સર્કિટ	તમામ ઓપરેશન માટે ટાઈમિંગ પુરં પાડે છે

ડાયાગ્રામ:



મેમરી ટ્રીક

“CRIMSON” - “CPU, RAM/ROM, I/O, મેમરી, સિરિયલ પોર્ટ, ઓસિલેટર, ઇન્ટરપ્ટ”

પ્રશ્ન 3(અ OR) [3 ગુણ]

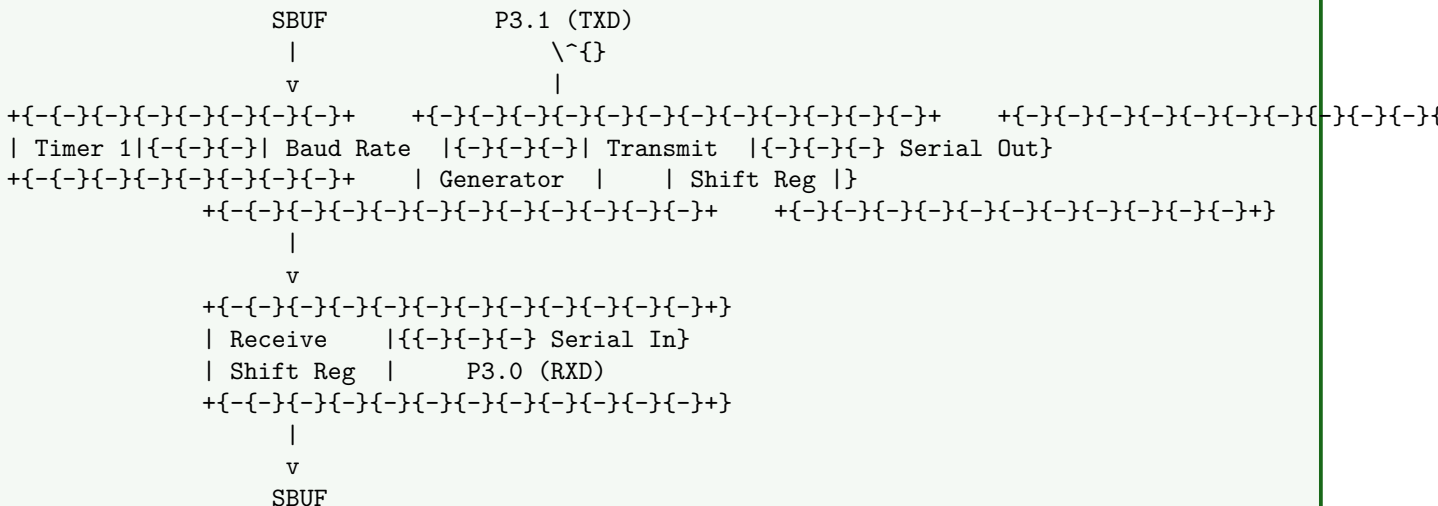
8051 ના કોઈપણ એક સીરિયલ કોમ્યુનિકેશન મોડને સમજાવો.

જવાબ

મોડ 1: 8-બિટ UART

ફીચર	વર્ણન
ફોર્મેટ	10 બિટ્સ (સ્ટાર્ટ બિટ, 8 ડેટા બિટ્સ, સ્ટોપ બિટ)
બોડ રેટ	વેરિએબલ, ટાઈમર 1 દ્વારા નક્કી થાય છે
ડેટા ડાયરેક્શન	કુલ-ડુપ્લેક્સ (એક સાથે ટ્રાન્સમિટ અને રિસીવ)
પિન્સ	ટ્રાન્સમિટ માટે TXD (P3.1), રિસીવ માટે RXD (P3.0)

ડાયાગ્રામ:



મેમરી ટ્રીક

“FADS” - “ફોર્મેટ 10-બિટ, ઓટો બોડ ટાઈમર 1 થી, ડુપ્લેક્સ મોડ, સ્ટાન્ડર્ડ UART”

પ્રશ્ન 3(બ OR) [4 ગુણ]

8051 માટે RXD, INT0, T0 અને PROG પિનનું ફંક્શન લખો.

