

Embedded System & Microcontroller Application (4351102) - Winter 2024 Solution - Gujarati

Milav Dabgar

November 21, 2024

પ્રશ્ન 1(a) [3 ગુણ]

ATmega32 ની વિશેષતાઓ લખો.

જવાબ

ATmega32 વિશેષતાઓ:

કોષ્ટક 1. ATmega32 Features

વિશેષતા	વર્ણન
આર્કિટેક્ચર	8-bit RISC પ્રોસેસર
મેમરી	32KB ફ્લેશ, 2KB SRAM, 1KB EEPROM
I/O પોર્ટ્સ	32 પ્રોગ્રામેબલ I/O પિન્સ
ટાઈમર્સ	3 ટાઈમર્સ (Timer0, Timer1, Timer2)
ADC	10-bit, 8-channel ADC
કમ્યુનિકેશન	USART, SPI, I2C (TWI)

- હાઇ પફોર્મન્સ: 16MHz પર 16 MIPS.
- લો પાવર: બહુવિધ સ્લીપ મોડ્સ.
- ઓપરેટિંગ વોલ્ટેજ: 2.7V થી 5.5V.

મેમરી ટ્રીક

“Architecture-RISC Memory-32KB Timers-3 I/O-32pins Communication-3types”

પ્રશ્ન 1(b) [4 ગુણ]

માઇક્રોકોલર પસંદ કરવા માટેના માપદંડો લખી સમજાવો.

જવાબ

પસંગીના માપદંડો:

કોષ્ટક 2. Selection Criteria

માપદંડ	વિચારણા
પફોર્માન્સ	સ્પીડ, ઇન્સ્ટ્રુક્શન સેટ, આર્કિટેક્ચર
મેમરી	RAM, ROM, EEPROM આવશ્યકતાઓ
I/O જરૂરિયાતો	પિન્સની સંખ્યા, સ્પેશિયલ ફંક્શન્સ
પાવર કન્જમ્પશન	બેટરી લાઇફ, સ્લીપ મોડ્સ
કિંમત	યુનિટ પ્રાઇસ, ડેવલપમેન્ટ કોસ્ટ
ડેવલપમેન્ટ ટૂલ્સ	કમ્પાઇલર, ડીબગર ઉપલબ્ધતા

- એપ્લિકેશન જરૂરિયાતો: રિયલ-ટાઇમ કન્સ્ટ્રેઇન્ટ્સ, પ્રોસેસિંગ નીડ્સ.
- પેકેજ સાઇઝ: ફાઇનલ પ્રોડક્ટમાં ર્પેસ લિમિટેશન્સ.
- પેરિફરલ સપોર્ટ: ADC, ટાઇમર્સ, કમ્પ્યુનિકેશન ઇન્ટરફેસ.

મેમરી ટ્રીક

“Performance Memory I/O Power Cost Development”

પ્રશ્ન 1(c) [7 ગુણા]

Embedded System ને વ્યાખ્યાયિત કરો. નાના, મધ્યમ અને વિશાળ Embedded System ની ઉપયોગિતાની યાદી બનાવો.

જવાબ

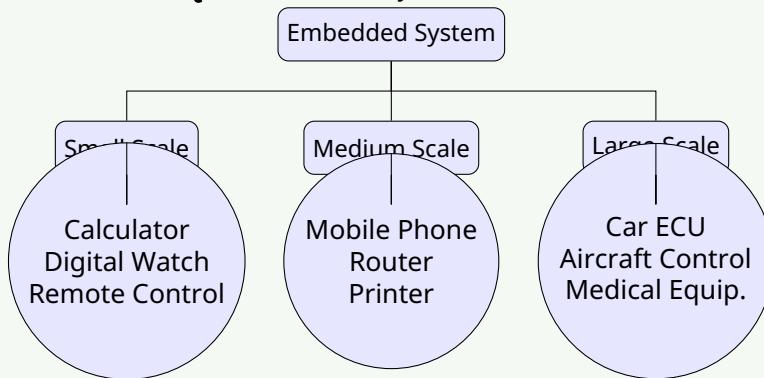
વ્યાખ્યા: Embedded System એ મોટા યાંત્રિક અથવા ઇલેક્ટ્રિકલ સિસ્ટમમાં ચોક્કસ કામ કરતું કમ્પ્યુટર સિસ્ટમ છે, જે વિશિષ્ટ કામો રિયલ-ટાઇમ મર્યાદા સાથે કરવા માટે ડિઝાઇન કરવામાં આવે છે.

એપ્લિકેશન્સ:

કોષ્ટક 3. Embedded System Applications

સિસ્ટમ પ્રકાર	મેમરી સાઇઝ	એપ્લિકેશન્સ
નાના સ્કેલ	<64KB	કેલ્ક્યુલેટર, ડિજિટલ વોચ, રમકડાં
મધ્યમ સ્કેલ	64KB-1MB	મોબાઇલ ફોન, રાઉટર, પ્રિન્ટર
વિશાળ સ્કેલ	>1MB	ઓટોમોબાઇલ, એરકાફ્ટ સિસ્ટમ, સેટેલાઇટ

આકૃતિ 1. Embedded System Classification



લાક્ષણીકરણાઓ:

- રિયલ-ટાઇમ ઓપરેશન: પ્રિડિક્ટેબલ રિસ્પોન્સ ટાઇમ.
- રિસોર્સ કન્સ્ટ્રેઇન્ટ્સ: મર્યાદિત મેમરી અને પ્રોસેસિંગ પાવર.
- ડિડિક્ટેડ ફંક્શનાલિટી: સિંગલ-પર્ફ્જ ડિઝાઇન.

