Emulator MikroprocesorJa Motorola M680X

(Inovacijski predlog)

KAZALO VSEBINE

[1 Povzetek iv](#_Toc150339831)

[2 Uvod 4](#_Toc150339832)

[3 RAZVOJ 1](#_Toc150339833)

[4 Model 1](#_Toc150339834)

[4.1 Uporabniški vmesnik 1](#_Toc150339835)

[4.1.1 Števec vrstic 1](#_Toc150339836)

[4.1.2 Polje za vpis assembly kode 1](#_Toc150339837)

[4.1.3 Polje za prikaz spomina 2](#_Toc150339838)

[5 ZAKLJUČEK 5](#_Toc150339839)

# Povzetek

Moj inovacijski predlog predstavlja razvoj emulatorja za mikro procesorja Motorola M6800 in Motorola M6803, ki omogoča uporabnikom, še posebej tistim, ki se želijo naučiti osnove nizko nivojskega programiranja in dela z »assembly« programskimi jeziki, edinstveno priložnost za izboljšanje njihovega razumevanja računalniške arhitekture. Emulator vsebuje uporabniku prijazen vmesnik s poljem za vnos »assembly« kode, prikaz spomina, orodja za sestavljanje in razstavljanje ukazov, orodja za izvajanje ukazov, zaslon, vhodne medpomnilnike, tabelo ukazov in njihove lastnosti, ter veliko nastavitev za olajšanje uporabe emulatorja.

# Uvod

S trenutno omejenimi možnostmi za simulacijo in izvajanje kode za mikro procesorja Motorola M6800 in M6803 se soočamo s pomanjkanjem virov za izobraževanje študentov, razvijalcev in ljubiteljev retro računalnikov. To pomanjkanje priložnosti je očitna težava, ki me je spodbudila k razvoju emulatorja. Moja rešitev bo prinesla koristi v obliki izboljšane uporabniške izkušnje pri učenju programiranja v assembly jeziku za omenjene procesorje, kar bo omogočilo lažje razumevanje računalniške arhitekture. Zato sem tudi izbral Motorola procesorje iz družine M68XX saj so ti zelo nezapleteni ter imajo malo število registrov in omejeno število dokazov. Moj emulator ima potencialne aplikacije v izobraževalnih ustanovah, kar dodatno krepi njegov pomen.

# RAZVOJ

Emulator je bil razvit z ogrodjem za programsko opermo »Qt«, ki je odprtokodno in zmogljivo okolje za razvoj grafičnih uporabniških vmesnikov ter ostale programske opereme. Večina emulatorja je bila izdelana z programskim jezikom c++. Nekaj postopkov razvoja sem avtomatiziral z skriptami napisanimi v programskem jeziku python.

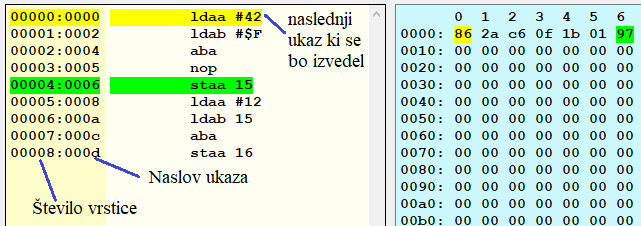
# Model

## Uporabniški vmesnik

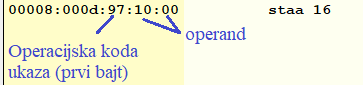
V aplikaciji »Qt Creator 10.0.2 (Community edition)« sem razvil uporabniški vmesnik s podanimi orodji.

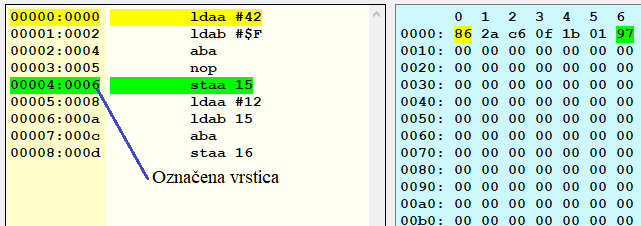
### Števec vrstic

Na levi strani glavnega okna je polje namenjeno štetju vrstic assembly programa. Vsaka vrstica tega polja ustreza številu vrstic polja za vpis assembly programa z indeksiranjem, ki se začne pri 0. Ko program trenutno ni sestavljen ali spomin ni razstavljen bodo v polju pisale samo vrsticam ustrezna števila. Če pa je program sestavljen oz. ukazi ustrezajo strojni kodi v spominu pa bo ob številu vrstic, ločeno s dvopičjem izpisan pomnilniški naslov, ki ustreza pomnilniški lokaciji kjer je zapisan prvi bajt ukaza oz. njegova operacijska koda. Takrat bo tudi vidna rumena označba, ki bo kazala na ukaz kateri se nahaja na trenutni vrednosti PC. To je tudi naslednji ukaz ki se bo izvedel.



