

Osnovi digitalne elektronike IR

vežba 3

Odsek za elektroniku

Elektrotehnički fakultet,
Univerzitet u Beogradu

2018/2019

1 Periferije za serijsku komunikaciju

- Serijska komunikacija
- USCI Periferija
 - UART Mod - Opis
 - UART Mod - Podešavanje takta
 - UART Mod - Registri

2 Primeri

- Korišćenje USCI periferije u UART modu
- Korišćenje prekida USCI periferije u UART modu

Pregled

1 Periferije za serijsku komunikaciju

- Serijska komunikacija
- USCI Periferija
 - UART Mod - Opis
 - UART Mod - Podešavanje takta
 - UART Mod - Registri

2 Primeri

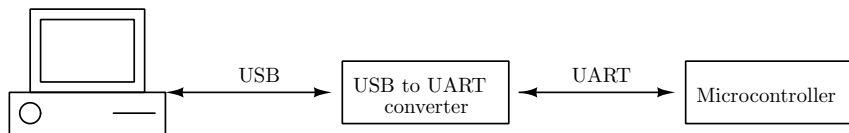
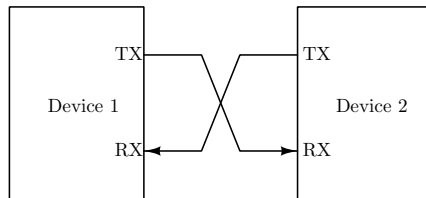
- Korišćenje USCI periferije u UART modu
- Korišćenje prekida USCI periferije u UART modu

Serial vs Parallel

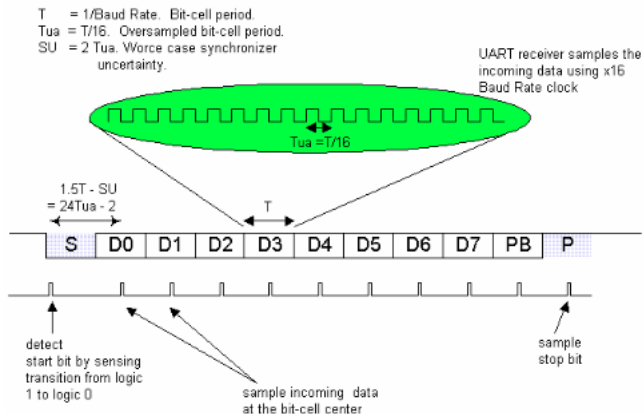
Paralelna komunikacija realizuje slanje više bita istovremeno. Kao takva zahteva više resursa (pinova paralelnog porta) ali omogućava slanje podataka većom brzinom. Neki od interfejsa paralelne komunikacije su: *ISA, ATA, SCSI, PCI*

Serijska komunikacija realizuje slanje jednog bita u posmatranom trenutku. Kao takva zahteva manje resursa ali se podaci šalju manjom brzinom. Neki od interfejsa za serijsku komunikaciju su: *RS-232, RS485, I2C, SPI, UART*

Implementacija UART komunikacije



UART data frame



USCI - Universal Serial Communication Interface

Serijska komunikacija na MSP430 realizovana je posredstvom USCI periferije

USCI moduli podržavaju više serijskih protokola

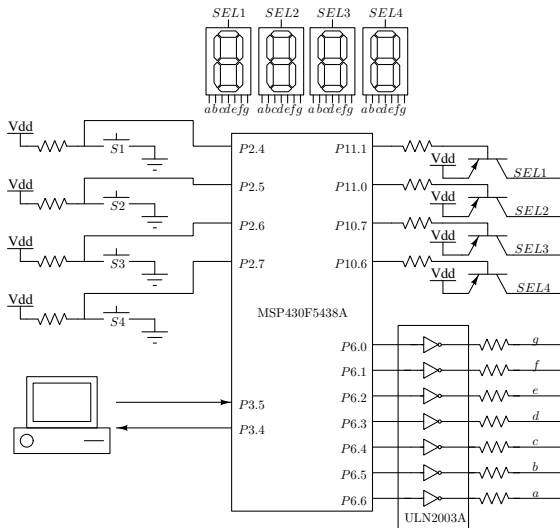
USCI_A x ($x = 0 - 3$)