મેમરી ટ્રીક

"Small-Calculator Medium-Mobile Large-Lifesupport"

OR

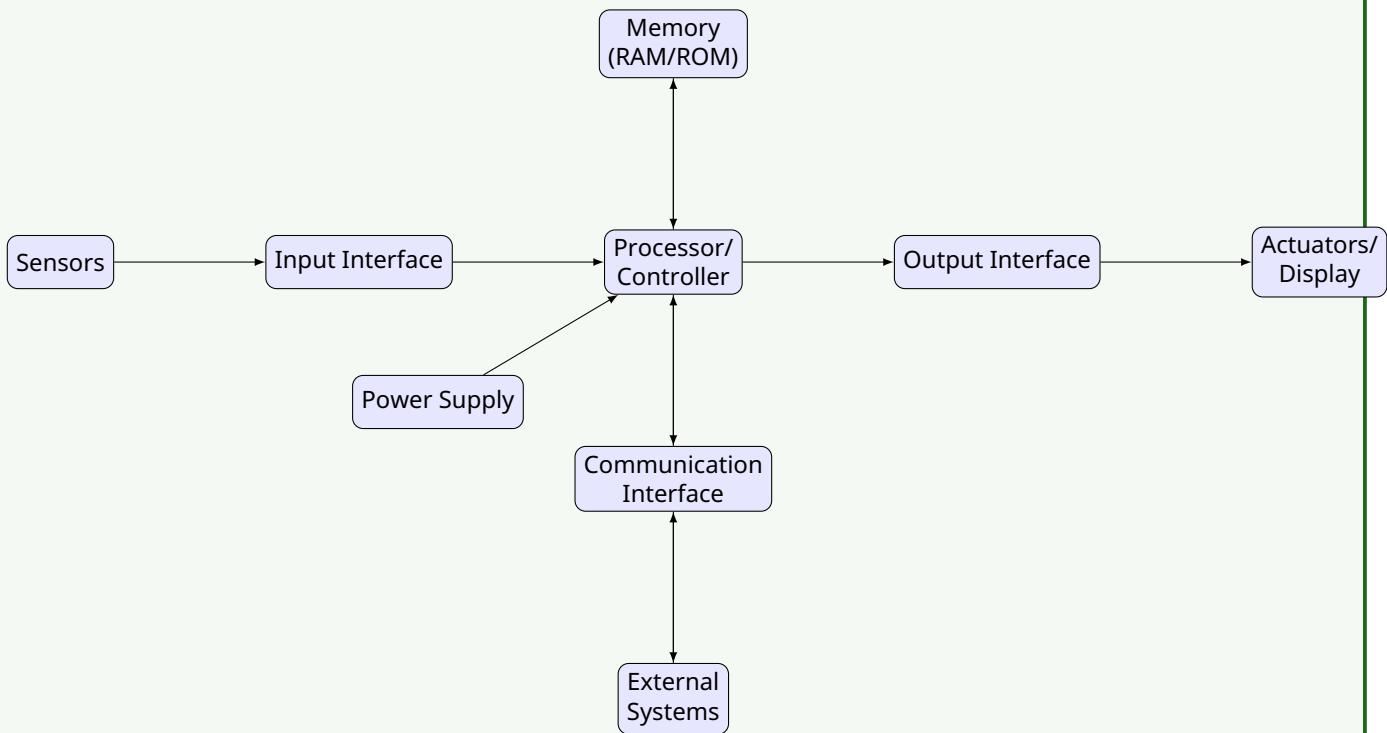
પ્રશ્ન 1(c) [7 ગુણ]

Embedded system નો સામાન્ય બ્લોક ડાયાગ્રામ દોરી સમજાવો.

જવાબ

સામાન્ય બ્લોક ડાયાગ્રામ:

આકૃતિ 2. General Block Diagram



બ્લોક ફંક્શન્સ:

કોષ્ટક 4. Block Functions

બ્લોક	કાર્ય
પ્રોસેસર	સેન્ટ્રલ પ્રોસેસિંગ યુનિટ (CPU/MCU).
ઇનપુટ ઇન્ટરફેસ	સેન્સર ડેટા એક્સિસિંગ, યુઝર ઇનપુટ.
આઉટપુટ ઇન્ટરફેસ	એક્ચ્યુઅટર કંટ્રોલ, ડિસ્પ્લે આઉટપુટ.
મેમરી	પ્રોગ્રામ સ્ટોરેજ, ડેટા સ્ટોરેજ.
કમ્પ્યુનિકેશન	બાહ્ય સિસ્ટમ કનેક્ટિવિટી.

- ઇનપુટ પ્રોસેસિંગ: ADC, ડિજિટલ ઇનપુટ કન્વિર્સિંગ.
- આઉટપુટ કંટ્રોલ: PWM, રિલે ડ્રાઇવર્સ, LED ડિસ્પ્લે.
- પાવર મેનેજમેન્ટ: વોલ્ટેજ રેગ્યુલેશન, પાવર ઓપ્ટિમાઇઝેશન.

મેમરી ટ્રીક

"Processor Input Output Memory Communication Power"

પ્રશ્ન 2(a) [3 ગુણ]

EEPROM નું પૂરું નામ લખો અને તેના વિશે સમજાવો.

જવાબ

પૂરું નામ: Electrically Erasable Programmable Read-Only Memory

EEPROM રજિસ્ટર્સ:

કોષ્ટક 5. EEPROM Registers

રજિસ્ટર	કાર્ય
EEAR	EEPROM Address Register
EEDR	EEPROM Data Register
EECR	EEPROM Control Register

- EEAR: EEPROM એક્સેસ માટે 10-bit એડ્રેસ (0-1023) હોય કરે છે.
- EEDR: રીડ/રાઇટ ઓપરેશન માટે ડેટા રજિસ્ટર.
- EECR: કંટ્રોલ બિટ્સ - EERE (Read Enable), EEWE (Write Enable).

મેમરી ટ્રીક

"Address-EEAR Data-EEDR Control-EECR"

પ્રશ્ન 2(b) [4 ગુણ]

ATmega32માં રીસેટ સર્કિટ વિશે સમજાવો.

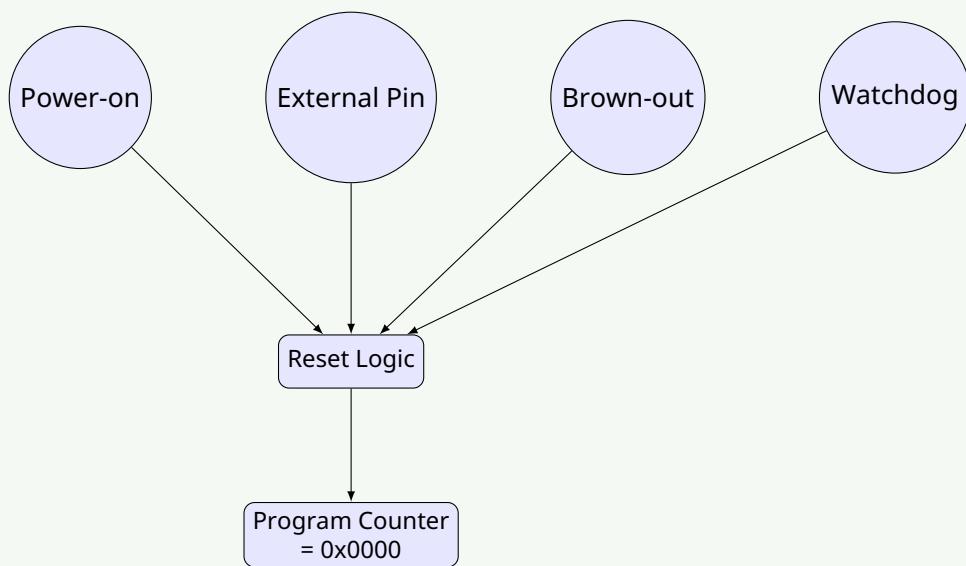
જવાબ

રીસેટ સોર્સ:

