

પ્રશ્ન 1(અ) [3 ગુણ]

રીઅલ ટાઇમ ઓપરેટિંગ સિસ્ટમની લાક્ષણિકતાઓની ચર્ચા કરો.

જવાબ:

કોષ્ટક: RTOS લાક્ષણિકતાઓ

લાક્ષણિકતા	વર્ણન
નિર્ધારિત વર્તન	અનુમાનિત પ્રતિસાદ સમય
સમય મર્યાદા	કઠિન અને નરમ ડેડલાઇન
પ્રાથમિકતા શેડ્યુલિંગ	પ્રાથમિકતા દ્વારા કાર્ય અમલ
સંસાધન વ્યવસ્થાપન	કાર્યક્ષમ મેમરી અને CPU ઉપયોગ

- નિર્ધારિત વર્તન:** સિસ્ટમ ગેરંટીવાળા સમય મર્યાદામાં પ્રતિસાદ આપે છે
- મલ્ટિટાસ્કિંગ સપોર્ટ:** બહુવિધ કાર્યો પ્રાથમિકતા સાથે સમાંતર ચાલે છે
- ઇન્ટરપ્ટ હેન્ડલિંગ:** બાહ્ય ઘટનાઓને ઝડપી પ્રતિસાદ

સ્મરણ સહાયક: "RTOS કાર્યો યોગ્ય રીતે વિતરિત કરે છે"

પ્રશ્ન 1(બ) [4 ગુણ]

AVR I/O પોર્ટ રજિસ્ટરનું વર્ણન કરો.

જવાબ:

કોષ્ટક: AVR I/O પોર્ટ રજિસ્ટર

રજિસ્ટર	કાર્ય	પ્રવેશ
DDRx	ડેટા દિશા રજિસ્ટર	વાંચો/લખો
PORTx	પોર્ટ આઉટપુટ રજિસ્ટર	વાંચો/લખો
PINx	પોર્ટ ઇનપુટ રજિસ્ટર	ફક્ત વાંચો

- DDRx રજિસ્ટર:** પિન દિશા નિયંત્રિત કરે છે (0=ઇનપુટ, 1=આઉટપુટ)
- PORTx રજિસ્ટર:** આઉટપુટ મૂલ્યો સેટ કરે છે અથવા pull-up રેઝિસ્ટર સક્રિય કરે છે
- PINx રજિસ્ટર:** ઇનપુટ ઓપરેશન માટે વર્તમાન પિન સ્થિતિ વાંચે છે

સ્મરણ સહાયક: "દિશા, પોર્ટ, પિન - DPP"

પ્રશ્ન 1(ક) [7 ગુણ]

વિવિધ AVR માઇક્રોકન્ટ્રોલરની સરખામણી કરો અને એમ્બેડેડ સિસ્ટમ માટે માઇક્રોકન્ટ્રોલર પસંદ કરવા માટે કયા પરિબલો ધ્યાનમાં લેવા જોઈએ?

જવાબ:

કોષ્ટક: AVR માઇક્રોકન્ટ્રોલર સરખામણી

લક્ષણ	ATmega8	ATmega32	ATmega128
Flash મેમરી	8KB	32KB	128KB
SRAM	1KB	2KB	4KB
EEPROM	512B	1KB	4KB
I/O પિન	23	32	53
ટાઇમર	3	3	4

પસંદગીના પરિબલો:

- પ્રોસેસિંગ સ્પીડ: એપ્લિકેશન માટે ક્લોક ફ્રીક્વન્સી જરૂરિયાત
- મેમરી જરૂરિયાત: પ્રોગ્રામ અને ડેટા સ્ટોરેજની જરૂર
- I/O જરૂરિયાત: ઇન્ટરફેસિંગ માટે જરૂરી પિનોની સંખ્યા
- પાવર વપરાશ: પોર્ટેબલ ઉપકરણો માટે બેટરી જીવનની વિચારણા
- કિંમત પરિબલ: બજેટ મર્યાદા અને વોલ્યુમ જરૂરિયાત
- ડેવલપમેન્ટ ટૂલ્સ: કમ્પાઇલર અને ડીબગરની ઉપલબ્ધતા

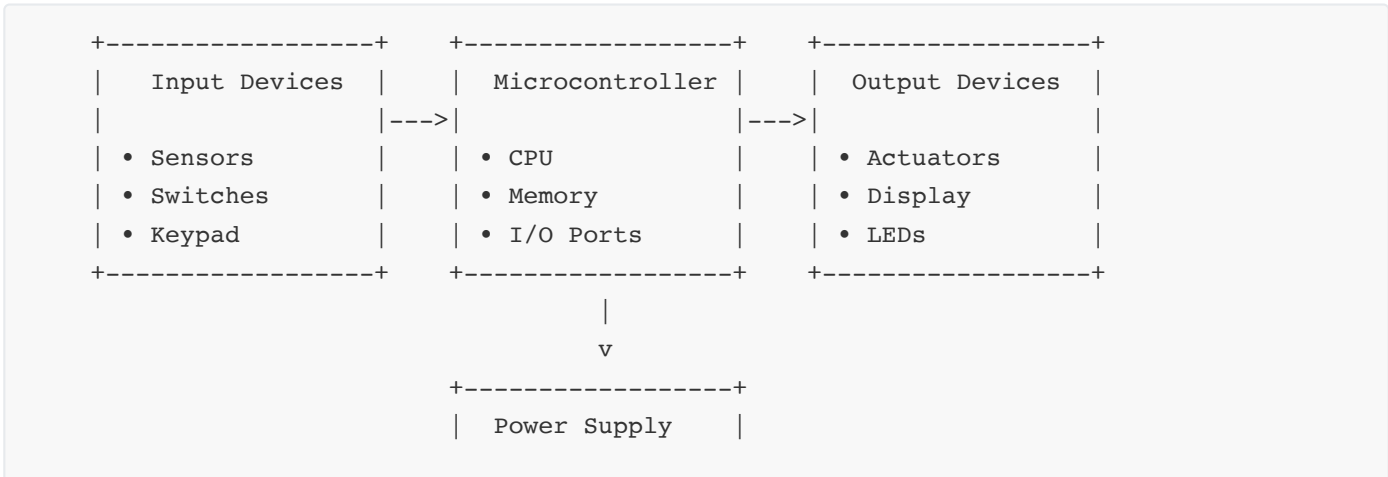
સ્મરણ સહાયક: "સ્પીડ, મેમરી, I/O, પાવર, કિંમત, ટૂલ્સ - SMIPCT"

પ્રશ્ન 1(ક અથવા) [7 ગુણ]

એમ્બેડેડ સિસ્ટમનો સામાન્ય બ્લોક ડાયાગ્રામ દોરો અને સમજાવો.

