

DOKUMENTACIJA PROJEKTA „MAGNETIC CARD READER“

Projekt iz kolegija
Ugradbeni računalni sustavi

Mentor: izv. prof. dr. sc. Mladen Tomić
Asistent: asist. Arian Skoki
Studenti: Mauro Copetti, Rea Prpić, Karlo Sečen

Tehnički fakultet Rijeka
Preddiplomski sveučilišni studij računarstva
30. kolovoz 2021.

SADRŽAJ

1.	SAŽETAK.....	3
2.	OPIS	4
3.	USART	5
a.	USART protokol	5
b.	USART UDR registar	6
c.	USART UCSRA registar	7
d.	USART UCSRB registar	9
e.	USART UCSRC registar	11
f.	USART UBRRL i UBRRH registri	14
4.	HARDVERSKJE KOMPONENTE.....	15
a.	ATmega16A mikrokontroler	15
b.	16x2 LCD ekran	17
e.	Shema spajanja komponenti	20
5.	ZAKLJUČAK.....	21
6.	LITERATURA.....	22

1. SAŽETAK

Ova dokumentacija opisuje izradu projekta „Magnetic Card Reader“ zadanog u sklopu kolegija Ugradbeni računalni sustavi na Tehničkom fakultetu Rijeka. Cilj tog kolegija je osposobiti studenta da razumije arhitekturu i načine korištenja mikrokontrolera, da razumije principe i koncepte programiranja ugradbenih sustava te da stekne znanje i praktično iskustvo u izvedbi sklopovske i programske komponente ugradbenih računalnih sustava.

Ovaj projekt izabrali smo jer smo bili zainteresirani za rad sustava čitanja kartica, poput sigurnosnih ili transakcijskih sustava. Sustavi za autentifikaciju su sve traženiji i potrebniji u svijetu tehnologije.

Na projektu je radio tim od troje studenata uz pomoć profesora i asistenta koji vode taj kolegij.

Projekt također predstavlja završni ispit kolegija Ugradbeni računalni sustavi te na praktičan način ispituje usvojenost znanja i iskustva stečenog na tom kolegiju.

2. OPIS

Osnovni zadatak ovog projekta je da napraviti sustav koji očitava podatke s kartice s magnetskom trakom te daje neku vrstu prikaza na 16x2 LCD ekranu. Čitač kartica kojeg smo koristili ima samo opciju čitanja podataka s kartice te se ne može koristiti za operacije pisanja podataka na karticu što je i prednost u ovom slučaju zbog mogućnosti oštećenja rada kartice. Kada LCD ekran prikaže podatke, zadržava ih dok se ponovno ne očita kartica ili se uređaj resetira ili ugasi.

Korištena je serijska USART komunikacija kao način komunikacije čitača kartice i mikrokontrolera. Da bi se mogla koristiti navedena komunikacija, potrebno je postaviti odgovarajuće registre u programskom kodu. Ti registri će biti opisani u nastavku dokumentacije.