કોષ્ટક 6. Reset Sources

રીસેટ પ્રકાર	દ્રિગ કન્ડિશન
પાલર-ઓન રીસેટ	VCC થ્રેશહોલ્ડ ઉપર વધે છે
એક્સટર્નલ રીસેટ	RESET પિન લો પુલ કરવામાં આવે છે
બ્રાઉન-આઉટ રીસેટ	VCC થ્રેશહોલ્ડ નીચે પડે છે
વોયડોગ રીસેટ	વોયડોગ ટાઇમર ઓવરફ્લો

આકૃતિ 3. Reset Logic



- રીસેટ જ્યુરેશન: મિનિમમ 2 કલોક સાઇકલ્વ્સ.
- રીસેટ વેક્ટર: પ્રોગ્રામ એક્ઝિક્યુશન એડેસ 0x0000 થી શરૂ થાય છે.
- હાર્ડવેર કનેક્શન: એક્સ્ટરનલ રોસેટ માટે પુલ-અપ રેજિસ્ટર જરૂરી.

મેમરી ટ્રીક

“Power-on External Brown-out Watchdog”

પ્રશ્ન 2(c) [7 ગુણા]

રિયલ ટાઈમ ઓપરેટિંગ સિસ્ટમની વ્યાખ્યા આપો અને તેની લાક્ષણિકતાઓ સમજાવો.

જવાબ

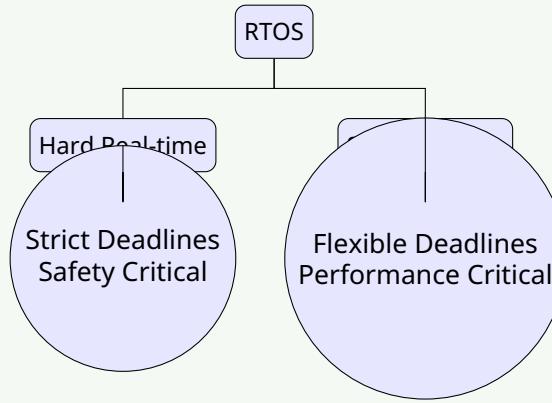
વ્યાખ્યા: Real Time Operating System (RTOS) એ એવું ઓપરેટિંગ સિસ્ટમ છે જે કડક ટાઈમિંગ કન્સ્ટ્રેઇન્ટ્સ અને પ્રિડિક્ટેબલ રિસ્પોન્સ ટાઈમ સાથે રિયલ-ટાઈમ એપ્લિકેશન્સ હેંડલ કરવા માટે ડિઝાઇન કરવામાં આવે છે.

લાક્ષણિકતાઓ:

કોષ્ટક 7. RTOS Characteristics

લાક્ષણિકતા	વર્ણન
ડિટર્મિનિસ્ટિક	પ્રિડિક્ટેબલ એક્ઝિક્યુશન ટાઈમ
પ્રીએમ્બિટ્વ	હાઇ પ્રાયોરિટી ટાસ્ક લો પ્રાયોરિટીને ઇન્ટરપાટ કરે છે
મલ્ટિટાસ્કિંગ	મલ્ટિપલ ટાસ્ક એક્ઝિક્યુશન
ફાસ્ટ રિસ્પોન્સ	મિનિમલ ઇન્ટરપાટ લેટન્સી
પ્રાયોરિટી-બેસ્ડ	પ્રાયોરિટી આધારિત ટાસ્ક શિજ્યુલિંગ
રિસોર્સ મેનેજમેન્ટ	એફિશિયન્ટ મેમરી અને CPU ઉપયોગ

આકૃતિ 4. RTOS Types



- ટાસ્ક શિજુલિંગ: રાઉન્ડ-રોબિન, પ્રાયોરિટી-બેસ્ડ અલોરિધમ્સ.
- ઇન્ટર-ટાસ્ક કાન્ફુન્ડિશન: સેમાફોર્સ, મેરેજ ક્યુ.
- મેમરી મેનેજમેન્ટ: પ્રિડિક્ટેબિલિટી માટે સ્ટેટિક એલોકેશન.

મેમરી ટ્રીક

“Deterministic Preemptive Multitasking Fast Priority Resource”

OR

પ્રશ્ન 2(a) [3 ગુણા]

AVR ફેબ્રિલી વિશે સમજાવો.

જવાબ

AVR ફેબ્રિલી વર્ગીકરણાઃ

કોષ્ટક 8. AVR Family

AVR પ્રકાર	વિશેષતાઓ
ATtiny	8-32 પિન્સ, બેસિક ફીચર્સ
ATmega	28-100 પિન્સ, કુલ ફીચર્સ
ATxmega	એડવાન્સ ફીચર્સ, DMA

- આર્કિટેક્ચર: 8-bit RISC, હાર્વર્ડ આર્કિટેક્ચર.
- ઇન્સ્ટ્રક્શન સેટ: 130+ ઇન્સ્ટ્રક્શન્સ, સિંગલ સાઇકલ એક્ઝિક્યુશન.
- મેમરી: ફ્લેશ પ્રોગ્રામ મેમરી, SRAM, EEPROM.

મેમરી ટ્રીક

“Tiny-basic mega-full Xmega-advanced”

OR

પ્રશ્ન 2(b) [4 ગુણા]

ATmega32માં કલોક સોર્સની પરંદગી માટે ફ્યૂઝ બિટ્સનું મહત્વ સમજાવો.

