

Elektrotehnički fakultet
Banja Luka

MIKROPROCESORSKI SISTEMI 2

BATTLESHIP

Ljubomir Grublješić
Marinko Grublješić
Duško Sopić

Sadržaj

1. Uvod	3
2. Hardver	4
2.1 Displej	5
2.2 Dekoder	5
2.3 Invertor	6
2.4. Otpornik.....	6
2.5. Protobord	7
2.6. Žice za spajanje.....	7
3. Softver	7

1. Uvod

Tema projekta je implementacija igre Battleship upotrebom dva mikrokontrolera 8051 povezana UART serijskim komunikacionim kablom i četiri matrična LED displeja 8x8. Potrebno je projektovati odgovarajući hardverski dizajn, a zatim napisati softverski modul koji će upravljati odgovarajućim hardverom. Za realizaciju projekta su nam potrebne odgovarajuće hardverske komponente, kao i softverki alati u kojem ćemo razvijati odgovarajući softver.

Battleship je igra pogađanja između dva igrača. Igra se na papiru na kojem su nacrtane mreže (najčešće 10x10), po dvije za svakog igrača. U mreži je ucrtana flota brodova, svaki igrač ima svoju flotu ucrtanu na jednoj mreži, a druga mreža predstavlja protivnikovu flotu koja je inicijalno prazna. Flota brodova se sastoji najčešće od četiri tipa brodova veličina od 1 do 4 polja, s tim da brodova veličine 1 ima 4 komada raspoređenih po mreži, veličine 2 sa 3 komada itd. Polja u mreži se identifikuju pomoću označenih redova (1-10) i kolona (A-J). Igra se tako da jedan igrač „gađa“ protivnikovu flotu tako što odabere polje (npr. E7) i rezultat upisuje u svoju mrežu koja predstavlja protivnikovu flotu zavisno od toga da li je bio pogodak ili promašaj, što mu protivnik sam kaže. Nakon njega igra protivnik na isti način. Pobjednik je onaj koji prvi potopi sve protivnikove brodove.

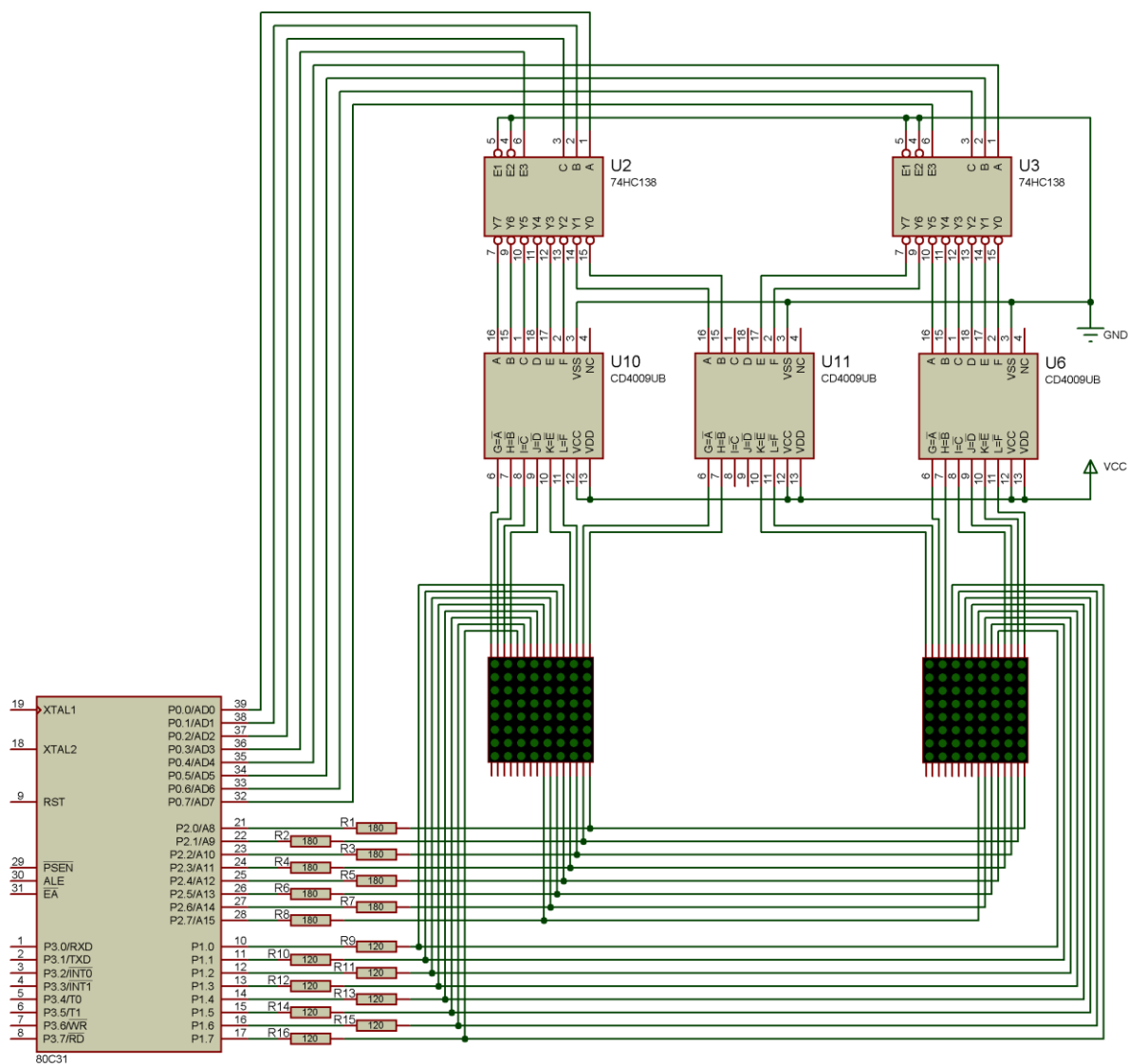
Na našem projektu mreža je veličine 8x8 zbog veličine matričnog displeja. Takođe, broj brodova (osim za brod veličine 4) je smanjen za po 1. Mi smo koristili sledeće brodove:

