

પ્રશ્ન 1(અ) [3 ગુણ]

માઇક્રોપ્રોસેસર અને માઇક્રોકન્ટ્રોલરની સરખામણી કરો.

જવાબ:

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(બ) [4 ગુણ]

RISC અને CISC ની સરખામણી કરો.

જવાબ:

ફીચર	RISC (રિડ્યુસ્ડ ઇન્સ્ટ્રક્શન સેટ કમ્પ્યુટર)	CISC (કોમ્પ્લેક્સ ઇન્સ્ટ્રક્શન સેટ કમ્પ્યુટર)
ઇન્સ્ટ્રક્શન	થોડી, સરળ ઇન્સ્ટ્રક્શન	ઘણી, જટિલ ઇન્સ્ટ્રક્શન
એક્ઝિક્યુશન ટાઈમ	ફિક્સ્ડ (1 ક્લોક 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)

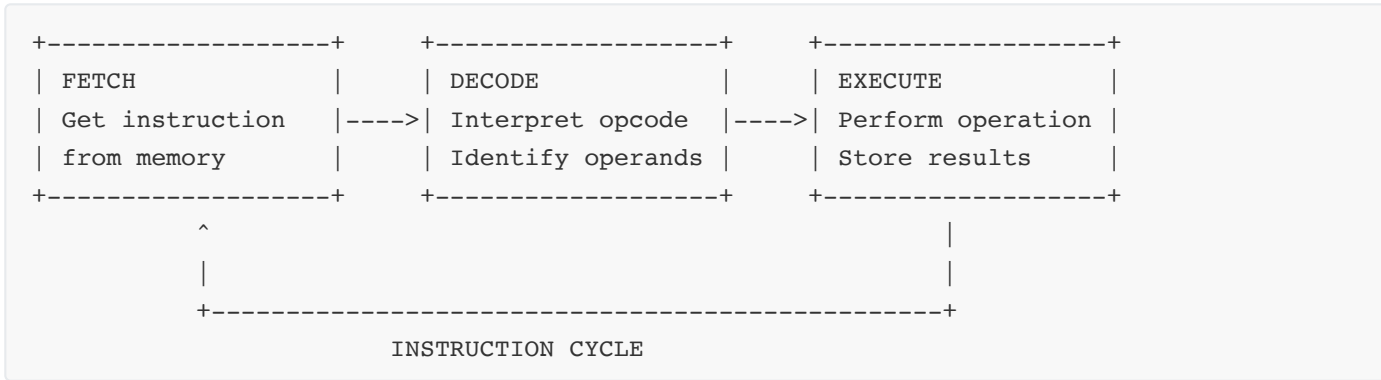
પ્રશ્ન 1(ક) [7 ગુણ]

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

જવાબ:

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)

સાચાગ્રામ:



મેમરી ટ્રીક: "મારો ઓલ્ડ Intel ચિપ ઓનલી મેક્સ ટૂબલ" (માઇક્રોપ્રોસેસર, ઓપરેન્ડ, ઇન્સ્ટ્રક્શન, ઓપકોડ, ALU, મશીન, ટી-સ્ટેટ)

પ્રશ્ન 1(ક OR) [7 ગુણ]

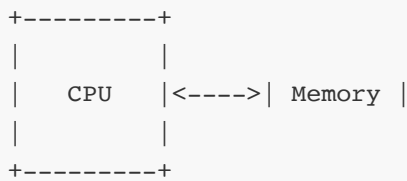
વોન-ન્યુમેન અને હાર્વર્ડ આર્કિટેક્ચરની તુલના કરો.

જવાબ:

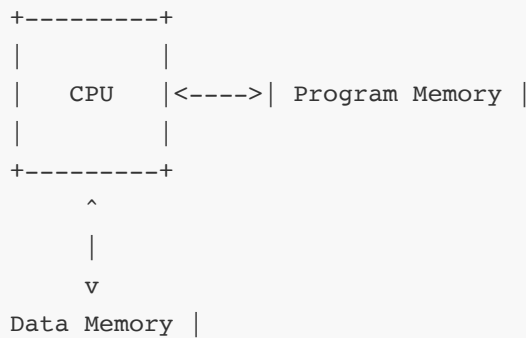
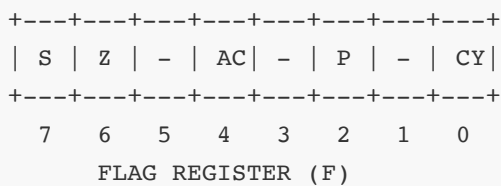
ફીચર	વોન-ન્યુમેન આર્કિટેક્ચર	હાર્વર્ડ આર્કિટેક્ચર
મેમરી બસ	ઇન્સ્ટ્રક્શન અને ડેટા માટે એક જ મેમરી બસ	પ્રોગ્રામ અને ડેટા મેમરી માટે અલગ બસ
એક્ઝિક્યુશન	સિક્વેન્શિયલ એક્ઝિક્યુશન	પેરેલલ ફેચ અને એક્ઝિક્યુટ શક્ય
સ્પીડ	બસ બોટલનેક ને કારણે ધીમું	સમાંતર એક્સેસને કારણે ઝડપી
જટિલતા	સરળ ડિઝાઇન	વધુ જટિલ ડિઝાઇન
ઉપયોગો	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

ડાયાગ્રામ:

Von-Neumann:

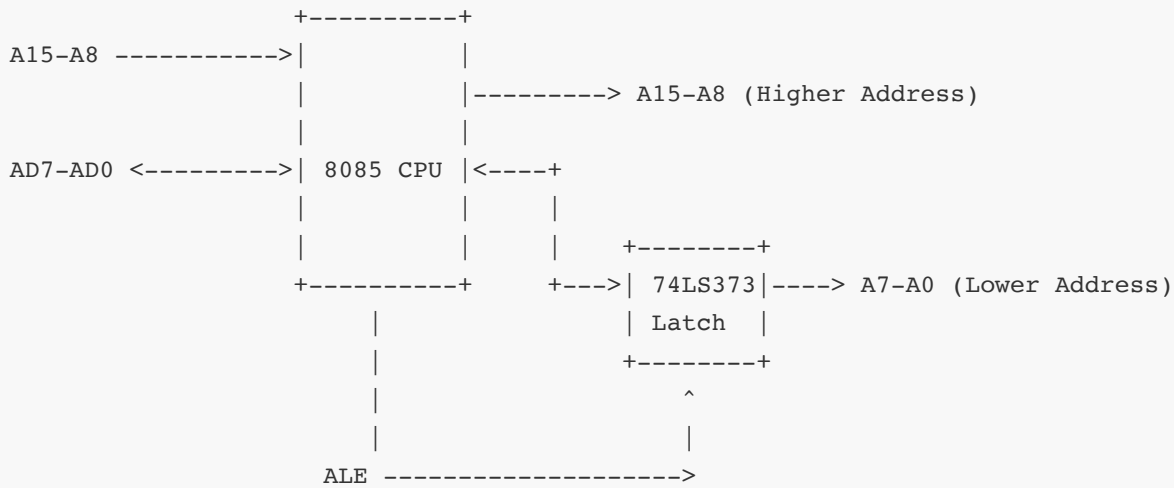


Harvard:

**મેમરી ટ્રીક:** "હાર્વર્ડ હંમેશા અલગ રસ્તા રાખે" (હાર્વર્ડમાં મેમરી પાથ અલગ હોય છે)**પ્રશ્ન 2(અ) [3 ગુણ]****8085 માઇક્રોપ્રોસેસરનું ફ્લેગ રજીસ્ટર દોરો અને તેને સમજાવો.****જવાબ:****ડાયાગ્રામ:**

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	ઓકિઝલિયરી કૅરી	જો બિટ 3 થી બિટ 4 માં કૅરી થાય તો સેટ થાય
P	પેરિટી	જો પરિણામમાં ઇવન પેરિટી હોય તો સેટ થાય
CY	કૅરી	જો બિટ 7 થી કૅરી કે બોરો થાય તો સેટ થાય

મેમરી ટ્રીક: "સરસ ઝોમ્બી આજે પણ ચાલે" (સાઈન, ઝીરો, ઓકિઝલિયરી, પેરિટી, કૅરી)**પ્રશ્ન 2(બ) [4 ગુણ]****8085 માઇક્રોપ્રોસેસર માટે એડ્રેસ અને ડેટાબસોનું ડી-મલ્ટીપ્લેક્સીંગ સમજાવો.****જવાબ:****ડાયાગ્રામ:**



- **જરૂરિયાત:** 8085 માં પિન બચાવવા માટે મલ્ટીપ્લેક્સ 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