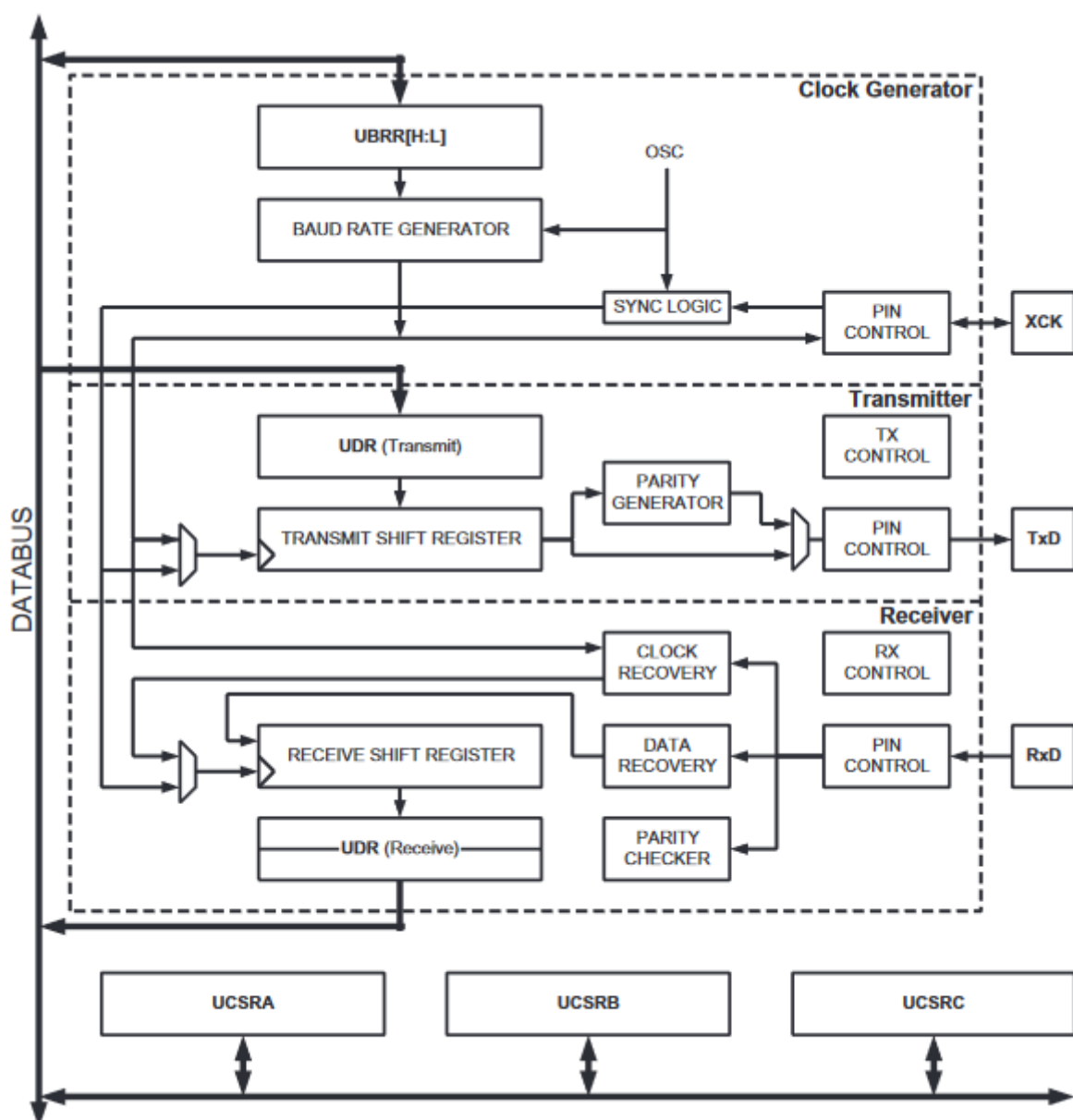
Ovaj projekt predstavlja neku osnovu za sustave za transakciju, sigurnost, autentifikaciju i slično, te bi se mogao nadograditi za korištenje u navedenim primjenama.

3. USART

a. USART protokol

USART(engl. Universal synchronous and asynchronous receiver-transmitter) je uređaj za uspostavljanje serijske komunikacije između dva uređaja.

Kako bi došlo do ikakve komunikacije, USART prvo treba inicijalizirati. Uobičajeno se taj proces sastoji od postavljanja baud rate-a, odnosno brzine slanja podataka u bitovima po sekundi, postavljanja formata podataka i omogućiti transmitter i/ili receiver ovisno o potrebama.

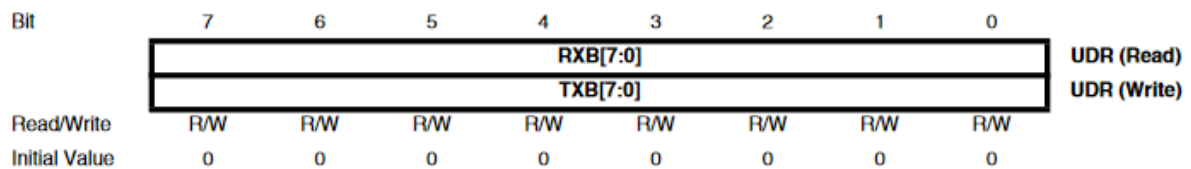


Slika 1: USART block diagram

b. USART UDR registar

USART Transmit Data Buffer Registrar i USART Receive Data Buffer Register dijele istu I/O adresu pod nazivom USART Data Register(UDR). Informacije koje se šalju sa UDR registra zapisane su u Transmit Data Buffer registru(TXB), dok prilikom čitanja informacija iz UDR registra se čitaju informacije iz Receive Data Buffer registra(RXD).