જવાબ

કલોક સોર્સ સિલેક્શન:

કોષ્ટક 9. Fuse Bits

ફ્યૂઝ બિટ્સ	કાર્ય
CKSEL3:0	કલોક સોર્સ સિલેક્શન
SUT1:0	સ્ટાર્ટ-અપ ટાઈમ સિલેક્શન

કલોક ઓપ્શન્સ:

કોષ્ટક 10. Clock Options

CKSEL મૂલ્ય	કલોક સોર્સ	ફીકવન્સી
0001	એક્સ્ટર્નલ ક્રિસ્ટલ	1-8 MHz
0010	એક્સ્ટર્નલ ક્રિસ્ટલ	8+ MHz
0100	ઇન્ટર્નલ RC	8 MHz
0000	એક્સ્ટર્નલ કલોક	યુઝર ડિફાઈન્ડ

- ક્રિસ્ટલ સિલેક્શન: એક્સ્ટર્નલ ક્રિસ્ટલ અને કેપેસિટર જરૂરી.
- RC ઓસિલેટર: બિલ્ટ-ઇન, ઓછું એક્ચ્યુરેટ પણ સુવિધાજનક.
- સ્ટાર્ટ-અપ ટાઈમ: ક્રિસ્ટલ સ્ટેબિલાઇઝેશનની મંજૂરી આપે છે.

મેમરી ટ્રીક

“Crystal RC Internal Start-up”

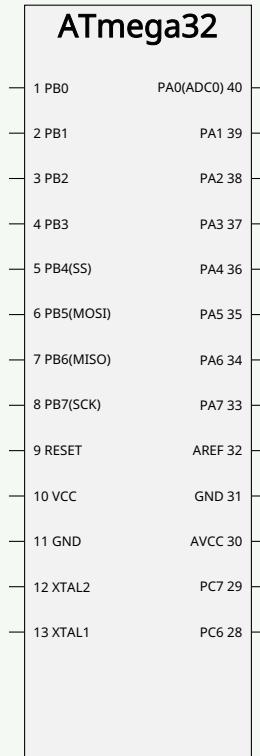
OR

પ્રશ્ન 2(c) [7 ગુણ]

ATmega32નો પિન ડાયાગ્રામ દોરી MISO, MOSI, SCK & AREF Pin નું કાર્ય સમજાવો.

જવાબ

ATmega32 પિન કન્ફિગરેશન:



પિન ફંક્શન્સ:

કોષ્ટક 11. Pin Functions

પિન	કાર્ય	વર્ણન
MOSI	Master Out Slave In	માસ્ટરથી સ્લેવમાં SPI ડેટા આઉટપુટ
MISO	Master In Slave Out	સ્લેવથી માસ્ટરમાં SPI ડેટા ઇનપુટ
SCK	Serial Clock	SPI કલોક સિગ્નલ
AREF	Analog Reference	ADC રેફરન્સ વૉલટેજ

- SPI કમ્યુનિકેશન: MOSI, MISO, SCK મળીને સીરિયલ ડેટા ટ્રાન્સફર માટે કામ કરે છે.
- ADC રેફરન્સ: AREF, ADC કન્વર્જન માટે સ્થિર વૉલટેજ રેફરન્સ પ્રદાન કરે છે.
- પિન માલિટિલેન્ડિસિંગ: આ પિન્સ GPIO તરીકે વૈકલ્પિક કાર્યો ધરાવે છે.

મેમરી ટ્રીક

“MOSI-out MISO-in SCK-clock AREF-reference”

પ્રશ્ન 3(a) [3 ગુણ]

ATmega32 માં DDR I/O રજિસ્ટરની ભૂમિકા સમજાવો.

જવાબ

DDR (Data Direction Register) કાર્યો:

કોષ્ટક 12. DDR Bit Settings

બિટ મૂલ્ય	પિન કન્ફિગરેશન
0	ઇનપુટ પિન
1	આઉટપુટ પિન

- પોર્ટ કંટ્રોલ: દરેક પોર્ટનું અનુરૂપ DDR (DDRA, DDRB, DDRC, DDRD) છે.
- બિટ-વાઈજ કંટ્રોલ: વ્યક્તિગત પિન દિશા કંટ્રોલ.
- ડિફોલ્ટ સ્થિતિ: રીસેટ પછી બધા પિન્સ ઇનપુટ (DDR = 0x00).

કોડ ઉદાહરણ:

```
1 DDRA = 0xFF; // બધા Port A પિન્સ આઉટપુટ તરીકે
2 DDRB = 0x0F; // PB0-PB3 આઉટપુટ, PB4-PB7 ઇનપુટ
```

મેમરી ટ્રીક

“Data Direction Register controls Input/Output”

પ્રશ્ન 3(b) [4 ગુણ]

Port B પરથી ડેટાને રીડ કરાવી Port C પર મોકલવા માટેનો AVR C પ્રોગ્રામ લખો.

જવાબ

પ્રોગ્રામ:

```
1 #include <avr/io.h>
2
3 int main(void)
4 {
5     unsigned char data;
6
7     // Port B ને ઇનપુટ તરીકે કન્ફિગિર કરો
8     DDRB = 0x00;
9
10    // Port C ને આઉટપુટ તરીકે કન્ફિગિર કરો
11    DDRC = 0xFF;
12
13    while(1)
14    {
15        // Port B થી ડેટા રીડ કરો
16        data = PINB;
17
18        // Port C પર ડેટા મોકલો
19        PORTC = data;
20    }
21
22    return 0;
23 }
```

પ્રોગ્રામ સમજૂતી:

- DDRB = 0x00: બધા Port B પિન્સને ઇનપુટ તરીકે સેટ કરે છે.
- DDRC = 0xFF: બધા Port C પિન્સને આઉટપુટ તરીકે સેટ કરે છે.
- PINB: Port B પિન્સની વર્તમાન સ્થિતિ રીડ કરે છે.
- PORTC: Port C આઉટપુટ પિન્સ પર ડેટા લખે છે.

મેમરી ટ્રીક

``Read-PINB Set-DDR Transfer-data Output-PORTC''

પ્રશ્ન 3(c) [7 ગુણ]

PORT B ના પિન નં 1 પર ડોર સેન્સર જોડાયેલ છે અને PORT C ના પિન નં 7 પર LED જોડાયેલ છે. દરવાજા ઉપર લાગેલા સેન્સરને મોનિટર કરતાં રહો અને જ્યારે દરવાજો ખુલે ત્યારે LED ચાલુ થાય તે માટેનો AVR C પ્રોગ્રામ લખો.