જવાબ:

આકૃતિ:



- Voltage Reg.
 - Battery
- +-----+

ઘટકો:

- ઇનપુટ વિભાગ: સેન્સર અને સ્વિચ સિસ્ટમને ડેટા પ્રદાન કરે છે
- પ્રોસેસિંગ યુનિટ: માઇક્રોકન્ટ્રોલર પ્રોગ્રામ ચલાવે છે અને ઓપરેશન કંટ્રોલ કરે છે
- આઉટપુટ વિભાગ: પરિણામો દર્શાવે છે અને બાહ્ય ઉપકરણો કંટ્રોલ કરે છે
- પાવર સપ્લાય: બધા ઘટકોને નિયંત્રિત પાવર પ્રદાન કરે છે
- મેમરી: પ્રોગ્રામ કોડ અને ડેટાને કાયમી ધોરણે સંગ્રહિત કરે છે
- કમ્યુનિકેશન: સીરીયલ/વાયરલેસ દ્વારા બાહ્ય સિસ્ટમ સાથે ઇન્ટરફેસ

સ્મરણ સહાયક: "ઇનપુટ, પ્રોસેસ, આઉટપુટ, પાવર, મેમરી, કમ્યુનિકેશન - IPOPMC"

પ્રશ્ન 2(અ) [3 ગુણ]

ATMega32 ના EEPROM સાથે SRAM ની સરખામણી કરો.

જવાબ:

કોષ્ટક: SRAM વિ EEPROM સરખામણી

પેરામીટર	SRAM	EEPROM
કદ	2KB	1KB
અસ્થિરતા	અસ્થિર	બિન-અસ્થિર
પ્રવેશ ઝડપ	ઝડપી	ધીમી
લેખન ચક્ર	અમર્યાદિત	100,000 ચક્ર

- ડેટા રીટેન્શન: SRAM પાવર-ઓફ પર ડેટા ખોવાય છે, EEPROM ડેટા જાળવે છે
- ઉપયોગ હેતુ: SRAM વેરિએબલ માટે, EEPROM કૉન્ફિગરેશન ડેટા માટે

સ્મરણ સહાયક: "SRAM ઝડપી પણ ભૂલી જાય, EEPROM ટકી રહે"

પ્રશ્ન 2(બ) [4 ગુણ]

ટાઇમર/કાઉન્ટર 0 ઓપરેશન મોડની સૂચિ બનાવો અને કોઈપણને સમજાવો.

જવાબ:

કોષ્ટક: Timer0 ઓપરેશન મોડ

મોડ	નામ	વર્ણન
0	સામાન્ય	0xFF સુધી ગણતરી, ઓવરફ્લો
1	PWM ફેઝ કરેક્ટ	ફેઝ કરેક્શન સાથે PWM
2	CTC	કંપેર પર ટાઇમર ક્લિયર
3	ફાસ્ટ PWM	ઉચ્ચ ફ્રીક્વન્સી PWM

સામાન્ય મોડ સમજૂતી:

- કાઉન્ટર ઑપરેશન: સતત 0x00 થી 0xFF સુધી ગણતરી કરે છે
- ઓવરફ્લો ફ્લેગ: કાઉન્ટર 0x00 પર ઓવરફ્લો થાય છે ત્યારે TOV0 ફ્લેગ સેટ થાય છે
- ઇન્ટરપ્ટ જનરેશન: ઓવરફ્લો કન્ડિશન પર ઇન્ટરપ્ટ જનરેટ કરી શકે છે

સ્મરણ સહાયક: "સામાન્ય ગણો, PWM પલ્સ કરે, CTC ક્લિયર કરે"

પ્રશ્ન 2(ક) [7 ગુણ]

સ્કેચ સાથે, ATmega32 ની દરેક પિનનું કાર્ય ઓળખો અને લખો.

જવાબ:

આકૃતિ: ATmega32 પિન કોન્ફિગરેશન

ATmega32			
+-----+			
(XCK/T0) PB0	1	40	PA0 (ADC0)
(T1) PB1	2	39	PA1 (ADC1)
(INT2/AIN0) PB2	3	38	PA2 (ADC2)
(OC0/AIN1) PB3	4	37	PA3 (ADC3)
(SS) PB4	5	36	PA4 (ADC4)
(MOSI) PB5	6	35	PA5 (ADC5)
(MISO) PB6	7	34	PA6 (ADC6)
(SCK) PB7	8	33	PA7 (ADC7)
RST	9	32	AREF
VCC	10	31	GND
GND	11	30	AVCC
XTAL2	12	29	PC7 (TOSC2)
XTAL1	13	28	PC6 (TOSC1)
(RXD) PD0	14	27	PC5 (TDI)
(TXD) PD1	15	26	PC4 (TDO)
(INT0) PD2	16	25	PC3 (TMS)
(INT1) PD3	17	24	PC2 (TCK)
(OC1B) PD4	18	23	PC1 (SDA)
(OC1A) PD5	19	22	PC0 (SCL)
(ICP1) PD6	20	21	PD7 (OC2)
+-----+			

પિન કાર્યો:

- **પોર્ટ A:** 8-બિટ ADC ઇનપુટ પિન (PA0-PA7)
- **પોર્ટ B:** SPI કમ્યુનિકેશન અને ટાઇમર કાર્યો
- **પોર્ટ C:** JTAG ઇન્ટરફેસ અને I2C કમ્યુનિકેશન
- **પોર્ટ D:** UART કમ્યુનિકેશન અને બાહ્ય ઇન્ટરપ્ટ
- **પાવર પિન:** VCC, GND, AVCC એનાલોગ સપ્લાય માટે
- **ક્રિસ્ટલ પિન:** XTAL1, XTAL2 બાહ્ય ઓસિલેટર માટે

સ્મરણ સહાયક: "એનાલોગ-A, બસ-B, કમ્યુનિકેશન-C, ડેટા-D"

પ્રશ્ન 2(અ અથવા) [3 ગુણ]

ATmega32 ની ડેટા મેમરીની રચના સમજાવો.

જવાબ:

કોષ્ટક: ATmega32 મેમરી ઓર્ગેનાઇઝેશન

મેમરી પ્રકાર	એડ્રેસ રેન્જ	કદ
રજિસ્ટર	0x00-0x1F	32 બાઇટ
I/O રજિસ્ટર	0x20-0x5F	64 બાઇટ
આંતરિક SRAM	0x60-0x25F	2048 બાઇટ

- **સામાન્ય હેતુ રજિસ્ટર:** અંકગણિત ઓપરેશન માટે R0-R31
- **I/O મેમરી જગ્યા:** પેરિફેરલ માટે કંટ્રોલ રજિસ્ટર
- **આંતરિક SRAM:** પ્રોગ્રામ એક્ઝિક્યુશન દરમિયાન વેરિએબલ સ્ટોરેજ

સ્મરણ સહાયક: "રજિસ્ટર, I/O, SRAM - RIS"

પ્રશ્ન 2(બ અથવા) [4 ગુણ]

ટાઇમર/કાઉન્ટર 0 ના TIFR અને TCCR રજિસ્ટર દોરો.