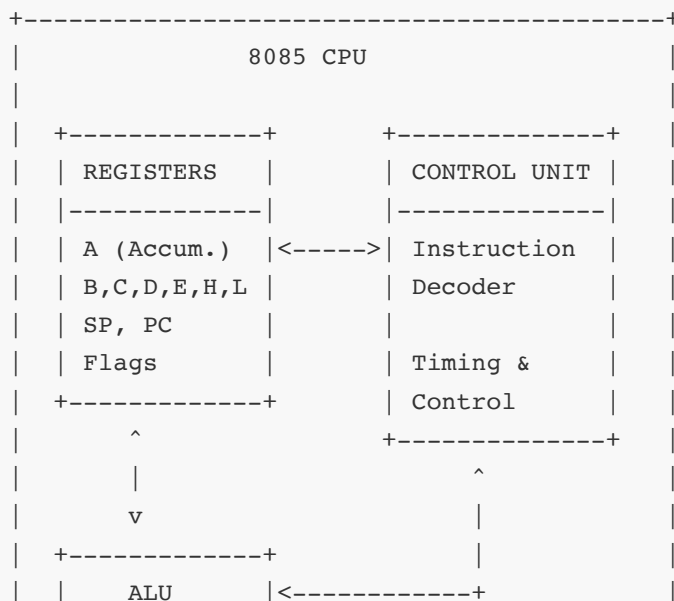
મેમરી ટ્રીક: "ALE પહેલા, ડેટા પછી" (એડ્રેસ લેય એનેબલ પહેલા એડ્રેસ પકડે, પછી ડેટા આવે)

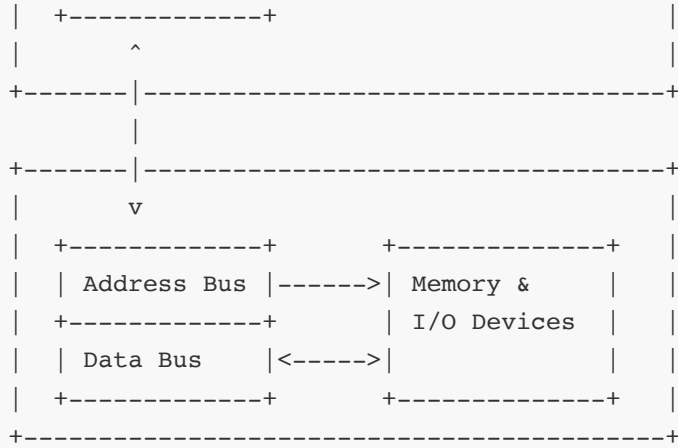
પ્રશ્ન 2(ક) [7 ગુણ]

આકૃતિની મદદથી 8085 માઇક્રોપ્રોસેસરના આર્કિટેક્ચરનું વર્ણન કરો.

જવાબ:

સાચાગ્રામ:





• 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)

પ્રશ્ન 2(અ OR) [3 ગુણ]

8085 માઇક્રોપ્રોસેસરનું બસ ઓર્ગેનાઈઝેશન સમજાવો.

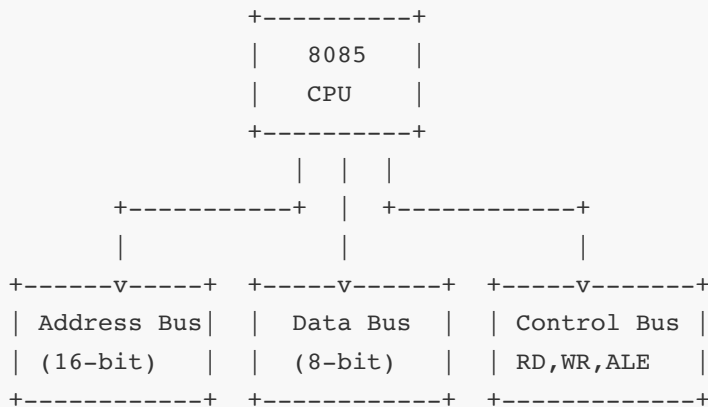
જવાબ:

બસ પ્રકાર	વિસ્તૃત	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:

- **RD:** રીડ સિગ્નલ (એક્ટિવ લો)
- **WR:** રાઈટ સિગ્નલ (એક્ટિવ લો)
- **ALE:** એડ્રેસ લેચ એનેબલ
- **IO/M:** I/O (હાઈ) અને મેમરી (લો) ઓપરેશન વચ્ચે ભેદ પાડે છે

સાચાગ્રામ:



મેમરી ટ્રીક: "ADC" - "એડ્રેસ શોધો, ડેટા ફરે, કંટ્રોલ ચલાવે"

પ્રશ્ન 2(બ OR) [4 ગુણ]

સમજાવો: પ્રોગ્રામ કાઉન્ટર અને સ્ટેક પોઈન્ટર

જવાબ:

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

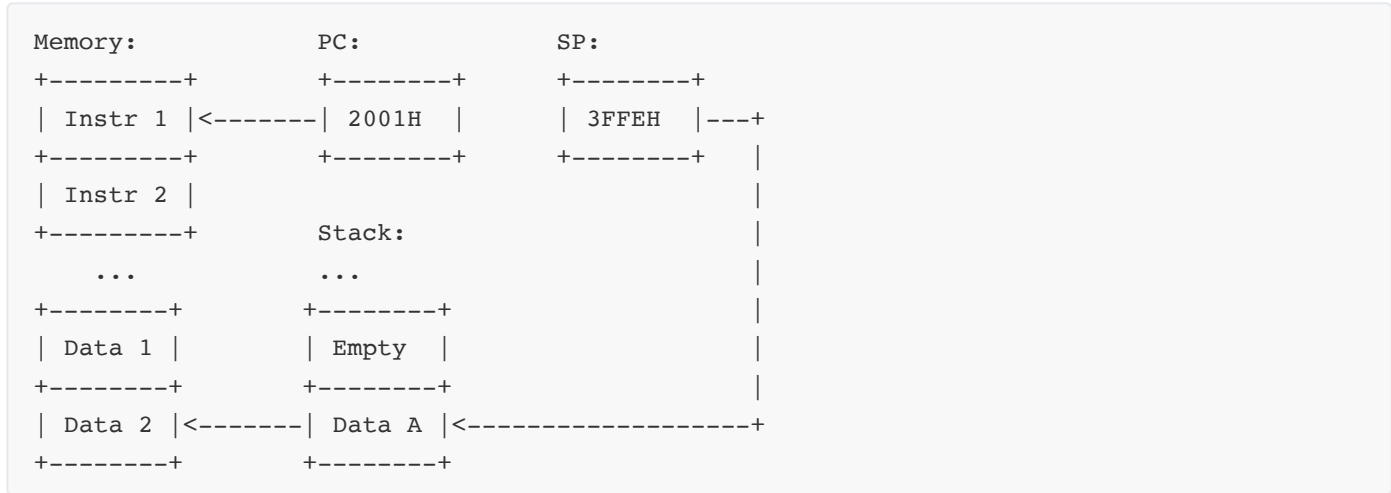
પ્રોગ્રામ કાઉન્ટર (PC):

- ઇન્સ્ટ્રક્શન ફ્રેચ 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)

સાચાગ્રામ:



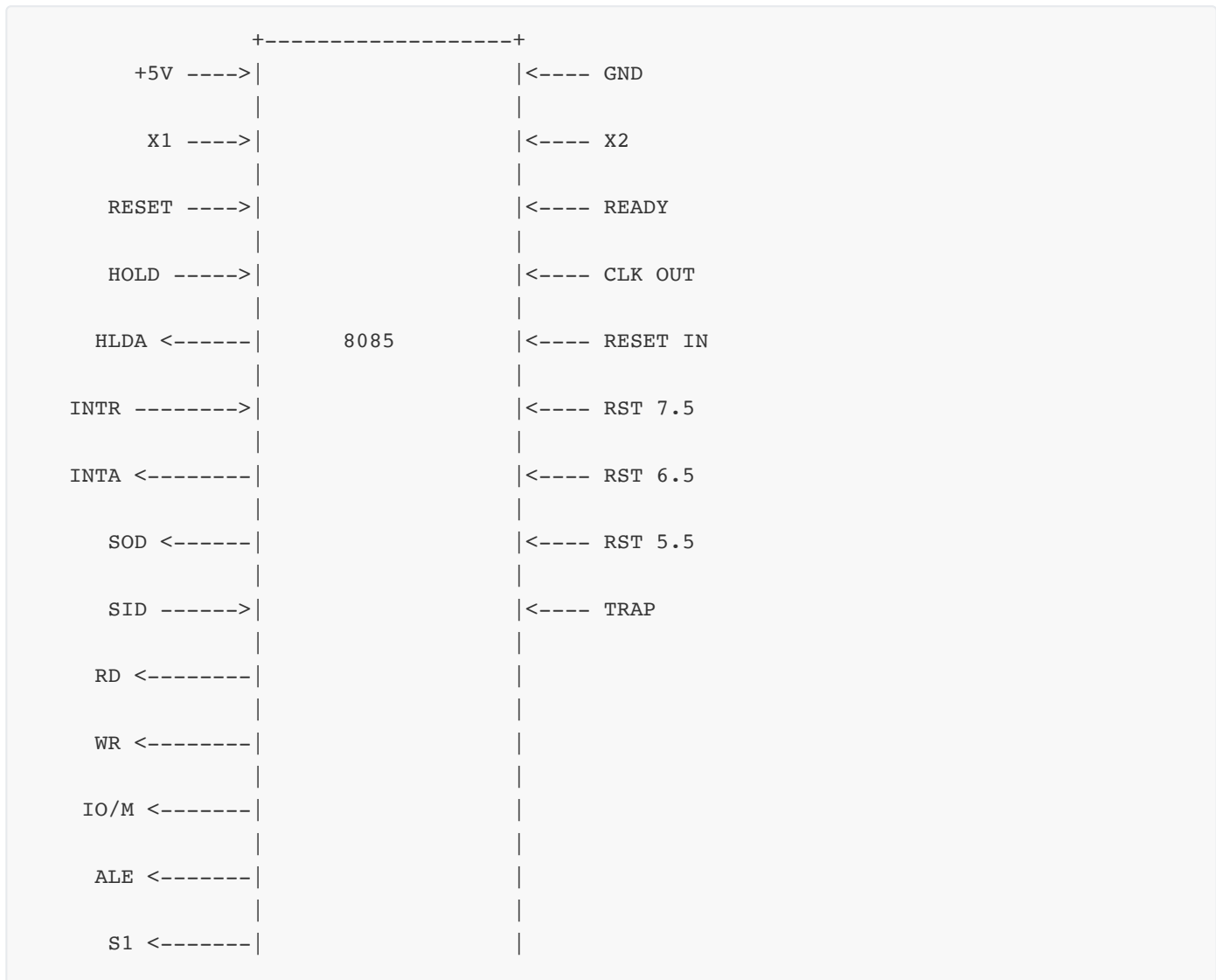
મેમરી ટ્રીક: "PC આગળ જુએ, SP સ્ટેક સંભાળે" (PC આગળું ઇન્સ્ટ્રક્શન જુએ છે, SP સ્ટેક મેનેજ કરે છે)

