

Embedded System (4343204) - Winter 2024 Solution

Milav Dabgar

December 03, 2024

પ્રશ્ન 1(અ) [3 ગુણ]

ATmega32 માં RAM, Flash અને EEPROM મેમરી કેટલી છે? માઇક્રોકન્ટ્રોલરમાં તેની જરૂરિયાત સમજાવો.

જવાબ

ATmega32 મેમરી સ્પેસિફિકેશન અને માઇક્રોકન્ટ્રોલર ઓપરેશનમાં તેનું મહત્વ:

- RAM: પ્રોગ્રામ એક્ઝિક્યુશન દરમિયાન વેરિએબલ્સ માટે ટેમ્પરરી સ્ટોરેજ
- Flash: પ્રોગ્રામ ઇન્સ્ટ્રક્શન્સ અને કોન્સ્ટન્ટ્સ માટે પરમેનન્ટ સ્ટોરેજ
- EEPROM: પાવર સાયકલ્સ પછી પણ જાળવી રાખવા જરૂરી એવા ડેટા માટે લાંબા ગાળાનું સ્ટોરેજ

મેમરી ટ્રીક

“રન માટે RAM, ફ્લેશ માટે Flash, હંમેશા માટે EEPROM”

પ્રશ્ન 1(બ) [4 ગુણ]

ATmega32 ની RAM મેમરીની ચર્ચા કરો.

- રજિસ્ટર ફાઇલ: પ્રથમ 32 લોકેશન્સ (0x0000-0x001F)
- I/O રજિસ્ટર્સ: સ્ટાન્ડર્ડ I/O સ્પેસ (0x0020-0x005F)
- એક્સટેન્ડેડ I/O: વધારાના પેરિફરલ રજિસ્ટર્સ (0x0060-0x00FF)
- ડેટા મેમરી: જનરલ પરપઝ SRAM (0x0100-0x085F)

મેમરી ટ્રીક

“રજિસ્ટર્સ, I/O, એક્સટેન્ડેડ, ડેટા - RAM ની કાર્યક્ષમ ડિઝાઇન”

પ્રશ્ન 1(ક) [7 ગુણ]

રિયલ ટાઇમ ઓપરેટિંગ સિસ્ટમની વ્યાખ્યાયિત કરો અને તેની લાક્ષણિકતાઓ સમજાવો.

જવાબ

રિયલ-ટાઇમ ઓપરેટિંગ સિસ્ટમ (RTOS) એ ચુસ્ત ટાઇમિંગ જરૂરિયાતો સાથે ડેટા અને ઇવેન્ટ્સ પ્રોસેસ કરવા માટે ડિઝાઇન કરાયેલ સ્પેશિયલાઇઝ્ડ ઓપરેટિંગ સિસ્ટમ છે.

- મલ્ટીટાસ્કિંગ: અનેક ટાસ્ક્સના કન્કરન્ટ એક્ઝિક્યુશનને સપોર્ટ કરે છે
- સ્મોલ ફૂટપ્રિન્ટ: મર્યાદિત રિસોર્સવાળા એમ્બેડેડ સિસ્ટમ્સ માટે ઓપ્ટિમાઇઝ્ડ
- ટાઇમ મેનેજમેન્ટ: માઇક્રોસેકન્ડ રેઝોલ્યુશન સાથે પ્રિસાઇઝ ટાઇમિંગ સર્વિસીસ
- કર્નલ સર્વિસીસ: ટાસ્ક કોઓર્ડિનેશન માટે IPC, મ્યુટેક્સ, સેમાફોર

મેમરી ટ્રીક

“ડિટર્મિનિસ્ટિક પ્રિએમિટિવ ટાસ્ક્સ રન ઓન સ્ટ્રિક્ટ ટાઇમલાઇન્સ”

પ્રશ્ન 1(ક OR) [7 ગુણ]

એમ્બેડેડ સિસ્ટમ શું છે? એમ્બેડેડ સિસ્ટમનો સામાન્ય બ્લોક ડાયાગ્રામ દોરો અને સમજાવો.

કોષ્ટક 3. એમ્બેડેડ સિસ્ટમ કોમ્પોનન્ટ્સ

કોમ્પોનન્ટ	ફંક્શન
પ્રોસેસિંગ યુનિટ	પ્રોગ્રામ ઇ-સ્ટ્રક્શન્સ એક્ઝિક્યુટ કરે છે (માઇક્રોકન્ટ્રોલર/માઇક્રોપ્રોસેસર)
મેમરી	પ્રોગ્રામ અને ડેટા સ્ટોર કરે છે (RAM, ROM, Flash)
ઇનપુટ/આઉટપુટ	બાહ્ય ડિવાઇસ સાથે ઇન્ટરફેસ કરે છે
કમ્યુનિકેશન	અન્ય સિસ્ટમ્સ અથવા નેટવર્ક્સ સાથે જોડાય છે
પાવર સપ્લાય	રેગ્યુલેટેડ પાવર પ્રદાન કરે છે
સેન્સર્સ	પર્યાવરણીય ડેટા એકત્રિત કરે છે

