

Osnove mikroprocesorske elektronike

Vaja 7: Realizacija avtomata stanj

Naloga:

Na novo napišite igrico "Ujemi voluharja", tokrat v obliki avtomata stanj

Program organizirajte tako, da bo glavna zanka enkrat vsako milisekundo poklicala podprograma "Igra" in "ServisnaOpravlila". To pomeni, da se morata oba podprograma vsakič zaključiti v manj kot eni milisekundi.

Podprogram "Igra" (ali "Game") preveri odziv uporabnika in ustrezno ukrepa glede na trenutno stanje. Načrt za podprogram "Igra" narišite v obliki diagrama prehajanja stanj (kot za avtomat stanj) in ga nato realizirajte s pomočjo ukaza "switch" in spremenljivke stanja "stanje".

Podprogram "ServisnaOpravlila" (ali "BackgroundServices") naj bere tipkovnico in osvežuje prikaz na LED diodah.

Komunikacijo med podprogramoma "Igra" in "ServisnaOpravlila" napravite preko globalnih spremenljivk (npr.: tocke, voluharji, zivljenja, nivo ...).

Specifikacija nadgrajene igre:

Igra naj ima dva načina delovanja:

- Enostaven: vsakič se prikaže en voluhar, igra ima 7 nivojev težavnosti.
- Napreden: vsakič se prikažeta dva voluharja, igra ima 7 nivojev težavnosti.

Enostaven način igre zaženemo s tipko T1 ali T2, napreden način igre zaženemo s tipko T3 ali T4.

Začetni prikaz (ko čakamo, da igralec začne igro): Vse LED diode utripajo dvakrat na sekundo.

Končni prikaz (prikaz doseženih točk po koncu igre):

- LED diode pomežiknejo petkrat, s hitrostjo petkrat na sekundo,
- prikaže se rezultat v binarni obliki,
- prvo sekundo ignoriramo tipke,
- po eni sekundi preidemo nazaj na začetek, ko igralec pritisne katerokoli tipko.
- Če tipka ni pritisnjena celo minuto, vseeno skočimo na začetek.

Prikaz na LED diodah med igro:

- 7, 6, 5: trenutni nivo težavnosti
- 4: izključena
- 3, 2, 1, 0: voluharji

Domača naloga:

1. Po vzoru napredne igre naredite še enostavno igro.
2. Na enak način kot oba tipa igre naredite še končni prikaz.