- UART
- Oblikovanje signala za IrDA komunikaciju
- Automatsko detektovanje brzine prenosa za LIN komunikaciju
- SPI

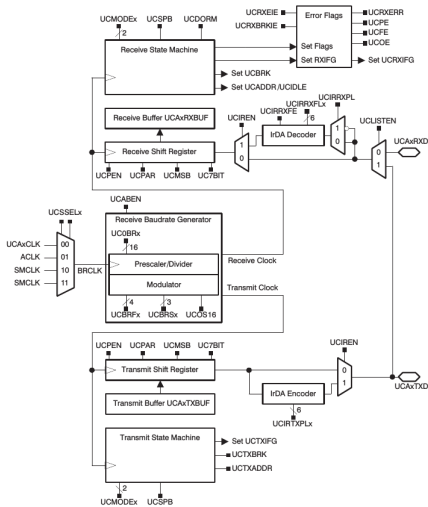
USCI_B x ($x = 0 - 3$)

- I²C
- SPI

Opis hardvera



UART mod 1/3



UART Mod 2/3

Bira se sa `UCAxCTL0.UCSYNC=0`

Podatak dužine 7 ili 8 bita i bit parnosti

Nezavisan rad prijemnika i predajnika

Baferisani prijemni i predajni registri

Ugrađena podrška za multiprocesorsku komunikaciju

UART mod 3/3

Mogućnost buđenja iz LPMx moda na startnu ivicu signala na ulazu u prijemnik

Mogućnost podešavanja brzine prenosa u širokom opsegu vrednosti i sa velikom rezolucijom

Hardverska detekcija grešaka u prenosu

Prekid za prijemnik i predajnik

UART prekidi

USCI poseduje jedan prekidni vektor koji se koristi i za prijemnik i za predajnik

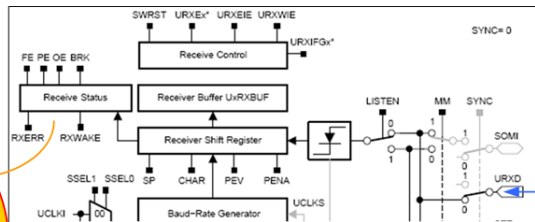
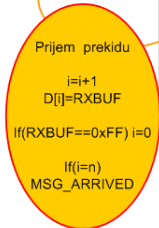
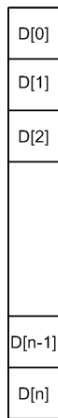
Prekid prijemnika se događa kada je podatak primljen i upisan u `UCAxRXBUF`

Prekid predajnika se događa kada je sadržaj predajnog registra `UCAxTXBUF` prebačen u izlazni pomerački registar i može se upisati novi podatak

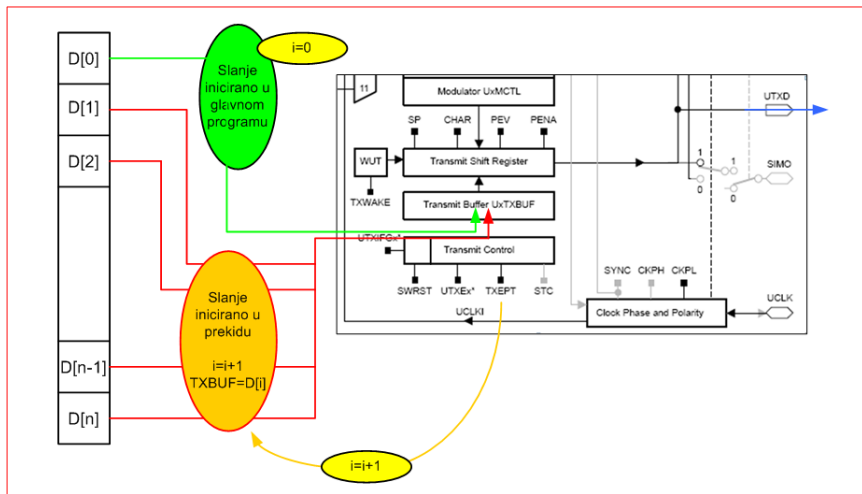
Registar `UCAxIV` sadrži informaciju o aktivnom zahtevu za prekid najvišeg nivoa