જગીય

પિન	ફંક્શન
RXD (P3.0)	રિસીવ ડેટા - સિરિયલ ડેટા રિસેપ્શન માટે સિરિયલ પોર્ટ ઇનપુટ પિન
INT0 (P3.2)	એક્સ્ટર્નલ ઇન્ટરપ્ટ 0 - બાહ્ય ઇન્ટરપ્ટ ટ્રિગર કરી શકે તેવો ઇનપુટ
T0 (P3.4)	ટાઈમર 0 - ટાઈમર/કાઉન્ટર 0 માટે બાહ્ય કાઉન્ટ ઇનપુટ
PROG (EA)	પ્રોગ્રામ એનેબલ - જ્યારે LOW હોય, ત્યારે CPU ને બાહ્ય મેમરીમાંથી કોડ ફેચ કરવા ફોર્સ કરે છે

ડાયાગ્રામ:

```

      8051 Pin Functions
+--[-][-][-][-][-][-][-][-][-][-][-][-][-][-][-][-][-]-+
|                                                                 |
|           +--[-][-][-][-][-]+                               |}
|   RXD {-|                |{-} PROG|}
|           |               |               |
|   INTO{-| 8051 |          |}
|           |               |               |
|   TO    {-|                |}
|           +--[-][-][-][-][-]+                               |}
|                                                                 |
+--[-][-][-][-][-][-][-][-][-][-][-][-][-][-][-][-][-]-+

```

મેમરી ટ્રીક

“RIPE” - “રિસીવ ડેટા, ઇન્ટરપ્ટ ટ્રિગર, પલ્સ કાઉન્ટિંગ, એક્સટર્નલ મેમરી”

પ્રશ્ન 3(ક OR) [7 ગુણ]

8051 માટે ALU, PC, DPTR, RS0, RS1, આંતરિક RAM અને આંતરિક ROM નું વર્ણન કરો.

જવાબ

ઘટક	વર્ણન
ALU	અર્થમેટિક લોજિક યુનિટ - ગાણિતિક અને લોજિકલ ઓપરેશન કરે છે
PC	પ્રોગ્રામ કાઉન્ટર - 16-બિટ રજિસ્ટર જે આગલી ઇન્સ્ટ્રક્શનને પોઇન્ટ કરે છે
DPTR	ડેટા પોઇન્ટર - 16-બિટ રજિસ્ટર (DPH+DPL) બાહ્ય મેમરી એડ્રેસિંગ માટે
RS0, RS1	PSW માં રજિસ્ટર બેંક સિલેક્ટ બિટ્સ - ચાર રજિસ્ટર બેંક્સમાંથી એક પસંદ કરે છે
આંતરિક RAM	128 બાઈટ્સ ઓન-ચિપ RAM (00H-7FH) વેરિએબલ્સ અને સ્ટેક માટે
આંતરિક ROM	4KB ઓન-ચિપ ROM (0000H-0FFFFH) પ્રોગ્રામ સ્ટોરેજ માટે

જવાબ

```
MOV A, \#76H ; 76H
MOV R0, \#32H ; 32H R0
ADD A, R0 ; R0 A (76H + 32H = A8H)
MOV R1, A ; R1 (A8H)
JNC DONE ;
MOV R2, \#01H ; , R2 1
DONE: NOP ;
```

ડાયાગ્રામ:

```
+[-][-][-][-][-][-]+ +[-][-][-][-][-][-]+ +[-][-][-][-][-][-][-][-]+
| 76H | + ? | 32H | = ? | A8H | + Carry Flag
+[-][-][-][-][-][-]+ +[-][-][-][-][-][-]+ +[-][-][-][-][-][-][-][-]+
```