પ્રશ્ન 2(ક OR) [7 ગુણ]

આકૃતિની મદદથી 8085 માઇક્રોપ્રોસેસરના પિન ડાયાગ્રામનું વર્ણન કરો.

જવાબ:

ડાયાગ્રામ:





પિન ગ્રુપ્સ:

1. **પાવર & ક્લોક:** Vcc, GND, X1, X2, CLK
2. **એડ્રેસ/ડેટા:** A8-A15, AD0-AD7 (મલ્ટીપ્લેક્સ)
3. **કંટ્રોલ:** ALE, RD, WR, IO/M
4. **ઇન્ટરપ્ટ:** INTR, 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 3(અ) [3 ગુણ]

સ્ટેક, સ્ટેક પોઇન્ટર અને સ્ટેક ઓપરેશન સમજાવો.

જવાબ:

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

સાચાગ્રામ:

Memory:	Stack Operations:	
Before PUSH:	PUSH B (76H):	POP H:
+-----+ <-- 2000H	+-----+	+-----+
+-----+ <-- 1FFFH	+-----+	+-----+
	76H <-- SP	<-- SP
+-----+ <-- 1FFE H	+-----+	+-----+
<-- SP		
+-----+	+-----+	+-----+

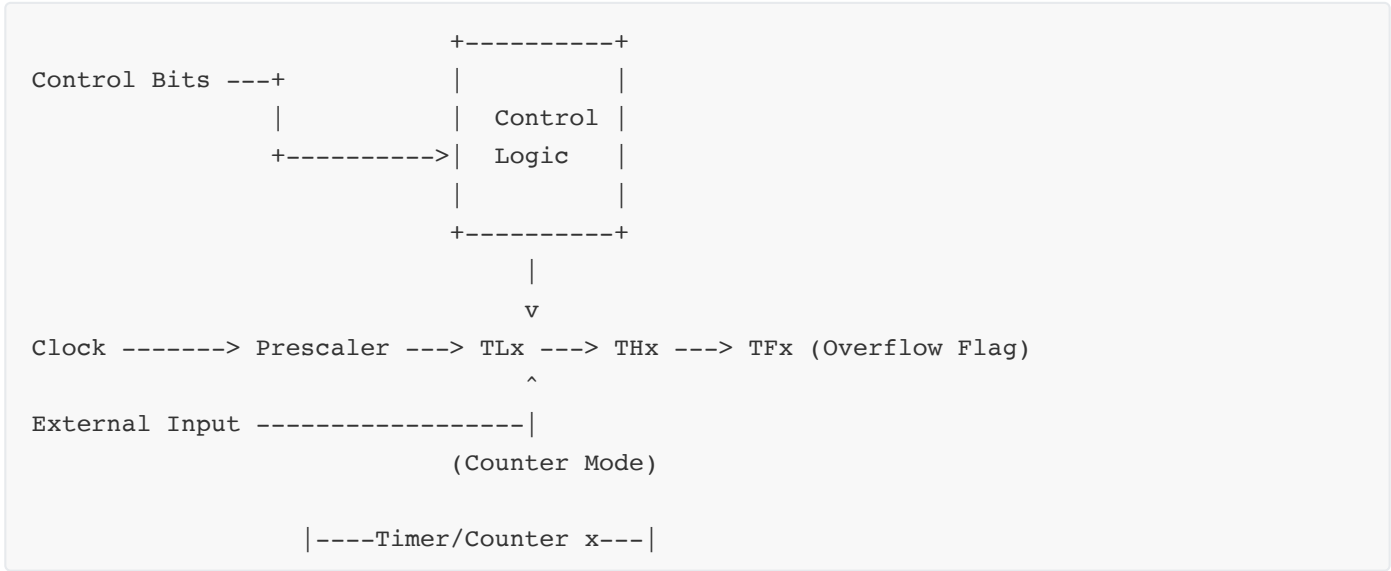
મેમરી ટ્રીક: "LIFO પુશ-પોપ કરે" (છેલ્લો અંદર-પહેલો બહાર, પુશ અને પોપ ઓપરેશન સાથે)

પ્રશ્ન 3(બ) [4 ગુણ]

8051 માઇક્રોકન્ટ્રોલરનો ટાઇમર્સ/કાઉન્ટર્સનો લોજિક ડાયાગ્રામ દોરો અને તેને સમજાવો.

જવાબ:

ડાયાગ્રામ:



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

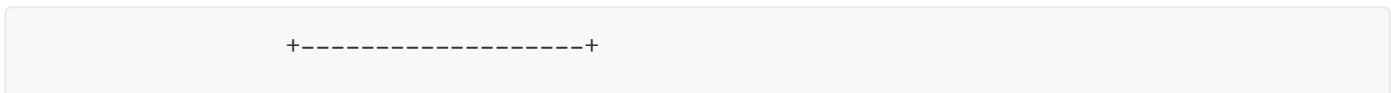
મેમરી ટ્રીક: "TIME-C" (ટાઇમર ઈનપુટ, મોડ સિલેક્ટ, એક્સટર્નલ કાઉન્ટ)

પ્રશ્ન 3(ક) [7 ગુણ]

આકૃતિની મદદથી 8051 માઇક્રોકન્ટ્રોલરનો પિન ડાયાગ્રામ સમજાવો.

જવાબ:

ડાયાગ્રામ:



VCC ---->		<----- GND
P1.0 <-->		<--> P3.0 (RXD)
P1.1 <-->		<--> P3.1 (TXD)
P1.2 <-->		<--> P3.2 (INT0)
P1.3 <-->		<--> P3.3 (INT1)
P1.4 <-->		<--> P3.4 (T0)
P1.5 <-->		<--> P3.5 (T1)
P1.6 <-->	8051	<--> P3.6 (WR)
P1.7 <-->		<--> P3.7 (RD)
RST ---->		<----- XTAL2
P2.0 <-->		<----- XTAL1
P2.1 <-->		<--> P0.0 (AD0)
P2.2 <-->		<--> P0.1 (AD1)
P2.3 <-->		<--> P0.2 (AD2)
P2.4 <-->		<--> P0.3 (AD3)
P2.5 <-->		<--> P0.4 (AD4)
P2.6 <-->		<--> P0.5 (AD5)
P2.7 <-->		<--> P0.6 (AD6)
		<--> P0.7 (AD7)
PSEN ---->		
ALE ---->		<----- EA
+-----+		

પિન ગ્રુપ્સ:

1. પોર્ટ પિન્સ:

- P0 (પોર્ટ 0): 8-બિટ બિડાયરેક્શનલ, મલ્ટીપ્લેક્સ 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. કંટ્રોલ સિગ્નલ્સ:

- RST: રીસેટ ઈનપુટ
- ALE: એડ્રેસ લેચ એનેબલ
- PSEN: પ્રોગ્રામ સ્ટોર એનેબલ
- EA: એક્સટર્નલ એક્સેસ

મેમરી ટ્રીક: "PORT-CAPS" (પોર્ટ્સ 0-3, ક્લોક, એડ્રેસ લેચ, પ્રોગ્રામ સ્ટોર, સપ્લાય)

પ્રશ્ન 3(અ OR) [3 ગુણ]