- એપ્લિકેશન-સ્પેસિફિક: ડેડિકેટેડ ટાસ્ક્સ માટે ડિઝાઇન કરાયેલ
- રિસોર્સ-કન્સ્ટ્રેઇન્ડ: મર્યાદિત પ્રોસેસિંગ પાવર અને મેમરી
- રિયલ-ટાઇમ: ટાઇમિંગ કન્સ્ટ્રેઇન્ટ્સની અંદર ઇવેન્ટ્સને પ્રતિસાદ આપે છે
- હાઇ રિલાયબિલિટી: નિષ્ફળતા વિના સતત ઓપરેટ કરવું જોઈએ

મેમરી ટ્રીક

“પ્રોસેસ, મેમરી, I/O - દરેક સિસ્ટમમાં હોવું જોઈએ”

પ્રશ્ન 2(અ) [3 ગુણ]

એમ્બેડેડ સિસ્ટમમાં કોઈપણ એપ્લિકેશન ડિઝાઇન માટે માઇક્રોકન્ટ્રોલર પસંદ કરવા માટે વિવિધ માપદંડો લખો.

જવાબ

યોગ્ય માઇક્રોકન્ટ્રોલર પસંદ કરવા માટે એપ્લિકેશન જરૂરિયાતો આધારિત અનેક માપદંડોનું મૂલ્યાંકન કરવું જરૂરી છે.

- એપ્લિકેશન રિકવાયરમેન્ટ્સ: એપ્લિકેશન માટે જરૂરી સ્પેસિફિક ફીચર્સ
- ડેવલપમેન્ટ એન્વાયરમેન્ટ: ઉપલબ્ધ કમ્પાઇલર્સ, ડિબગર્સ, લાઇબ્રેરીઝ
- ફ્યુચર એક્સપાન્શન: ભવિષ્યના એન્હાન્સમેન્ટ્સ માટે સ્કેલેબિલિટી

મેમરી ટ્રીક

“પરફોર્મન્સ મેમરી પાવર I/O કોસ્ટ”

પ્રશ્ન 2(બ) [4 ગુણ]

TCCR0 રજિસ્ટર દોરો અને સમજાવો.

- WGM01:0: ટાઇમર ઓપરેટિંગ મોડ નક્કી કરે છે (નોર્મલ, CTC, PWM)
- COM01:0: OC0 પિન આઉટપુટ બિહેવિયર કંટ્રોલ કરે છે
- CS02:0: ક્લોક સોર્સ અને પ્રીસ્કેલર વેલ્યુ પસંદ કરે છે

D

કોષ્ટક 5. TCCR0 બિટ ફંક્શનસ

બિટ્સ	નામ	ફંક્શન
7	FOC0	ફોર્સ આઉટપુટ કમ્પેર
6,3	WGM01:0	વેવફોર્મ જનરેશન મોડ
5,4	COM01:0	કમ્પેર મેચ આઉટપુટ મોડ
2,1,0	CS02:0	ક્લોક સિલેક્ટ (પ્રીસ્કેલર)

મેમરી ટ્રીક

“ફોર્સ વેવફોર્મ કમ્પેર ક્લોક સિલેક્ટ”

પ્રશ્ન 2(ક) [7 ગુણ]

ATmega32 ના ટાઇમરોની યાદી બનાવો અને કોઈપણ એક ટાઇમરના Modes ને વિગતવાર સમજાવો.

જવાબ

ATmega32માં વિવિધ ક્ષમતાઓ અને ઓપરેટિંગ મોડ્સ સાથે અનેક ટાઇમર્સ છે. ટાઇમર0 ઓપરેટિંગ મોડ્સ:

- નોર્મલ મોડ:
 - કાઉન્ટર 0 થી 255 સુધી વધે છે પછી 0 પર ઓવરફ્લો થાય છે
 - ઓવરફ્લો ઇન્ટરપ્ટ જનરેટ થઈ શકે છે
 - સરળ ટાઇમિંગ અને ડિલે જનરેશન માટે વપરાય છે
- CTC (ક્લિયર ટાઇમર ઓન કમ્પેર) મોડ:
 - કાઉન્ટર OCR0 વેલ્યુ પર પહોંચે ત્યારે રીસેટ થાય છે
 - પ્રિસાઇઝ ફિક્સ-સી જનરેશન માટે ઉપયોગી
 - કમ્પેર મેચ ઇન્ટરપ્ટ જનરેટ થઈ શકે છે
- ફાસ્ટ PWM મોડ:
 - કાઉન્ટર 0 થી 255 સુધી ગણે છે
 - આઉટપુટ ઓવરફ્લો અને કમ્પેર મેચ પર ટોગલ થાય છે
 - હાઇ ફ્રિક્વન્સી PWM જનરેશન
- ફેઝ કરેક્ટ PWM મોડ:
 - કાઉન્ટર ઉપર પછી નીચે (0→255→0) ગણે છે
 - સિમેટ્રિક PWM વેવફોર્મ જનરેશન
 - ફાસ્ટ PWM કરતાં ઓછી ફ્રિક્વન્સી પણ વધુ સારી રેઝોલ્યુશન

મેમરી ટ્રીક

“નોર્મલ કમ્પેર ફાસ્ટ ફેઝ - ટાઇમર મોડ્સ મેટર”

પ્રશ્ન 2(અ OR) [3 ગુણ]

વિવિધ એમ્બેડેડ સિસ્ટમ એપ્લિકેશન-સની સૂચિ બનાવો. કોઈપણ એકને ટૂંકમાં સમજાવો.

