**Projektni zadatak iz Sistemskog Softvera**

## ***Zadatak I* – Asembler**

### Opis rešenja:

Relaizovani program prevodi uneti tekstualni fajl sa asemblerskim kodom definisanim za zadat sistem prevodi u objektne relokatibilne fajlove po redukovanoj verziji ELF formata opisanoj u daljem tekstu uputstva. Parsiranje ulaznog fajla se vrši uz pomoć GNUovih alata Flex i Bison.

Kako bi prilagodili ELF format opisanoj mašini definisali smo sledeće strukture:

Elf16\_Header – Koristi se za identifikaciju formata fajla i nosi osnovne informacije o sadržaju fajla

struct Elf16\_Header {

Byte format[ELF\_FORMAT\_LEN] = { 'E', 'L', 'F', '\_', '1', '6' }; // Koristi se za identifikaciju formata

Elf16\_File\_Type type; // Tip fajla (relokatibilan ili izvršni)

Offs phoffs; // Program header table offset (ne koristi se u okviru asemblera)

Word phentries; // Broj ulaza u program header table

Offs shoffs; // Ofset od pocetka fajla do tabele sekcija

Word shentries; // Broj ulaza u tabelu sekcija

Word symtabndx; // Indeks sekcija koja predstavlja tabelu simbola

Word strndx; // Indeks sekcija koja sadrži imena simbola (niz karaktera koji se završava nultim karakterom)

Word shstrndx; // Indeks sekcije koja sadrži imena sekcija (niz karaktera koji se završava nultim karakterom)

};

Elf16\_SH\_Entry – Predstavlja jedan ulaz u tabelu sekcija

struct Elf16\_SH\_Entry {

Offs name; // Ofset u okviru shstr sekcije do imena sekcije

Elf16\_Section\_Type type; // Tip sekcije: UND, PROGBITS, SYMTAB, STRTAB, REL

Offs offs; // Ofset od pocetka fajla do pocetka sekcije

Word size; // Velicina sekcije

Word rel; // Uparena relokaciona sekcija (ako postoji, ako ne – UND)

};

Elf16\_ST\_Entry – Predstavlja jedan ulaz u tabelu simbola

struct Elf16\_ST\_Entry {

Offs name; // Ofset u okviru str sekcije do imena simbola

Word value; // Vrednost simbola (ako je definisan, ako ne – 0)

Elf16\_Sym\_Bind bind; // Vezivanje simbola – lokalno ili globalno

Elf16\_Sym\_Type type; // Tip simbola: NOTYPE ili SECTION

Word shndx; // Indeks sekcije u okviru tabela sekcija kojoj ovaj simbol pripada (ako nije definisan – UND)

};

Elf16\_RT\_Entry – Predstavlja jedan ulaz u relokacionim tabelama

struct Elf16\_RT\_Entry {

Offs offs; // Ofset na kom se nalazi vrednost koju treba popraviti u okviru sekcije kojoj pripada ovaj relokacioni zapis

Elf16\_Rel\_Type type; // Tip relokacionog zapisa: ERT\_8, ERT\_16 ili ERT\_PC16

Word stndx; // Indeks simbola u okviru tabele simbola na koji se relokacioni zapis odnosi

};

### Prevođenje

Program se prevodi pomoću GNU Make alata izvršavanjem komande *make* u kontekstu korenog direktorijuma projekta.

### Testovi

Napomena: Navedeni testovi su ekvivalentni testovima korišćenim za testiranje drugog projektnog zadatka i iz tog razloga nazivi testova se odnose na testiranje opisanog računarskog sistema ali kroz same testove su pokazane funkcionalnosti svih aspekata opisanog asemblera.