Calculation:

```
76H = 0111 0110
+ 32H = 0011 0010
[-][-][-][-][-][-][-][-][-][-][-][-][-][-][-][-]
A8H = 1010 1000
```

મેમરી ટ્રીક

"LASER" - "લોડ A, સ્ટોર સેકન્ડ નંબર, એક્ઝિક્યુટ એડિશન, રિઝલ્ટ સ્ટોર"

પ્રશ્ન 4(ક) [7 ગુણ]

એડ્રેસિંગ મોડ શું છે? તેને 8051 માટે વર્ગીકૃત કરો.

જવાબ

એડ્રેસિંગ મોડ: ઇન્સ્ટ્રક્શન માટે ઓપરેન્ડ/ડેટાનું સ્થાન સ્પષ્ટ કરવાની પદ્ધતિ.

એડ્રેસિંગ મોડ	વર્ણન	ઉદાહરણ
રજિસ્ટર	ઓપરેન્ડ રજિસ્ટરમાં	MOV A, R0 (R0 ને A માં મુવ કરે)
ડાયરેક્ટ	ઓપરેન્ડ ચોક્કસ મેમરી લોકેશન પર	MOV A, 30H (30H પરથી ડેટા A માં મુવ કરે)
રજિસ્ટર ઇન્ડાયરેક્ટ	રજિસ્ટરમાં ઓપરેન્ડનું એડ્રેસ	MOV A, @R0 (R0 માં સ્ટોર એડ્રેસ પરથી ડેટા A માં મુવ કરે)
ઈમીડિયેટ	ઓપરેન્ડ ઇન્સ્ટ્રક્શનનો ભાગ છે	MOV A, #55H (A માં 55H લોડ કરે)
ઇન્ડેક્સ	બેઝ એડ્રેસ + ઓફસેટ	MOVC A, @A+DPTR (A+DPTR પર કોડ બાઈટ મેળવે)
બિટ	વ્યક્તિગત બિટ એડ્રેસેબલ	SETB P1.0 (પોર્ટ 1 ના બિટ 0 ને સેટ કરે)
ઇમ્પ્લાઈડ/ઇનહેરન્ટ	ઓપરેન્ડ ઇન્સ્ટ્રક્શન દ્વારા સૂચિત	RRC A (A ને કેરી સાથે જમણી બાજુ રોટેટ કરે)

ડાયાગ્રામ:

Register	Direct	Indirect
MOV A, R5	MOV A, 40H	MOV A, @R1
A R5	A 40H	A X
		R1 = X

મેમરી ટ્રીક

"RIDDIB" - "રજિસ્ટર, ઇમીડિયેટ, ડાયરેક્ટ, ડેટા ઇન્ડાયરેક્ટ, ઇન્ડેક્સ, બિટ"

પ્રશ્ન 4(અ OR) [3 ગુણ]

08H અને 02H નો ગુણાકાર કરવા માટે એસેમ્બલી ભાષામાં પ્રોગ્રામ વિકસાવો.

જવાબ

```
MOV A, \#08H ; 08H
MOV B, \#02H ; 02H B
MUL AB ; A B (B:A = )
MOV R0, A ; {- R0 (10H)}
MOV R1, B ; {- R1 (00H)}
```

ડાયાગ્રામ:

Before MUL AB:	After MUL AB:
A: 08H	A: 10H (08H 02H = 10H)
B: 02H	B: 00H (High byte = 00H)

મેમરી ટ્રીક

"LMSR" - "લોડ નંબર્સ, મલ્ટિપ્લાય, સ્ટોર રિઝલ્ટ"

પ્રશ્ન 4(બ) [4 ગુણ]

76H માંથી 32H બાદ કરવા માટે એસેમ્બલી ભાષામાં પ્રોગ્રામ વિકસાવો.