જવાબ

એમ્બેડેડ સિસ્ટમ્સ વિવિધ ડોમેઇન-સમાં અનેક એપ્લિકેશન-સમાં જોવા મળે છે.

સ્માર્ટ હોમ ઓટોમેશન સિસ્ટમ: સ્માર્ટ હોમ સિસ્ટમ ઘરેલું ઉપકરણોને મોનિટર અને કંટ્રોલ કરવા માટે એમ્બેડેડ કન્ટ્રોલર્સનો ઉપયોગ કરે છે. સેન્સર્સ તાપમાન અને મોશન જેવી પર્યાવરણીય સ્થિતિઓને ડિટેક્ટ કરે છે, જ્યારે માઇક્રોકન્ટ્રોલર્સ આ ડેટાને પ્રોસેસ કરે છે અને HVAC સિસ્ટમ્સ, લાઇટિંગ અને સિક્યુરિટી ડિવાઇસીસ જેવા એક્ઝ્યુએટર્સને કંટ્રોલ કરે છે. સિસ્ટમને ઓટોનોમસ ઓપરેશન અથવા સ્માર્ટફોન એપ્સ દ્વારા યુઝર કંટ્રોલ માટે પ્રોગ્રામ કરી શકાય છે, જે સુવિધા, એનર્જી એફિશિયન્સી અને એન્હાન્સ્ડ સિક્યુરિટી પ્રદાન કરે છે.

મેમરી ટ્રીક

“કન્ઝ્યુમર્સ ઓટોમેટ ઇન્ડસ્ટ્રી મેડિકલ કમ્યુનિકેશન્સ એરોસ્પેસ”

પ્રશ્ન 2(બ OR) [4 ગુણ]

ATmega32 માઇક્રોકન્ટ્રોલરમાં DDRA, PINA અને PORTA રજિસ્ટરનાં કાર્ય સમજાવો.

જવાબ

ત્રણ રજિસ્ટર્સ ATmega32માં પોર્ટ A ના ઓપરેશનને કંટ્રોલ કરે છે, દરેક અલગ હેતુ ધરાવે છે.
કન્ફિગરેશન ઉદાહરણો:

bRRA = 0xFF; // All pins output PORTA = 0xA5; // Set alternating pattern (10100101)
DDRA = 0x00; // All pins input PORTA = 0xFF; // Enable pull-ups data = PINA; // Read pin states

- બિટ-લેવલ કંટ્રોલ: દરેક બિટ સંબંધિત પિનને કંટ્રોલ કરે છે
- એટોમિક ઓપરેશન્સ: વ્યક્તિગત બિટ્સ મોડિફાઇ કરી શકાય છે
- રીડ-મોડિફાઇ-રાઇટ: સામાન્ય ઓપરેશન પેટર્ન

મેમરી ટ્રીક

“ડિરેક્શન ડિટરમાઇન્સ, પોર્ટ પ્રોવાઇડ્સ, PIN પર્સીવ્સ”

પ્રશ્ન 2(ક OR) [7 ગુણ]

ATmega32 નું સ્ટેટસ રજિસ્ટર દોરો અને તેને વિગતવાર સમજાવો.

જવાબ

ATmega32માં સ્ટેટસ રજિસ્ટર (SREG) એરિથમેટિક ઓપરેશન્સથી પ્રભાવિત પ્રોસેસર સ્ટેટસ ફ્લેગ્સ ધરાવે છે અને ઇન્ટરપ્રુટ્સને કંટ્રોલ કરે છે.

- એરિથમેટિક ફ્લેગ્સ: રિઝલ્ટ સ્ટેટસ દર્શાવે છે
- કન્ડિશનલ બ્રાન્ચીસ: બ્રાન્ચ ઇન્સ્ટ્રક્શન્સ દ્વારા ઉપયોગ કરાય છે
- ઇન્ટરપ્રુટ કંટ્રોલ: I-બિટ બધા ઇન્ટરપ્રુટ્સને એનેબલ/ડિસેબલ કરે છે
- એક્સેસ મેથડ્સ: IN/OUT ઇન્સ્ટ્રક્શન્સ દ્વારા ડાયરેક્ટલી એક્સેસબલ

મેમરી ટ્રીક

“ઇન્ટરપ્રુટ ટ્રેક હાફ સાઇન ઓવરફ્લો નેગેટિવ ઝીરો કેરી”

પ્રશ્ન 3(અ) [3 ગુણ]

AVR માઇક્રોકન્ટ્રોલરના હાર્ડવેર્સ આર્કિટેક્ચર પર ટૂંક નોંધ લખો.