# **EQU**

|  |  |
| --- | --- |
| Ulazni fajl text.s | |
| .extern d, data\_out  .global b, \_start  .equ a, d - '0'  .equ b, '2'  .equ sym0, a + b - d  .equ sym1, a + b - d + c  .section text  \_start:  mov $b, data\_out  mov $0, %r0  mov sym1(%r0), data\_out  jmp \*sym\_quit(%pc)  .section data  c:  .word 'a'  .word 'b'  .section end  .equ sym\_quit, quit  quit:  halt  .end | #./tests/equ/text.o  #sh\_offs sh\_entries sym\_tab\_ndx str\_tab\_ndx sh\_str\_ndx  0x0113 8 4 5 7  #sections  # name type offs size link ndx  0x0000 UND 0x0018 0 0 0 #  0x0001 PROGBITS 0x0018 23 6 1 # text  0x0006 PROGBITS 0x002f 4 0 2 # data  0x000b PROGBITS 0x0033 1 0 3 # end  0x000f SYMTAB 0x0034 96 0 4 # .symtab  0x0017 STRTAB 0x0094 59 0 5 # .strtab  0x001f REL 0x00cf 18 0 6 # text.rel  0x0028 STRTAB 0x00e1 50 0 7 # .shstrtab  #text  0x0000: 64 00 32 00 80 00 00 64 00 00 00 20  0x000c: 64 60 02 00 80 00 00 2c 6e e9 ff  #data  0x0000: 61 00 62 00  #end  0x0000: 04  #.symtab  # name value type bind shndx ndx  0x0000 0x0000 NOTYPE LOCAL UND 0 #  0x0001 0x0000 SECTION LOCAL 1 1 # text  0x0006 0x0000 SECTION LOCAL 2 2 # data  0x000b 0x0000 SECTION LOCAL 3 3 # end  0x000f 0x0002 NOTYPE LOCAL ABS 4 # sym0  0x0014 0x0000 NOTYPE GLOBAL 1 5 # \_start  0x001b 0x0032 NOTYPE GLOBAL ABS 6 # b  0x001d 0x0000 NOTYPE GLOBAL UND 7 # data\_out  0x0026 0x0002 NOTYPE LOCAL 2 8 # sym1  0x002b 0x0000 NOTYPE LOCAL 2 9 # c  0x002d 0x0000 NOTYPE LOCAL 3 10 # quit  0x0032 0x0000 NOTYPE LOCAL 3 11 # sym\_quit  #.strtab  0x0000: textdataendsym0\_star  0x0019: tbdata\_outsym1cquit  0x0032: sym\_quit  #text.rel  #offset type stndx ndx  0x000e ERT\_16 2 0  0x0005 ERT\_16 7 1  0x0011 ERT\_16 7 2  #.shstrtab  0x0000: textdataend.symtab.s  0x0019: trtabtext.rel.shstrtab |

