

<p>Blic1, OSUR, <i>student</i>:</p> <p>Neki projekt se sastoji od dvije datoteke s izvornim kodom <code>main.c</code> i <code>obrada.c</code> te zaglavljem <code>obrada.h</code> koje se uključuje u obje datoteke. Nadopuniti sljedeći Makefile. Dopune mogu biti i u već zadanom dijelu (ali i ne moraju).</p> <pre>CFLAGS = -DX=2 #-MMD LDFLAGS = -O2 LDLIBS = -lpthread prog: main.o obrada.o     \$(CC) \$(LDFLAGS) main.o obrada.o \$(LDLIBS) -o prog #    \$(CC) \$(LDFLAGS) \$^ \$(LDLIBS) -o \$@ #narančasto umjesto plavog, otkomentirati  main.o: main.c obrada.h  obrada.o: obrada.c obrada.h  #-include *.d #zeleno umjesto crvenog, otkomentirati</pre>	<p>Blic1, OSUR, <i>student</i>:</p> <p>Neki projekt se sastoji od tri datoteke s izvornim kodom <code>prvi.c</code>, <code>drugi.c</code> i <code>treci.c</code> te zaglavljem <code>zagi.h</code> koje se uključuje u sve datoteke. Nadopuniti sljedeći Makefile. Dopune mogu biti i u već zadanom dijelu (ali i ne moraju).</p> <pre>CFLAGS = -DY=2 #-MMD prog: prvi.o drugi.o treci.o     \$(CC) prvi.o drugi.o treci.o -o prog #    \$(CC) \$^ -o \$@ #narančasto umjesto plavog, otkomentirati  prvi.o: prvi.c zagi.h  drugi.o: drugi.c zagi.h  treci.o: treci.c zagi.h  #-include *.d #zeleno umjesto crvenog, otkomentirati</pre>
<p>Blic2-A, OSzUR, <i>student</i>:</p> <p>a) Nadopuniti postupak prihvata prekida (ono što radi sam procesor):</p> <ol style="list-style-type: none"> <li>1. zabrani daljnje prekidanje</li> <li>2. prebaci se u prekidni način rada</li> <li>3. <a href="#">PC i SR spremi na prekidni stog</a></li> <li>4. u PC stavi adresu prekidnog potprograma</li> </ol> <p>b) Koje sve informacije <b>moraju</b> biti zapisane u zaglavlju <b>zauzetog</b> bloka kojim upravlja neki algoritam dinamičkog upravljanja spremnikom (npr. algoritam prvi odgovarajući)?</p> <p><a href="#">veličina bloka</a>  <a href="#">oznaka zauzetosti</a></p>	<p>Blic2-B, OSzUR, <i>student</i>:</p> <p>a) Instrukcija za povratak iz prekida (npr. IRET) radi:</p> <ol style="list-style-type: none"> <li>1. <a href="#">s prekidna stoga obnavlja PC i SR</a></li> <li>2. prebaci se u način rada prekinute dretve (zapisan u RS)</li> <li>3. dozvoli prekidanje</li> </ol> <p>b) Koje sve informacije <b>moraju</b> biti zapisane u zaglavlju <b>slobodnog</b> bloka kojim upravlja neki algoritam dinamičkog upravljanja spremnikom (npr. algoritam prvi odgovarajući)?</p> <p><a href="#">veličina bloka</a>  <a href="#">oznaka zauzetosti</a>  <a href="#">kazaljka na sljedeći slobodni blok</a></p>
<p>Blic3-A, OSzUR, <i>student</i>:</p> <p>1. U izvornom kodu datoteka koje ostvaruju upravljački program za naprave nalazi se na samom početku datoteke <code>#ifdef IME</code> (i na kraju <code>#endif</code>). Zašto? Što to omogućuje?</p> <p>To omogućuje opcionalno uključivanje zadane datoteke u projekt. Ako se makro <code>IME</code> ne definira (npr. preko <code>-D IME</code>) onda je za prevoditelj datoteka prazna – neće se uključiti njen kod.</p> <p>2. Zašto se napravama pristupa preko OS a ne izravno iz programa?</p> <p>- potrebne su privilegirane instrukcije  - mogući problemi kada se koristi višedretvenost (paralelno korištenje)</p>	<p>Blic3-B, OSzUR, <i>student</i>:</p> <p>1. Navesti osnovne operacije koje mora ostvariti upravljački program neke naprave.</p> <p><a href="#">inicijaliziraj-napravu, pošalji-podatke-napravi, pročitaj-podatke-s-naprave, dohvati-status-naprave, deaktiviraj-napravu, funkcija-za-prihvat-prekida-naprave</a></p> <p>2. U opisniku naprave <code>kdevice_t</code> koji koristi jezgra nalazi se i element <code>ref_cnt</code> (reference counter)? Čemu on služi (i u drugim opisnicima)?</p> <p><a href="#">ref_cnt</a> broji koliko ostalih objekata ima referencu (kazaljku) na ovaj objekt. Objekt se ne bi smio osloboditi (<code>kfree</code>) dok god neki drugi objekt pamti njegovu adresu.</p>
<p>Blic4-A, OSzUR, <i>student</i>:</p> <p>1. Neko brojilo odbrojava od učitane vrijednosti do nule frekvencijom od 100 kHz. U nekom trenutku pročitana vrijednost u brojilu iznosila je 34251. Koliko će još vremena proteći (u mikrosekundama) dok brojilo ne izazove prekid?  <math>T_1 = 1/100 \text{ kHz} = 10 \mu\text{s}</math>  <math>T = 34251 * T_1 = 342510 \mu\text{s} = 0,34251 \text{ s}</math></p> <p>2. Upravljački program nekog ugrađenog sustava je:</p> <pre>while(1) {     *wdt = 10000;     obavi_korak_upravljanja(); }</pre> <p>Sustav koristi nadzorni alarm (watchdog timer) na adresi <code>wdt</code>. Što će se dogoditi ako <code>obavi_korak_upravljanja()</code> traje više od 10000 otkucaja brojila nadzornog alarma? (Odgovoriti na poledini.)  Kad <code>wdt</code> dođe do nule poslati će signal prekida na RESET ulaz procesora i na taj način resetirati cijeli sustav</p>	<p>Blic4-B, OSzUR, <i>student</i>:</p> <p>1. Neko 32-bitovno brojilo odbrojava od učitane vrijednosti do nule frekvencijom <math>2^{20}</math> Hz. Može li se brojilo programirati da ono izazove prekid za 10 sekundi? Ako da, koju vrijednost treba učitati u brojilo?  Otkucaja u 10 sekundi: <math>10 * 2^{20}</math> je još uvijek manje od <math>2^{32}</math> =&gt; može  U brojilo treba učitati <math>10 * 2^{20}</math></p> <p>2. Upravljački program nekog ugrađenog sustava je:</p> <pre>while(1) {     *wdt = 10000;     obavi_korak_upravljanja(); }</pre> <p>Sustav koristi nadzorni alarm (watchdog timer) na adresi <code>wdt</code>. Što će se dogoditi ako <code>obavi_korak_upravljanja()</code> traje manje od 10000 otkucaja brojila nadzornog alarma? (Odgovoriti na poledini.)  Ako funkcija traje kraće, <code>wdt</code> će se nanovo učitati i neće resetirati sustav</p>