જવાબ:

આકૃતિ: Timer0 રજિસ્ટર

TIFR (ટાઇમર ઇન્ટરપ્ટ ફ્લેગ રજિસ્ટર)

```

+---+---+---+---+---+---+---+---+
| - | - | - | - | - | - | OCF2 | TOV2 | TOV0 | OCF0 | TOV1 | OCF1A | ICF1 | OCF1B |
+---+---+---+---+---+---+---+---+
    7     6     5     4     3     2     1     0

```

TCCR0 (ટાઇમર/કાઉન્ટર કંટ્રોલ રજિસ્ટર 0)

```

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

```

બિટ કાર્યો:

- **TOV0:** Timer0 ઓવરફ્લો ફ્લેગ બિટ
- **OCF0:** Timer0 આઉટપુટ કંપેર મેચ ફ્લેગ
- **CS02:CS00:** પ્રીસ્કેલર માટે ક્લોક સિલેક્ટ બિટ
- **WGM01:WGM00:** વેવફોર્મ જનરેશન મોડ બિટ

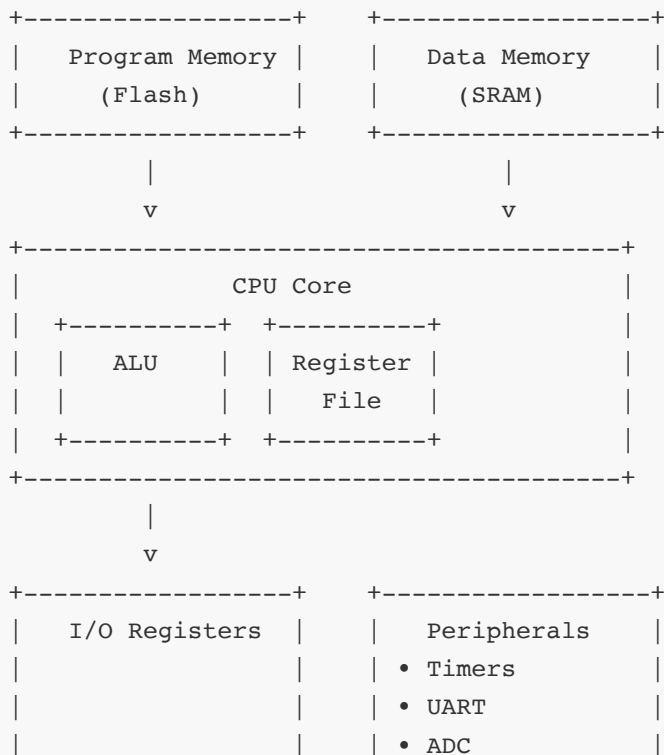
સમરણ સહાયક: "TIFR ફ્લેગ બતાવે, TCCR ક્લોક કંટ્રોલ કરે"

પ્રશ્ન 2(ક અથવા) [7 ગુણ]

AVR માઇક્રોકન્ટ્રોલરનો સામાન્ય બ્લોક ડાયાગ્રામ દોરો અને સમજાવો.

જવાબ:

આકૃતિ: AVR આર્કિટેક્ચર



+-----+ +-----+

ઘટકો:

- **CPU કોર:** ઇન્સ્ટ્રક્શન એક્ઝિક્યુટ કરે છે અને સિસ્ટમ ઓપરેશન કંટ્રોલ કરે છે
- **પ્રોગ્રામ મેમરી:** બિન-અસ્થિર flash માં એપ્લિકેશન કોડ સ્ટોર કરે છે
- **ડેટા મેમરી:** વેરિએબલ અને સ્ટેક માટે અસ્થાયી સ્ટોરેજ
- **ALU:** અંકગણિત અને તાર્કિક ઓપરેશન કરે છે
- **રજિસ્ટર ફાઇલ:** 32 સામાન્ય-હેતુ વર્કિંગ રજિસ્ટર
- **I/O સિસ્ટમ:** બાહ્ય હાર્ડવેર ઘટકો સાથે ઇન્ટરફેસ
- **પેરિફેરલ:** બિલ્ટ-ઇન મોડ્યુલ જેમ કે ટાઇમર, UART, ADC

સ્મરણ સહાયક: "CPU પ્રોગ્રામ, ડેટા, I/O, પેરિફેરલ કંટ્રોલ કરે - CPDIP"

પ્રશ્ન 3(અ) [3 ગુણ]

10 ms વિલંબ સાથે સતત પોર્ટ B ના તમામ બિટ્સને ટૉગલ કરવા માટે AVR C પ્રોગ્રામ લખો.

જવાબ:

```
#include <avr/io.h>
#include <util/delay.h>

int main()
{
    DDRB = 0xFF;          // પોર્ટ B ને આઉટપુટ તરીકે સેટ કરો

    while(1)
    {
        PORTB = 0xFF;      // બધા બિટ હાઇ સેટ કરો
        _delay_ms(10);     // 10ms વિલંબ
        PORTB = 0x00;      // બધા બિટ લો સેટ કરો
        _delay_ms(10);     // 10ms વિલંબ
    }
}
```

મુખ્ય મુદ્દાઓ:

- **DDRB = 0xFF:** પોર્ટ B ના બધા પિનને આઉટપુટ તરીકે કોન્ફિગર કરે છે
- **PORTB ટૉગલ:** 0xFF અને 0x00 વચ્ચે બદલાય છે

સ્મરણ સહાયક: "DDR દિશા, PORT આઉટપુટ"

પ્રશ્ન 3(બ) [4 ગુણ]

MAX232 નું કાર્ય સમજાવો.

જવાબ:

કોષ્ટક: MAX232 કાર્યો

કાર્ય	વર્ણન
લેવલ કન્વર્ઝન	TTL થી RS232 વોલ્ટેજ લેવલ
ચાર્જ પંપ	+5V સપ્લાયથી ±10V જનરેટ કરે છે
લાઇન ડ્રાઇવર	બે ટ્રાન્સમિટ ડ્રાઇવર
લાઇન રિસીવર	બે રિસીવ રિસીવર

- વોલ્ટેજ કન્વર્ઝન: 0-5V TTL ને ±12V RS232 લેવલમાં કન્વર્ટ કરે છે
- સીરીયલ કમ્યુનિકેશન: માઇક્રોકન્ટ્રોલરને PC સાથે કમ્યુનિકેટ કરવા સક્ષમ બનાવે છે
- ડ્યુઅલ ચેનલ: બે-દિશાવાળી કમ્યુનિકેશનને સમાંતર સપોર્ટ કરે છે

સ્મરણ સહાયક: "MAX232 માઇક્રોકન્ટ્રોલરને PC સાથે મળાવે છે"

પ્રશ્ન 3(ક) [7 ગુણ]

કેટલાક વિલંબ સાથે સતત PORTC ના તમામ બિટ્સને ટોંગલ કરવા માટે AVR C પ્રોગ્રામ લખો. વિલંબ જનરેટ કરવા માટે પ્રીસ્કેલર વિકલ્પ વગર અને ટાઇમર 0, મોડ 0 નો ઉપયોગ કરવો.

