

Spisak tema za izradu projekta u školskoj 2018/2019. godini

Pravila za izradu projekata:

1. Projekat se bira sa liste predloženih projekata, ili svaki projektni tim predlaže svoj projekat. Ukoliko predloženi projekat odobre predmetni nastavnici, projektni tim može započeti izradu projekta.
2. Trajanje izrade projekta nije vremenski ograničeno, ali se očekuje da se projekat završi u toku tekuće školske godine, odnosno najkasnije do oktobra 2019. godine.
3. Veličina projektnog tima nije striktno definisana, ali preporučeni broj članova projektnog tima iznosi 4 člana. Projektni tim ne može biti veći od 4 člana, ali može biti manji.
4. Za svaki od predloženih projekata navedena je i maksimalna ocena koja se može dobiti prilikom izrade datog projekta, kao i maksimalan broj članova projektnog tima.
5. Navedena maksimalna ocena znači da je to ujedno i maksimalna ocena koja se može dobiti i na svakom od predmeta koje pokriva dati projekat, kao i na završnom Bachelor radu, ukoliko se projekat koristi kao osnova za izradu Bachelor rada.
6. Prilikom izrade projekta očekuje se da svaki od članova projektnog tima učestvuje u svakoj od faza realizacije projekta (system level dizajn, RTL dizajn, funkcionalna verifikacija, razvoj Linux drajvera, razvoj korisničke aplikacije, pisanje tehničke dokumentacije, itd.). Ono što je dozvoljeno jeste da postoji jedna glavna osoba za svaku od ovih faza, koja će biti odgovorna za kontrolisanje toka realizacije te faze.
7. Prilikom odbrane projekta, svaki od članova projektnog tima učestvuje u odbrani svake od faza projekta, i u zavisnosti od pruženih odgovora dobija odgovarajuću ocenu za tu fazu, koja ujedno znači i da je to ocena praktičnog dela odgovarajućeg ispita kojim je posmatrana faza pokrivena.
8. Projekat se može braniti po fazama, kako se koja kompletira. Nakon uspešne odbrane jedne faze projekta studentima se upisuju bodovi iz praktičnog dela ispita koji je pokriven dotičnom fazom.
9. Projekat, ili jedna od njegovih faza, se smatra odbranjenim tek kada svaki od članova projektnog tima dobije prelaznu ocenu. Dok se to ne desi, niko od članova tima ne može smatrati da je uspešno odbranio projekat i neće mu biti zaključena ocena iz predmeta koji je pokriven posmatranom fazom.
10. Ukoliko projekat ukjučuje veći broj predmeta, odbrana se izvodi za svaki predmet odvojeno, a tako se dobijaju i ocene. Da bi se položio jedan predmet, čije je gradivo pokriveno projektom, potrebno je odbraniti deo projekta koji se odnosi na dotični predmet.

Lista projekata za školsku 2018/2019. godinu:

1. **Digital Audio Equalizer System**, baziran na korišćenju filter banki (4 studenta, maksimalna ocena 10)
2. **Digital Audio Equalizer System**, baziran na FFT transformaciji (4 studenta, maksimalna ocena 10)
3. **Audio Scrambler System** (4 studenta, maksimalna ocena 10)
4. **Audio Compressed File Player System** (4 studenta, maksimalna ocena 10)
5. **Video/Image Filtering System** (4 studenta, maksimalna ocena 10)
6. **Video Motion Detection System** (4 studenta, maksimalna ocena 10)
7. **Motion JPEG Compression System** (4 studenta, maksimalna ocena 10)
8. **Data Encryption System** (4 studenta, maksimalna ocena 8)

Prilikom izrade svakog projekta potrebno je izvršiti sledeće zadatke:

1. Kreirati izvršnu specifikaciju kompletnog sistema (koristeći na primer, MATLAB, C, C++, Python, ili neki drugi programski jezik, u dogovoru sa predmetnim nastavnicima).
2. Izvršiti podelu čitavog sistema na hardversku i softversku komponentu i kreirati virtualnu platformu kompletnog sistema (koristeći SystemC jezik).
3. Projektovati potreban broj IP jezgara, ili koristiti postojeća IP jezgra, neophodan za kompletno implementiranje hardverske komponente sistema, koristeći RT metodologiju projektovanja IP jezgara i neki od jezika za modelovanje hardvera (VHDL, Verilog, SystemVerilog). **NAPOMENA:** Studenti moraju samostalno isprojektovati barem jedno IP jezgro. Barem jedno od korišćenih IP jezgara potrebno je ugraditi u virtualnu platformu, razvijenu u tački 2.
4. Razviti verifikaciono okruženje, korišćenjem UVM metodologije i SystemVerilog jezika, i izvršiti verifikaciju korektnog rada svakog IP jezgra koje je razvijeno u koraku 3.
5. Zapakovati sva IP jezgra razvijena u koraku 3, korišćenjem Vivado IP Packager alata.
6. Razviti ARM bazirani embeded sistem, koji se može implementirati na ZYBO/ZedBoard razvojnom sistemu, a koji implementira kompletnu hardversku komponentu sistema.
7. Razviti i verifikovati potrebne Linux drajvere za svako od IP jezgara razvijenih u koraku 3. Za postojeća IP jezgra, koja su neophodna za ostvarivanje željene funkcionalnosti a nisu razvijena u koraku 3, odabrati i verifikovati neki od postojećih Linux drajvera.
8. Razviti i verifikovati korisničku aplikaciju, pod Linux operativnim sistemom, koja će implementirati softversku komponentu sistema koji se razvija.
9. Napisati tehničku dokumentaciju kompletnog sistema, koja uključuje:

- 9.1. Opis izvršne specifikacije.
 - 9.2. Opis kriterijuma i načina HW/SW particionisanja. kao i samo particionisanje.
 - 9.3. Opis i način korišćenja razvijene virtualne platforme.
 - 9.4. Opis svakog IP jezgra koje je razvijeno (funkcionalnost koju IP jezgro implementira, blok dijagram IP jezgra, interfejsi IP jezgra, registarska mapa IP jezgra, potrebni hardverski resursi za implementaciju i performanse IP jezgra, itd.).
 - 9.5. Opis i način korišćenja svakog verifikacionog okruženja (struktura verifikacionog okruženja, funkcija svake od verifikacionih komponenti, odabrani način generisanja stimulusa i vršenja provera, spisak svih testova koji su izvršeni prilikom verifikacije, opis implementiranog sistema za prikljupljanje verifikacione pokrivenosti i analiza prikupljene verifikacione pokrivenosti, itd.).
 - 9.6. Opis razvijenog ARM baziranog sistema.
 - 9.7. Opis i način korišćenja svakog Linux drajvera koji je razvijen.
 - 9.8. Opis i način korišćenja korisničke aplikacije.
 - 9.9. Opis korišćenja čitavog embeded sistema koji je razvijen, kao i rezultate hardverskog testiranja (spisak testiranih funkcija na razvojnom sistemu).
 - 9.10. Analizu postignutih performansi, kao i analizu mogućih unapređenja razvijenog sistema.
10. Demonstrirati ispravan rad čitavog sistema na nekom od razvojnih sistema.