U transmit buffer registar se može pisati jedino kada je UDRE zastavica u UCSRA registru postavljena na logičku jedinicu, u protivnom će se sve informacije koje su zapisane u transmit buffer registru ignorirati od strane samog transmittera. Kada se podatci zapišu u transmit buffer i rad transmitta-a je omogućen, transmitta će učitati podatke u shift registar kada je prazan. Nakon toga se podatci serijski šalju preko TXD pina.



Slika 2: UDR registar

c. USART UCSRA registar

Bit	7	6	5	4	3	2	1	0	
	RXC	TXC	UDRE	FE	DOR	PE	U2X	MPCM	UCSRA
Read/Write	R	R/W	R	R	R	R	R/W	R/W	
Initial Value	0	0	1	0	0	0	0	0	

Slika 3: UCSRA registar

USART Control and Status Register A(UCSRA) je registar koji se sastoji od 8 bitova.

Bit 7 - RXC: USART Receive Complete:

Ovaj bit predstavlja zastavicu koja je u stanju logičko 1 ako postoje nepročitani podatci u receive buffer registru, a nalazi se u stanju logičko 0 kada je receive buffer registar prazan odnosno ne sadrži nepročitane podatke. Ako je rad receiver-a onemogućen onda će RXC automatski biti postavljen u stanje logičko 0. RXC zastavica također se može koristiti za generiranje Receive Complete prekida.

Bit 6 - TXC: USART Transmit Complete:

Ova zastavica je postavljena u 1 kada nema novih podataka u transmit buffer registru. TXC zastavica se automatski postavlja u 0 kada se izvrši Transmit Complete prekid ili se može manualno postaviti na 0. Također TXC zastavica može generirati Transmit Complete prekid.

Bit 5 - UDRE: USART Data Register Empty:

UDRE zastavica pokazuje ako je transmit buffer registar spreman primiti nove podatke. Ako je UDRE postavljen u 1 onda je buffer registar prazan, dakle spreman da se u njega zapišu podatci. UDRE može generirati Data Register Empty prekid. UDRE se postavlja na 1 nakon što se dogodi reset kako bi pokazao da je transmitter spreman.

Bit 4 - FE: Frame Error:

Ovaj bit je postavljen u 1 ako sljedeći znak u buffer registru ima frame error, odnosno kada je stop bit sljedećeg znaka u buffer registru jednak 0.

Bit 3 - DOR: Data OverRun:

Ovaj bit je postavljen u 1 ako je otkriven Data OverRun uvjet. Data OverRun se događa kada je receive buffer registar pun, a novi znak čeka u receive shift registru. Ovaj bit uvijek treba postaviti u 0 kada se piše u UCSRA.

Bit 2 - PE: Parity Error:

Ovaj bit se postavlja u 1 ako je sljedeći znak pri zaprimanju imao Parity Error, a parity checking je bio omogućen (UPM = 1). Uvijek postaviti ovaj bit na 0 kada se piše u UCSRA.

Bit 1 - U2X: Double the USART Transmission Speed:

Ovaj bit se koristi samo u asinkronom načinu rada, odnosno postavlja se na 0 ako se koristi sinkroni način rada. Ako je ovaj bit postavljen u 1 prepoloviti će djelitelja baud rate-a, čime će efektivno poduplati brzinu prijenosa podataka za asinkroni način rada.

Bit 0 - MPCM: Multi-processor Communication Mode:

Ovaj bit omogućava Multi-processor Communication način rada.

d. USART UCSRB registar

Bit	7	6	5	4	3	2	1	0	
	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8	UCSRB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Slika 4: UCSRB registar

USART Control and Status Register B(UCSRA) je registar koji se sastoji od 8 bitova.

Bit 7 - RXCIE: RX Complete Interrupt Enable:

Postavljanjem ovog bita na 1 omogućuju se prekidi na RXC zastavici. USART Receive Complete prekid će se generirati samo ako su RXCIE bit, globalna zastavica prekida u SREG i RXC bit u UCSRA postavljeni na 1.