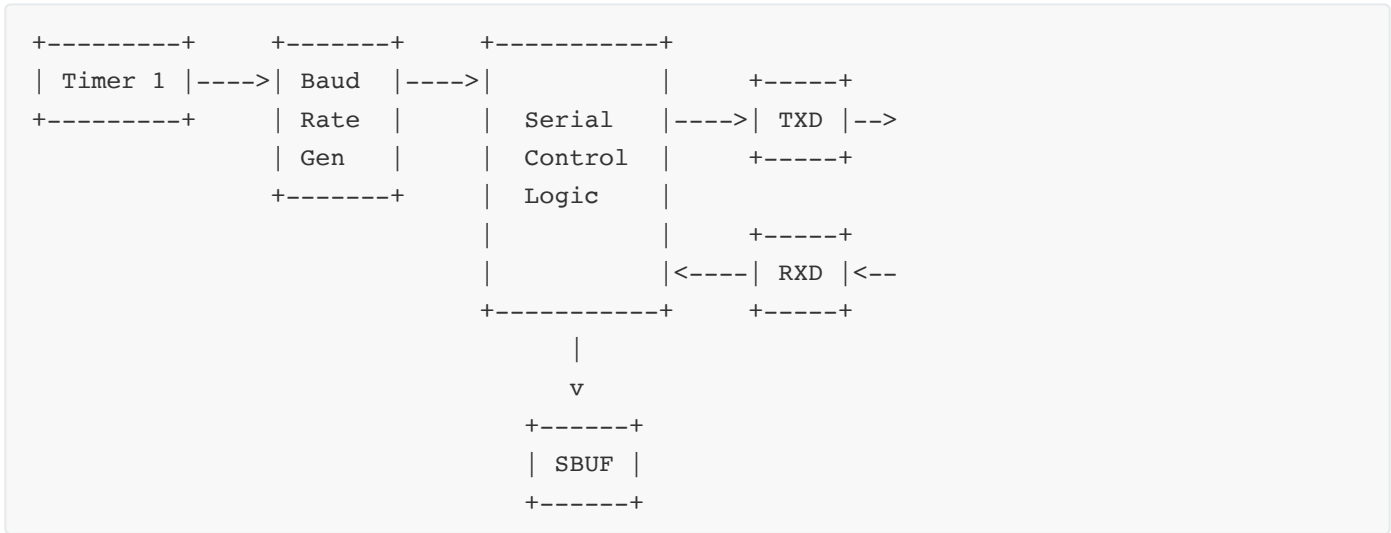
8051 માઇક્રોકન્ટ્રોલર માટે સીરિયલ કોમ્યુનિકેશન મોડ્સ સમજાવો.

મોડ	વર્ણન	મોડ રેટ	ડેટા બિટ્સ
મોડ ૦	શિફ્ટ રજિસ્ટર	ફ્રિક્વન્સી (FOSC/12)	8 બિટ્સ
મોડ ૧	8-બિટ UART	વેરિએબલ	10 બિટ્સ (8+સ્ટાર્ટ+સ્ટોપ)
મોડ ૨	9-બિટ UART	ફ્રિક્વન્સી (FOSC/32 અથવા FOSC/64)	11 બિટ્સ (9+સ્ટાર્ટ+સ્ટોપ)
મોડ ૩	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

સાચા શ્રામ:



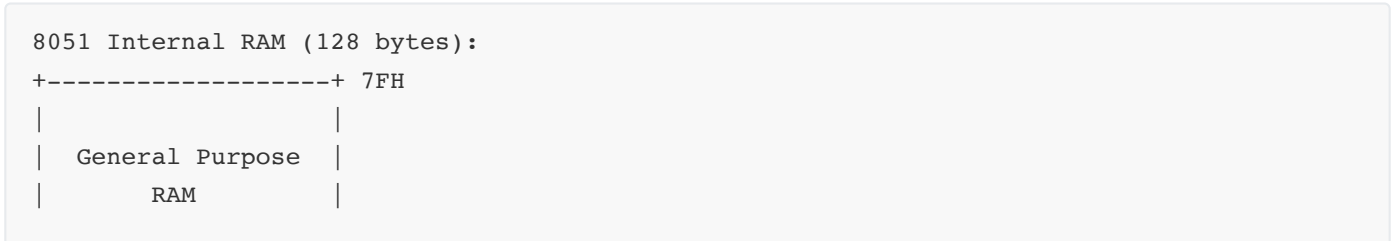
મેમ્બરી ટ્રીક: "SMART" (સીરિયલ મોડ્સ આર રેટ એન્ડ ટાઈમિંગ પર આધારીત)

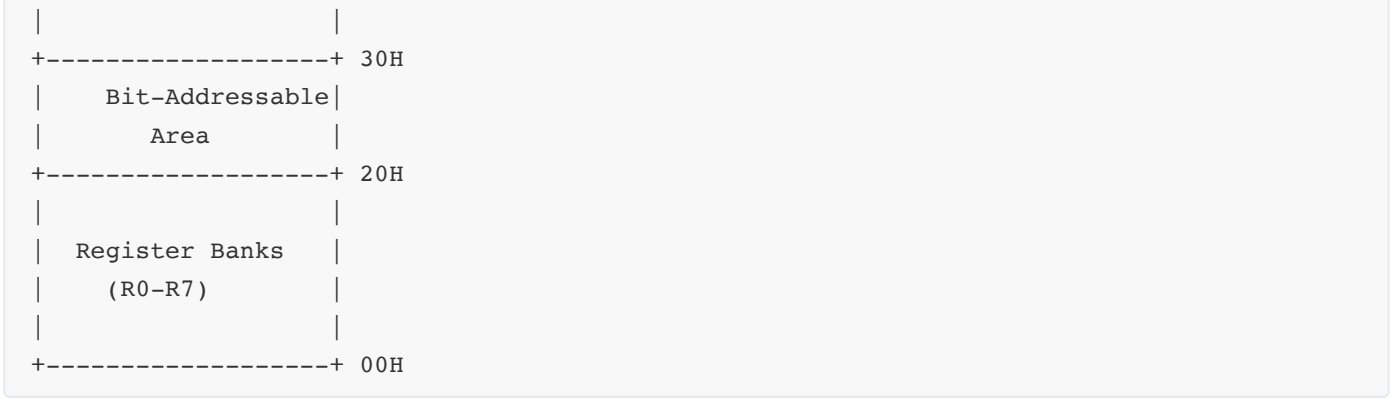
પ્રશ્ન 3(બ OR) [4 ગુણ]

8051 માઇક્રોકન્ટ્રોલરનું ઈન્ટર્નલ રેમ ઓર્ગેનાઇઝેશન સમજાવો.

જવાબ:

સાયાગ્રામ:





મેમરી રીજન	એડ્રેસ રેન્જ	વર્ણન
રજિસ્ટર બેન્ક્સ	00H-1FH	8 રજિસ્ટર (R0-R7) ની ચાર બેન્ક્સ (0-3)
બિટ-એડ્રેસેબલ	20H-2FH	16 બાઈટ્સ (128 બિટ્સ) વ્યક્તિગત રીતે એડ્રેસ કરી શકાય
જનરલ પર્પઝ	30H-7FH	વેરિએબલ્સ માટે સ્કેચ 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

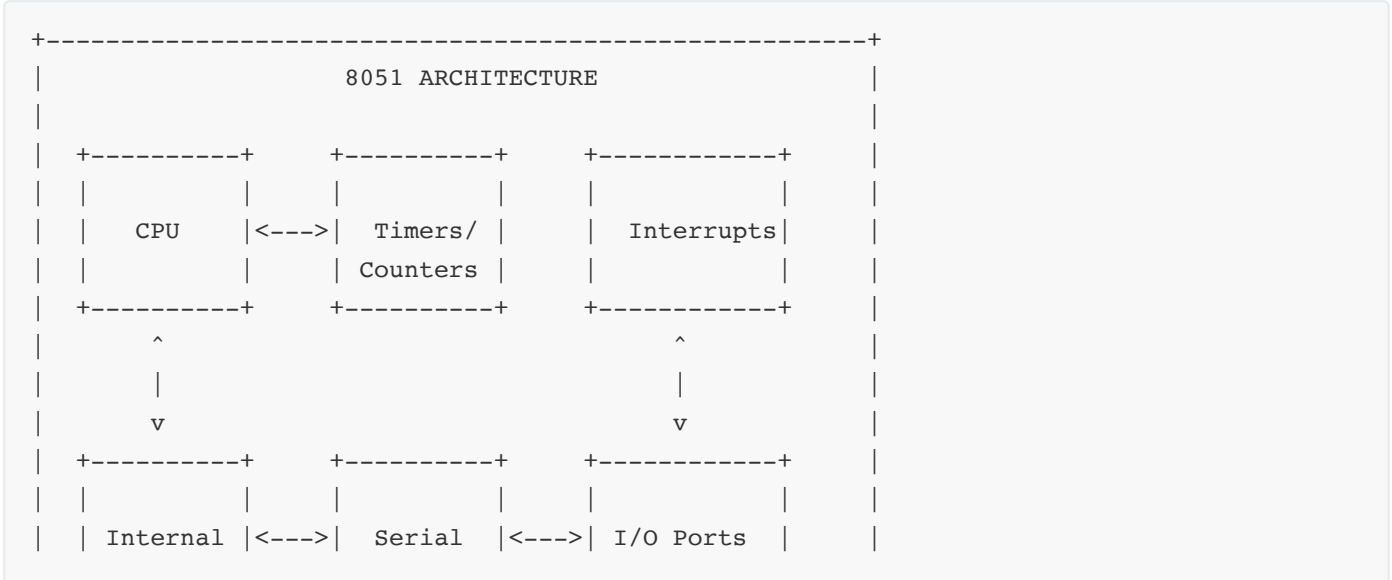
મેમરી ટ્રીક: "RGB-S" (રજિસ્ટર્સ, જનરલ પર્પઝ, બિટ-એડ્રેસેબલ, SFRs)