જવાબ

હાર્ડવેર્સ આર્કિટેક્ચર એ AVR માઇક્રોકન્ટ્રોલર્સનો મૂળભૂત ડિઝાઇન સિદ્ધાંત છે, જે પ્રોગ્રામ અને ડેટા મેમરીને અલગ કરે છે.

- Separate Buses: પ્રોગ્રામ અને ડેટા મેમરી માટે સ્વતંત્ર બસો
- Parallel Access: એક્સાઇઝ ઇન્સ્ટ્રક્શન્સ ફેચ કરી શકે છે અને ડેટા એક્સેસ કરી શકે છે
- Performance: મેમરી બોટલનેક્સ દૂર કરીને એક્ઝિક્યુશન સ્પીડ વધારે છે
- Different Widths: પ્રોગ્રામ મેમરી 16-બિટ વર્ડ્સમાં, ડેટા મેમરી 8-બિટ બાઇટ્સમાં ગોઠવાય છે

મેમરી ટ્રીક

“પ્રોગ્રામ અને ડેટા પાથ અલગ છે”

પ્રશ્ન 3(બ) [4 ગુણ]

સીરીયલ કોમ્યુનિકેશન (RS232) સાથે સંકળાયેલા રજીસ્ટરોની યાદી બનાવો અને તેને ATmega32 સાથે ઇન્ટરફેસ કરવાના પગલાં સમજાવો.

જવાબ

ATmega32 સીરીયલ કમ્યુનિકેશન માટે USART (યુનિવર્સલ સિંક્રનસ એસિંક્રોનસ રિસીવર ટ્રાન્સમીટર) નો ઉપયોગ કરે છે. RS232 ઇન્ટરફેસ કરવાના સ્ટેપ્સ:

- હાર્ડવેર કનેક્શન:
 - ATmega32 ના TXD (PD1) અને RXD (PD0) ને MAX232 સાથે જોડો
 - MAX232 ને RS232 પોર્ટ અથવા કનેક્ટર સાથે જોડો
- USART ઇનિશિયલાઇઝ કરો:
 - બોડ રેટ સેટ કરો (UBRR)
 - ફ્રેમ ફોર્મેટ સેટ કરો (ડેટા બિટ્સ, પેરિટી, સ્ટોપ બિટ્સ)
 - ટ્રાન્સમીટર અને/અથવા રિસીવર એનેબલ કરો
- ડેટા ટ્રાન્સમિશન/રિસેપ્શન:
 - ઓપરેશન પહેલાં સ્ટેટસ ફ્લેગ્સ ચેક કરો
 - ટ્રાન્સમિટ કરવા માટે UDR માં રાઇટ કરો
 - રિસીવ કરવા માટે UDR માંથી રીડ કરો

મેમરી ટ્રીક

“કનેક્ટ, કન્ફિગર બોડ, એનેબલ, ટ્રાન્સમિટ/રિસીવ”

પ્રશ્ન 3(ક) [7 ગુણ]

AVR C પ્રોગ્રામિંગમાં બીટ-વાઈઝ લોજિકલ ઓપરેશન્સ જરૂરી ઉદાહરણો સાથે સમજાવો.

જવાબ

બિટ-વાઈઝ ઓપરેશન્સ બાઇટ અથવા વર્ડમાં વ્યક્તિગત બિટ્સને મેનિપ્યુલેટ કરે છે, જે એમ્બેડેડ પ્રોગ્રામિંગ માટે આવશ્યક છે. ઉદાહરણ: બિટ્સ સેટ અને ક્લિયર કરવા

```

1 // Set bit 3 of PORTB
2 PORTB |= (1 << 3); // PORTB = PORTB | 0b00001000
3
4 // Clear bit 5 of PORTB
5 PORTB &= ~(1 << 5); // PORTB = PORTB & 0b11011111
6
7 // Toggle bit 2 of PORTB
8 PORTB ^= (1 << 2); // PORTB = PORTB ^ 0b00000100
9
10 // Check if bit 4 is set
11 if (PINB & (1 << 4)) {
12     // Bit 4 is set
13 }
```

મેમરી ટ્રીક

“AND ક્લિયર કરે, OR સેટ કરે, XOR ટોગલ કરે, શિફ્ટ મલ્ટિપ્લાય/ડિવાઇડ કરે”

પ્રશ્ન 3(અ OR) [3 ગુણ]

ATmega32 માઇક્રોકન્ટ્રોલર માટે RESET સર્કિટ સમજાવો.

જવાબ

જ્યારે પાવર આપવામાં આવે અથવા સિસ્ટમ રિસેટ દરમિયાન રિસેટ સર્કિટ ATmega32 નું યોગ્ય ઇનિશિયલાઇઝેશન સુનિશ્ચિત કરે છે.