Bit 6 - TXCIE: TX Complete Interrupt Enable:

Postavljanjem ovog bita na 1 omogućuju se prekidi na TXC zastavici. USART Transmit complete prekid bit će generiran samo ako su TXCIE bit, globalna zastavica prekida u SREG i TXC bit u UCSRA postavljeni na 1.

Bit 5 - UDRIE: USART Data Register Empty Interrupt Enable:

Postavljanjem ovog bita na 1 omogućuju se prekidi na UDRE zastavici. Data Register Empty prekid bit će generiran samo ako su UDRIE bit, globalna zastavica prekida u SREG i UDRE bit u UCSRA postavljeni na 1.

Bit 4 - RXEN: Receiver Enable:

Postavljanjem ovog bita na 1 omogućuje se rad USART receiver-a.

Bit 3 - TXEN: Transmitter Enable:

Postavljanjem ovog bita na 1 omogućuje se rad USART transmitter-a.

Bit 2 - UCSZ2: Character Size:

UCSZ2 u kombinaciji sa UCSZ1:0 bitovima postavlja veličinu podatkovnih bitova (veličinu znakova) koje primaju i šalju receiver i transmitter.

Bit 1 - RXB8: Receive Data Bit 8:

RXB8 je deveti podatkovni bit primljenog znaka kada se radi sa 9 podatkovnih bitova.

Bit 0 - TXB8: Transmit Data Bit 8:

TXB8 je deveti podatkovni bit znaka koji se šalje kada se radi sa 9 podatkovnih bitova.

e. USART UCSRC registar

Bit	7	6	5	4	3	2	1	0	
	URSEL	UMSEL	UPM1	UPM0	USBS	UCSZ1	UCSZ0	UCPOL	UCSRC
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	1	0	0	0	0	1	1	0	

Slika 5: UCSRC registar

USART Control and Status Register C(UCSRA) je registar koji se sastoji od 8 bitova.

Bit 7 - URSEL: Register Select:

Ovaj bit bira između pristupanja UCSRC ili UBRRH registru. Postavljen je u 1 kada pristupa UCSRC. URSEL mora bit postavljen na 1 ako se želi pisati u UCSRC registar.

Bit 6 - UMSEL: USART Mode Select:

Postavljanje ovog bita na 1 označava korištenje sinkronog načina rada, u suprotnom se koristi asinkroni način rada.

Bit 5:4 - UPM1:0: Parity Mode:

Ovi bitovi omogućuju i postavljaju tip pariteta.

UPM1	UPM0	Parity Mode
0	0	Disabled
0	1	Reserved
1	0	Enabled, Even Parity
1	1	Enabled, Odd Parity

Slika 6: UPM postavke bitova

Bit 3 - USBS: Stop Bit Select:

Ovaj bit bira broj stop bitova koje će transmitter ubaciti u frame. Receiver ignorira ovu postavku.

USBS	Stop Bit(s)
0	1-bit
1	2-bit

Slika 7: USBS postavke bitova

Bit 2:1 - UCSZ1:0: Character Size:

UCSZ1:0 bitovi u kombinaciji sa UCSZ2 bitom postavlja broj podatkovnih bitova(veličinu znaka) u okviru koje receiver i transmitter koriste.

UCSZ2	UCSZ1	UCSZ0	Character Size
0	0	0	5-bit
0	0	1	6-bit
0	1	0	7-bit
0	1	1	8-bit
1	0	0	Reserved
1	0	1	Reserved
1	1	0	Reserved
1	1	1	9-bit

Slika 8: UCSZ postavke bitova

Bit 0 - UCPOL: Clock Polarity:

Ovaj bit se koristi samo za sinkroni način rada. Postavite ovaj bit na 0 kada se koristi asinkroni način rada. UCPOL bit postavlja vezu između promjene podatkovnog izlaza, uzorka podatkovnog ulaza i sinkronog clock-a (XCK).