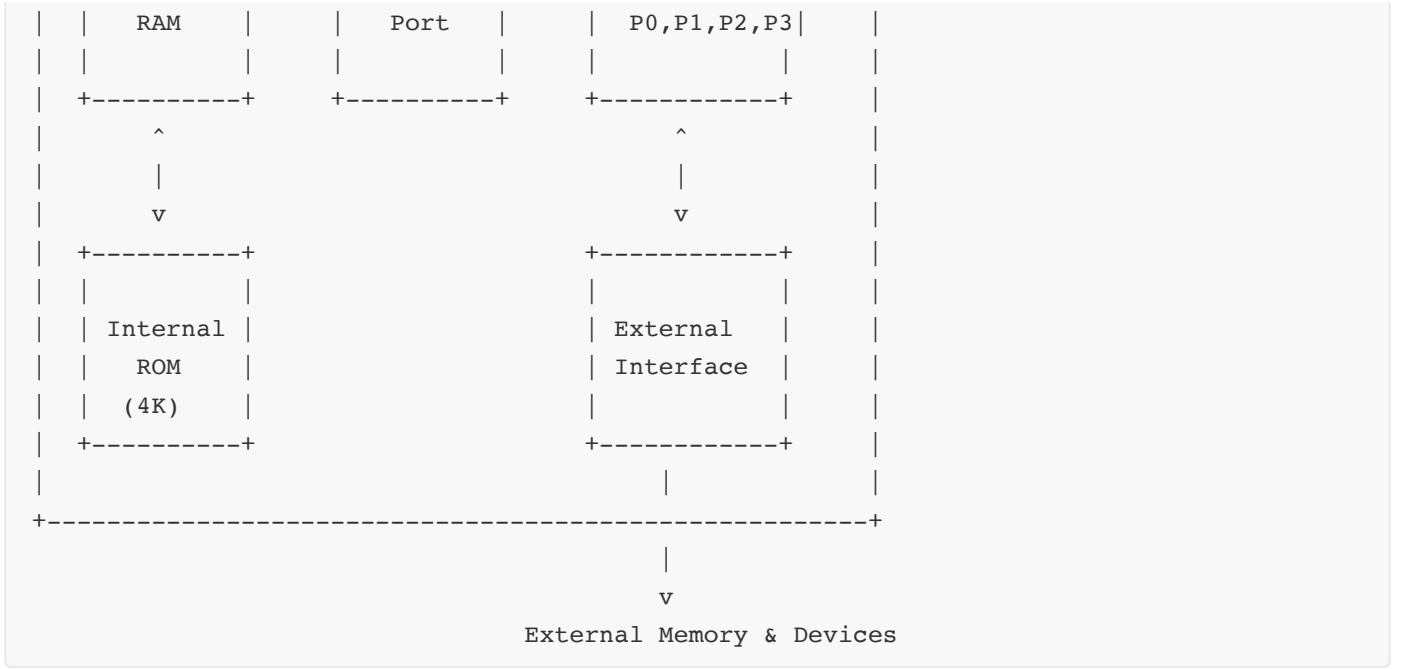
પ્રશ્ન 3(ક OR) [7 ગુણ]

આકૃતિની મદદથી 8051 માઇક્રોકંટ્રોલરનું આર્કિટેક્ચર સમજાવો.

જવાબ:

સાચાગ્રામ:





મુખ્ય ઘટકો:

- **CPU:** ALU, રજિસ્ટર્સ અને કંટ્રોલ 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
- **ઇન્ટરફેસ:** બે પ્રાયોરિટી લેવલ સાથે પાંચ ઇન્ટરફેસ સોર્સ

મેમરી ટ્રીક: "BASICS" (બસ, આર્કિટેક્ચર વિથ CPU, સીરિયલ પોર્ટ, I/O પોર્ટ્સ, કાઉન્ટર/ટાઈમર, સ્પેશિયલ ફંક્શન્સ)

પ્રશ્ન 4(અ) [3 ગુણ]

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:

```
; Exchange lower nibbles of R5 and R6
MOV A, R5      ; Copy R5 to accumulator
ANL A, #0FH    ; Mask upper nibble (keep only lower nibble)
MOV B, A       ; Save R5's lower nibble in B

MOV A, R6      ; Copy R6 to accumulator
ANL A, #0FH    ; Mask upper nibble (keep only lower nibble)
MOV C, A       ; Save temporarily in a free register (R7)

MOV A, R5      ; Get R5 again
```

```

ANL A, #F0H ; Keep only upper nibble of R5
ORL A, C    ; Combine with lower nibble from R6
MOV R5, A   ; Store result back in R5

MOV A, R6   ; Get R6 again
ANL A, #F0H ; Keep only upper nibble of R6
ORL A, B    ; Combine with lower nibble from R5
MOV R6, A   ; Store result back in R6

```

સાચાગ્રામ:

Initially:	After Exchange:
R5: 1010 1100	R5: 1010 0011
↑↑↑↑ ↑↑↑↑	↑↑↑↑ ↑↑↑↑
+--+	
+--v	v
	From R6
v v	v
R6: 0011 0011	R6: 0011 1100

મેમરી ટ્રીક: "MAMS" (માસ્ક, એન્ડ, મુવ, સ્વેપ)

પ્રશ્ન 4(બ) [4 ગુણ]

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

જવાબ:

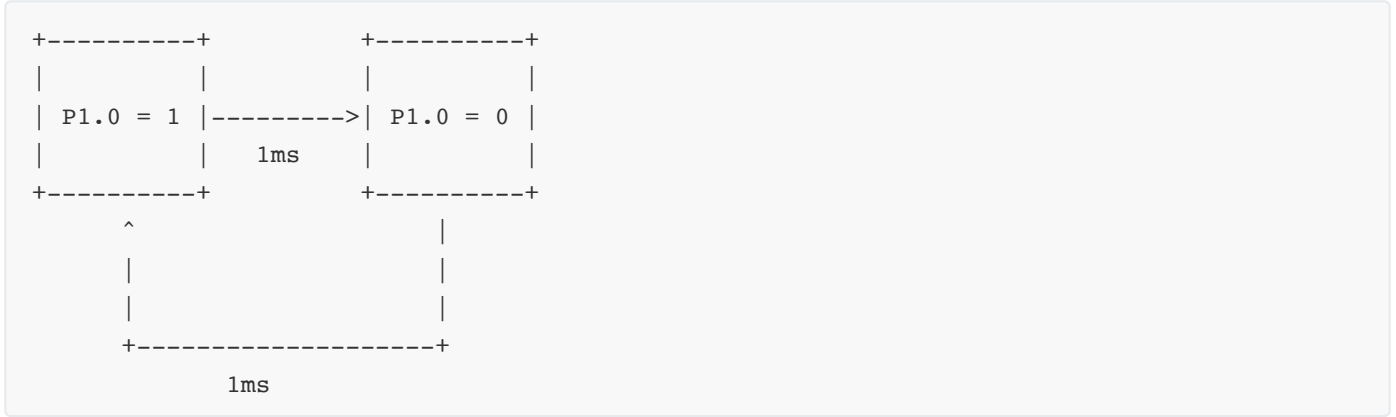
```

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

DELAY: MOV R7, #2 ; Load R7 for outer loop (2)
DELAY1: MOV R6, #250 ; Load R6 for inner loop (250)
DELAY2: NOP ; No operation (consume time)
      NOP ; Additional delay
      DJNZ R6, DELAY2 ; Decrement R6 & loop until zero
      DJNZ R7, DELAY1 ; Decrement R7 & loop until zero
      RET ; Return from subroutine

```

સાચાગ્રામ:



મેમરી ટ્રીક: "TCDL" (ટોગલ, કોલ, ડિલે, લૂપ)