જવાબ

```
MOV A, \#32H ; 32H
MOV R0, \#76H ; 76H R0
CLR C ; ( )
SUBB A, R0 ; A R0 (32H {- 76H = BCH})
MOV R1, A ; R1 (BCH, {-44H })
```

ડાયાગ્રામ:

32H	76H	BCH	(represents {-44H})

```
    32H = 0011 0010  
{- 76H = 0111 0110}  
{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}{-}  
    BCH = 1011 1100 (two's complement of 44H)
```

“LESS” - “લોડ ફર્સ્ટ નંબર, એનેબલ બોરો (CLR C), સબટ્રેક્ટ, સ્ટોર”

Instruction set ના પ્રકારોની સૂચિ બનાવો. કોઈપણ ત્રણને એક ઉદાહરણ સાથે સમજાવો.

ઇન્સ્ટ્રક્શન ગ્રુપ	વર્ણન	ઉદાહરણ
અર્થમેટિક	ગાણિતિક ઓપરેશન	ADD A, R0 (R0 ને A માં ઉમેરો)
લોજિકલ	લોજિકલ ઓપરેશન	ANL A, #0FH (A ને 0FH સાથે AND કરો)
ડેટા ટ્રાન્સફર	લોકેશન વચ્ચે ડેટા ખસેડો	MOV A, R7 (R7 ને A માં ખસેડો)
બ્રાન્ચ	પ્રોગ્રામ ફ્લો બદલો	JNZ LOOP (જો A શૂન્ય ન હોય તો જમ્પ કરો)
બિટ મેનિપ્યુલેશન	વ્યક્તિગત બિટ પર ઓપરેશન	SETB P1.0 (પોર્ટ 1 ના બિટ 0 ને સેટ કરો)
મશીન કંટ્રોલ	પ્રોસેસર ઓપરેશન કંટ્રોલ	NOP (કોઈ ઓપરેશન નહીં)

સમજાવેલા ઇન્સ્ટ્રક્શન્સ:

1. ડેટા ટ્રાન્સફર ઇન્સ્ટ્રક્શન્સ:

- રજિસ્ટર્સ, મેમરી, અથવા I/O પોર્ટ્સ વચ્ચે ડેટા ખસેડે છે
- ઉદાહરણ: MOV A, 30H - મેમરી લોકેશન 30H માંથી એક્યુમ્યુલેટરમાં ડેટા ખસેડે છે
- ઓપરેશન: $A \leftarrow [30H]$

2. અર્થમેટિક ઇન્સ્ટ્રક્શન્સ:

- ઉમેરવું, બાદ કરવું વગેરે જેવા ગાણિતિક ઓપરેશન કરે છે
- ઉદાહરણ: $ADD\ A, R0 - R0$ ની સામગ્રી એક્યુમ્યુલેટરમાં ઉમેરે છે
- ઓપરેશન: $A \leftarrow A + R0$

3. લોજિકલ ઇન્સ્ટ્રક્શન્સ:

- AND, OR, XOR, NOT જેવા લોજિકલ ઓપરેશન કરે છે
- ઉદાહરણ: `ANL A, #0FH` - અપર નિબલ માસ્ક કરે છે (માત્ર લોઅર નિબલ રાખે છે)
- ઓપરેશન: $A \leftarrow A \text{ AND } 0FH$

સાચાગ્રામ:

8051 Instruction Types

[illegible]