V primeru da je vklopljena nastavitev »napredne informacije programa« pa bodo ob pomnilniški lokaciji bili izpisani vsi bajti v katere se ukaz na ustrezni vrstici sestavi ločeni z dvopičjem.

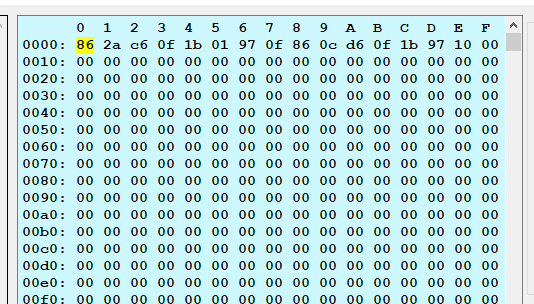


Klik na vrstico tega polja bo pod pogojem, da je program sestavljen, obarval ukaz, število vrstice ter naslov prvega bajta tega ukaza v spominu z zeleno barvo. Ta oznaka bo ostala dokler se program ne spremeni ali uporabnik ne pritisne desni gumb miške na to polje.

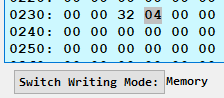
### Polje za vpis assembly kode

Ob števcu vrstic je polje za vpis assembly kode. Vpis v polje je mogoč, ko je emulator nastavljen na način vpisa kode.

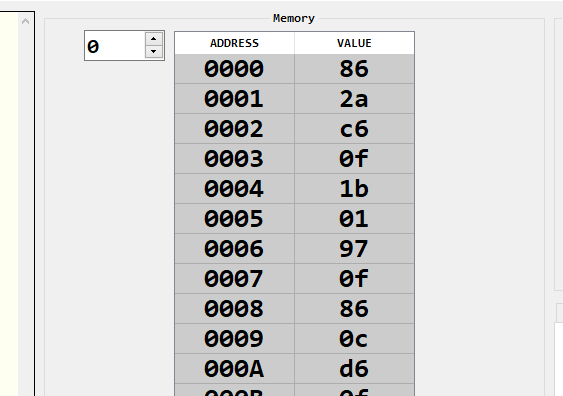
### Polje za prikaz spomina

Polje za prikaz spomina, ki je na sredini glavnega okna v podobi tabele prikazuje vse pomnilniške celice ter vrednosti ki jih hranijo v šestnajstiškem številskem sistemu. 

V načinu pisanja v spomin bo viden kazalec za spremembo pomnilniške lokacije. Uporabnik ga premika z puščicami na tipkovnici in vrednost celice zapiše z števkami šestnajstiškega številskega sistema.



Enostavni spomin je prikazan v obliki tabele. V levem stolpcu je pomnilniški naslov, v desnem pa vrednost spomina na tem naslovu. Prikazani naslovi se spreminjajo s števcem, ki je na levi strani tabele.



### Registri

Desno zgoraj je polje ki prikazuje vsebine registrov procesorja M6800, ti so akumulator A, akumulator B, indeksni register X, programski števec, kazalec sklada in zastavice stanja(Half-carry, interrupt flag, negative, zero, overflow, carry)

### Stran z večmi zavihki

Desno spodaj je stran za izbiro večih zavihkov.