- prekid prijemnika ima viši prioritet od prekida predajnika

Interrupt driven receiving



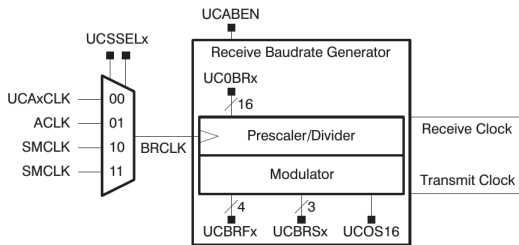
Interrupt driven transmitting



Primer prekidne rutine

```
USCI_UART_ISR
    ADD    &UCA0IV, PC    ; Add offset to jump table
    RETI   ; Vector 0: No interrupt
    JMP    RXIFG_ISR     ; Vector 2: RXIFG
TXIFG_ISR
    ...      ; Task starts here
    RETI   ; Return
RXIFG_ISR
    ...      ; Task starts here
    RETI   ; Return
```

Baud rate generator



Podržana su dva režima rada koji se biraju sa UCOS16

Faktor N kojim se deli učestanost $BRCLK$ u cilju dobijanja željene bitske učestanosti se dobija kao

$$N = \frac{f_{BRCLK}}{\text{Baudrate}}$$

Low-Frequency režim (UCOS16=0)

Omogućava generisanje željenog *baud rate*-a na osnovu izvora takta niske učestanosti - smanjena potrošnja, mada mogu da se koriste i izvori visoke učestanosti

Maksimalan *baud rate* je jedna trećina učestanosti takta BRCLK

Podешavanje

$$UCBR_x = \text{INT}(N)$$

$$UCBR_{SX} = \text{round} [(N - \text{INT}(N)) \times 8]$$

Oversampling režim (UCOS16=1)

Omogućava generisanje željenog *baud rate*-a na osnovu izvora takta visoke učestanosti

Vrši se *oversampling* sa 16 puta većom učestanosti

Maksimalan *baud rate* je jedna šesnaestina učestanosti takta BRCLK

Podešavanje

$$UCBR_x = \text{INT}(N/16)$$

$$UCBRF_x = \text{round} [(N/16 - \text{INT}(N/16)) \times 16]$$

UCAxCTL0

Figure 34-12. UCAxCTL0 Register

7	6	5	4	3	2	1	0
UCPEN	UCPAR	UCMSB	UC7BIT	UCSPB	UCMODEx		UCSYNC
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0

Modify only when UCSWRST = 1.

Table 34-7. UCAxCTL0 Register Description

Bit	Field	Type	Reset	Description
7	UCPEN	RW	0h	Parity enable 0b = Parity disabled 1b = Parity enabled. Parity bit is generated (UCAxTXD) and expected (UCAxRXD). In address-bit multiprocessor mode, the address bit is included in the parity calculation.
6	UCPAR	RW	0h	Parity select. UCPAR is not used when parity is disabled. 0b = Odd parity 1b = Even parity
5	UCMSB	RW	0h	MSB first select. Controls the direction of the receive and transmit shift register. 0b = LSB first 1b = MSB first
4	UC7BIT	RW	0h	Character length. Selects 7-bit or 8-bit character length. 0b = 8-bit data 1b = 7-bit data
3	UCSPB	RW	0h	Stop bit select. Number of stop bits. 0b = One stop bit 1b = Two stop bits
2-1	UCMODEx	RW	0h	USCI mode. The UCMODEx bits select the asynchronous mode when UCSYNC = 0. 00b = UART mode 01b = Idle-line multiprocessor mode 10b = Address-bit multiprocessor mode 11b = UART mode with automatic baud-rate detection
0	UCSYNC	RW	0h	Synchronous mode enable 0b = Asynchronous mode 1b = Synchronous mode

UCAxCTL1

Figure 34-13. UCAxCTL1 Register

7	6	5	4	3	2	1	0
UCSSELx	UCRXEIE	UCBRKIE	UCDORM	UCTXADDR	UCTXBRK	UCSWRST	
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-1

Modify only when UCSWRST = 1.