મેમરી ટ્રીક

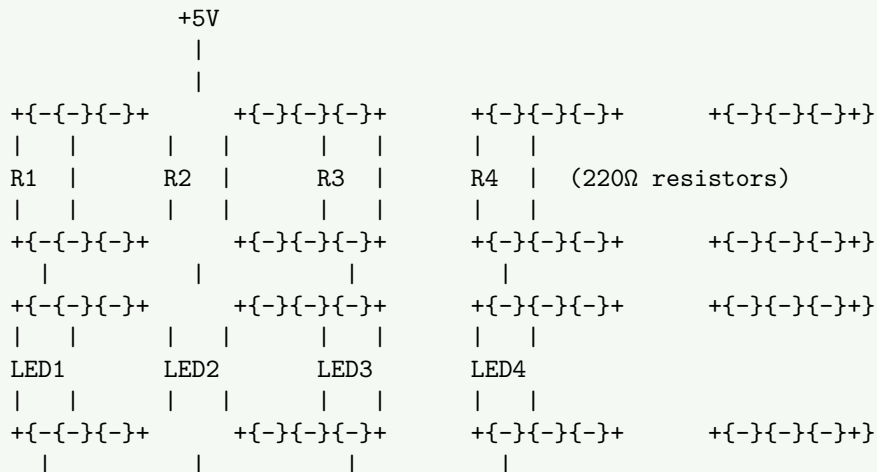
“BALDM” - “બ્રાન્ચ, અર્થમેટિક્સ, લોજિકલ, ડેટા ટ્રાન્સફર, મશીન કંટ્રોલ”

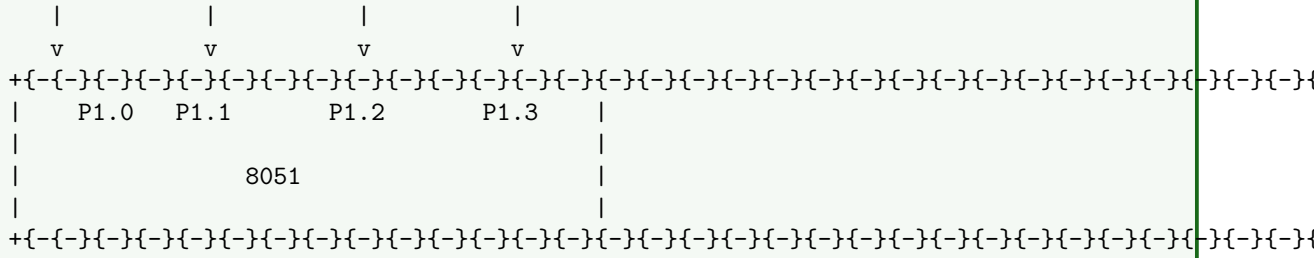
પ્રશ્ન 5(અ) [3 ગુણ]

8051 માઇક્રોકન્ટ્રોલર સાથે ચાર એલઇડીનું ઇન્ટરફેસિંગ દોરો.

ଉଦାହ

સાધાગ્રામ:





જરૂરી ઘટકો:

- 8051 માઇક્રોકન્ટ્રોલર
- ચાર LED
- ચાર કરંટ લિમિટિંગ રેસિસ્ટર (220Ω)
- પાવર સપ્લાય