|  |  |
| --- | --- |
| Ulazni fajl ivt\_tab.s | |
| .extern \_start  .global data\_in, data\_out  .section ivt\_tab  .word \_start  .word \_invalid\_instruction  .word \_timer  .word \_terminal  .section ivt\_tab.text  .equ data\_in, 0xFF00  .equ data\_out, 0xFF02  \_invalid\_instruction:  iret  \_timer:  iret  \_terminal:  iret  .end | #./tests/equ/ivt\_tab.o  #sh\_offs sh\_entries sym\_tab\_ndx str\_tab\_ndx sh\_str\_ndx  0x0113 7 3 4 6  #sections  # name type offs size link ndx  0x0000 UND 0x0018 0 0 0 #  0x0001 PROGBITS 0x0018 8 5 1 # ivt\_tab  0x0009 PROGBITS 0x0020 3 0 2 # ivt\_tab.text  0x0016 SYMTAB 0x0023 72 0 3 # .symtab  0x001e STRTAB 0x006b 84 0 4 # .strtab  0x0026 REL 0x00bf 24 0 5 # ivt\_tab.rel  0x0032 STRTAB 0x00d7 60 0 6 # .shstrtab  #ivt\_tab  0x0000: 00 00 00 00 01 00 02 00  #ivt\_tab.text  0x0000: 0c 0c 0c  #.symtab  # name value type bind shndx ndx  0x0000 0x0000 NOTYPE LOCAL UND 0 #  0x0001 0x0000 SECTION LOCAL 1 1 # ivt\_tab  0x0009 0x0000 SECTION LOCAL 2 2 # ivt\_tab.text  0x0016 0xff00 NOTYPE GLOBAL ABS 3 # data\_in  0x001e 0x0002 NOTYPE LOCAL 2 4 # \_terminal  0x0028 0xff02 NOTYPE GLOBAL ABS 5 # data\_out  0x0031 0x0001 NOTYPE LOCAL 2 6 # \_timer  0x0038 0x0000 NOTYPE GLOBAL UND 7 # \_start  0x003f 0x0000 NOTYPE LOCAL 2 8 # \_invalid\_instruction  #.strtab  0x0000: ivt\_tabivt\_tab.textdat  0x0019: a\_in\_terminaldata\_out\_  0x0032: timer\_start\_invalid\_ins  0x004b: truction  #ivt\_tab.rel  #offset type stndx ndx  0x0002 ERT\_16 2 0  0x0004 ERT\_16 2 1  0x0006 ERT\_16 2 2  0x0000 ERT\_16 7 3  #.shstrtab  0x0000: ivt\_tabivt\_tab.text.sy  0x0019: mtab.strtabivt\_tab.rel  0x0032: .shstrtab |