UCPOL	Transmitted Data Changed (Output of TxD Pin)	Received Data Sampled (Input on RxD Pin)
0	Rising XCK Edge	Falling XCK Edge
1	Falling XCK Edge	Rising XCK Edge

Slika 9: UCPOL postavke bitova

f. USART UBRRL i UBRRH registri

Bit	15	14	13	12	11	10	9	8	
	URSEL	—	—	—	UBRR[11:8]				UBRRH
	UBRR[7:0]								UBRRL
	7	6	5	4	3	2	1	0	
Read/Write	R/W	R	R	R	R/W	R/W	R/W	R/W	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

Slika 10: UBRRL i UBRRH registri

UBRRH registri dijele istu I/O lokaciju sa UCSRC registrom.

Bit 15 - URSEL: Register Select:

Ovaj bit bira između pristupanja UCSRC ili UBRRH registru. Postavljen je u 0 kada pristupa UBRRH. URSEL mora biti postavljen na 0 ako se želi pisati u UBRRH registar.

Bit 14:12 - Reserved Bits:

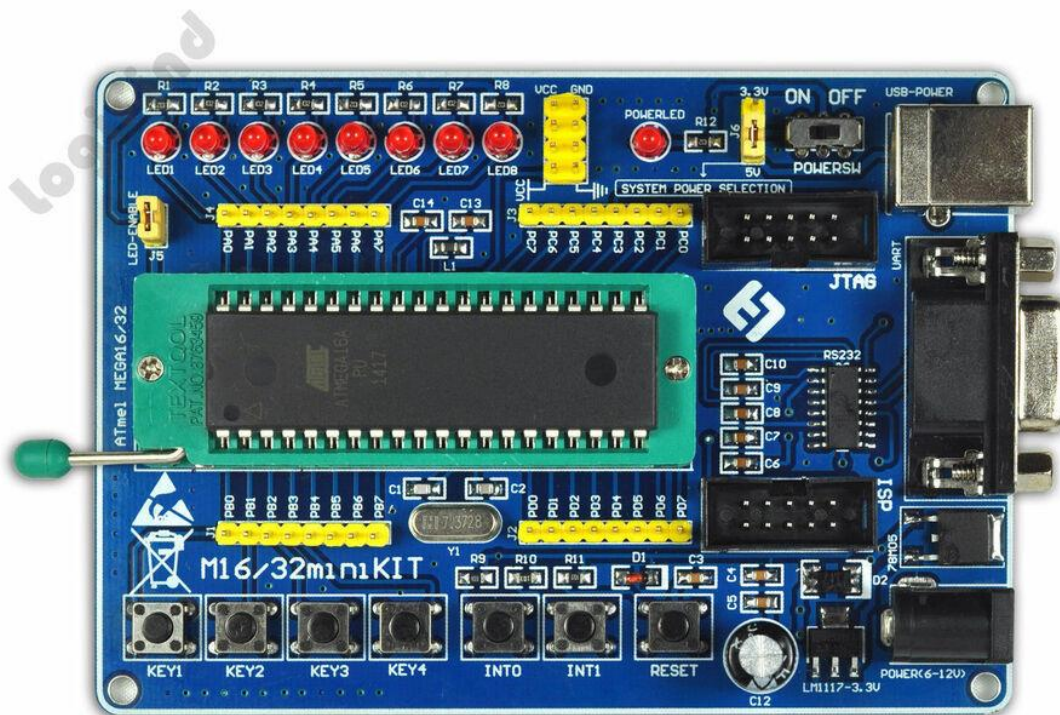
Ovi bitovi su rezervirani za neke buduće funkcije i moraju biti postavljeni na 0.

Bit 11:0 - UBRR11:0: USART Baud Rate Register:

Ovo je 12-bitni registar koji sadrži USART baud rate. UBRRH sadrži 4 najbitnija (MSB) bita dok UBRRL sadrži 8 najmanje bitna (LSB) bita.

4. HARDVERSKJE KOMPONENTE

a. ATmega16A mikrokontroler



Slika 11: ATmega16A mikrokontroler

Specifikacije:

- Napredna RISC arhitektura
 - 32 8-bitna registra opće namjene direktno spojena na aritmetičko-logičku jedinicu
 - koristi Harvard arhitekturu sa zasebnim memorijama i sabirnicama za program i podatke
 - postiže propusnost od oko blizu 1 MIPS po MHz omogućavajući dizajneru sustava optimizaciju potrošnje energije u odnosu na brzinu obrade podataka
- Memorija
 - Tri memorijska prostora:
 - Za podatke, 1KB internog SRAM-a
 - Za program, 16KB samoprogramirajuće Flash memorije s izdržljivošću od najmanje 100,000 write/erase ciklusa
 - Za pohranu podataka, 512B EEPROM - s izdržljivošću od najmanje 100,000 write/erase ciklusa