Table 34-8. UCAxCTL1 Register Description

Bit	Field	Type	Reset	Description
7-6	UCSSELx	RW	0h	USCI clock source select. These bits select the BRCLK source clock. 00b = UCAxCLK (external USCI clock) 01b = ACLK 10b = SMCLK 11b = SMCLK
5	UCRXEIE	RW	0h	Receive erroneous-character interrupt enable 0b = Erroneous characters rejected and UCRXIFG is not set. 1b = Erroneous characters received set UCRXIFG.
4	UCBRKIE	RW	0h	Receive break character interrupt enable 0b = Received break characters do not set UCRXIFG. 1b = Received break characters set UCRXIFG.
3	UCDORM	RW	0h	Dormant. Puts USCI into sleep mode. 0b = Not dormant. All received characters set UCRXIFG. 1b = Dormant. Only characters that are preceded by an idle-line or with address bit set UCRXIFG. In UART mode with automatic baud-rate detection, only the combination of a break and synch field sets UCRXIFG.
2	UCTXADDR	RW	0h	Transmit address. Next frame to be transmitted is marked as address, depending on the selected multiprocessor mode. 0b = Next frame transmitted is data. 1b = Next frame transmitted is an address.
1	UCTXBRK	RW	0h	Transmit break. Transmits a break with the next write to the transmit buffer. In UART mode with automatic baud-rate detection, 055h must be written into UCAxTXBUF to generate the required break/synch fields. Otherwise, 0h must be written into the transmit buffer. 0b = Next frame transmitted is not a break. 1b = Next frame transmitted is a break or a break/synch.
0	UCSWRST	RW	1h	Software reset enable 0b = Disabled. USCI reset released for operation. 1b = Enabled. USCI logic held in reset state.

UCA_xIFGFigure 34-24. UCA_xIFG RegisterTable 34-19. UCA_xIFG Register Description

Bit	Field	Type	Reset	Description
7-2	Reserved	R	0h	Reserved. Always reads as 0.
1	UCTXIFG	RW	1h	Transmit interrupt flag. UCTXIFG is set when UCA _x TXBUF empty. 0b = No interrupt pending 1b = Interrupt pending
0	UCRXIFG	RW	0h	Receive interrupt flag. UCRXIFG is set when UCA _x RXBUF has received a complete character. 0b = No interrupt pending 1b = Interrupt pending

UCAxIE

Figure 34-23. UCAxIE Register



Table 34-18. UCAxIE Register Description

Bit	Field	Type	Reset	Description
7-2	Reserved	R	0h	Reserved. Always reads as 0.
1	UCTXIE	RW	0h	Transmit interrupt enable 0b = Interrupt disabled 1b = Interrupt enabled
0	UCRXIE	RW	0h	Receive interrupt enable 0b = Interrupt disabled 1b = Interrupt enabled

Pregled

1 Periferije za serijsku komunikaciju

- Serijska komunikacija
- USCI Periferija
 - UART Mod - Opis
 - UART Mod - Podešavanje takta
 - UART Mod - Registri

2 Primeri

- Korišćenje USCI periferije u UART modu
- Korišćenje prekida USCI periferije u UART modu

1. Zadatak - Prijem i slanje cifara putem UARTa

Zadatak

Potrebno je napisati program koji na sedmosegmentnom displeju ispisuje cifre koje korisnik unese u serijsku konzolu na računaru. Pritiskom na taster S4 na računar se šalje cifra koja je trenutno ispisana na sedmosegmentnom displeju.