- **Active-Low RESET:** માઇક્રોકન્ટ્રોલરને રિસેટ કરવા માટે લો (low) રાખવામાં આવે છે
- **External Reset:** મેન્યુઅલ રિસેટ બટન RESET પિનને ગ્રાઉન્ડ સાથે જોડે છે
- **Power-on Reset:** જ્યારે પાવર પ્રથમ વખત આપવામાં આવે ત્યારે ઓટો-રિસેટ થાય છે
- **Brown-out Detection:** વોલ્ટેજ થ્રેશોલ્ડથી નીચે જાય ત્યારે રિસેટ થાય છે
- **Watchdog Timer:** સોફ્ટવેર મેલફંક્શન પર રિસેટ થાય છે

મેમરી ટ્રીક

“પુલ અપ, પુશ બટન, પાવર સ્ટાર્ટસ, વોલ્ટેજ ડ્રોપ્સ”

પ્રશ્ન 3(બ OR) [4 ગુણ]

EEPROM સાથે સંકળાયેલા રજિસ્ટરોની યાદી બનાવો અને ATmega32 ના EEPROM ને ઇન્ટરફેસ કરવાના સ્ટેપ્સ લખો.

જવાબ

ATmega32 પાસે ડેડિકેટેડ રજિસ્ટર્સ સાથે ઓન-ચિપ EEPROM છે.

EEPROM ઇન્ટરફેસ કરવાના સ્ટેપ્સ:

1. પૂર્ણ થવાની રાહ જુઓ: અગાઉનું રાઇટ ઓપરેશન પૂર્ણ થયું છે કે કેમ તે તપાસો (EECR માં EERE બિટ)
2. એડ્રેસ સેટ કરો: EEARH:EEARL માં એડ્રેસ લોડ કરો (16-બિટ એડ્રેસ)
3. રીડ અથવા રાઇટ ઓપરેશન:
 - રીડ માટે: EECR માં EERE બિટ સેટ કરો, પછી EEDR વાંચો
 - રાઇટ માટે: EEDR માં ડેટા લખો, પછી EECR માં EEMWE અને EEWB બિટ્સ સેટ કરો
4. પૂર્ણ થવાની રાહ જુઓ: EEWB બિટ શૂન્ય ન થાય ત્યાં સુધી પોલ કરો

મેમરી ટ્રીક

“રાહ જુઓ, એડ્રેસ, ડેટા, કંટ્રોલ, રાહ જુઓ”

પ્રશ્ન 3(ક OR) [7 ગુણ]

પોર્ટ C.2 પિન પર સતત 1KHz નો સ્ક્વેર વેવ જનરેટ કરવા માટે C પ્રોગ્રામ લખો. ડિલે બનાવવા માટે ટાઇમર 0, નોર્મલ મોડ અને 1:8 પ્રી-સ્કેલરનો ઉપયોગ કરો. XTAL = 8 MHz ધારો.

જવાબ

કોડ ઇમ્પ્લીમેન્ટેશન:

```

1 #include <avr/io.h>
2
3 int main(void)
4 {
5     // Configure PORTC.2 as output
6     DDRC |= (1 << 2); // Set PC2 as output
7
8     // Timer0 configuration - Normal mode, 1:8 prescaler
9     TCCR0 = (0 << WGM01) | (0 << WGM00) | (0 << CS02) | (1 << CS01) | (0 << CS00);
10
11     // Calculate timer value for 1KHz (500us period, 250us half-period)
12     // 8MHz/8 = 1MHz timer clock, 250 cycles for 250us
13     // 256-250 = 6 (starting value for 250us)
14
15     while (1)
16     {
17         // Toggle PORTC.2
18         PORTC ^= (1 << 2);
19
20         // Reset timer
21         TCNT0 = 6;
22
23         // Wait until timer overflows
24         while (!(TIFR & (1 << TOV0)));
25
26         // Clear overflow flag
27         TIFR |= (1 << TOV0);
28     }
29
30     return 0;
31 }

```

- ફ્રિક્વન્સી ગણતરી: $1\text{KHz} = 1000\text{Hz} = 1\text{ms}$ પિરિયડ $= 500\mu\text{s}$ હાફ-પિરિયડ
- ટાઇમર કલોક: $8\text{MHz} \div 8 = 1\text{MHz} = 1\mu\text{s}$ પ્રતિ ટિક
- ટાઇમર ટિક્સ: $250\mu\text{s} \div 1\mu\text{s} = 250$ ટિક્સ
- પ્રારંભિક મૂલ્ય: $256 - 250 = 6$ (250 ટિક્સ પછી ઓવરફ્લો માટે)

પ્રશ્ન 4(અ) [3 ગુણ]

ATmega32 સાથે SPI આધારિત device ઇન્ટરફેસિંગ ડાયાગ્રામ દોરો અને સમજાવો.