જવાબ

પ્રોગ્રામ:

```

1 #include <avr/io.h>
2
3 int main(void)
4 {
5     // PB1 ને ઇનપુટ તરીકે કન્ફિર કરો ડોર( સેન્સર)
6     DDRB &= ~(1<<1); // બટિ 1 ક્લાયિર કરો
7
8     // PC7 ને આઉટપુટ તરીકે કન્ફિર કરો (LED)
9     DDRC |= (1<<7); // બટિ 7 સેટ કરો
10
11    // PB1 માટે પુલાયમ- એરેબલ કરો
12    PORTB |= (1<<1);
13
14    while(1)
15    {
16        // ડોર સેન્સરની સૂચિચેક કરો
17        if(PINB & (1<<1))
18        {
19            // દરવાજો બંધ - LED બંધ કરો
20            PORTC &= ~(1<<7);
21        }
22        else
23        {
24            // દરવાજો ખુલ્લો - LED ચાલુ કરો
25            PORTC |= (1<<7);
26        }
27    }
28
29    return 0;
30 }
```

હાર્ડવેર કનેક્શન:

- ડોર સેન્સર: PB1 અને GND વર્ચે જોડાયેલ.
- LED: કરન્ટ લિમિટિંગ રેઝિસ્ટર દ્વારા PC7 સાથે જોડાયેલ.

મેમરી ટ્રીક

``Door-sensor Configure-pins Open-check LED-control''

OR

પ્રશ્ન 3(a) [3 ગુણ]

AVR C પ્રોગ્રામ ના ડેટા ટાઇપની ચર્ચા કરો.

જવાબ

AVR C ડેટા ટાઇપ્સ:

કોષ્ટક 13. Data Types

ડેટા ટાઇપ	સાઇઝ	રેન્જ
char	8-bit	-128 થી 127
unsigned char	8-bit	0 થી 255
int	16-bit	-32768 થી 32767
unsigned int	16-bit	0 થી 65535
long	32-bit	-2^{31} થી $2^{31}-1$
float	32-bit	IEEE 754 ફોર્મેટ

- મેમરી એફિશિયન્સી: સૌથી નાનો ધોગ્ય ડેટા ટાઇપ વાપરો.
- અનસાઇન્ડ ટાઇપ્સ: ફક્ત પોઝિટિવ વેલ્યુ માટે, રેન્જ બમાવે છે.
- બિટ ફિલ્ડ્સ: સ્પેચિફિક બિટ-વિલ્યુ વેરિએબલ્સ ડિફાઇન કરી શકાય છે.

મેમરી ટ્રીક

“Char-8bit Int-16bit Long-32bit Float-32bit Unsigned-positive”

OR

પ્રશ્ન 3(b) [4 ગુણ]

સિરિયલ કોમ્યુનિકેશન પ્રોટોકોલ સમજાવો.

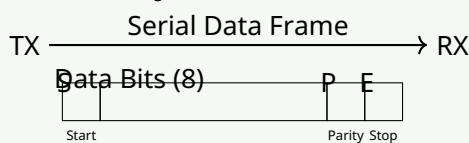
જવાબ

સિરિયલ કોમ્યુનિકેશન પેરામીટર્સ:

કોષ્ટક 14. Serial Parameters

પેરામીટર	વર્ણન
બોડ રેટ	ડેટા ટ્રાન્સમિશન સ્પીડ (બિટ્સ/સેકન્ડ)
ડેટા બિટ્સ	ડેટા બિટ્સની સંખ્યા (5-9)
પેરિટી	અએર ચેકિંગ (None, Even, Odd)
સ્ટોપ બિટ્સ	હેમના અંતનું માર્કર (1 અથવા 2)

આકૃતિ 5. Serial Frame



- એસિંકોન્સ: કોઈ કલોક સિચલ નથી, સ્ટાર્ટ/સ્ટોપ બિટ્સ વાપરે છે.
- RS232 સ્ટાન્ડર્ડ: $\pm 12V$ લેવલ્સ, TTL લેવલ્સમાં કન્વર્ટ થાય છે.
- સામાન્ય બોડ રેટ્સ: 9600, 19200, 38400, 115200.

મેમરી ટ્રીક

``Baud-rate Data-bits Parity-check Stop-bits''

OR

પ્રશ્ન 3(c) [7 ગુણ]

Port B ના પિન નં. 0 અને પિન નં. 1 ને રીડ કરી નીચે આપેલા ટેબલ પ્રમાણે ASCII કેરેક્ટર Port D પર મોકલાવા માટેનો AVR C પ્રોગ્રામ લખો

જવાબ

ટુથ ટેબલ અમલીકરણ:

કોષ્ટક 15. Truth Table

Pin1	Pin0	ઇનપુટ મૂલ્ય	ASCII આઉટપુટ
0	0	0x00	'0' (0x30)
0	1	0x01	'1' (0x31)
1	0	0x02	'2' (0x32)
1	1	0x03	'3' (0x33)

પ્રોગ્રામ:

```

1 #include <avr/io.h>
2
3 int main(void)
4 {
5     unsigned char input;
6
7     // PB1 અને PB0 ને ઇનપુટ તરીકે કન્ફિગર કરો
8     DDRB &= ~((1<<1)|(1<<0));
9
10    // Port D ને આઉટપુટ તરીકે કન્ફિગર કરો
11    DDRD = 0xFF;
12
13    // PB1 અને PB0 માટે પુલઅપ- એનેબલ કરો
14    PORTB |= (1<<1)|(1<<0);
15
16    while(1)
17    {
18        // PB1 અને PB0 રીડ કરો
19        input = PINB & 0x03; // અન્ય બટિસ માસ્ક કરો
20
21        switch(input)
22        {
23            case 0x00: // Pin1=0, Pin0=0
24                PORTD = '0'; // ASCII '0' = 0x30
25                break;
26
27            case 0x01: // Pin1=0, Pin0=1
28                PORTD = '1'; // ASCII '1' = 0x31
29                break;
30
31            case 0x02: // Pin1=1, Pin0=0
32                PORTD = '2'; // ASCII '2' = 0x32
33                break;