# **Instrukcije (2B)**

|  |  |
| --- | --- |
| Ulazni fajl text.s | |
| .global \_start  .extern data\_out  .section text  \_start:  push char\_b  push char\_a  call write\_func  pop var  pop var  xchg char\_a, char\_b  push char\_b  push char\_a  call write\_func  pop var  pop var  mov $0, var  call write\_out  add $1, var  call write\_out  sub $-1, var  call write\_out  mul $4, var  call write\_out  div $2, var  call write\_out  not $0xfff8, var  call write\_out  and $9, var  call write\_out  or $6, var  call write\_out  xor $0xe, var  call write\_out  shr var, $2  call write\_out  shl $1, var  call write\_out  halt  .equ char\_a\_offs, 2  .equ char\_b\_offs, 4  write\_func:  mov char\_a\_offs(%sp), data\_out  mov char\_b\_offs(%sp), data\_out  mov $0xA, data\_out  ret  write\_out:  add $'0', var  mov var, data\_out  mov $0xA, data\_out  sub $'0', var  ret  var:  .word 0  char\_a:  .word 'a'  char\_b:  .word 'b'  .end | #./tests/instructions/text.o  #sh\_offs sh\_entries sym\_tab\_ndx str\_tab\_ndx sh\_str\_ndx  0x02cc 6 2 3 5  #sections  # name type offs size link ndx  0x0000 UND 0x0018 0 0 0 #  0x0001 PROGBITS 0x0018 226 4 1 # text  0x0006 SYMTAB 0x00fa 88 0 2 # .symtab  0x000e STRTAB 0x0152 85 0 3 # .strtab  0x0016 REL 0x01a7 252 0 4 # text.rel  0x001f STRTAB 0x02a3 41 0 5 # .shstrtab  #text  0x0000: 4c 80 e0 00 4c 80 de 00 24 00 a9 00  0x000c: 54 80 dc 00 54 80 dc 00 5c 80 de 00  0x0018: 80 e0 00 4c 80 e0 00 4c 80 de 00 24  0x0024: 00 a9 00 54 80 dc 00 54 80 dc 00 64  0x0030: 00 00 00 80 dc 00 24 00 bf 00 6c 00  0x003c: 01 00 80 dc 00 24 00 bf 00 74 00 ff  0x0048: ff 80 dc 00 24 00 bf 00 7c 00 04 00  0x0054: 80 dc 00 24 00 bf 00 84 00 02 00 80  0x0060: dc 00 24 00 bf 00 94 00 f8 ff 80 dc  0x006c: 00 24 00 bf 00 9c 00 09 00 80 dc 00  0x0078: 24 00 bf 00 a4 00 06 00 80 dc 00 24  0x0084: 00 bf 00 ac 00 0e 00 80 dc 00 24 00  0x0090: bf 00 c4 80 dc 00 00 02 00 24 00 bf  0x009c: 00 bc 00 01 00 80 dc 00 24 00 bf 00  0x00a8: 04 64 6c 02 00 80 00 00 64 6c 04 00  0x00b4: 80 00 00 64 00 0a 00 80 00 00 14 6c  0x00c0: 00 30 00 80 dc 00 64 80 dc 00 80 00  0x00cc: 00 64 00 0a 00 80 00 00 74 00 30 00  0x00d8: 80 dc 00 14 00 00 61 00 62 00  #.symtab  # name value type bind shndx ndx  0x0000 0x0000 NOTYPE LOCAL UND 0 #  0x0001 0x0000 SECTION LOCAL 1 1 # text  0x0006 0x0000 NOTYPE GLOBAL UND 2 # data\_out  0x000f 0x0004 NOTYPE LOCAL ABS 3 # char\_b\_offs  0x001b 0x0000 NOTYPE GLOBAL 1 4 # \_start  0x0022 0x0002 NOTYPE LOCAL ABS 5 # char\_a\_offs  0x002e 0x00de NOTYPE LOCAL 1 6 # char\_a  0x0035 0x00dc NOTYPE LOCAL 1 7 # var  0x0039 0x00a9 NOTYPE LOCAL 1 8 # write\_func  0x0044 0x00e0 NOTYPE LOCAL 1 9 # char\_b  0x004b 0x00bf NOTYPE LOCAL 1 10 # write\_out  #.strtab  0x0000: textdata\_outchar\_b\_off  0x0019: s\_startchar\_a\_offschar  0x0032: \_avarwrite\_funcchar\_b  0x004b: write\_out  #text.rel  #offset type stndx ndx  0x000a ERT\_16 1 0  0x0025 ERT\_16 1 1  0x0038 ERT\_16 1 2  0x0043 ERT\_16 1 3  0x004e ERT\_16 1 4  0x0059 ERT\_16 1 5  0x0064 ERT\_16 1 6  0x006f ERT\_16 1 7  0x007a ERT\_16 1 8  0x0085 ERT\_16 1 9  0x0090 ERT\_16 1 10  0x009b ERT\_16 1 11  0x00a6 ERT\_16 1 12  0x000e ERT\_16 1 13  0x0012 ERT\_16 1 14  0x0029 ERT\_16 1 15  0x002d ERT\_16 1 16  0x0034 ERT\_16 1 17  0x003f ERT\_16 1 18  0x004a ERT\_16 1 19  0x0055 ERT\_16 1 20  0x0060 ERT\_16 1 21  0x006b ERT\_16 1 22  0x0076 ERT\_16 1 23  0x0081 ERT\_16 1 24  0x008c ERT\_16 1 25  0x0094 ERT\_16 1 26  0x00a2 ERT\_16 1 27  0x00c4 ERT\_16 1 28  0x00c8 ERT\_16 1 29  0x00d9 ERT\_16 1 30  0x0006 ERT\_16 1 31  0x0016 ERT\_16 1 32  0x0021 ERT\_16 1 33  0x0002 ERT\_16 1 34  0x0019 ERT\_16 1 35  0x001d ERT\_16 1 36  0x00ae ERT\_16 2 37  0x00b5 ERT\_16 2 38  0x00bc ERT\_16 2 39  0x00cb ERT\_16 2 40  0x00d2 ERT\_16 2 41  #.shstrtab  0x0000: text.symtab.strtabtex  0x0019: t.rel.shstrtab |