- veličine 4 x 1
- veličine 3 x 1
- veličine 2 x 2
- veličine 1 x 4

2. Hardver

Za realizaciju projekta koristili smo ploču *Easy8051 v6*. Razvojna ploča *Easy8051 v6* je pogodna za programiranje *Atmel 8051* mikrokontrolera, kao i za dizajniranje i testiranje *8051* projekata. Ovaj razvojni sistem uključuje *on-board* programer omogućavajući interfejs između mikrokontrolera i PC-a. *Easy8051 v6* sadrži mnoge uređaje, poput 128x64 grafičkog LCD displeja, alfanumeričkog 2x16 LCD displeja, 2x16 LCD displeja direktno ugrađenog na ploču, 4x4 tastature i sl. koji imaju pristup *8051* mikrokontrolerima i omogućavaju simulaciju ciljanog uređaja.

Komponente smo spojili po sledećoj šemi:

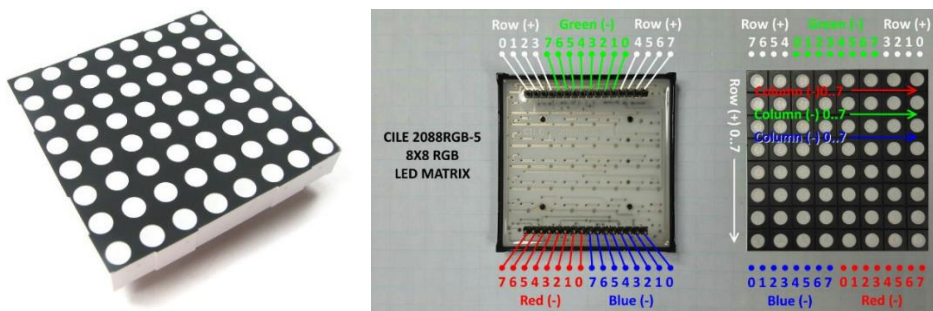


Što se tiče samih komponenti, za jednog igrača koristili smo:

- *Easy8051 v6*
- *CILE 2088-5* LED displej x2
- Dekoder *74HC138* x2
- Invertor *CD4009UB* x3
- 120 Ω otpornik x8
- 180 Ω otpornik x8
- Protoboard *EIC-406*
- Žice za spajanje

2.1 Displej

LED displej se sastoji iz matrice 8×8 *RGB LED* dioda. Svako polje matrice se sastoji od 3 *LED* diode (crvena, zelena i plava). Displej ima 32 pina za povezivanje. Za omogućavanje reda matrice dovodi se na odgovarajući pin visok logički nivo, a za uključivanje određene diode koristi se nizak logički nivo.



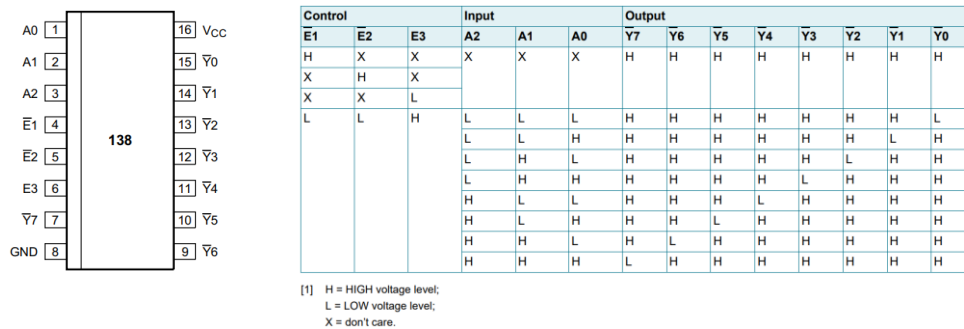
Slika 1.a) LED displej 1.b) Raspored pinova displeja