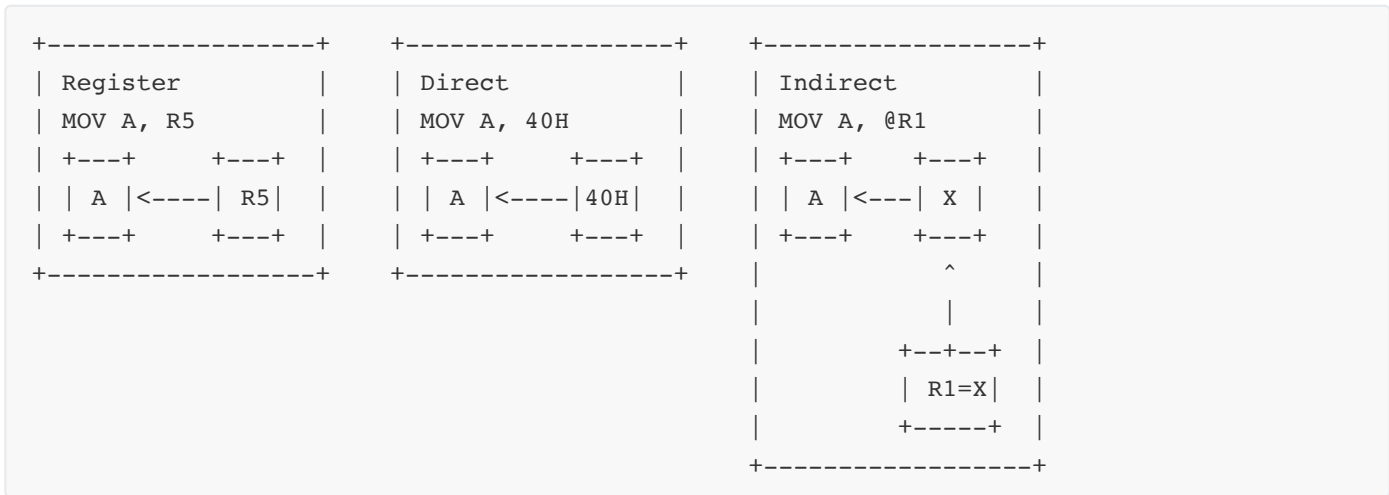
પ્રશ્ન 4(ક) [7 ગુણ]

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

જવાબ:

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

ડાયાગ્રામ:



મેમરી ટ્રીક: "RIDDIBM" (રજિસ્ટર, ઇમીડિયેટ, ડાયરેક્ટ, ડેટા, ઇન્ડાયરેક્ટ, બિટ, ઇમ્પ્લિઇડ)

પ્રશ્ન 4(અ OR) [3 ગુણ]

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:

```
MOV A, R2      ; Move R2 to accumulator
ADD A, R3      ; Add R3 to accumulator
MOV DPTR, #2040H ; Set DPTR to external RAM address 2040H
MOVX @DPTR, A  ; Store the result (LSB) at 2040H

MOV A, #00H    ; Clear accumulator
ADDC A, #00H   ; Add carry flag to accumulator
INC DPTR       ; Increment DPTR to 2041H
MOVX @DPTR, A  ; Store the result (MSB) at 2041H
```

સાચાગ્રામ:

R2	R3	External RAM
+-----+	+-----+	+-----+
25H	45H	2040H: 6A (25H + 45H = 6AH)
+-----+	+-----+	+-----+
		2041H: 00 (No carry)
v	v	+-----+
+-----+		
ADD		
+-----+		

મેમરી ટ્રીક: "MASIM" (મુવ, એડ, સ્ટોર, ઇન્ક્રિમેન્ટ, મુવ એગેન)

પ્રશ્ન 4(બ OR) [4 ગુણ]

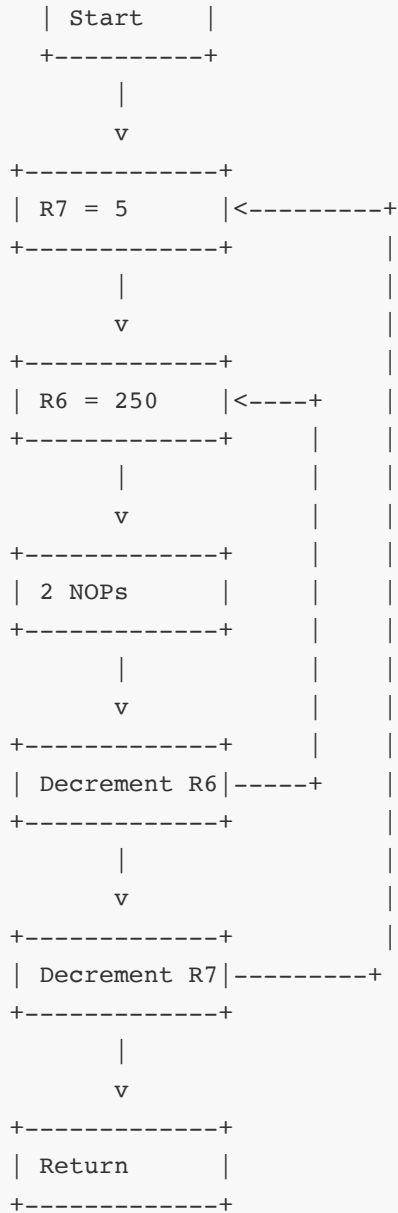
12 MHz ની ક્રિસ્ટલ ફ્રિક્વન્સી સાથે 8051 માઇક્રોકંટ્રોલર માટે, 5ms નો ડિલે જનરેટ કરો.

જવાબ:

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

સાચાગ્રામ:

+-----+



ગણતરી:

- 12MHz ક્રિસ્ટલ = 1μs મશીન સાયકલ
- ઇનર લૂપ: 2 NOPs (2μs) + DJNZ (2μs) = 4μs પ્રતિ ઇટરેશન
- 250 ઇટરેશન × 4μs = 1000μs = 1ms
- આઉટર લૂપ: 5 ઇટરેશન × 1ms = 5ms

મેમરી ટ્રીક: "LOON-5" (લૂપ નેસ્ટેડ ફોર 5ms)

પ્રશ્ન 4(ક OR) [7 ગુણ]

8051 માઇક્રોકન્ટ્રોલર માટે કોઈપણ સાત એરિથમેટિક ઇન્સ્ટ્રક્શન ઉદાહરણ સાથે સમજાવો.

જવાબ:

ઇન્સ્ટ્રક્શન	ફંક્શન	ઉદાહરણ	ફ્લોગ અસર
ADD A,src	સોર્સને A માં ઉમેરે	ADD A,R0 ($A=A+R0$)	C, OV, AC
ADDC A,src	સોર્સ + કેરી A માં ઉમેરે	ADDC A,#25H ($A=A+25H+C$)	C, OV, AC
SUBB A,src	સોર્સ + બોરો A માંથી બાદ કરે	SUBB A,@R1 ($A=A-@R1-C$)	C, OV, AC
INC	1 વધારે	INC R3 ($R3=R3+1$)	કોઈ નહીં
DEC	1 ઘટાડે	DEC A ($A=A-1$)	કોઈ નહીં
MUL AB	A અને B ગુણાકાર કરે	MUL AB ($B:A=A\times B$)	C, OV
DIV AB	A ને B વડે ભાગે	DIV AB ($A=ભાગફળ, B=શેષ$)	C, OV

સાચાગ્રામ:

<pre> +-----+ ADD A,R0 A = 25H, R0 = 15H A = 25H + 15H A = 3AH +-----+ </pre>	<pre> +-----+ MUL AB A = 05H, B = 03H B:A = 05H × 03H B = 00H, A = 0FH +-----+ </pre>	<pre> +-----+ DIV AB A = 14H, B = 05H A = 14H ÷ 05H A = 04H, B = 00H +-----+ </pre>
---	---	---

મેમરી ટ્રીક: "ACID-IBM" (એડ, કેરી એડ, ઇન્ક, ડેક, મલ, બોરો સબ્ટ્રેક્ટ, ડિવાઈડ)

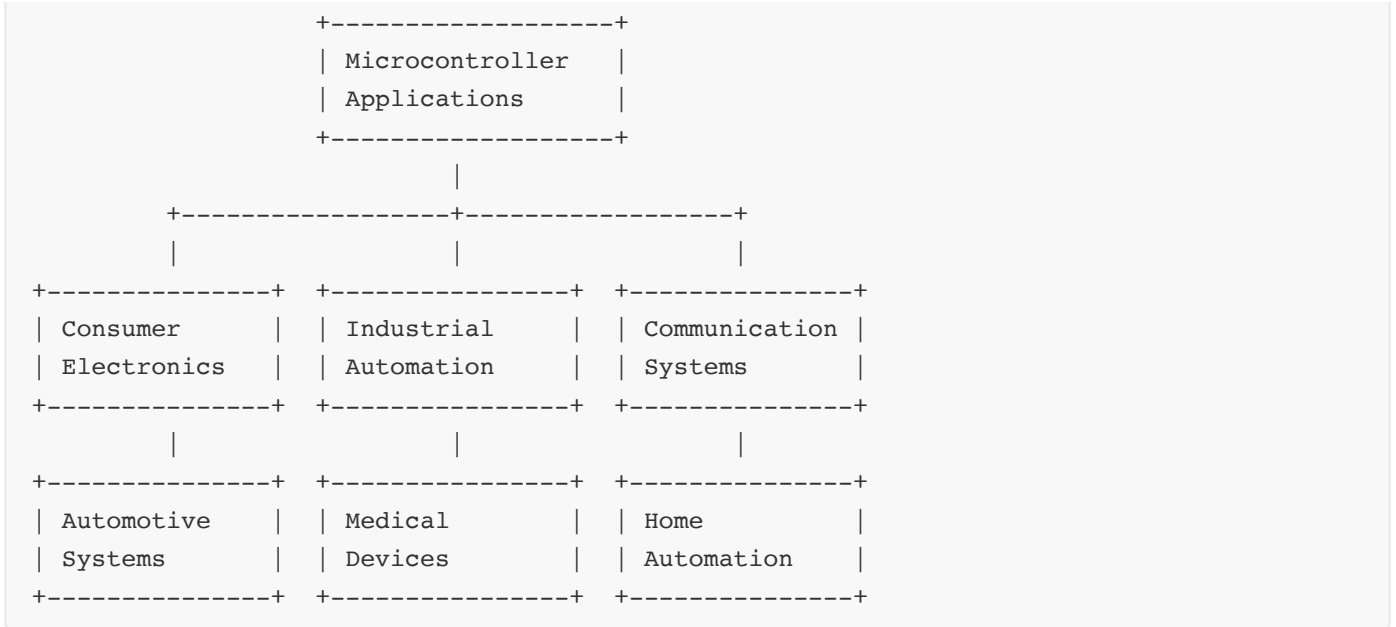
પ્રશ્ન 5(અ) [3 ગુણ]

વિવિધ ક્ષેત્રોમાં માઇક્રોકંટ્રોલરની એપ્લિકેશનોની સૂચિ બનાવો.