- Periferija
 - Dva 8-bitna timera
 - Jedan 16-bitni timer

- Ostale značajke
 - JTAG sučelje
 - Interni i eksterni prekidi
 - 10-bitni ADC konverter (engl. analog-to-digital converter)
 - 32 programibilne I/O linije

b. 16x2 LCD ekran



Slika 12: 16x2 LCD ekran

LCD je ekran koji koristi svojstva modulacije svjetlosti tekućih kristala u kombinaciji s polarizatorima. Kristali ne emitiraju svjetlost nego koriste pozadinsko osvjetljenje ili reflektor kako bi dobili jednobojne prikaze ili slike u boji. LCD ima prostora za 32 znaka te je 4-bitni, što znači da se bitovi prenose 4 po 4 bita. Način rada 4 po 4 koristi samo bitove DB4 do DB7 te se DB0 do DB3 ne koriste. Bajtovi se LCD-u prenose u 2 koraka: prvo viša 4 bita zatim niža 4 bita.

c. SU-90 Magnetic Card Reader



Slika 13: SU-90 Magnetic Card Reader

SU-90 magnetni čitač kartica je uređaj sa magnetskom trakom na poleđini te ima mogućnost provlačenja kartice u oba smjera, očitavanje tih podataka te LED lampicu koja označava napajanje te uspješno ili neuspješno očitavanje provučene kartice. SU-90 koristimo iz razloga što podatke šalje preko RS232 signala, a napajanje dobiva preko PS2 konektora. Novija varijanta istog čitača pod imenom MSR-90 ne bi bila prigodna za ovaj projekt iz razloga što ima samo USB konektor i zato smo koristili SU-90 čitač za ovaj projekt.

Kartice koje može pročitati rade na principu magnetskih traka koje sadrže podatke. Kartice mogu sadržavati dvije ili čak tri trake podataka različitih veličina i vrijednosti. Podaci se spremaju tako da se mijenja magnetizam sitnih čestica koje su bazirane na željezu. Velika većina kartica je napravljena po ISO standardu te imaju sličan ili isti format podataka.

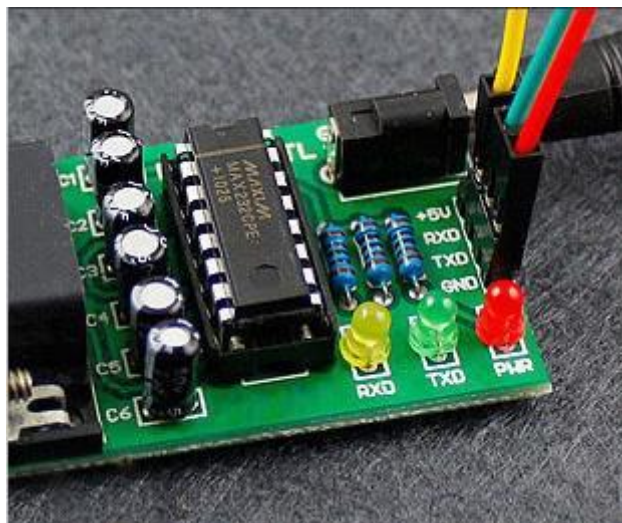
Na prvoj traci nalazi se 210 bitova po inču, a znakovi su veličine 7 bitova te se na toj traci nalazi 79 alfanumeričkih znakova.

Na drugoj traci gustoća je 75 bitova po inču, a veličina znakova je 5 bitova što daje veličinu od 40 numeričkih znakova na traci 2.

Na trećoj traci ponovno imamo gustoću 210 bit po inču sa veličinom znakova od 5 bitova što u konačnici rezultira sa 107 numeričkih znakova koji stanu na traku 3.