મેમરી ટ્રીક

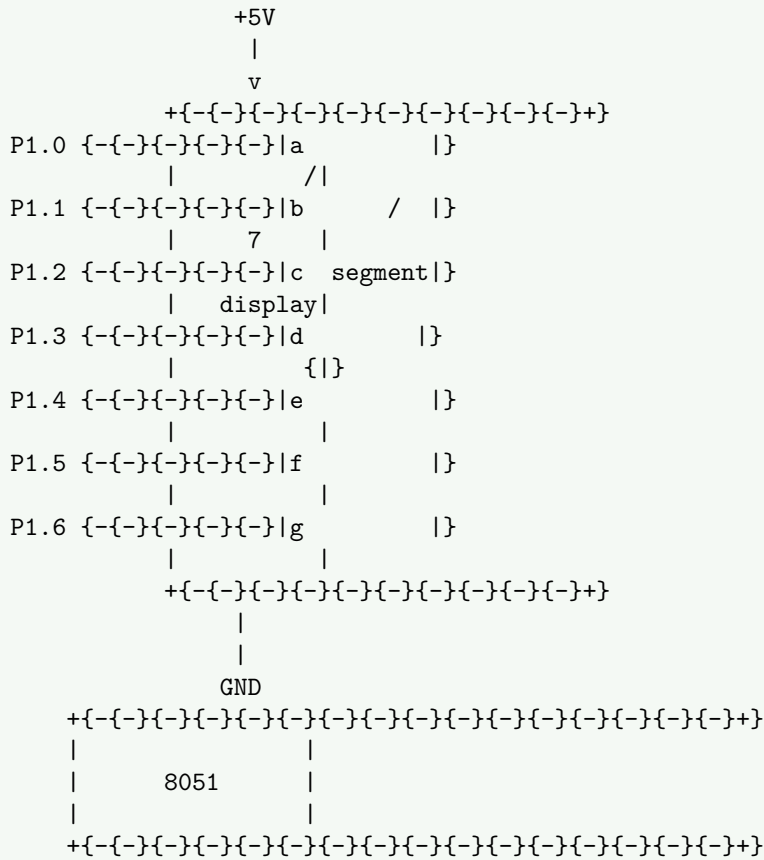
“PALS” - “પોર્ટ પિન, એક્ટિવ-લો કન્ટ્રોલ, LED, સિમ્પલ સર્કિટ”

પ્રશ્ન 5(બ) [4 ગુણ]

8051 માઇક્રોકન્ટ્રોલર સાથે 7 સેગમેન્ટ એલઇડીનું ઇન્ટરફેસિંગ દોરો.

જવાબ

ડાયાગ્રામ:



જરૂરી ઘટકો:

- 8051 માઇક્રોકન્ટ્રોલર
- 7-સેગમેન્ટ LED ડિસ્પ્લે (કોમન કેથોડ)
- સાત કરંટ લિમિટિંગ રેસિસ્ટર (નથી બતાવેલા)
- પાવર સપ્લાય

કોડ ઉદાહરણ:


```

; 0{-9
DIGITS: DB 3FH, 06H, 5BH, 4FH, 66H, 6DH, 7DH, 07H, 7FH, 6FH

; 5
MOV A, \#6DH ; 5
MOV P1, A ; P1

```

મેમરી ટ્રીક

“SPACE-7” - “સેવન પિન્સ, પેટર્ન સેગમેન્ટ, અર્થિંગ કોમન, ઇઝી ડિસ્પ્લે”

પ્રશ્ન 5(ક) [7 ગુણ]

8051 માઇક્રોકન્ટ્રોલર સાથે DAC નું ઇન્ટરફેસિંગ સમજાવો અને જરૂરી પ્રોગ્રામ લખો.

જવાબ

ડાયાગ્રામ:

```

+{-{-{-{-{-{-{-{-{-{-{-}+
|          |          |          |          |          |
|          |{-{-}P1.0{-}P1.7{-{-}|D0{-}D7   |          |
| 8051     |          |{-{-}Output{-{-}| Filter |{-{-{-} Analog}
|          |          | DAC0808|          |          Output
|          |{-{-}P3.0{-{-{-{-{-{-{-{-{-}|CS   |          |
|          |          |          |          |          |
+{-{-{-{-{-{-{-{-{-{-{-}+
|
|          |          |          |          |          |
|          |{-{-}+{-{-{-{-{-}+
|          |{-5V   |
|          | +5V   |
|          | GND   |
|          |          |
+{-{-{-{-{-{-{-{-{-{-{-}+

```

જરૂરી ઘટકો:

- 8051 માઇક્રોકન્ટ્રોલર
- DAC0808 (8-બિટ ડિજિટલ-ટુ-એનાલોગ કન્વર્ટર)
- આઉટપુટ બફરિંગ માટે ઓપરેશનલ એમ્પ્લિફાયર
- સ્મુથિંગ માટે RC ફિલ્ટર
- પાવર સપ્લાય

કનેક્શન:

- P1.0-P1.7 \rightarrow D0 – D7(8 –)
- P3.0 \rightarrow CS()