જવાબ:

ક્ષેત્ર	એપ્લિકેશન્સ
કન્ઝ્યુમર ઇલેક્ટ્રોનિક્સ	ટીવી, વોશિંગ મશીન, માઇક્રોવેવ, રિમોટ કંટ્રોલ
ઓટોમોટિવ	એન્જિન કંટ્રોલ, એન્ટી-લોક બ્રેકિંગ, એરબેગ સિસ્ટમ
ઇન્ડસ્ટ્રિયલ	ઓટોમેશન, રોબોટિક્સ, પ્રોસેસ કંટ્રોલ
મેડિકલ	પેશન્ટ મોનિટરિંગ, મેડિકલ ઇન્સ્ટ્રુમેન્ટ્સ, ઇમ્પ્લાન્ટ્સ
હોમ ઓટોમેશન	સ્માર્ટ લાઇટિંગ, સિક્યુરિટી સિસ્ટમ, HVAC કંટ્રોલ
કમ્યુનિકેશન	મોબાઇલ ફોન, રાઉટર્સ, મોડેમ્સ
એરોસ્પેસ	નેવિગેશન સિસ્ટમ, ફ્લાઇટ કંટ્રોલ, સેટેલાઇટ સિસ્ટમ

સાચાગ્રામ:



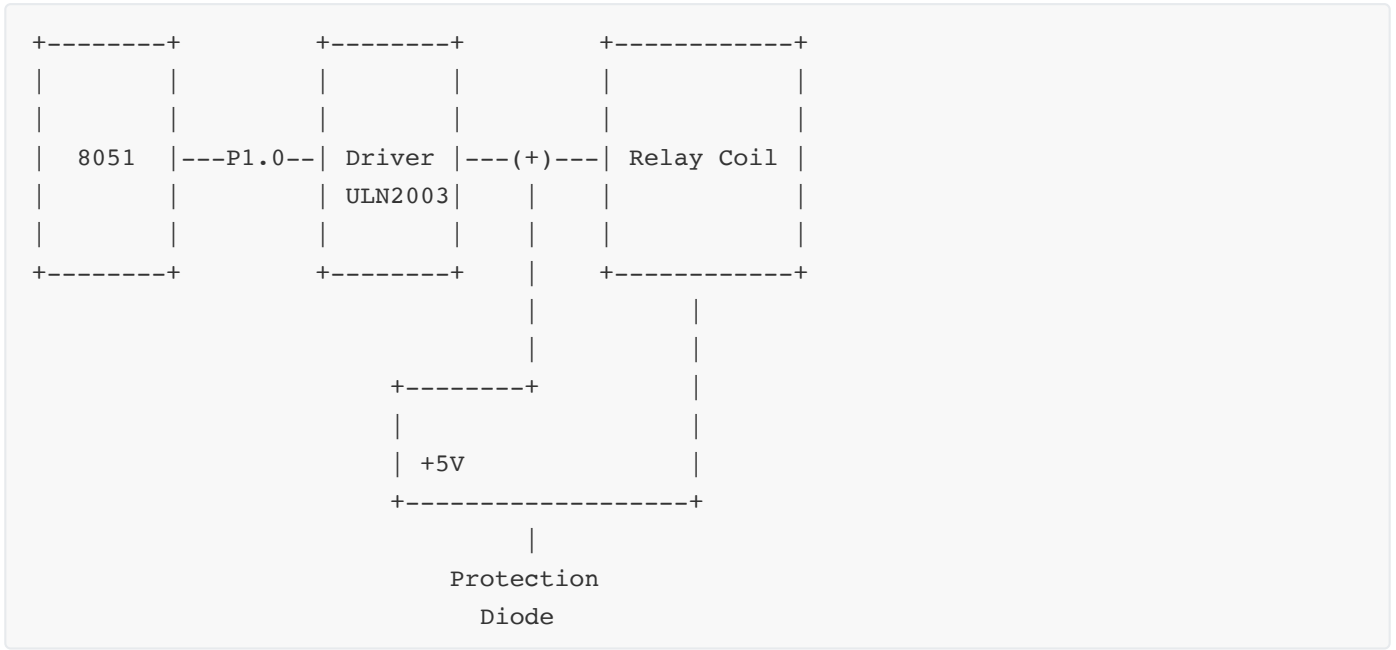
મેમરી ટ્રીક: "CHAIM-MA" (કન્સ્યુમર, હોમ, ઓટોમોટિવ, ઇન્ડસ્ટ્રિયલ, મેડિકલ, મોબાઇલ, એરોસ્પેસ)

પ્રશ્ન 5(બ) [4 ગુણ]

8051 માઇક્રોકંટ્રોલર સાથે રિલે ઇન્ટરફેસ કરો.

જવાબ:

સાચાગ્રામ:



જરૂરી ઘટકો:

- 8051 માઇક્રોકંટ્રોલર
- ULN2003 અથવા સમાન ડ્રાઇવર IC
- રિલે (5V અથવા 12V)
- પ્રોટેક્શન ડાયોડ (1N4007)

- પાવર સપ્લાય

કાર્યપ્રણાલી:

1. 8051 P1.0 થી કંટ્રોલ સિગ્નલ મોકલે છે
2. ડ્રાઇવર રિલે ચલાવવા માટે કરંટ એમ્પ્લિફાય કરે છે
3. પ્રોટેક્શન ડાયોડ બેક EMF નુકસાનથી બચાવે છે
4. રિલે કનેક્ટેડ ડિવાઇસ સ્વિચ કરે છે

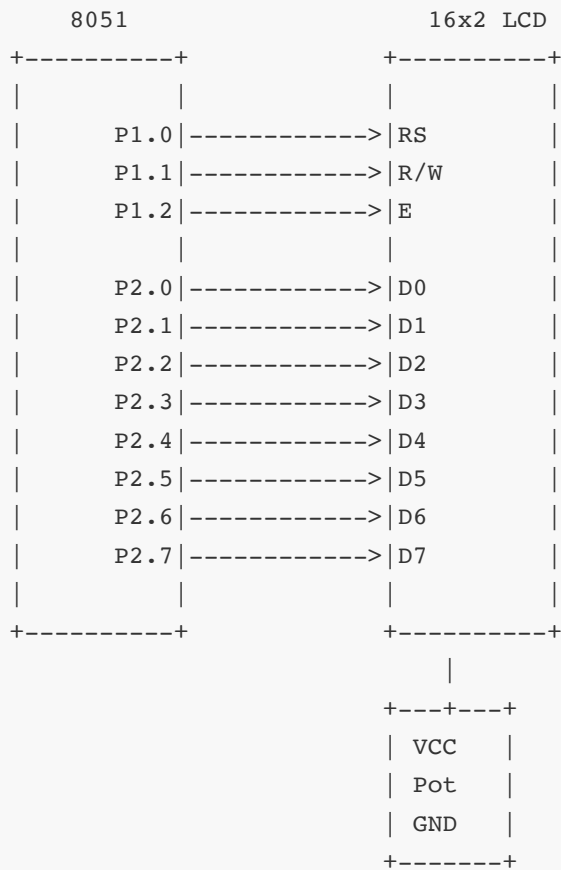
મેમરી ટ્રીક: "DRIPS" (ડ્રાઇવર, રિલે, ઇનપુટ ફોમ μC , પ્રોટેક્શન ડાયોડ, સ્વિચિંગ)

પ્રશ્ન 5(ક) [7 ગુણ]

8051 માઇક્રોકંટ્રોલર સાથે LCD ઇન્ટરફેસ કરો.