જવાબ:

```
#include <avr/io.h>

void timer0_delay()
{
    TCNT0 = 0;           // કાઉન્ટર ઇનિશિયલાઇઝ કરો
    TCCR0 = 0x01;        // કોઈ પ્રીસ્કેલર નહીં, સામાન્ય મોડ
    while(!(TIFR & (1<<TOV0))); // ઓવરફ્લો માટે રાહ જુઓ
    TIFR |= (1<<TOV0);    // ઓવરફ્લો ફ્લેગ ક્લિયર કરો
    TCCR0 = 0;           // ટાઇમર સ્ટોપ કરો
}

int main()
{
    DDRC = 0xFF;         // પોર્ટ C આઉટપુટ તરીકે

    while(1)
    {
        PORTC = 0xFF;    // બધા બિટ હાઇ
        for(int i=0; i<100; i++)
            timer0_delay(); // બહુવિધ વિલંબ

        PORTC = 0x00;    // બધા બિટ લો
        for(int i=0; i<100; i++)
```



```

        timer0_delay(); // બહુવિધ વિલંબ
    }
}

```

મુખ્ય લક્ષણો:

- **Timer0 સામાન્ય મોડ:** 0 થી 255 સુધી ગણે છે પછી ઓવરફ્લો
- **કોઈ પ્રીસ્કેલર નહીં:** ટાઇમર સિસ્ટમ ક્લોક સ્પીડે ચાલે છે
- **ઓવરફ્લો ડિટેક્શન:** TOV0 ફ્લેગ ટાઇમર ઓવરફ્લો દર્શાવે છે
- **વિલંબ જનરેશન:** બહુવિધ ટાઇમર ચક્ર દૃશ્યમાન વિલંબ બનાવે છે

સ્મરણ સહાયક: "ટાઇમર ગણે, ઓવરફ્લો ફ્લેગ, વિલંબ જનરેટ કરે"

પ્રશ્ન 3(અ અથવા) [3 ગુણ]

EEPROM ના સ્થાન 0X011F માં #30h સ્ટોર કરવા માટે AVR C પ્રોગ્રામ લખો.

જવાબ:

```

#include <avr/io.h>
#include <avr/eeprom.h>

int main()
{
    eeprom_write_byte((uint8_t*)0x011F, 0x30);
    return 0;
}

```

વૈકલ્પિક પદ્ધતિ:

```

#include <avr/io.h>

int main()
{
    while(EECR & (1<<EEWE)); // અગાઉના લેખન માટે રાહ જુઓ
    EEAR = 0x011F;           // એડ્રેસ સેટ કરો
    EEDR = 0x30;             // ડેટા સેટ કરો
    EECR |= (1<<EEMWE);      // માસ્ટર લેખન સક્ષમ
    EECR |= (1<<EEWE);       // લેખન સક્ષમ
}

```

સ્મરણ સહાયક: "એડ્રેસ, ડેટા, માસ્ટર, લેખન - ADMW"

પ્રશ્ન 3(બ અથવા) [4 ગુણ]

C માં AVR પ્રોગ્રામિંગ માટે વિવિધ ડેટા પ્રકારોની ચર્ચા કરો.

જવાબ:

કોષ્ટક: AVR C ડેટા પ્રકાર

ડેટા પ્રકાર	કદ	રેન્જ
char	1 બાઇટ	-128 થી 127
unsigned char	1 બાઇટ	0 થી 255
int	2 બાઇટ	-32768 થી 32767
unsigned int	2 બાઇટ	0 થી 65535
long	4 બાઇટ	-2^{31} થી $2^{31}-1$
float	4 બાઇટ	IEEE 754 ફોર્મેટ

- **મેમરી કાર્યક્ષમતા:** સૌથી નાના યોગ્ય ડેટા પ્રકારની પસંદગી કરો
- **Unsigned પ્રકાર:** જ્યારે નેગેટિવ મૂલ્યોની જરૂર ન હોય ત્યારે ઉપયોગ કરો
- **Integer અંકગણિત:** ફ્લોટિંગ-પોઇન્ટ ઓપરેશન કરતાં ઝડપી

સ્મરણ સહાયક: "મેમરી કાર્યક્ષમતા માટે યોગ્ય કદ પસંદ કરો"

પ્રશ્ન 3(ક અથવા) [7 ગુણ]

સીરીયલ ડેટા ટ્રાન્સમિશન માટે AVR C પ્રોગ્રામ્સ લખો.

જવાબ:

```
#include <avr/io.h>

void uart_init(unsigned int baud)
{
    UBRRH = (unsigned char)(baud>>8);
    UBRL = (unsigned char)baud;
    UCSRB = (1<<TXEN); // ટ્રાન્સમિટર સક્ષમ કરો
    UCSRC = (1<<URSEL) | (3<<UCSZ0); // 8-બિટ ડેટા
}

void uart_transmit(unsigned char data)
{
    while(!(UCSRA & (1<<UDRE))); // ખાલી બફર માટે રાહ જુઓ
    UDR = data; // ડેટા મોકલો
}

void uart_send_string(char *str)
{
    while(*str)
    {
```

```

        uart_transmit(*str++);
    }
}

int main()
{
    uart_init(51);                // 8MHz પર 9600 baud

    while(1)
    {
        uart_send_string("Hello World\r\n");
        for(long i=0; i<100000; i++); // વિલંબ
    }
}

```

મુખ્ય ઘટકો:

- **બોડ રેટ સેટિંગ:** UBRR રજિસ્ટર કમ્યુનિકેશન સ્પીડ સેટ કરે છે
- **ટ્રાન્સમિટ સક્ષમ:** TXEN બિટ UART ટ્રાન્સમિટર સક્ષમ કરે છે
- **ડેટા ટ્રાન્સમિશન:** UDR રજિસ્ટર ટ્રાન્સમિટ કરવાનો ડેટા હોલ્ડ કરે છે
- **બફર ચેક:** UDRE ફ્લેગ ટ્રાન્સમિટ બફર ખાલી દર્શાવે છે

સ્મરણ સહાયક: "ઇનિટ, સક્ષમ, ચેક, ટ્રાન્સમિટ - IECT"

પ્રશ્ન 4(અ) [3 ગુણ]

ADMUX રજિસ્ટર સમજાવો.

જવાબ:

કોષ્ટક: ADMUX રજિસ્ટર બિટ્સ

બિટ	નામ	કાર્ય
REFS1:0	રેફરન્સ સિલેક્ટ	વોલ્ટેજ રેફરન્સ પસંદગી
ADLAR	લેફ્ટ એડજસ્ટ	પરિણામ ડાબે એડજસ્ટમેન્ટ
MUX4:0	ચેનલ સિલેક્ટ	ADC ઇનપુટ ચેનલ પસંદગી

- **રેફરન્સ વોલ્ટેજ:** આંતરિક/બાહ્ય વોલ્ટેજ રેફરન્સ પસંદ કરે છે
- **પરિણામ ફોર્મેટ:** ADLAR બિટ 10-બિટ પરિણામ એલાઇનમેન્ટ એડજસ્ટ કરે છે
- **ચેનલ પસંદગી:** MUX બિટ્સ કયા ADC પિનને વાંચવો તે પસંદ કરે છે