જવાબ

SPI (સીરિયલ પેરિફેરલ ઇન્ટરફેસ) એક સિંક્લસ સીરિયલ કમ્યુનિકેશન પ્રોટોકોલ છે જેનો ઉપયોગ ATmega32 ને પેરિફેરલ ડિવાઇસ સાથે જોડવા માટે થાય છે.

- MOSI (Master Out Slave In): માસ્ટર થી સ્લેવ ડેટા
- MISO (Master In Slave Out): સ્લેવ થી માસ્ટર ડેટા
- SCK (Serial Clock): માસ્ટર દ્વારા પૂરી પાડવામાં આવતી સિંક્નાઇઝેશન કલોક
- SS (Slave Select): ચોક્કસ સ્લેવ ડિવાઇસ પસંદ કરવા માટે એક્ટિવ-લો સિગ્નલ

મેમરી ટ્રીક

“માસ્ટર આઉટપુટ્સ, સ્લેવ ઇનપુટ્સ, કલોક કીપ્સ સિંક્નાઇઝેશન”

પ્રશ્ન 4(બ) [4 ગુણ]

ATmega32 સાથે ULN2803 નો ઉપયોગ કરીને Relay નું ઇન્ટરફેસિંગ દોરો અને સમજાવો.

જવાબ

ULN2803 એ ડાર્લિંગ્ટન ટ્રાન્ઝિસ્ટર જોડીઓનો એરે છે જેનો ઉપયોગ માઇક્રોકંટ્રોલર પિનમાંથી રિલે જેવા હાઇ-કરંટ ડિવાઇસને ડ્રાઇવ કરવા માટે થાય છે.

- કરંટ એમ્પ્લિફિકેશન: ULN2803 ચેનલ દીઠ 500mA સુધી સિંક કરી શકે છે
- વોલ્ટેજ આઇસોલેશન: બિલ્ટ-ઇન ડાયોડ્સ ઇન્ડક્ટિવ લોડ સામે રક્ષણ આપે છે
- મલ્ટીપલ ચેનલો: એક પેકેજમાં 8 ડાર્લિંગ્ટન જોડીઓ
- હાઇ વોલ્ટેજ રેટિંગ: આઉટપુટ પર 50V સુધી હેન્ડલ કરી શકે છે

મેમરી ટ્રીક

“લો કરંટ કંટ્રોલ્સ હાઇ કરંટ લોડ્સ”

પ્રશ્ન 4(ક) [7 ગુણ]

ATmega32 ના ADC0 (પિન 40) પર જોડાયેલ LM35 નો ઇન્ટરફેસિંગ ડાયાગ્રામ દોરો અને પોર્ટ B પર ડિજિટલ પરિણામ દર્શાવવા માટે AVR C પ્રોગ્રામ લખો. (ADC નો 8-બીટ મોડમાં ઉપયોગ કરો).

જવાબ

LM35 એ પ્રિસિઝન ટેમ્પરેચર સેન્સર છે જે તાપમાનના પ્રમાણમાં એનાલોગ વોલ્ટેજ આઉટપુટ આપે છે.
C Program:

```

1 #include <avr/io.h>
2 #include <util/delay.h>
3
4 int main(void)
5 {
6     // Configure PORTB as output for displaying result
7     DDRB = 0xFF;
8
9     // Configure ADC
10    // REFS0=1: AVCC reference
11    // ADLAR=1: Left adjust for 8-bit reading from ADCH
12    // MUX=00000: ADC0
13    ADMUX = (0 << REFS1) | (1 << REFS0) | (1 << ADLAR) |
14            (0 << MUX4) | (0 << MUX3) | (0 << MUX2) | (0 << MUX1) | (0 << MUX0);
15
16    // Enable ADC and set prescaler to 128
17    ADCSRA = (1 << ADEN) | (1 << ADPS2) | (1 << ADPS1) | (1 << ADPS0);
18
19    while (1)
20    {
21        // Start conversion
22        ADCSRA |= (1 << ADSC);
23
24        // Wait for conversion to complete
25        while (ADCSRA & (1 << ADSC));
26
27        // Display result on PORTB (8-bit from ADCH)
28        PORTB = ADCH;
29
30        // Wait before next reading
31        _delay_ms(500);
32    }
33
34    return 0;
35 }

```

- તાપમાન ગણતરી: LM35 10mV/°C આઉટપુટ આપે છે
- ADC કન્ફિગરેશન: સરળ 8-બિટ રીડિંગ માટે લેફ્ટ-એડજસ્ટેડ
- રેઝોલ્યુશન: 8-બિટ મોડનો ઉપયોગ કરીને 5V રેફરન્સ સાથે આશરે 1°C રેઝોલ્યુશન મળે છે
- રેન્જ: 0-255°C રેન્જ માપી શકે છે (8-બિટ રજિસ્ટર દ્વારા મર્યાદિત)

મેમરી ટ્રીક

“કનેક્ટ, કન્ફિગર, કન્વર્ટ, કેપ્ચર, ડિસ્પ્લે”

પ્રશ્ન 4(અ OR) [3 ગુણ]

પોર્ટ A ના PA0 પિનને સતત મોનિટર કરવા માટે એક AVR C પ્રોગ્રામ લખો. જો તે HIGH હોય, તો પોર્ટ C ના PC0 પિન પર HIGH મોકલો; અન્યથા, પોર્ટ C ના PC0 પિન પર LOW મોકલો.