1. Prvi je okno konzole ki prikazuje napake, opozorila in informacije ki jih emulator posreduje uporabniku.
2. Drugi vsebuje orodja za odpravljanje napak, ki vsebuje prevajalnik številskih sistemov ter sistem prelomnih točk ki omogočajo uporabniku samodejno ustavljanje izvajanja ukazov ob določenem dogodku ali pogoju. Na primer, program se lahko ustavi če po kateri register vseboval zaželeno vrednost, lahko se ustavi na določeni vrstici/ukazu ali pa, ko neka celica v spominu vsebuje zaželeno vrednost.
3. Tretji vsebuje večino nastavitev emulatorja. Možne nastavitve so:
   * Nastavitve, ki določajo kaj se bo prikazalo v oknu konzole.
   * Nastavitev, ki omogoča prikaz podrobnejših informacij vsakega sestavljenega ukaza.
   * Nastavitev, ki določa v kakšnem številskem sistemu bodo napisane vrednosti registrov.
   * Nastavitev, ki določa če bodo ukazi samodejno sestavljeni ko je program zagnan po spremembi ukazov.
   * Nastavitev, ki določa če bo se emulator samodejno ponastavil na stanje pred zadnjim izvrševanjem.
   * Nastavitev, ki določa, ali se procesor ravna po ciklični hitrosti ukazov in izvaja z določenim številom ciklov na sekundo ali pa sledi uporabnikovi nastavitvi ukazov na sekundo, pri čemer ciklična hitrost ni upoštevana.
   * Nastavitev, ki določa če bo uporabniku prikazan standardni prikaz spomina ki ima 16 pomnilniških naslovov in njihovih vrednosti v eni vrstici, ali enostavnejši prikaz ki prikazuje 20 poljubnih zaporednih pomnilniških naslovov in vrednosti katere vsebujejo v tabeli.
   * Nastavitev, ki omogoča nastavljanje delovanja emulatora med pisanjem kode ter neposredno v spomin, to posledično tudi določa če bo emulator kodo ali ukaze sestavljal ali razstavljal. Na spodnjem delu glavnega okna se pojavi gumb ki prikazuje trenutni način delovanja emulatorja ter ponuja uporabniku da ga zamenja. Privzeta nastavitev je da deluje na način sestavljanja, to je da uporabnik piše ukaze v assemblyu ter jih prevajalec sestavi v strojno kodo, druga nastavitev je način razstavljanja, takrat je uporabniku omogočeno spreminjanje strojne kode ter razstavljanje spomina, to je da se strojna koda v spominu prevede v assembly ukaze ki se bodo izpisali v polju za assembly program.
   * Nastavitvi za samodejno premikanje polja na ukaz kateri se trenutno izvršuje.
   * Nastavitev, ki določa kje bo prikazan zaslon. Če je nastavitev nastavljena na »glavno okno« bo zaslon prikazan na glavnem oknu med polju za spomin in polju registrov, ko bo ta imel zadosti prostora, oziroma ko bo glavno okno dovolj široko. Če je nastavitev nastavljena na »zunanje okno«, bo zaslon prikazan zunaj glavnega okna v obliki okna za dialog.
4. Četrti zavihek vsebuje opis in navodila za uporabo emulatorja
5. Peti vsebuje tabelo z vsemi ukazi mikroprocesorjev Motorola M6800 in Motorola M6803. Ukazi, operacijske kode, število ciklov, velikost ter opisi zadnje omenjenega so obarvani rdeče.

Na spodnji strani glavnega okna so gumbi ter izbirni meniji za olajšano in uporabniku prijaznejšo uporabo emulatorja. Ti so:

1. Gumb »Sestavi/Razstavi« bo če je način pisanja nastavljen na »pisanje kode« ukaze, ki so trenutno napisani v polju za pisanje ukazov sestavil in prevedel v strojno kodo, ki bo pregledna v polju za spomin. Če pa je način pisanja nastavljen na pisanje v spomin, bo ta gumb razstavil strojno kodo zapisano v spominu ter jo zapisal v polju za ukaze.
2. Meni za izbiro različice Motorola procesorjev. Trenutno emulator podpira dve različici Motorola M6800 in Motorola M6803, ampak je tako zasnovan da omogoča nadaljnjo razširitev ter dodajanje novih procesorjev. Emulator bo se ravnal, sestavljal razstavljal ter izvajal ukaze glede na izbran procesor.
3. Gumb za nalaganje, bo glede na izbiro načina pisanja iz zunanje datoteke naložil assembly kodo ali pa spomin.
4. Gumb za shranjevanje, bo glede na izbiro načina pisanja srahnil assembly kodo ali pa trenutni spomin v datoteko.
5. Gumb za zamenjavo načina delovanja emulatorja. Privzeto je ta gumb nedosegljiv lahko se pa prikaže s nastavitvijo »dovoli zamenjavo načina pisanja«. Desno od gumba je napis trenutnega načina.
6. Gumb za ponastavitev ponastavi emulator na stanje kakšnega je bil po zadnjem sestavljanju ali pred zadnjim izvajanjem.
7. Gumb »korak« ki izvede en ukaz z procesorjem izbranim v meniju za izbiro različice procesorja.
8. Gumb »Zaženi/Ustavi« v primeru, da procesor trenutno ne izvaja ukazov zažene samodejno izvajanje ukazov. V primeru, da procesor trenutno izvaja ukaze pa ga ta gumb začasno ustavi. Uporabnik lahko izvajanje vedno nadaljuje z vnovičnim pritiskom na ta gumb. Gumb deluje v skladu z menijem hitrosti ter nastavitvijo za izbiro načina izvajanja. Procesor bo samodejno prenehal izvajanje kadar naleti na strojni kod 0x00. Če je programski števec(PC) trenutno vsebuje pomnilniško lokacijo, ki vsebuje 0x00 in če je nastavitev »samodejna ponastavitev« vklopljena bo se emulator ponastavil.
9. Meni za nastavitev hitrosti izvajanja. Prikazuje število ukazov ali ciklov na sekundo. Možna je izbira od 1 do 1000 ter »maksimum«. Maksimum je včasih neenakomeren ampak je najhitrejša izbira. Te vrednosti so le približek saj je hitrost odvisna od zmogljivosti uporabnikovega računalnika.