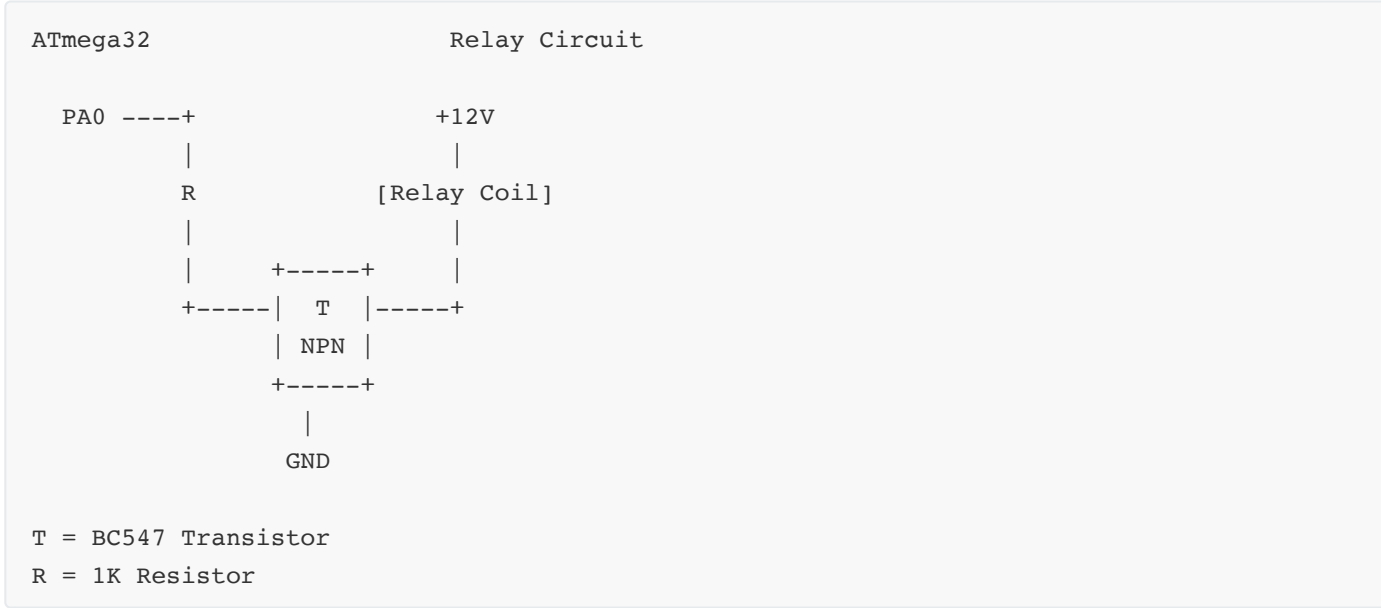
સ્મરણ સહાયક: "રેફરન્સ, એડજસ્ટ, ચેનલ - RAC"

પ્રશ્ન 4(બ) [4 ગુણ]

ATmega32 સાથે ઇન્ટરફેસિંગ રિલે દોરો અને સમજાવો.

જવાબ:

આકૃતિ: રિલે ઇન્ટરફેસિંગ



ઘટકો:

- ટ્રાન્ઝિસ્ટર સ્વિચ: BC547 NPN ટ્રાન્ઝિસ્ટર ઇલેક્ટ્રોનિક સ્વિચ તરીકે કામ કરે છે
- બેઝ રેઝિસ્ટર: $1K\Omega$ માઇક્રોકન્ટ્રોલરથી બેઝ કરન્ટ મર્યાદિત કરે છે
- રિલે કોઇલ: 12V રિલે બાહ્ય હાઇ-પાવર ઉપકરણો ઓપરેટ કરે છે
- પ્રોટેક્શન ડાયોડ: બેક EMF થી બચાવવા માટે ફ્રીવ્હીલિંગ ડાયોડ

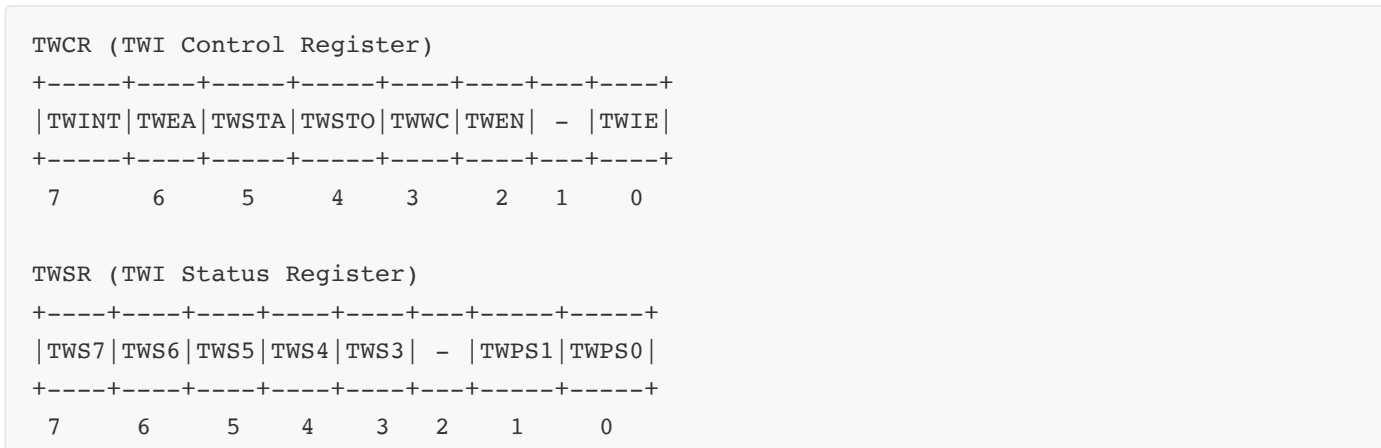
સમરણ સહાયક: "માઇક્રો ટ્રાન્ઝિસ્ટર કંટ્રોલ કરે, ટ્રાન્ઝિસ્ટર રિલે કંટ્રોલ કરે"

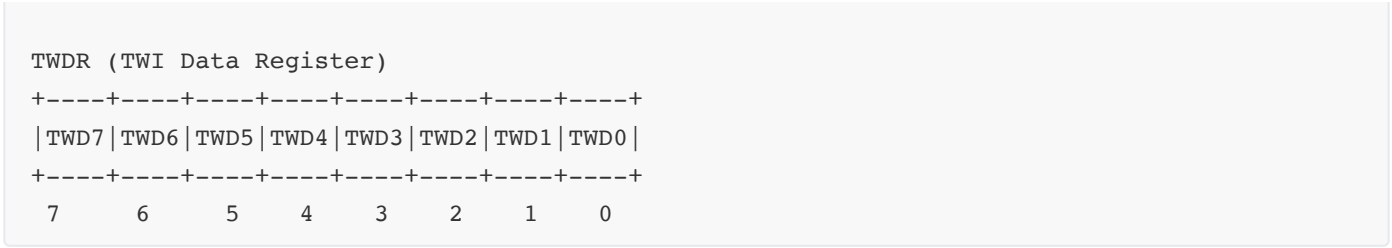
પ્રશ્ન 4(ક) [7 ગુણ]

AVR માં TWI રજિસ્ટર દોરો અને સમજાવો.