જવાબ

કોડ ઇમ્પ્લીમેન્ટેશન:

```

1 #include <avr/io.h>
2
3 int main(void)
4 {
5     // Configure PA0 as input
6     DDRA &= ~(1 << PA0);
7
8     // Enable pull-up resistor on PA0
9     PORTA |= (1 << PA0);
10
11    // Configure PC0 as output
12    DDRC |= (1 << PC0);
13
14    while (1)
15    {
16        // Check if PA0 is HIGH
17        if (PINA & (1 << PA0))
18        {
19            // Set PC0 HIGH
20            PORTC |= (1 << PC0);
21        }
22        else
23        {
24            // Set PC0 LOW
25            PORTC &= ~(1 << PC0);
26        }
27    }
28
29    return 0;
30 }

```

- ઇનપુટ કન્ફિગરેશન: પુલ-અપ રેઝિસ્ટર સાથે ઇનપુટ તરીકે સેટ કરેલ છે
- સતત મોનિટરિંગ: ઇન્ક્રિનિટ લૂપ પિન સ્ટેટ ચેક કરે છે
- આઉટપુટ એક્શન: PC0 PA0 સ્ટેટને મિરર કરે છે
- કાર્યક્ષમ કોડ: પિન મોનિટરિંગ માટે સરળ કન્ડિશનલ સ્ટેટમેન્ટ

મેમરી ટ્રીક

“કન્ફિગર, મોનિટર, મિરર”

પ્રશ્ન 4(બ OR) [4 ગુણ]

ATmega32 પિન ડાયાગ્રામ દોરો અને Vcc, AVcc અને Aref પિનના કાર્ય લખો.

જવાબ

ATmega32 માં DIP પેકેજમાં 40 પિન્સ ગોઠવાયેલી છે, જેમાં પાવર સાપ્લાય પિન્સ અલગ કાર્ય ધરાવે છે.

- VCC: ડિજિટલ લોજિક અને I/O પોર્ટ્સને પાવર આપે છે
- AVCC: VCC ના $\pm 0.3V$ ની અંદર હોવું આવશ્યક છે, ભલે ADC બિનઉપયોગી હોય
- AREF: ADC માટે વૈકલ્પિક એક્સટર્નલ રેફરન્સ, અન્યથા AVCC સાથે કનેક્ટ કરો

મેમરી ટ્રીક

“VCC કોર સર્કિટ માટે, AVCC એનાલોગ માટે, AREF રેફરન્સ માટે”

પ્રશ્ન 4(ક OR) [7 ગુણ]

ATmega32 સાથે MAX7221 નું ઇન્ટરફેસિંગ ડાયાગ્રામ દોરો અને સમજાવો.

જવાબ

MAX7221 એ LED ડિસ્પ્લે ડ્રાઇવર IC છે જે SPI કમ્યુનિકેશનનો ઉપયોગ કરીને ATmega32 સાથે ઇન્ટરફેસ કરે છે. ઇન્ટરફેસિંગ સ્ટેપ્સ:

- SPI ઇનિશિયલાઇઝ કરો: માસ્ટર મોડ, કલોક પોલેરિટી/ફેઝ કન્ફિગર કરો, SS હાઇ સેટ કરો.
- MAX7221 ઇનિશિયલાઇઝ કરો: ડિકોડ મોડ, સ્કેન લિમિટ, ઇન્ટેન્સિટી સેટ કરો, ડિસ્પ્લે ચાલુ કરો.
- ડેટા મોકલો: SS લો પુલ કરો, એડ્રેસ/ડેટા મોકલો, SS હાઇ પુલ કરો.

પ્રશ્ન 5(અ) [3 ગુણ]

L293D મોટર ડ્રાઇવર IC નો પિન ડાયાગ્રામ દોરો અને સમજાવો.

જવાબ

L293D એ DC મોટર્સના બાયડાયરેક્શનલ કંટ્રોલ માટે ડિઝાઇન કરાયેલ ક્વાડ્રુપલ હાફ-H ડ્રાઇવર છે.

- Dual H-Bridges: બે DC મોટર્સને સ્વતંત્ર રીતે કંટ્રોલ કરી શકે છે
- Heat Sink: ગ્રાઉન્ડ પિન્સ હીટ ડિસિપેશન પ્રદાન કરે છે
- High Current: પ્રતિ ચેનલ 600mA સુધી ડ્રાઇવ કરી શકે છે
- Protection Diodes: ઇન્ડક્ટિવ લોડ્સ માટે ઇન્ટરનલ ફ્લાયબેક ડાયોડ્સ

મેમરી ટ્રીક

“એનેબલ, ઇનપુટ, આઉટપુટ, પાવર”

પ્રશ્ન 5(બ) [4 ગુણ]

ADMUX રજિસ્ટર દોરો અને સમજાવો.