```

```

34     case 0x03: // Pin1=1, Pin0=1
35         PORTD = '3'; // ASCII '3' = 0x33
36         break;
37     }
38 }
39 return 0;
40
41
42 }
```

મેમરી ટ્રીક

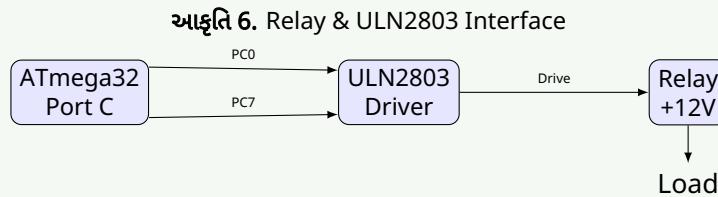
``Mask-inputs ASCII-conversion Truth-table Switch-case''

પ્રશ્ન 4(a) [3 ગુણ]

ATmega32 સાથે રિલે ફ્રાઇવર ULN2803નું ઇન્ટરફેસિંગ ડાયાગ્રામ દોરો.

જવાબ

રિલે ઇન્ટરફેસ ડાયાગ્રામ:



- **ULN2803:** ડાલિંગન ટ્ર૔ન્ઝિસ્ટર એરે, કરન્ટ એમિલાફિકેશન.
- પ્રોટેક્શન ડાયોડ્સ: ઇન્ડિકેટવ લોડ્સ માટે બિલ્ટ-ઇન ફિલાયબેક ડાયોડ્સ.
- રિલે કોઈલ: 12V જરૂરી, ULN2803 આઉટપુટ દ્વારા કંટ્રોલ.

મેમરી ટ્રીક

``ULN-driver Port-control Current-amplify''

પ્રશ્ન 4(b) [4 ગુણ]

પોલિંગ મેથડથી A/D કન્વર્ટરને પ્રોગ્રામ કરવા માટેના સ્ટેપ્સ લખો.

જવાબ

ADC પ્રોગ્રામ્બિંગ સ્ટેપ્સ:

કોષ્ટક 16. ADC Steps

સ્ટેપ	કિયા
1	ADMUX રજિસ્ટર કન્ફિગર કરો (રેફરન્સ, ચેનલ)
2	ADCSRA રજિસ્ટર કન્ફિગર કરો (એનેબલ, પ્રીસ્કેલર)
3	કન્વર્જન સ્ટાર્ટ કરો (ADSC બિટ સેટ કરો)
4	કન્વર્જન પૂર્ણ થવાની રાહ જુઓ (ADIF ફલેગ પોલ કરો)
5	ADCL અને ADCH થી પરિણામ રીડ કરો

કોડ:

```

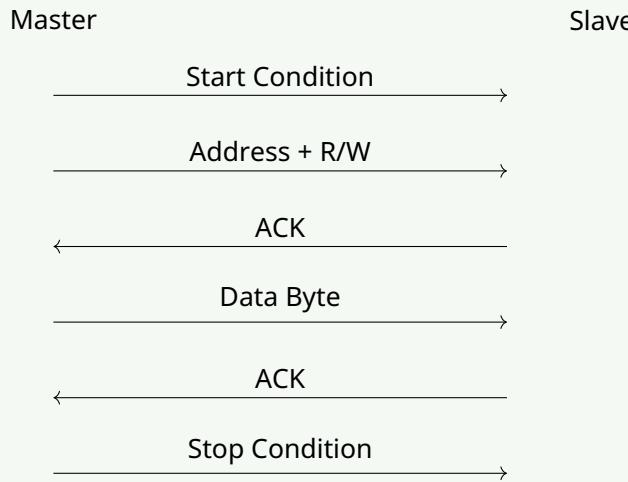
1 // સ્ટેપ 1: ADMUX કન્ફિગર કરો
2 ADMUX = (1<<REFS0); // AVCC રેફરન્સ, ચેનલ 0
3
4 // સ્ટેપ 2: પરીસ્કેલર સાથે ADC એનેબલ કરો
5 ADCSRA = (1<<ADEN)|(1<<ADPS2)|(1<<ADPS1)|(1<<ADPS0);
6
7 // સ્ટેપ 3: કન્વર્જન સ્ટાર્ટ કરો
8 ADCSRA |= (1<<ADSC);
9
10 // સ્ટેપ 4: પૂર્ણતાની રાહ જુઓ
11 while(!(ADCSRA & (1<<ADIF)));
12
13 // સ્ટેપ 5: પરિણામ રીડ કરો
14 result = ADC; // ADCL અને ADCH નું સંયોજન

```

મેમરી ટ્રીક**``Configure-ADMUX Configure-ADCSRA Start-conversion Wait-complete Read-result''****પ્રશ્ન 4(c) [7 ગુણ]****I2C-Two Wire Serial Interface (TWI) પ્રોટોકોલ વિસ્તારવાર સમજાવો.****જવાબ****I2C પ્રોટોકોલ ફીચર્સ:****કોષ્ટક 17. I2C Features**

ફીચર	વર્ણન
બે વાયર	SDA (ડેટા) અને SCL (કલોક)
માલિટિ-માસ્ટર	બહુવિધ માસ્ટર બસ કંટ્રોલ કરી શકે છે
એડ્રેસિંગ	7-bit અથવા 10-bit ડિવાઇસ એડ્રેસ
બાઇડાયરેક્શનલ	બંને દિશામાં ડેટા ફલો

આકૃતિ 7. I2C Sequence



- સ્ટાર્ટ કન્ડિશન: SCL હાઇ હોય ત્યારે SDA લો જાય છે.
- એડ્રેસ ફેમ: 7-bit એડ્રેસ + R/W બિટ.
- ડેટા ફેમ: 8-bit ડેટા + ACK/NACK.
- સ્ટોપ કન્ડિશન: SCL હાઇ હોય ત્યારે SDA હાઇ જાય છે.

રજિસ્ટર્સ: TWCR, TWDR, TWAR, TWSR.

મેમરી ટ્રીક

``Start-Address-Data Control-Status-Address''

OR

પ્રશ્ન 4(a) [3 ગુણા]

8-બિટ ટાઇમરનો ઉપયોગ કરી DC મોટરની સ્પીડ કંટ્રોલ કરવા માટે કોઈ પણ એક PWM મોડ સમજાવો.