|  |  |
| --- | --- |
| Ulazni fajl ivt\_tab.s | |
| .extern \_start  .global data\_in, data\_out  .section ivt\_tab  .word \_start  .word \_invalid\_instruction  .word \_timer  .word \_terminal  .section ivt\_tab.text  .equ data\_in, 0xFF00  .equ data\_out, 0xFF02  \_invalid\_instruction:  iret  \_timer:  iret  \_terminal:  iret  .end | #./tests/equ/ivt\_tab.o  #sh\_offs sh\_entries sym\_tab\_ndx str\_tab\_ndx sh\_str\_ndx  0x0113 7 3 4 6  #sections  # name type offs size link ndx  0x0000 UND 0x0018 0 0 0 #  0x0001 PROGBITS 0x0018 8 5 1 # ivt\_tab  0x0009 PROGBITS 0x0020 3 0 2 # ivt\_tab.text  0x0016 SYMTAB 0x0023 72 0 3 # .symtab  0x001e STRTAB 0x006b 84 0 4 # .strtab  0x0026 REL 0x00bf 24 0 5 # ivt\_tab.rel  0x0032 STRTAB 0x00d7 60 0 6 # .shstrtab  #ivt\_tab  0x0000: 00 00 00 00 01 00 02 00  #ivt\_tab.text  0x0000: 0c 0c 0c  #.symtab  # name value type bind shndx ndx  0x0000 0x0000 NOTYPE LOCAL UND 0 #  0x0001 0x0000 SECTION LOCAL 1 1 # ivt\_tab  0x0009 0x0000 SECTION LOCAL 2 2 # ivt\_tab.text  0x0016 0xff00 NOTYPE GLOBAL ABS 3 # data\_in  0x001e 0x0002 NOTYPE LOCAL 2 4 # \_terminal  0x0028 0xff02 NOTYPE GLOBAL ABS 5 # data\_out  0x0031 0x0001 NOTYPE LOCAL 2 6 # \_timer  0x0038 0x0000 NOTYPE GLOBAL UND 7 # \_start  0x003f 0x0000 NOTYPE LOCAL 2 8 # \_invalid\_instruction  #.strtab  0x0000: ivt\_tabivt\_tab.textdat  0x0019: a\_in\_terminaldata\_out\_  0x0032: timer\_start\_invalid\_ins  0x004b: truction  #ivt\_tab.rel  #offset type stndx ndx  0x0002 ERT\_16 2 0  0x0004 ERT\_16 2 1  0x0006 ERT\_16 2 2  0x0000 ERT\_16 7 3  #.shstrtab  0x0000: ivt\_tabivt\_tab.text.sy  0x0019: mtab.strtabivt\_tab.rel  0x0032: .shstrtab |