Format u kojem su spremljeni podatci također je definiran. Prva traka započinje sa znakom koji se naziva start sentinel i on označava početak podataka. U većini slučajeva to je znak postotka ("%"). Nakon start sentinela na prvoj traci slijedi format code znak nakon kojega dolazi PAN (Primary Account No.). Između svakog seta podataka nalazi se field separator. Za prvu traku to je znak "^". Konačno nakon svih podataka dolazi end sentinel koji označava kraj trake. To je gotovo uvijek znak "?". Nakon end sentinela slijedi još longitudinal redundancy check(LRC) koji provjerava ukoliko se dogodila kakva greška sa podacima. Ista struktura vrijedi i za druge trake sa manjim izmjenama u znakovima koji se koriste za field separator i start sentinel i slično.

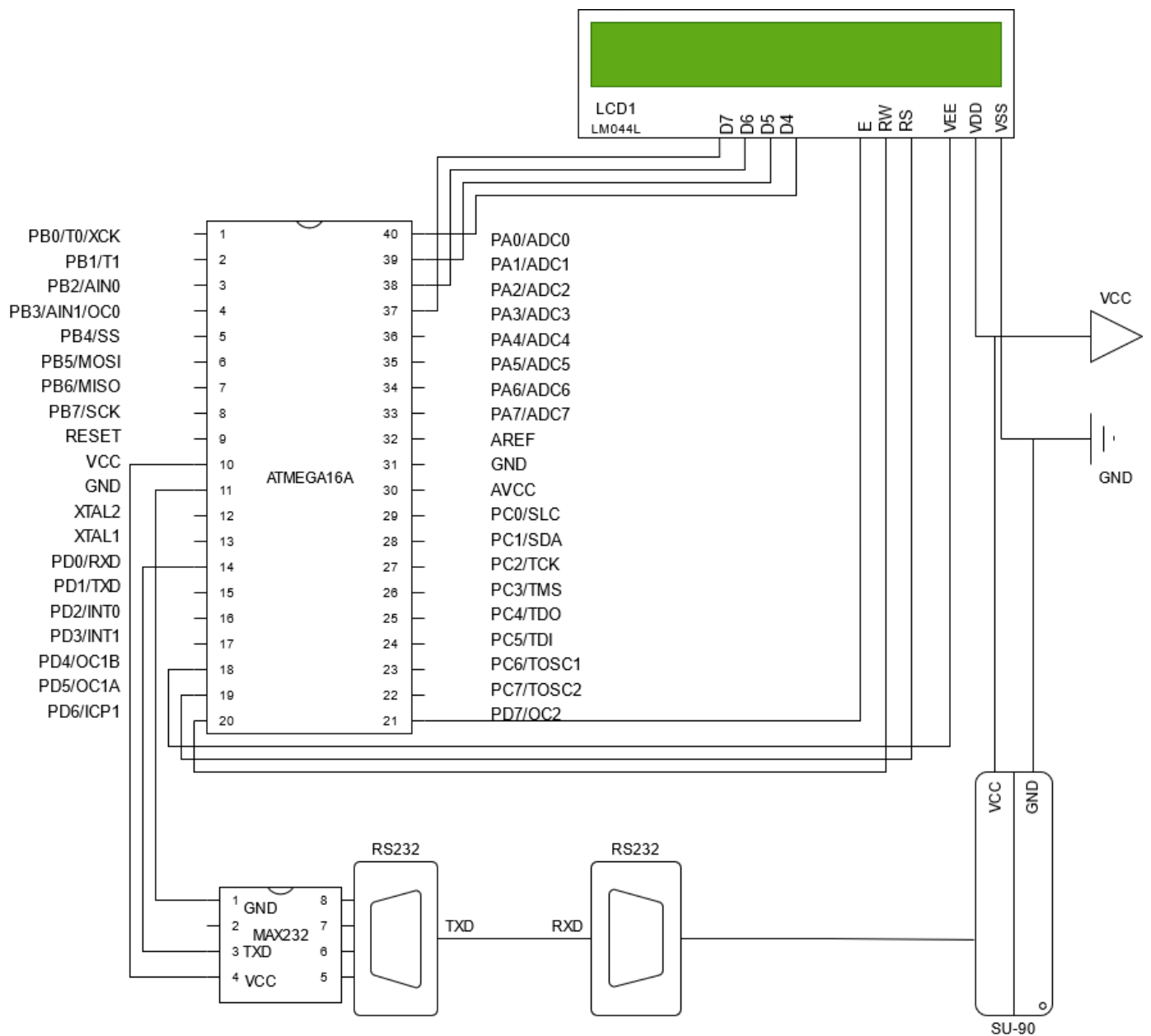
d. DB9 breakout board sa MAX232 čipom



Slika 14: DB9 breakout board sa MAX232 čipom

Kako bismo mogli očitati podatke sa magnetnog čitača, bio nam je potreban nekakav "prevoditelj", za ovakve svrhe se najčešće koristi MAX232. Mi smo nabavili MAX232 čip integriran na DB9 breakout board-u. Breakout board na sebi je imao četiri pina potrebna za operaciju: VCC, GND, RXD, TXD; te je imao i tri LED lampice koje su označavale napajanje, primanje podataka i slanje podataka. Isprva smo koristili RXD pin na DB9 konektoru board-a, ali nismo dobivali nikakva očitavanja, nakon što smo proučili shemu spajanja smo saznali da treba koristiti TDX pin jer TDX vodi u MAX232 gdje se analogni RS232 signal pretvara u TTL signal koji naš mikrokontroler može interpretirati i iskoristiti.

e. Shema spajanja komponenti



Slika 15: Shema spajanja komponenti

5. ZAKLJUČAK

Izrada projekta je bio proces u više koraka, za svaki od koraka bi se dalo navesti probleme na koje smo naišli, rješenja koja smo probali, stranice na internetu koje smo posjetili, stranci koji su se bavili istim stvarima i imali iste probleme. Moglo bi se reći da je projekt bio izazov rješavanja praktičnog problema više nego izazov znanja. Koristili smo sve izvore koje smo mogli, sve savjete koji su nam bili dostupni. Naša očekivanja su bila da će nam najteži dio bit sam programski kod projekta te da ćemo najviše problema naići tu, ali to nije bio slučaj. Datasheet našeg