Napomena

Komunikacija se odvija putem UARTa koji ima boudrate 9200bps, gde se prenosi 8 bita sa jednim stop bitom i bez bita parnosti (8N1)

1. Zadatak - Prijem i slanje cifara putem UARTa

Rešenje

Ovaj primer ilustruje način korišćenja USCI periferije u UART modu. Da bi primanje i slanje karaktera bilo moguće neophodno je ispravno inicijalizovati ovu periferiju. Inicijalizacija podrazumeva sledeći niz koraka:

- Selektovati alternativnu funkciju na pinovima mikrokontrolera koji se koriste kao TX i RX pinovi UARTa (Videti *DataSheet*)
- Periferiju dovesti u stanje softverskog reseta setovanjem odgovarajućeg bita u kontrolnom registru
- Podesiti izvor takta
- Podesiti boudrate
- Izvesti periferiju iz stanja reseta brisanjem odgovarajućeg bita u kontrolnom registru

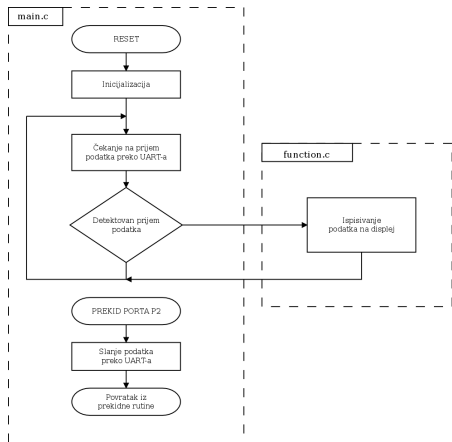
1. Zadatak - Prijem i slanje cifara putem UARTa

Slanje i prijem

Slanje jednog bajta putem UARTa vrši se upisom vrednosti u UCAXTX registar. Bajt primljen putem UART smešten je u UCAXRX registar. Algoritam koji opisuje slanje i prijem podataka putem uarta prikazan je na slici

Kôd

Videti primer *v3-z1-uart* u materijalima



2. Zadatak - Prijem i slanje cifara putem UARTa

Zadatak

Napisati program koji ima istu funkcionalnost kao i zadatak 1. ali u ovom slučaju prijem cifre treba da bude realizovan u prekidnoj rutini.

Napomena

Komunikacija se odvija putem UARTa koji ima boudrate 9200bps, gde se prenosi 8 bita sa jednim stop bitom i bez bita parnosti (8N1)

2. Zadatak - Prijem i slanje cifara putem UARTa

Rešenje

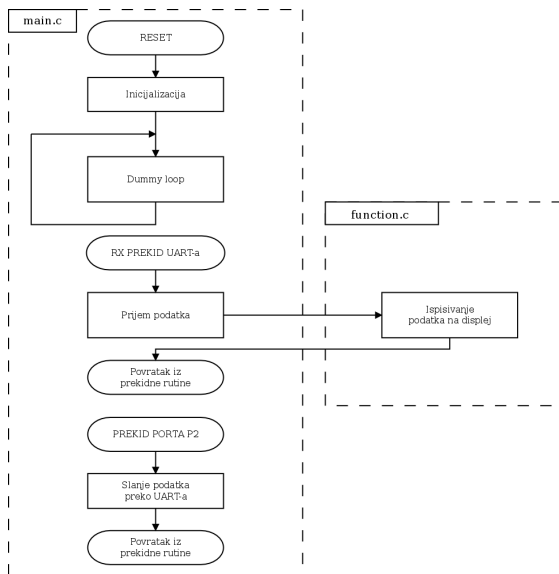
U ovom primeru ilustrovan je način korišćenja USCI prekida nakon prijema podatka. Da bi korišćenje USCI periferije u UART modu bilo moguće neophodno je uraditi niz koraka u cilju inicijalizacije periferije. Potrebno je naglasiti da je u ovom primeru u koracima inicijalizacije dodat još jedan korak koji dozvoljava prekide USCI periferije.

Algoritam koji opisuje način realizacije ovog primera prikazan je na slici na narednom slajdu

Kôd

Videti primer *v3-z2-uart-isr* u materijalima

2. Zadatak - Prijem i slanje cifara putem UARTa



Kraj trećeg dela...