જવાબ:

આકૃતિ: TWI રજિસ્ટર સ્ટ્રક્ચર





રજિસ્ટર કાર્યો:

- **TWCR:** TWI ઓપરેશન અને ઇન્ટરપ્ટ હેન્ડલિંગ કંટ્રોલ કરે છે
- **TWSR:** સ્ટેટસ માહિતી અને પ્રીસ્કેલર સેટિંગ પ્રદાન કરે છે
- **TWDR:** ટ્રાન્સમિશન/રિસેપ્શન માટે ડેટા હોલ્ડ કરે છે
- **TWAR:** સ્લેવ તરીકે ઓપરેટ કરતી વખતે સ્લેવ એડ્રેસ સેટ કરે છે
- **TWBR:** TWI કમ્યુનિકેશન માટે બિટ રેટ સેટ કરે છે
- **TWINT:** ઇન્ટરપ્ટ ફ્લેગ 1 લખીને ક્લિયર થાય છે
- **Start/Stop:** TWSTA અને TWSTO I2C કન્ડિશન કંટ્રોલ કરે છે

સ્મરણ સહાયક: "કંટ્રોલ, સ્ટેટસ, ડેટા, એડ્રેસ, બિટ રેટ - CSDAB"

પ્રશ્ન 4(અ અથવા) [3 ગુણ]

ADCSRA રજિસ્ટર સમજાવો.

જવાબ:

કોષ્ટક: ADCSRA રજિસ્ટર બિટ્સ

બિટ	નામ	કાર્ય
ADEN	ADC સક્ષમ	ADC મોક્યુલ સક્ષમ કરે છે
ADSC	કન્વર્ઝન શરૂ કરો	ADC કન્વર્ઝન શરૂ કરે છે
ADATE	ઓટો ટ્રિગર	ઓટો ટ્રિગર મોડ સક્ષમ કરે છે
ADIF	ઇન્ટરપ્ટ ફ્લેગ	ADC કન્વર્ઝન પૂર્ણ ફ્લેગ
ADIE	ઇન્ટરપ્ટ સક્ષમ	ADC ઇન્ટરપ્ટ સક્ષમ કરે છે
ADPS2:0	પ્રીસ્કેલર	ADC ક્લોક પ્રીસ્કેલર સેટ કરે છે

- **ADC કંટ્રોલ:** ADEN ADC સક્ષમ કરે છે, ADSC કન્વર્ઝન શરૂ કરે છે
- **ઇન્ટરપ્ટ સિસ્ટમ:** કન્વર્ઝન પૂર્ણ થાય ત્યારે ADIF ફ્લેગ સેટ થાય છે

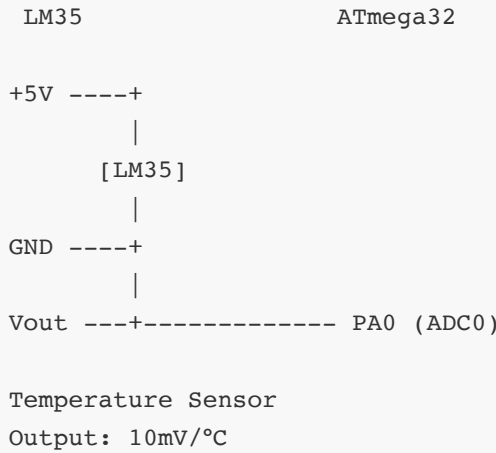
સ્મરણ સહાયક: "સક્ષમ, શરૂ, ટ્રિગર, ઇન્ટરપ્ટ, પ્રીસ્કેલ - ESTIP"

પ્રશ્ન 4(બ અથવા) [4 ગુણ]

ATmega32 સાથે LM35 નું ઇન્ટરફેસિંગ દોરો અને સમજાવો.

જવાબ:

આકૃતિ: LM35 ઇન્ટરફેસિંગ



કનેક્શન વિગતો:

- પાવર સપ્લાય: LM35 ને +5V અને ગ્રાઉન્ડ કનેક્શનની જરૂર છે
- આઉટપુટ વોલ્ટેજ: પ્રતિ ડિગ્રી સેલ્સિયસ 10mV ઉત્પન્ન કરે છે
- ADC ઇનપુટ: LM35 આઉટપુટને ADC ચેનલ (PA0) સાથે કનેક્ટ કરો
- ટેમ્પરેચર ગણતરી: $^{\circ}\text{C} = (\text{ADC_Value} \times 5000\text{mV}) / (1024 \times 10\text{mV})$

કોડ ઉદાહરણ:

```
float temp = (adc_read() * 5.0 * 100.0) / 1024.0;
```

સ્મરણ સહાયક: "LM35 પ્રતિ ડિગ્રી 10mV આપે છે"

પ્રશ્ન 4(ક અથવા) [7 ગુણ]

ATmega32 સાથે MAX7221 નો ઉપયોગ કરીને બહુવિધ 7-સેગમેન્ટ ડિસ્પ્લેના ઇન્ટરફેસિંગ દોરો અને સમજાવો.

જવાબ:

આકૃતિ: MAX7221 ઇન્ટરફેસિંગ

ATmega32	MAX7221	7-Segment Displays
PB5 (MOSI)	----- DIN	DIG0 ---- Display 1
PB7 (SCK)	----- CLK	DIG1 ---- Display 2
PB4 (SS)	----- CS	DIG2 ---- Display 3
		DIG3 ---- Display 4
+5V	----- VCC	DIG4 ---- Display 5
GND	----- GND	DIG5 ---- Display 6
		DIG6 ---- Display 7

```
SEGA ---- Common segments
SEGB      to all displays
SEGC
SEGD
SEGE
SEGF
SEGG
SEGDP
```

લક્ષણો:

- **SPI કમ્યુનિકેશન:** કંટ્રોલ માટે સીરીયલ પેરિફેરલ ઇન્ટરફેસ ઉપયોગ કરે છે
- **બહુવિધ ડિસ્પ્લે:** 8 સુધી સેવન-સેગમેન્ટ ડિસ્પ્લે કંટ્રોલ કરે છે
- **ઓટોમેટિક સ્કેનિંગ:** MAX7221 મલ્ટિપ્લેક્સિંગ ઓટોમેટિક હેન્ડલ કરે છે
- **બ્રાઇટનેસ કંટ્રોલ:** સોફ્ટવેર-કંટ્રોલ્ડ બ્રાઇટનેસ લેવલ
- **ડીકોડ મોડ:** બિલ્ટ-ઇન BCD થી 7-સેગમેન્ટ ડીકોડર
- **ઓછા ઘટકો:** જરૂરી બાહ્ય ઘટકો ઘટાડે છે

મુખ્ય રજિસ્ટર:

- **ડીકોડ મોડ રજિસ્ટર:** BCD ડીકોડિંગ સક્ષમ/અક્ષમ કરે છે
- **ઇન્ટેન્સિટી રજિસ્ટર:** ડિસ્પ્લે બ્રાઇટનેસ કંટ્રોલ કરે છે
- **સ્કેન લિમિટ રજિસ્ટર:** સક્રિય ડિસ્પ્લેની સંખ્યા સેટ કરે છે
- **શટડાઉન રજિસ્ટર:** સામાન્ય ઓપરેશન અથવા શટડાઉન મોડ

સ્મરણ સહાયક: "SPI બહુવિધ ડિસ્પ્લે માટે સીરીયલ ડેટા મોકલે છે"

પ્રશ્ન 5(અ) [3 ગુણ]

SPCR રજિસ્ટર સમજાવો.