જવાબ

ADMUX (ADC મલ્ટિપ્લેક્સર સિલેક્શન રજિસ્ટર) ATmega32માં એનાલોગ ચેનલ સિલેક્શન અને રિઝલ્ટ ફોર્મેટ કંટ્રોલ કરે છે.

- REFS1:0 સેટિંગ્સ: 00=AREF, 01=AVCC, 11=ઇન્ટરનલ 2.56V
- ચેનલ સિલેક્શન: MUX3:0 કયા ADC ઇનપુટને કનેક્ટ કરવું તે સિલેક્ટ કરે છે
- રિઝલ્ટ એલાઇનમેન્ટ: ADLAR=1 રિઝલ્ટને લેફ્ટ શિફ્ટ કરે છે (8-બિટ રીડિંગ્સ માટે)

મેમરી ટ્રીક

“રેફરન્સ, એલાઇનમેન્ટ, મલ્ટિપ્લેક્સર”

પ્રશ્ન 5(ક) [7 ગુણ]

સ્માર્ટ સિંગાઈ પદ્ધતિ સમજાવો.

જવાબ

સ્માર્ટ સિંચાઈ સિસ્ટમ પર્યાવરણીય પરિસ્થિતિઓના આધારે વનસ્પતિ ખેતી માટે પાણીનું કાર્યક્ષમ રીતે વ્યવસ્થાપન કરવા એમ્બેડેડ ટેકનોલોજીનો ઉપયોગ કરે છે.

- **Automated Watering:** જ્યારે સોઇલ મોઇશ્ચર થ્રેશોલ્ડથી નીચે જાય ત્યારે જ વનસ્પતિઓને પાણી આપે છે
- **Weather Adaptation:** ફોરકાસ્ટના આધારે વોટરિંગ શેડ્યુલ એડજસ્ટ કરે છે
- **Zone Control:** અલગ અલગ વિસ્તારોમાં અલગ અલગ વોટરિંગ શેડ્યુલ
- **Water Conservation:** ઓપ્ટિમલ ગ્રોથ માટે મિનિમમ જરૂરી પાણીનો ઉપયોગ

મેમરી ટ્રીક

“સેન્સ, ડિસાઇડ, કન્ઝર્વ, ગ્રો”

પ્રશ્ન 5(અ OR) [3 ગુણ]

L293D મોટર ડ્રાઇવરનો ઉપયોગ કરીને ATmega32 સાથે DC મોટરને ઇન્ટરફેસ કરવા માટે સર્કિટ ડાયાગ્રામ દોરો.

જવાબ

સર્કિટ DC મોટરને બાયડાયરેક્શનલ કંટ્રોલ માટે L293D મારફતે ATmega32 સાથે કનેક્ટ કરે છે.
કંટ્રોલ લોજિક:

મેમરી ટ્રીક

“એનેબલ અને ડિરેક્શન કંટ્રોલ મોટર”

પ્રશ્ન 5(બ OR) [4 ગુણ]

ATmega32 સાથે I2C આધારિત device ઇન્ટરફેસિંગ ડાયાગ્રામ દોરો અને સમજાવો.

જવાબ

I2C (ઇન્ટર-ઇન્ટિગ્રેટેડ સર્કિટ) એ માઇક્રોકન્ટ્રોલર સાથે મલ્ટિપલ ડિવાઇસ કનેક્ટ કરવા માટે ટુ-વાયર સીરિયલ બસ છે.

- **SDA:** સીરિયલ ડેટા લાઇન (બાયડાયરેક્શનલ)
- **SCL:** સીરિયલ કલોક લાઇન (માસ્ટર દ્વારા જનરેટ)
- **Pull-up Resistors:** બંને લાઇન્સ પર જરૂરી

મેમરી ટ્રીક

“સ્ટાર્ટ, એડ્રેસ, એકનોલેજ, ડેટા, સ્ટોપ”

પ્રશ્ન 5(ક OR) [7 ગુણ]

IoT આધારિત હોમ ઓટોમેશન સિસ્ટમ સમજાવો.

જવાબ

IoT-આધારિત હોમ ઓટોમેશન સિસ્ટમ ઘરના ઉપકરણોને રિમોટ મોનિટરિંગ અને કંટ્રોલ માટે ઇન્ટરનેટ સાથે કનેક્ટ કરે છે.

- **Remote Access:** ગમે ત્યાંથી કંટ્રોલ
- **Voice Control:** આસિસ્ટન્ટ્સ સાથે ઇન્ટિગ્રેશન
- **Energy Management:** કન્ઝમ્પશન ઓપ્ટિમાઇઝેશન
- **Automation:** શેડ્યુલિંગ અને સીન સેટિંગ

મેમરી ટ્રીક

``કનેક્ટ, કંટ્રોલ, મોનિટર, ઓટોમેટ, લર્ન``