# **Instrukcije (1B)**

|  |  |
| --- | --- |
| Ulazni fajl text.s | |
| .global \_start  .extern data\_out  .section text  \_start:  push char\_b  push char\_a  call write\_func  pop var  pop var  xchgb char\_a, char\_b  push char\_b  push char\_a  call write\_func  pop var  pop var  movb $0, var  call write\_out  addb $1, var  call write\_out  subb $-1, var  call write\_out  mulb $4, var  call write\_out  divb $2, var  call write\_out  notb $0xf8, var  call write\_out  andb $9, var  call write\_out  orb $6, var  call write\_out  xorb $0xe, var  call write\_out  shrb var, $2  call write\_out  shlb $1, var  call write\_out  halt  .equ char\_a\_offs, 2  .equ char\_b\_offs, 4  write\_func:  movb char\_a\_offs(%sp), data\_out  movb char\_b\_offs(%sp), data\_out  movb $0xA, data\_out  ret  write\_out:  addb $'0', var  movb var, data\_out  movb $0xA, data\_out  subb $'0', var  ret  var:  .word 0  char\_a:  .byte 'a'  char\_b:  .byte 'b'  .end | #./tests/instructions\_byte/text.o  #sh\_offs sh\_entries sym\_tab\_ndx str\_tab\_ndx sh\_str\_ndx  0x02bb 6 2 3 5  #sections  # name type offs size link ndx  0x0000 UND 0x0018 0 0 0 #  0x0001 PROGBITS 0x0018 209 4 1 # text  0x0006 SYMTAB 0x00e9 88 0 2 # .symtab  0x000e STRTAB 0x0141 85 0 3 # .strtab  0x0016 REL 0x0196 252 0 4 # text.rel  0x001f STRTAB 0x0292 41 0 5 # .shstrtab  #text  0x0000: 4c 80 d0 00 4c 80 cf 00 24 00 9e 00  0x000c: 54 80 cd 00 54 80 cd 00 58 80 cf 00  0x0018: 80 d0 00 4c 80 d0 00 4c 80 cf 00 24  0x0024: 00 9e 00 54 80 cd 00 54 80 cd 00 60  0x0030: 00 00 80 cd 00 24 00 b3 00 68 00 01  0x003c: 80 cd 00 24 00 b3 00 70 00 ff 80 cd  0x0048: 00 24 00 b3 00 78 00 04 80 cd 00 24  0x0054: 00 b3 00 80 00 02 80 cd 00 24 00 b3  0x0060: 00 90 00 f8 80 cd 00 24 00 b3 00 98  0x006c: 00 09 80 cd 00 24 00 b3 00 a0 00 06  0x0078: 80 cd 00 24 00 b3 00 a8 00 0e 80 cd  0x0084: 00 24 00 b3 00 c0 80 cd 00 00 02 24  0x0090: 00 b3 00 b8 00 01 80 cd 00 24 00 b3  0x009c: 00 04 60 6c 02 00 80 00 00 60 6c 04  0x00a8: 00 80 00 00 60 00 0a 80 00 00 14 68  0x00b4: 00 30 80 cd 00 60 80 cd 00 80 00 00  0x00c0: 60 00 0a 80 00 00 70 00 30 80 cd 00  0x00cc: 14 00 00 61 62  #.symtab  # name value type bind shndx ndx  0x0000 0x0000 NOTYPE LOCAL UND 0 #  0x0001 0x0000 SECTION LOCAL 1 1 # text  0x0006 0x0000 NOTYPE GLOBAL UND 2 # data\_out  0x000f 0x0004 NOTYPE LOCAL ABS 3 # char\_b\_offs  0x001b 0x0000 NOTYPE GLOBAL 1 4 # \_start  0x0022 0x0002 NOTYPE LOCAL ABS 5 # char\_a\_offs  0x002e 0x00cf NOTYPE LOCAL 1 6 # char\_a  0x0035 0x00cd NOTYPE LOCAL 1 7 # var  0x0039 0x009e NOTYPE LOCAL 1 8 # write\_func  0x0044 0x00d0 NOTYPE LOCAL 1 9 # char\_b  0x004b 0x00b3 NOTYPE LOCAL 1 10 # write\_out  #.strtab  0x0000: textdata\_outchar\_b\_off  0x0019: s\_startchar\_a\_offschar  0x0032: \_avarwrite\_funcchar\_b  0x004b: write\_out  #text.rel  #offset type stndx ndx  0x000a ERT\_16 1 0  0x0025 ERT\_16 1 1  0x0037 ERT\_16 1 2  0x0041 ERT\_16 1 3  0x004b ERT\_16 1 4  0x0055 ERT\_16 1 5  0x005f ERT\_16 1 6  0x0069 ERT\_16 1 7  0x0073 ERT\_16 1 8  0x007d ERT\_16 1 9  0x0087 ERT\_16 1 10  0x0091 ERT\_16 1 11  0x009b ERT\_16 1 12  0x000e ERT\_16 1 13  0x0012 ERT\_16 1 14  0x0029 ERT\_16 1 15  0x002d ERT\_16 1 16  0x0033 ERT\_16 1 17  0x003d ERT\_16 1 18  0x0047 ERT\_16 1 19  0x0051 ERT\_16 1 20  0x005b ERT\_16 1 21  0x0065 ERT\_16 1 22  0x006f ERT\_16 1 23  0x0079 ERT\_16 1 24  0x0083 ERT\_16 1 25  0x008b ERT\_16 1 26  0x0097 ERT\_16 1 27  0x00b7 ERT\_16 1 28  0x00bb ERT\_16 1 29  0x00ca ERT\_16 1 30  0x0006 ERT\_16 1 31  0x0016 ERT\_16 1 32  0x0021 ERT\_16 1 33  0x0002 ERT\_16 1 34  0x0019 ERT\_16 1 35  0x001d ERT\_16 1 36  0x00a3 ERT\_16 2 37  0x00aa ERT\_16 2 38  0x00b0 ERT\_16 2 39  0x00be ERT\_16 2 40  0x00c4 ERT\_16 2 41  #.shstrtab  0x0000: text.symtab.strtabtex  0x0019: t.rel.shstrtab |