જવાબ:

કોષ્ટક: SPCR રજિસ્ટર બિટ્સ

- સ્મરણ સહાયક: "ઇન્ટરપ્ટ, સક્ષમ, ડેટા, માસ્ટર, ક્લોક સેટિંગ્સ - IEDMC"**

પ્રશ્ન 5(બ) [4 ગુણ]

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

જવાબ:

આફ્રતિ: DC મોટર ઇન્ટરફેસિંગ

```

ATmega32                L293D                        DC Motor

PA0 ----- IN1      OUT1 -----+
PA1 ----- IN2      OUT2 -----+   [Motor]
                                   |    M
+5V ----- VCC1     VCC2 ----- +12V |
GND ----- GND      GND ----- GND  |
PA2 ----- EN1                               |
                                           |
Input Logic Table:                       |
IN1  IN2  Motor                          |
  0    0  Stop                           |
  0    1  CCW                             |
  1    0  CW                              |
  1    1  Brake                          |

```

ઘટકો:

- **L293D ડ્રાઇવર:** મોટર કંટ્રોલ માટે કરન્ટ એમ્પ્લિફિકેશન પ્રદાન કરે છે
- **પાવર સપ્લાય:** લૉજિક માટે +5V, મોટર પાવર માટે +12V
- **કંટ્રોલ સિગ્નલ:** IN1, IN2 મોટરની દિશા નક્કી કરે છે

- સક્ષમ પિન: EN1 મોટર ઓન/ઓફ અને સ્પીડ (PWM) કંટ્રોલ કરે છે

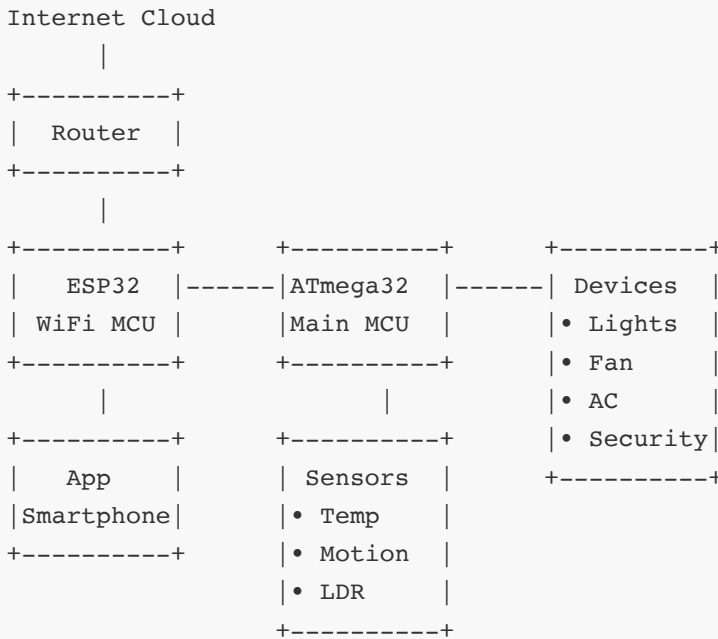
સ્મરણ સહાયક: "લૉજિક દિશા કંટ્રોલ કરે, સક્ષમ સ્પીડ કંટ્રોલ કરે"

પ્રશ્ન 5(ક) [7 ગુણ]

IoT આધારિત હોમ ઓટોમેશન સિસ્ટમ સમજાવો.

જવાબ:

આકૃતિ: IoT હોમ ઓટોમેશન સિસ્ટમ



સિસ્ટમ ઘટકો:

- ઇન્ટરનેટ કનેક્ટિવિટી: WiFi મોડ્યુલ સિસ્ટમને ઇન્ટરનેટ સાથે કનેક્ટ કરે છે
- મોબાઇલ એપ્લિકેશન: રિમોટ કંટ્રોલ અને મોનિટરિંગ માટે યુઝર ઇન્ટરફેસ
- સેન્સર નેટવર્ક: ઓટોમેશન માટે ટેમ્પરેચર, મોશન, લાઇટ સેન્સર
- કંટ્રોલ ઉપકરણો: રિલે ઘરના ઉપકરણો અને લાઇટ કંટ્રોલ કરે છે
- સેન્ટ્રલ કંટ્રોલર: માઇક્રોકંટ્રોલર કમાન્ડ અને સેન્સર ડેટા પ્રોસેસ કરે છે
- ક્લાઉડ સેવાઓ: ડેટા સ્ટોર કરે છે અને રિમોટ એક્સેસ સક્ષમ કરે છે

લક્ષણો:

- રિમોટ કંટ્રોલ: ઇન્ટરનેટ દ્વારા ગમે ત્યાંથી ઉપકરણો કંટ્રોલ કરો
- ઓટોમેશન: સેન્સર રીડિંગ આધારે ઓટોમેટિક કંટ્રોલ
- એનર્જી સેવિંગ: સ્માર્ટ શેડ્યુલિંગ પાવર વપરાશ ઘટાડે છે
- સુરક્ષા મોનિટરિંગ: સુરક્ષા માટે મોશન સેન્સર અને કેમેરા
- ડેટા લૉગિંગ: વિશ્લેષણ માટે ઐતિહાસિક ડેટા સ્ટોરેજ

સ્મરણ સહાયક: "ઇન્ટરનેટ ફોનને ઘરના ઉપકરણો સાથે જોડે છે - IPHD"

પ્રશ્ન 5(અ અથવા) [3 ગુણ]

SPSR રજિસ્ટર સમજાવો.

જવાબ:

કોષ્ટક: SPSR રજિસ્ટર બિટ્સ

બિટ	નામ	કાર્ય
SPIF	ઇન્ટરપ્ટ ફ્લેગ	SPI ટ્રાન્સફર પૂર્ણ ફ્લેગ
WCOL	રાઇટ કોલિશન	ડેટા કોલિશન એરર ફ્લેગ
SPI2X	ડબલ સ્પીડ	SPI ક્લોક રેટ બમણી કરે છે

- ટ્રાન્સફર પૂર્ણ: SPIF ફ્લેગ SPI ટ્રાન્સમિશન સમાપ્ત થયું દર્શાવે છે
- કોલિશન ડિટેક્શન: WCOL ફ્લેગ રાઇટ કોલિશન થયું બતાવે છે
- સ્પીડ કંટ્રોલ: SPI2X સેટ કરવાથી કમ્યુનિકેશન સ્પીડ બમણી થાય છે

સ્મરણ સહાયક: "ફ્લેગ, કોલિશન, સ્પીડ - FCS"

પ્રશ્ન 5(બ અથવા) [4 ગુણ]

L293D મોટર ડ્રાઇવર IC નો પિન ડાયાગ્રામ દોરો અને સમજાવો.