સોલ્યુશન વેવ જનરેટ કરવા માટે પ્રોગ્રામ:

```

START:  MOV R0, \#00H ; R0  0
LOOP:   MOV P1, R0    ; DAC
        CALL DELAY    ;
        INC R0         ;
        SJMP LOOP     ;

DELAY:  MOV R7, \#50   ;
DELAY1: MOV R6, \#255  ;
DELAY2: DJNZ R6, DELAY2 ; R6
        DJNZ R7, DELAY1 ; R7
        RET           ;

```

કાર્યપ્રણાલી:

1. ડિજિટલ વેલ્યુ પોર્ટ 1 પર આઉટપુટ કરવામાં આવે છે
2. DAC 8-બિટ ડિજિટલ વેલ્યુને પ્રપોર્શનલ એનાલોગ વોલ્ટેજમાં કન્વર્ટ કરે છે
3. ફિલ્ટર આઉટપુટ સિગ્નલને સ્મૂથ કરે છે
4. પ્રોગ્રામ આઉટપુટ વેલ્યુને વધારીને સોલ્યુશન વેવ બનાવે છે

મેમરી ટ્રીક

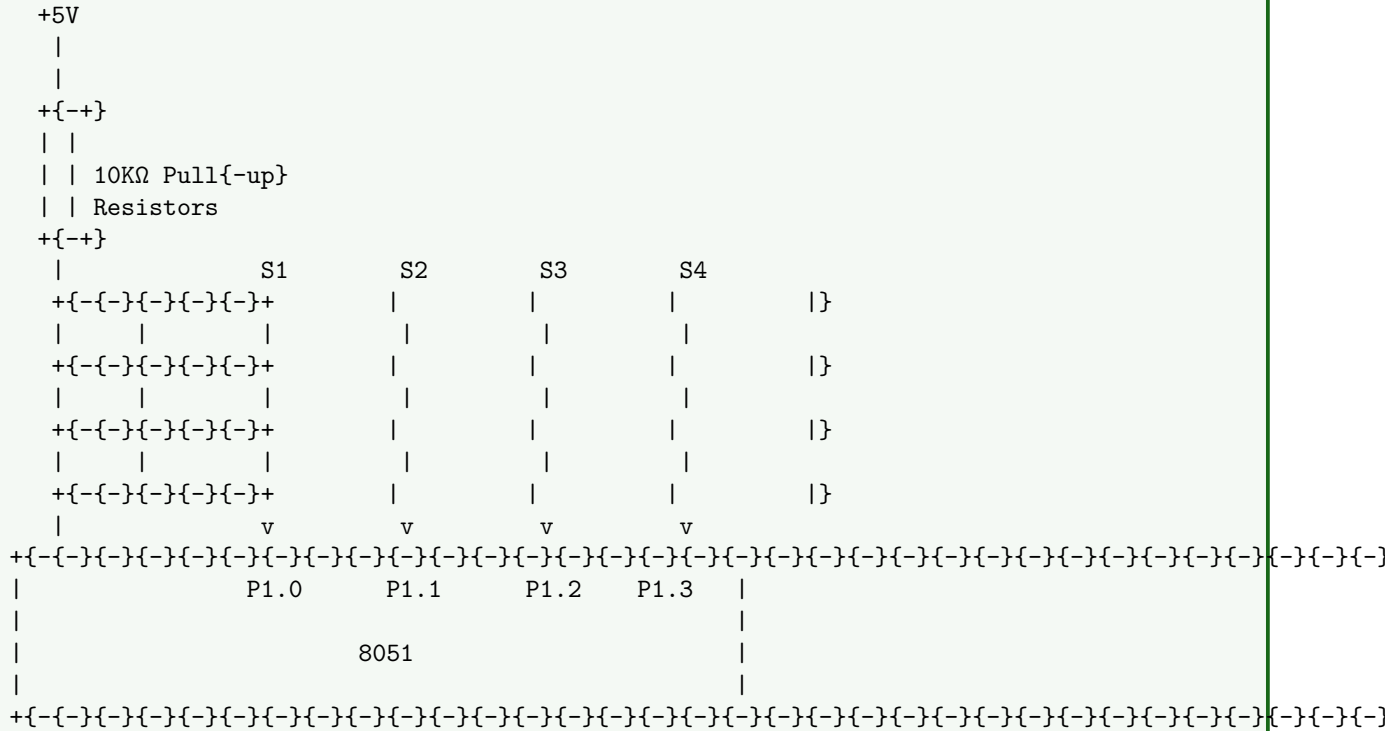
"DICAF" - "ડિજિટલ ઇનપુટ, ઇન્ક્રિમેન્ટ, કન્વર્ટ ટુ એનાલોગ, એમ્પ્લિફાય, ફિલ્ટર"