|  |  |
| --- | --- |
| Ulazni fajl ivt\_tab.s | |
| .extern \_start  .global data\_in, data\_out  .section ivt\_tab  .word \_start  .word \_invalid\_instruction  .word \_timer  .word \_terminal  .section ivt\_tab.text  .equ data\_in, 0xFF00  .equ data\_out, 0xFF02  \_invalid\_instruction:  iret  \_timer:  iret  \_terminal:  iret  .end | #./tests/equ/ivt\_tab.o  #sh\_offs sh\_entries sym\_tab\_ndx str\_tab\_ndx sh\_str\_ndx  0x0113 7 3 4 6  #sections  # name type offs size link ndx  0x0000 UND 0x0018 0 0 0 #  0x0001 PROGBITS 0x0018 8 5 1 # ivt\_tab  0x0009 PROGBITS 0x0020 3 0 2 # ivt\_tab.text  0x0016 SYMTAB 0x0023 72 0 3 # .symtab  0x001e STRTAB 0x006b 84 0 4 # .strtab  0x0026 REL 0x00bf 24 0 5 # ivt\_tab.rel  0x0032 STRTAB 0x00d7 60 0 6 # .shstrtab  #ivt\_tab  0x0000: 00 00 00 00 01 00 02 00  #ivt\_tab.text  0x0000: 0c 0c 0c  #.symtab  # name value type bind shndx ndx  0x0000 0x0000 NOTYPE LOCAL UND 0 #  0x0001 0x0000 SECTION LOCAL 1 1 # ivt\_tab  0x0009 0x0000 SECTION LOCAL 2 2 # ivt\_tab.text  0x0016 0xff00 NOTYPE GLOBAL ABS 3 # data\_in  0x001e 0x0002 NOTYPE LOCAL 2 4 # \_terminal  0x0028 0xff02 NOTYPE GLOBAL ABS 5 # data\_out  0x0031 0x0001 NOTYPE LOCAL 2 6 # \_timer  0x0038 0x0000 NOTYPE GLOBAL UND 7 # \_start  0x003f 0x0000 NOTYPE LOCAL 2 8 # \_invalid\_instruction  #.strtab  0x0000: ivt\_tabivt\_tab.textdat  0x0019: a\_in\_terminaldata\_out\_  0x0032: timer\_start\_invalid\_ins  0x004b: truction  #ivt\_tab.rel  #offset type stndx ndx  0x0002 ERT\_16 2 0  0x0004 ERT\_16 2 1  0x0006 ERT\_16 2 2  0x0000 ERT\_16 7 3  #.shstrtab  0x0000: ivt\_tabivt\_tab.text.sy  0x0019: mtab.strtabivt\_tab.rel  0x0032: .shstrtab |