જવાબ

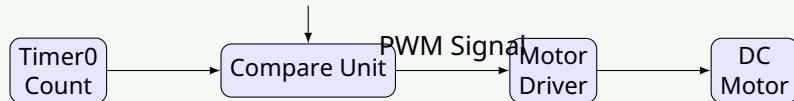
ફાસ્ટ PWM મોડ (મોડ 3):

કોષ્ટક 18. Fast PWM

પેરામીટર	મૂલ્ય
WGM બિટ્સ	WGM01=1, WGM00=1
TOP મૂલ્ય	0xFF (255)
રેજાલ્યુશન	8-bit
ફીકવન્સી	$f_{clk}/(256 \times prescaler)$

આકૃતિ 8. PWM Motor Control

OCRO Value



- ઇયુટી સાઇકલ કંટ્રોલ: OCRO મૂલ્ય મોટરની સ્પીડ નક્કી કરે છે.
- મોટર કંટ્રોલ: વધારે ઇયુટી સાઇકલ = વધારે સ્પીડ.

મેમરી ટ્રીક

``Fast-PWM Timer0 OCR0-control''

OR

પ્રશ્ન 4(b) [4 ગુણ]

SPI ડિવાઇસમાંથી ડેટા રીડ કરવા માટેના સ્ટેપ્સ લખો.

જવાબ

SPI રીડ સ્ટેપ્સ:

કોષ્ટક 19. SPI Steps

સ્ટેપ	ક્રિયા
1	SPI કંટ્રોલ રજિસ્ટર (SPCR) કન્ફિગર કરો
2	સ્લેવ સિલેક્ટ કરવા માટે SS પિન લો કરો
3	SPDR માં ડમી ડેટા લખો
4	ટ્રાન્સમિશન પૂર્ણ થવાની રાહ જુઓ (SPIF ફ્લેગ)
5	SPDR થી રિસીવ કરેલો ડેટા રીડ કરો
6	સ્લેવ ડિસિલેક્ટ કરવા માટે SS પિન હાઇ કરો

કોડ:

```

1 // SPI ને માસ્ટર તરીકે કન્ફિગર કરો
2 SPCR = (1<<SPE)|(1<<MSTR)|(1<<SPR0);
3
4 // સ્લેવ સલિક્ટ કરો
5 PORTB &= ~(1<<SS);
6
7 // ડમી બાઈટ મોકલો
8 SPDR = 0xFF;
9
10 // પૂર્ણતાની રાહ જુઓ
11 while(!(SPSR & (1<<SPIF)));
12
13 // ડેટા રીડ કરો
14 data = SPDR;
15
16 // સ્લેવ ડસિલેક્ટ કરો
17 PORTB |= (1<<SS);

```

મેમરી ટ્રીક

``Configure Select Write-dummy Wait Read-data Deselect''

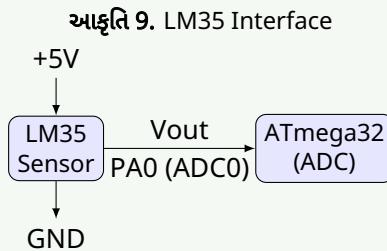
OR

પ્રશ્ન 4(c) [7 ગુણ]

ATmega32 સાથે LM35 ઇન્ટરફેસિંગ ડાયાગ્રામ દોરી સમજાવો.

જવાબ

LM35 ઇન્ટરફેસિંગ:



સ્પેસિફિકેશન્સ:

કોષ્ટક 20. LM35 Specs

પેરામીટર	મૂલ્ય
આઉટપુટ	10mV/°C
રેન્જ	0°C થી 100°C
સપ્લાય	4V થી 30V
એક્યુરરી	±0.5°C

કેલ્ક્યુલેશન:

$$Temp = \frac{ADC \times 5000mV}{1024 \times 10mV/{}^\circ C}$$

મેમરી ટ્રીક

“Voltage-output ADC-conversion Reference-5V Calculation-formula”

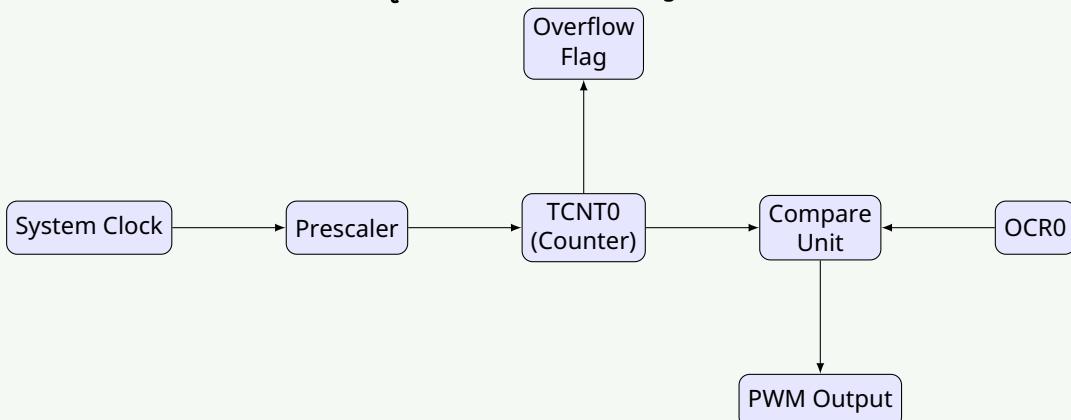
પ્રશ્ન 5(a) [3 ગુણા]

Timer 0 માટે working બ્લોક ડાયાગ્રામ દોરો.

જવાબ

Timer 0 બ્લોક ડાયાગ્રામ:

આકૃતિ 10. Timer 0 Block Diagram



- પ્રીસ્કેલર: કલોક ડિવિઝન (1, 8, 64, 256, 1024).

- કાઉન્ટર: 8-bit અપ કાઉન્ટર (0-255).
- કોમ્પેર યુનિટ: કાઉન્ટરને OCR0 સાથે કોમ્પેર કરે છે.

મેમરી ટ્રીક

``Prescaler Counter Compare Overflow''

પ્રશ્ન 5(b) [4 ગુણ]

ATmega32 સેથે MAX7221 ઇન્ટરફેસિંગ ડાયાગ્રામ દોરો.

જવાબ

MAX7221 ઇન્ટરફેસ:



- ડિસ્પ્લે ડ્રાઇવર: 8-digit 7-segment LED ડ્રાઇવર.
- SPI ઇન્ટરફેસ: સીરિયલ ડેટા ઇનપુટ (DIN, CLK, CS).
- ફીર્ચર્સ: કરન્ટ કંટ્રોલ, BCD ડિકોડ.