પ્રશ્ન 5(અ OR) [3 ગુણ]

8051 માઇક્રોકન્ટ્રોલર સાથે ચાર સ્વીચોનું ઇન્ટરફેસિંગ દોરો.

જવાબ

ડાયાગ્રામ:



જરૂરી ઘટકો:

- 8051 માઇક્રોકન્ટ્રોલર
- ચાર પુશ બટન (નોર્મલી ઓપન)
- પુલ-અપ રેસિસ્ટર્સ (10KΩ)
- પાવર સપ્લાય

કાર્યપ્રણાલી:

- સ્વિચ દબાવવા પર ગ્રાઉન્ડ સાથે જોડાય છે
- સ્વિચ ઓપન હોય ત્યારે પોર્ટ પિન HIGH (1) વાંચે છે
- સ્વિચ દબાવેલ હોય ત્યારે પોર્ટ પિન LOW (0) વાંચે છે

મેમરી ટ્રીક

"PIPS" - "પુલ-અપ્સ, ઇનપુટ પિન્સ, પ્રેસ ફોર ઝીરો, સ્વિચિસ"

પ્રશ્ન 5(બ) [4 ગુણ]

8051 માઇક્રોકન્ટ્રોલર સાથે સ્ટેપર મોટરનું ઇન્ટરફેસિંગ દોરો.

જવાબ

ડાયાગ્રામ:



- P3.0 → CS()
- P3.1 → RD()
- P3.2 → WR()

એનાલોગ ઇનપુટ વાંચવા માટે પ્રોગ્રામ:

```

START:  MOV P1, \#0FFH      ; P1

READ:   CLR P3.0            ; ADC      (CS = 0)
        CLR P3.2            ;          (WR = 0)
        NOP                 ;
        NOP                 ;
        SETB P3.2           ; WR = 1

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

        CLR P3.1            ;          RD = 0
        MOV A, P1           ;
        SETB P3.1           ; RD = 1
        SETB P3.0           ; ADC      (CS = 1)

PROCESS:
        ;                   ;
        ;                   : R0
        MOV R0, A

        SJMP READ          ;

```

કાર્યપ્રણાલી:

1. કંટ્રોલર સ્ટાર્ટ કન્વર્ઝન સિગ્નલ મોકલે છે
2. ADC એનાલોગ ઇનપુટને 8-બિટ ડિજિટલ વેલ્યુમાં કન્વર્ટ કરે છે
3. કંટ્રોલર કન્વર્ઝન પૂર્ણ થયા પછી ડિજિટલ વેલ્યુ વાંચે છે
4. પ્રોગ્રામ જરૂરિયાત મુજબ ડિજિટલ વેલ્યુને પ્રોસેસ કરે છે

મેમરી ટ્રીક

``CARSW" - ``કન્વર્ટ એનાલોગ, રીડ ડિજિટલ, સ્ટાર્ટ કન્વર્ઝન, વેઇટ ફોર કમ્પીશન"