Zbog ograničenja ploče sa brojem pinova, odlučili smo da ne koristimo plavu boju i na taj način kontroliramo displej sa 24 pina. Koristili smo crvenu i zelenu boju, a za treću boju smo koristili njihovu kombinaciju, žutu. Pošto smo koristili dva displeja, povećali smo broj pinova, pa smo morali da uvedemo i dekodere koje smo spojili na pinove za omogućavanje reda matrice (*Row (+)* – Slika 1.b). Iscrtavanje smo radili tako što smo prvo omogućili jedan displej, a onda drugi. Dok je omogućen jedan displej kroz *for* petlju smo omogućavali red po red i uključivali odgovarajuće diode u tom redu tako što smo pin za tu odgovarajuću boju doveli na nizak logički nivo.

2.2 Dekoder

Koristili smo dekodere *74HC138*, to je 3/8 demux. Na sledećem linku se nalazi datasheet: https://assets.nexperia.com/documents/data-sheet/74HC_HCT138.pdf. Za svaki displej koristili smo po jedan demultiplekser, dok smo radili iscrtavanje na displeje, prvo smo omogućili jedan, pa onda drugi. Na *VCC* smo doveli visok logički nivo sa mikrokontrolera. Na *GND*, *E1* i *E2* smo doveli nizak logički nivo, a sa *E3* smo kontrolisali uključivanje i

isključenje dekodera. Četiri pina sa mikrokontrolera smo doveli na $A0$, $A1$, $A2$ i $E3$ ulaze dekodera, a druga četiri na iste ulaze drugog dekodera i na taj način smo uticali na njegove izlaze. Dekoder radi po sledećoj šemi:

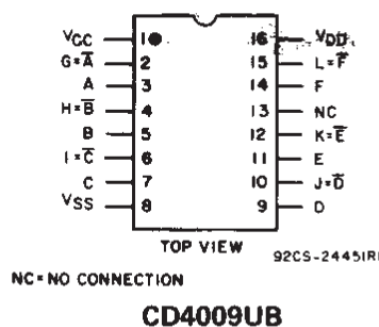


Slika 2.a) Šema dekodera 2.b) Tabela rada dekodera

Pošto smo dekodeer koristili za omogućavanje displeja, a on daje nizak logički nivo na odgovarajući izlaz, a nama je potreban visoki, morali smo da dodamo inverter prije pinova na displeju za omogućavanje reda.

2.3 Invertor

Koristili smo inverter *CD4009UB*. Na sledećem linku se nalazi datasheet: https://assets.nexperia.com/documents/data-sheet/74HC_HCT04.pdf. Na ulaze VCC i VDD smo spojili visok logički nivo sa ploče, a na VSS nizak logički nivo. Ulaze A , B , C , D , E i F smo spojili na izlaze dekodera, a G , H , I , J , K i L smo spojili na displej. Dekoder ima osam izlaza, a inverter šest, pa smo na dva dekodera koristili tri invertora.



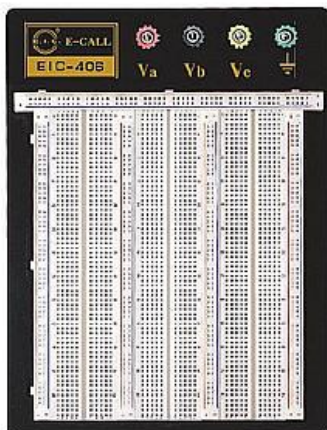
Slika 3. Šema dekodera

2.4 Otpornik

Za ograničenje struje kod displeja koristili smo otpornike od $120\ \Omega$ za zelenu boju i od $180\ \Omega$ za crvenu.

2.5. Protobord

Za povezivanje svih komponenti koristili smo protoboard *EIC-406*.



Slika 4. Protoboard

2.6. Žice za spajanje

Za povezivanje komponenti koristili smo 6 flet kablova i žice iz *LAN* kabla, jer su bile odgovarajuće debljine za spajanje u protobord.

3. Softver

Sa pisanje i kompajliranje koda smo koristili razvojno okruženje *mikroC PRO*, a za učitavanje koda na ploču smo koristili *8051Flash Programmer*. Alate smo preuzeli sa kursa predmeta. Pored ovog dokumenta biće dostupno detaljno uputstvo u *pdf* formatu za korištenje ovih alata koji su preuzeti sa sajta fakulteta.