mikrokontrolera nam je pružio sve potrebno kako bismo kodirali mikrokontroler da prima podatke. Magnetni čitač koristi stariju tehnologiju te podatke šalje serijski što znači da je signal analogan te ako bi direktno spojili čitač i mikrokontroler dobili bi, kako smo to nazvali, "gibberish" (puno nerazumljivih znakova). Potreban je bio nekakav posrednik između čitača i mikrokontrolera, a taj posrednik je bio MAX232 čip.

Nakon što smo dodali MAX232 između mikrokontrolera i čitača isprva nismo dobivali nikakav izlaz, problem je bio u pinovima, ispitivanjem svakog pina otkrili smo koji je potreban za slanje signala kroz MAX232 te sa shemom smo zaključili zašto je to slučaj. Posljednji korak je bio dodavanje dodatne funkcionalnosti te uljepšavanje koda. Finalni produkt nam ispisuje podatke zapisane na zdravstvenim, bankovnim te karticama dopunskog osiguranja te imamo funkcionalnost da ponovnim prolazom kartice LCD se resetira te pokazuje podatke sa druge kartice.

Projekt nam je dao uvid u serijske spojeve, pokazao nam je kako analogni signal funkcionira te što moramo koristiti kako bismo dobili digitalni signal koji drugi uređaji mogu interpretirati. Naime moglo bi se reći da analogni signali i serijski konektori polako odumiru, što možda je slučaj kod moderne elektronike prosječnih potrošača, ali to ne može biti dalje od istine. Za druge svrhe, serijski konektori i uređaji su itekako i dalje prisutni u tvrtkama i ostalim profesionalnim okruženjima gdje zbog svoje jednostavnosti pružaju sigurnost i preciznost. RS232 i njegov standard će ostati još aktualan neko vrijeme dokle god postoje uređaji koji koriste analogni signal i serijski konektor.

Puno smo naučili u procesu izrade projekta, koliko smo naučili o softveru toliko i o hardveru koji koristimo. Izrada nečega ovakvog je odlična prilika da dobijemo nekakvo "hands on", praktično iskustvo koje spaja programiranje, testira dobiveno znanje te nam daje stvarne probleme koje moramo riješiti na koji god način znamo koristeći sve resurse i znanje koje nam je dostupno. Cijenimo novonastalo iskustvo i znanje te priliku da ga dobijemo radom na ovome projektu.

6. LITERATURA

- ATmega16A Datasheet
- [SU-90 RS232 Interface Magnetic card reader Datasheet](#)