મેમરી ટ્રીક

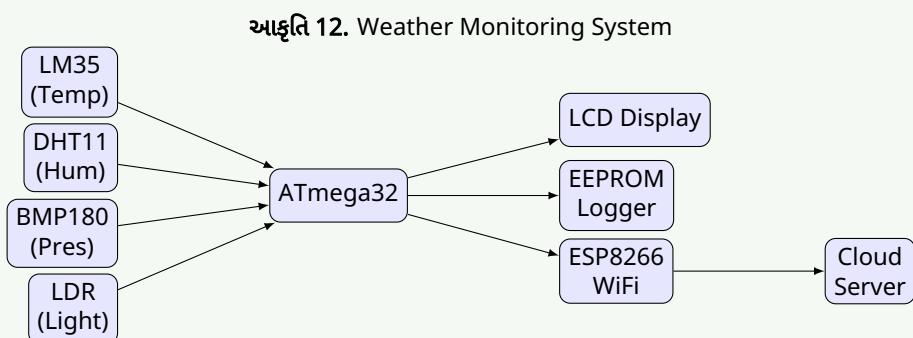
``SPI-interface Current-control Decode-mode Initialize-setup Scan-limit''

પ્રશ્ન 5(c) [7 ગુણ]

વેધર મોનિટરિંગ સિસ્ટમ સમજાવો.

જવાબ

વેધર મોનિટરિંગ સિસ્ટમ:



કોમ્પોનેન્ટ્સ:

કોષ્ટક 21. Components

કોમ્પોનેન્ટ	કાર્ય
LM35	ટેમ્પરેચર માપન
DHT11	લ્યુમિડિટી અને ટેમ્પરેચર
BMP180	વાતાવરણીય દબાણ
ESP8266	WiFi કનેક્ટિવિટી

- રિયલ-ટાઇમ મોનિટરિંગ: સતત સેન્સર ડેટા કલેક્શન.
- રિમોટ એક્સેસ: કલાઉડ અપલોડ માટે WiFi.
- ઓલર્ટ સિસ્ટમ: થેશાહોલ્ડ-બેસ્ડ વોર્નિંગ્સ.

મેમરી ટ્રીક

``Sensors Monitoring Alert Remote Temperature Weather''

OR

પ્રશ્ન 5(a) [3 ગુણા]

ટાઇમર/કાઉન્ટર કંટ્રોલ રજિસ્ટર TCCR0 દોરી સમજાવો.

જવાબ

TCCR0 રજિસ્ટર:

કાણક 22. TCCR0 Layout

7	6	5	4	3	2	1	0
FOC0	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00

- FOC0: Force Output Compare.
- WGM01:00: Waveform Generation Mode (Normal, PWM, CTC).
- COM01:00: Compare Output Mode.
- CS02:00: Clock Select (પ્રીસ્કૉલર સિલેક્શન).

મેમરી ટ્રીક

``Force Waveform Compare Clock-Select''

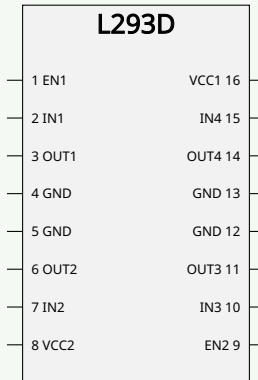
OR

પ્રશ્ન 5(b) [4 ગુણા]

મોટર ડ્રાઇવર L293D નું કાર્ય સમજાવો.

જવાબ

L293D મોટર ડ્રાઇવર:



- ફીચર્સ: ડ્યુઅલ H-બ્રિજ, ચેનલ દીઠ 600mA.
- ઓપરેશન: ડાયરેક્શન અને સ્પીડ (PWM) કંટ્રોલ કરે છે.
- સપ્લાય: VCC1 લોજિક (5V), VCC2 મોટર (36V સુધી).

મેમરી ટ્રીક

“Dual-channel H-bridge Input-control Enable-PWM”

OR

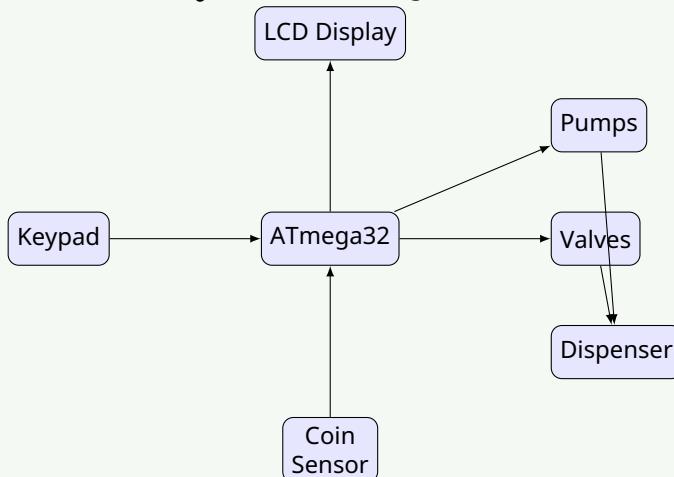
પ્રશ્ન 5(c) [7 ગુણ]

ઓટોમેટિક જૂસ વેન્ડિંગ મશીન સમજાવો.

જવાબ

ઓટોમેટિક જૂસ વેન્ડિંગ મશીન:

આકૃતિ 13. Juice Vending Machine



ઓપરેશન:

- સિલેક્શન: યુઝર કીપેડ વાયા જૂસ સિલેક્ટ કરે છે.
 - પેમેન્ટ: કોઇન સેન્સર પેમેન્ટ વેલિટેડ કરે છે.
 - પ્રોસેસિંગ: MCU મિક્રોસૉપ્ટ માર્કે પંચ/વાલ્વ એક્ટિવેટ કરે છે.
 - ડિસ્પેન્સિંગ: જૂસ ડિસ્પેન્સ થાય છે, LCD પર મેસેજ.
- ફીચર્સ: મલ્ટિપલ ફ્લેવર્સ, ઇન્વેન્ટરી મોનિટરિંગ, ઓટોમેટેડ કલીનિંગ.

મેમરી ટ્રીક

"Juice-selection User-interface Mixing-control Payment-system Sensors-monitoring"