જવાબ:

આકૃતિ: L293D પિન કોન્ફિગરેશન

L293D (16-pin DIP)			
+-----+			
EN1	1	16	VCC1
IN1	2	15	IN4
OUT1	3	14	OUT4
GND	4	13	GND
GND	5	12	GND
OUT2	6	11	OUT3
IN2	7	10	IN3
VCC2	8	9	EN2
+-----+			

પિન કાર્યો:

- સક્ષમ પિન (EN1, EN2): PWM દ્વારા મોટર ઓન/ઓફ અને સ્પીડ કંટ્રોલ કરે છે
- ઇનપુટ પિન (IN1-IN4): માઇક્રોકન્ટ્રોલરથી લૉજિક ઇનપુટ

- **આઉટપુટ પિન (OUT1-OUT4):** મોટર માટે હાઇ કરન્ટ આઉટપુટ
- **પાવર સપ્લાય (VCC1):** IC ઓપરેશન માટે +5V લૉજિક સપ્લાય
- **મોટર સપ્લાય (VCC2):** મોટર પાવર માટે +12V સપ્લાય
- **ગ્રાઉન્ડ પિન:** હીટ ડિસિપેશન માટે બહુવિધ ગ્રાઉન્ડ કનેક્શન

લક્ષણો:

- **ડ્યુઅલ H-બ્રિજ:** બે DC મોટર સમાંતર કંટ્રોલ કરી શકે છે
- **કરન્ટ કેપેસિટી:** પ્રતિ ચેનલ 600mA, 1.2A પીક
- **પ્રોટેક્શન:** મોટર પ્રોટેક્શન માટે બિલ્ટ-ઇન ફ્લાયબેક ડાયોડ

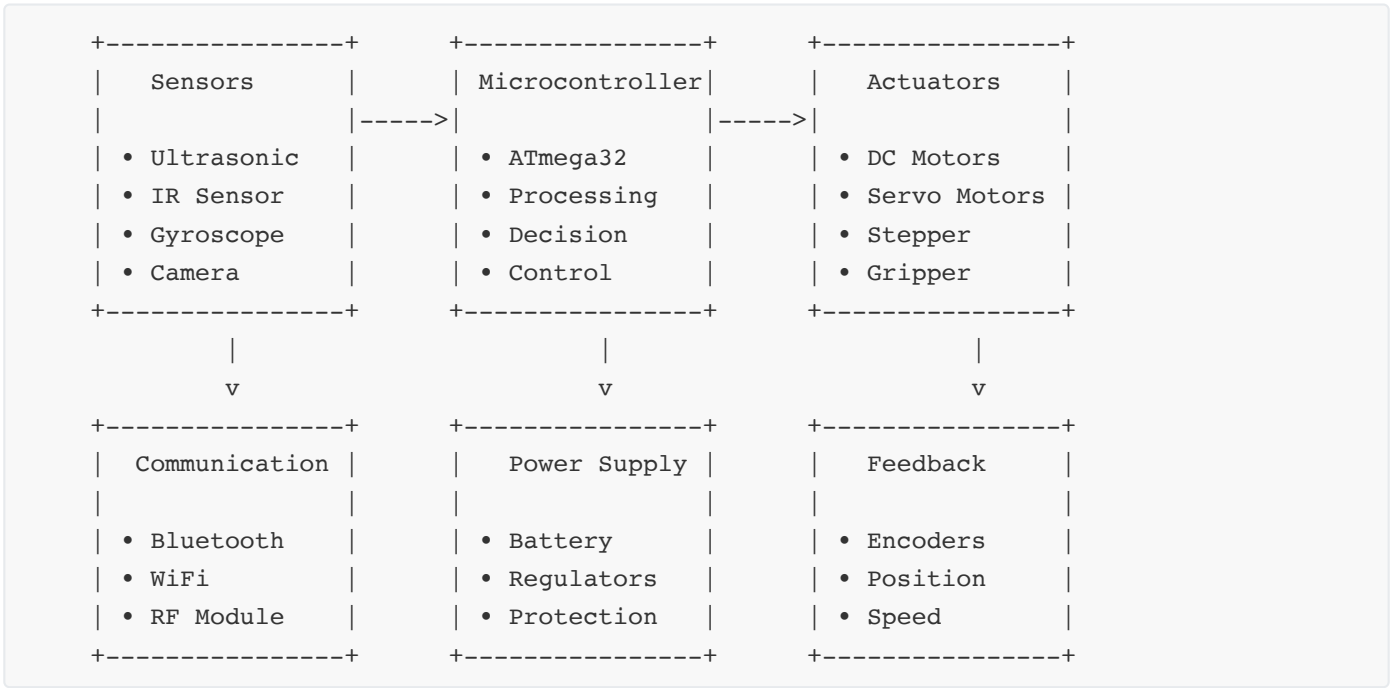
સ્મરણ સહાયક: "સક્ષમ, ઇનપુટ, આઉટપુટ, પાવર - EIOP"

પ્રશ્ન 5(ક અથવા) [7 ગુણ]

મોટરાઇઝ્ડ કંટ્રોલ રોબોટિક્સ સિસ્ટમ સમજાવો.

જવાબ:

આકૃતિ: રોબોટિક્સ કંટ્રોલ સિસ્ટમ



સિસ્ટમ ઘટકો:

કોષ્ટક: રોબોટિક્સ સિસ્ટમ એલિમેન્ટ્સ

ઘટક	કાર્ય	ઉદાહરણો
સેન્સર	પર્યાવરણ સેન્સિંગ	અલ્ટ્રાસોનિક, IR, કેમેરા
કંટ્રોલર	નિર્ણય લેવો	ATmega32, Arduino
એક્ઝ્યુએટર	ભૌતિક હલનચલન	મોટર, સર્વો
કમ્યુનિકેશન	રિમોટ કંટ્રોલ	બ્લૂટૂથ, WiFi
પાવર	એનર્જી સપ્લાય	બેટરી, રેગ્યુલેટર
ફીડબેક	પોઝિશન સેન્સિંગ	એન્કોડર, જાયરોસ્કોપ

કંટ્રોલ અલ્ગોરિથમ:

- સેન્સ:** સેન્સર ઉપયોગ કરીને પર્યાવરણથી ડેટા એકત્રિત કરો
- પ્રોસેસ:** સેન્સર ડેટાનું વિશ્લેષણ કરો અને નિર્ણયો લો
- એક્ટ:** નિર્ણયો આધારે મોટર અને એક્ઝ્યુએટર કંટ્રોલ કરો
- ફીડબેક:** વાસ્તવિક હલનચલન મોનિટર કરો અને કંટ્રોલ એડજસ્ટ કરો
- કમ્યુનિકેટ:** સ્ટેટસ મોકલો અને વાયરલેસ કમાન્ડ રિસીવ કરો

એપ્લિકેશન:

- સ્વાયત નેવિગેશન:** રોબોટ સેન્સર ઉપયોગ કરીને સ્વતંત્ર રીતે મૂવ કરે છે
- ઓબ્જેક્ટ મેનિપ્યુલેશન:** પિક અને પ્લેસ કાર્યો માટે ગ્રિપર કંટ્રોલ
- રિમોટ ઓપરેશન:** વાયરલેસ કમ્યુનિકેશન દ્વારા મેન્યુઅલ કંટ્રોલ
- પાથ ફોલોવિંગ:** લાઇન ફોલોવિંગ અથવા પૂર્વનિર્ધારિત 3ટ નેવિગેશન
- ઓબ્સ્ટેકલ એવોઇડન્સ:** અવરોધોની આસપાસ ડાયનેમિક પાથ પ્લાનિંગ

પ્રોગ્રામિંગ સ્ટ્રક્ચર:

```
while(1) {  
    read_sensors();  
    process_data();  
    make_decision();  
    control_motors();  
    check_feedback();  
    communicate_status();  
}
```

સ્મરણ સહાયક: "સેન્સ, પ્રોસેસ, એક્ટ, ફીડબેક, કમ્યુનિકેટ - SPACF"