# ZAKLJUČEK

Ta seminarska je malo podrobnejši vpogled v delovanje naprav USB in njihovih standardov. Na spletu res ni veliko seminarskih o USB v slovenščini še sploh pa ne tako podrobnih. V tej seminarski sem izpustil varnostne grožnje, kompatibilnost z drugimi standardi in protokole saj te stvari niso najpomembnejše po mojem mnenju.

Viri besedila

1. *Master/slave(Technology)* 2022 [online]. Wikipedija, The Free Encyclopedia. Pridobljeno 7. 3. 2022 na spletnem naslovu: https://en.wikipedia.org/wiki/Master/slave\_(technology)
2. *USB* 2022 [online]. Wikipedija, The Free Encyclopedia. Pridobljeno 7. 3. 2022 na spletnem naslovu: https://en.wikipedia.org/wiki/USB
3. *Universal Serial Bus Specification* 2022 [online]. Universal Serial Bus Specification. Pridobljeno 7. 3. 2022 na spletnem naslovu: https://fl.hw.cz/docs/usb/usb10doc.pdf
4. *USB Hardware* 2022 [online]. Wikipedija, The Free Encyclopedia. Pridobljeno 7. 3. 2022 na spletnem naslovu: https://en.wikipedia.org/wiki/USB\_hardware
5. *USB 3.0* 2022 [online]. Wikipedija, The Free Encyclopedia. Pridobljeno 7. 3. 2022 na spletnem naslovu: https://en.wikipedia.org/wiki/USB\_3.0
6. *USB-C* 2022 [online]. Wikipedija, The Free Encyclopedia. Pridobljeno 7. 3. 2022 na spletnem naslovu: https://en.wikipedia.org/wiki/USB-C
7. *Star network* 2022 [online]. Wikipedija, The Free Encyclopedia. Pridobljeno 7. 3. 2022 na spletnem naslovu: https://en.wikipedia.org/wiki/Star\_network
8. *USB(Communications)* 2022 [online]. Wikipedija, The Free Encyclopedia. Pridobljeno 7. 3. 2022 na spletnem naslovu: https://en.wikipedia.org/wiki/USB\_(Communications)
9. *Univerzalno serijsko vodilo* 2022 [online]. Wikipedija, prosta enciklopedija. Pridobljeno 7. 3. 2022 na spletnem naslovu: https://sl.wikipedia.org/wiki/Univerzalno\_serijsko\_vodilo
10. *What is Micro USB Pinout and Types* 2022 [online]. eTechnophiles. Pridobljeno 7. 3. 2022 na spletnem naslovu: https://www.etechnophiles.com/micro-usb-pinout-features/
11. *Univerzalno serijsko vodilo* 2022 [online]. Wikipedija, prosta enciklopedija. Pridobljeno 7. 3. 2022 na spletnem naslovu: https://sl.wikipedia.org/wiki/Univerzalno\_serijsko\_vodilo
12. *What is USB-C* 2022 [online]. electronicsnotes Pridobljeno 7. 3. 2022 na spletnem naslovu: https://www.electronics-notes.com/articles/connectivity/usb-universal-serial-bus/usb-c-connector.php
13. *USB 3.2 Specification Language Usage Guidelines from USB-IF* [online]. USB-IF Pridobljeno 7. 3. 2022 na spletnem naslovu: https://www.usb.org/sites/default/files/usb\_3\_2\_language\_product\_and\_packaging\_guidelines\_final.pdf

Viri slik

1. *USB* 2022 [online]. Wikipedija, The Free Encyclopedia. Pridobljeno 7. 3. 2022 na spletnem naslovu: https://en.wikipedia.org/wiki/USB
2. *What is Micro USB Pinout and Types* 2022 [online]. eTechnophiles. Pridobljeno 7. 3. 2022 na spletnem naslovu: https://www.etechnophiles.com/micro-usb-pinout-features/
3. *USB(Communications)* 2022 [online]. Wikipedija, The Free Encyclopedia. Pridobljeno 7. 3. 2022 na spletnem naslovu: https://en.wikipedia.org/wiki/USB\_(Communications)