જવાબ:

ડાયાગ્રામ:



કનેક્શન:

- **કંટ્રોલ લાઇન્સ:**
 - P1.0 → RS (રજિસ્ટર સિલેક્ટ)
 - P1.1 → R/W (રીડ/રાઇટ)
 - P1.2 → E (એનેબલ)

- ડેટા લાઇન્સ:

- P2.0-P2.7 → D0-D7 (8-બિટ ડેટા બસ)

LCD ઇનિશિયલાઇઝ કરવાનો કોડ:

```
MOV A, #38H      ; 2 lines, 5x7 matrix
ACALL COMMAND    ; Send command

MOV A, #0EH      ; Display ON, cursor ON
ACALL COMMAND    ; Send command

MOV A, #01H      ; Clear LCD
ACALL COMMAND    ; Send command

MOV A, #06H      ; Increment cursor
ACALL COMMAND    ; Send command
```

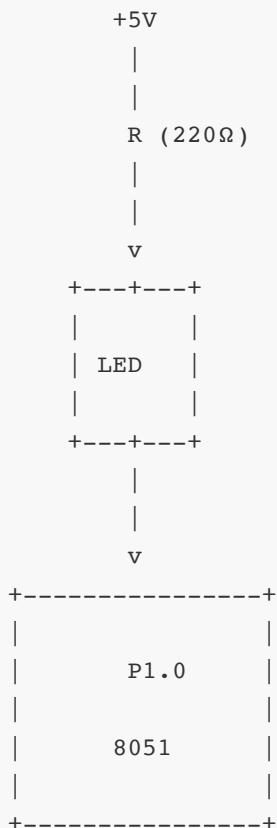
મેમરી ટ્રીક: "CIDER-8" (કંટ્રોલ લાઇન્સ, ઇનિશિયલાઇઝ, ડેટા બસ, એનેબલ, રજિસ્ટર સિલેક્ટ, 8-બિટ મોડ)

પ્રશ્ન 5(અ OR) [3 ગુણ]

8051 માઇક્રોકન્ટ્રોલર સાથે LED નું ઇન્ટરફેસિંગ દોરો.

જવાબ:

ડાયાગ્રામ:



જરૂરી ઘટકો:

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

કાર્યપ્રણાલી:

- એક્ટિવ-લો કન્ફિગરેશન: પિન = 0 ત્યારે LED ON
- P1.0 LED ને કરંટ લિમિટિંગ રેસિસ્ટર મારફતે ડ્રાઇવ કરે છે
- મહત્તમ કરંટ પિન દીઠ 20mA નથી વધવો જોઈએ

LED બ્લિંકિંગ માટે કોડ:

```
MAIN: CLR P1.0      ; Turn ON LED (active low)
      CALL DELAY    ; Wait
      SETB P1.0     ; Turn OFF LED
      CALL DELAY    ; Wait
      SJMP MAIN     ; Repeat
```

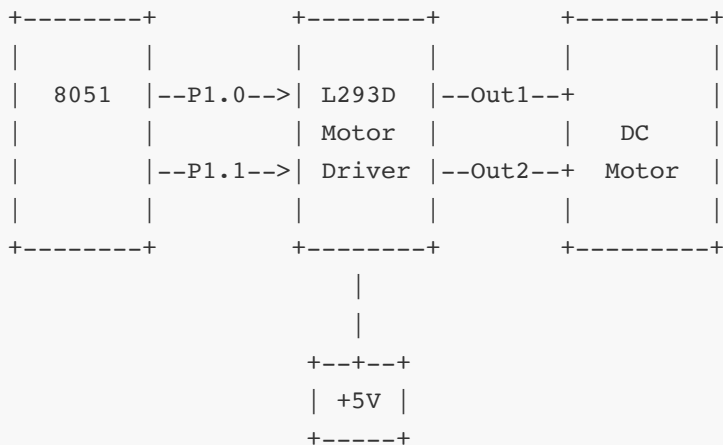
મેમરી ટ્રીક: "CIRCLE" (કરંટ લિમિટિંગ રેસિસ્ટર, IO પિન, કેથોડ ટુ LED, LED ટુ અર્થ/ગ્રાઉન્ડ)

પ્રશ્ન 5(બ OR) [4 ગુણ]

8051 માઇક્રોકન્ટ્રોલર સાથે ડીસી મોટર ઇન્ટરફેસ કરો.

જવાબ:

ડાયાગ્રામ:



જરૂરી ઘટકો:

- 8051 માઇક્રોકન્ટ્રોલર
- L293D મોટર ડ્રાઇવર IC
- ડીસી મોટર
- પાવર સપ્લાય

કંટ્રોલ લોજિક:

P1.0	P1.1	મોટર એક્શન
0	0	સ્ટોપ (બ્રેક)
0	1	ક્લોકવાઇઝ
1	0	કાઉન્ટર-ક્લોકવાઇઝ
1	1	સ્ટોપ (ફ્રી-રનિંગ)

મોટર કંટ્રોલ માટે કોડ:

```

MOV P1, #02H ; P1.0=0, P1.1=1 (Clockwise)
CALL DELAY ; Run for some time
MOV P1, #01H ; P1.0=1, P1.1=0 (Counter-clockwise)
CALL DELAY ; Run for some time
MOV P1, #00H ; P1.0=0, P1.1=0 (Stop)

```

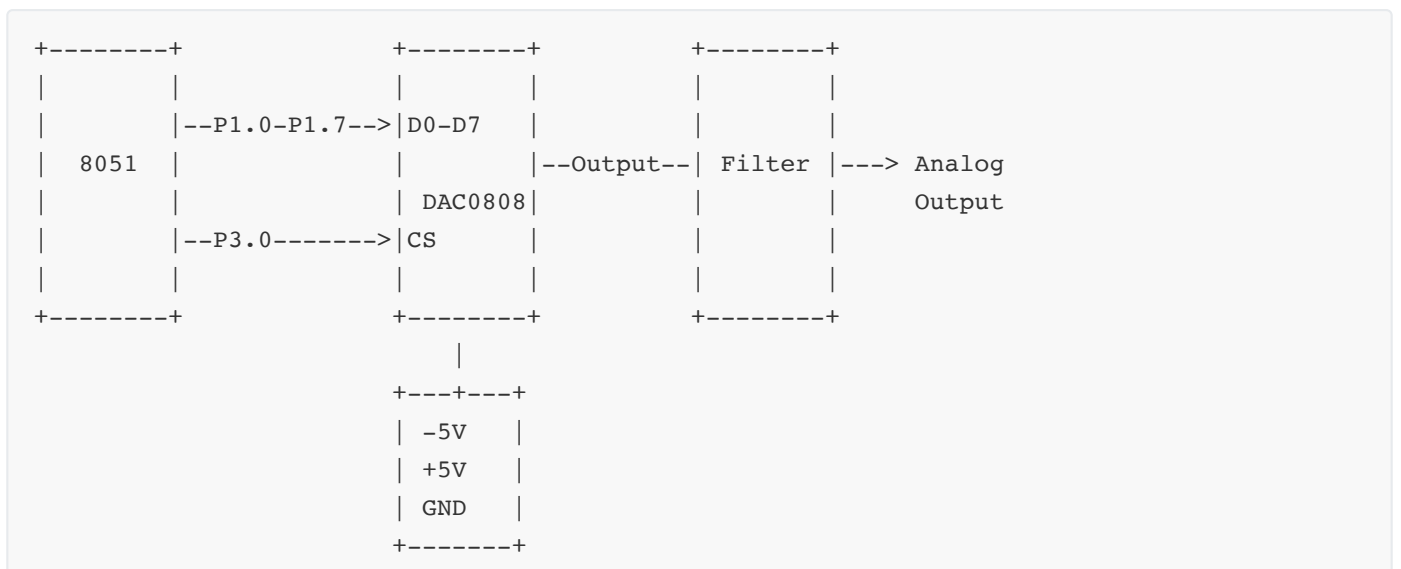
મેમરી ટ્રીક: "DICER" (ડ્રાઇવર ચિપ, ઇનપુટ ફોમ μ C, કંટ્રોલ લોજિક, એનેબલ મોટર, રોટેશન)

પ્રશ્ન 5(ક OR) [7 ગુણ]

8051 માઇક્રોકન્ટ્રોલર સાથે DAC0808 ઇન્ટરફેસ કરો.

જવાબ:

સાચાગ્રામ:

**જરૂરી ઘટકો:**

- 8051 માઇક્રોકન્ટ્રોલર
- DAC0808 (8-બિટ ડિજિટલ-ટુ-એનાલોગ કન્વર્ટર)
- ઓપરેશનલ એમ્પ્લિફાયર (આઉટપુટ બફરિંગ માટે)

- RC ફિલ્ટર (સ્મુથિંગ માટે)
- રેફરન્સ વોલ્ટેજ સોર્સ

કનેક્શન્સ:

- P1.0-P1.7 → D0-D7 (8-બિટ ડિજિટલ ઇનપુટ)
- P3.0 → CS (ચિપ સિલેક્ટ)
- DAC આઉટપુટ → ફિલ્ટર → ફાઇનલ એનાલોગ આઉટપુટ

રેમ્પ સિગ્નલ જનરેશન માટે સેમ્પલ કોડ:

```
START: MOV R0, #00H      ; Start from 0
LOOP:  MOV P1, R0        ; Output value to DAC
       CALL DELAY       ; Wait
       INC R0           ; Increment value
       SJMP LOOP        ; Loop to create ramp
```

ઉપયોગો:

- વેવફોર્મ જનરેશન
- પ્રોગ્રામેબલ વોલ્ટેજ સોર્સ
- મોટર સ્પીડ કંટ્રોલ
- ઓડિયો એપ્લિકેશન્સ

મેમરી ટ્રીક: "DACR" (ડિજિટલ ઇનપુટ, એનાલોગ આઉટપુટ, કન્વર્ઝન, રેફરન્સ વોલ્ટેજ)