# **Prekid prekidna rutina pri nailasku na nekorektnu instrukciju**

|  |  |
| --- | --- |
| Ulazni fajl text.s | |
| .global \_start  .section text  \_start:  div $0, \_start  .end | #./tests/interrupt\_invalid\_ins/text.o  #sh\_offs sh\_entries sym\_tab\_ndx str\_tab\_ndx sh\_str\_ndx  0x0071 6 2 3 5  #sections  # name type offs size link ndx  0x0000 UND 0x0016 0 0 0 #  0x0001 PROGBITS 0x0016 7 4 1 # text  0x0006 SYMTAB 0x001d 24 0 2 # .symtab  0x000e STRTAB 0x0035 13 0 3 # .strtab  0x0016 REL 0x0042 6 0 4 # text.rel  0x001f STRTAB 0x0048 41 0 5 # .shstrtab  #text  0x0000: 84 00 00 00 80 00 00  #.symtab  # name value type bind shndx ndx  0x0000 0x0000 NOTYPE LOCAL UND 0 #  0x0001 0x0000 SECTION LOCAL 1 1 # text  0x0006 0x0000 NOTYPE GLOBAL 1 2 # \_start  #.strtab  0x0000: text\_start  #text.rel  #offset type stndx ndx  0x0005 ERT\_16 1 0  #.shstrtab  0x0000: text.symtab.strtabtex  0x0019: t.rel.shstrtab |

|  |  |
| --- | --- |
| Ulazni fajl ivt\_tab.s | |
| .extern \_start  .global data\_in, data\_out  .section ivt\_tab  .word \_start  .word \_invalid\_instruction  .word \_timer  .word \_terminal  .section ivt\_tab.text  .equ data\_in, 0xFF00  .equ data\_out, 0xFF02  \_invalid\_instruction:  mov $0, %r0  loop:  mov msg(%r0), data\_out  add $1, %r0  cmp msg(%r0), $0  jne loop  halt  quit:  halt  \_timer:  iret  \_terminal:  iret  .section ivt\_tab.data  msg:  .byte 'i'  .byte 'n'  .byte 'v'  .byte 'a'  .byte 'l'  .byte 'i'  .byte 'd'  .byte ' '  .byte 'i'  .byte 'n'  .byte 's'  .byte 't'  .byte 'r'  .byte 'u'  .byte 'c'  .byte 't'  .byte 'i'  .byte 'o'  .byte 'n'  .byte 0xa  .byte 0  .end | #./tests/interrupt\_invalid\_ins/ivt\_tab.o  #sh\_offs sh\_entries sym\_tab\_ndx str\_tab\_ndx sh\_str\_ndx  0x01ae 9 4 5 8  #sections  # name type offs size link ndx  0x0000 UND 0x0016 0 0 0 #  0x0001 PROGBITS 0x0016 8 7 1 # ivt\_tab  0x0009 PROGBITS 0x001e 32 6 2 # ivt\_tab.text  0x0016 PROGBITS 0x003e 21 0 3 # ivt\_tab.data  0x0023 SYMTAB 0x0053 104 0 4 # .symtab  0x002b STRTAB 0x00bb 111 0 5 # .strtab  0x0033 REL 0x012a 18 0 6 # ivt\_tab.text.rel  0x0044 REL 0x013c 24 0 7 # ivt\_tab.rel  0x0050 STRTAB 0x0154 90 0 8 # .shstrtab  #ivt\_tab  0x0000: 00 00 00 00 1e 00 1f 00  #ivt\_tab.text  0x0000: 64 00 00 00 20 64 60 00 00 80 02 ff  0x000c: 6c 00 01 00 20 8c 60 00 00 00 00 00  0x0018: 3c 00 05 00 04 04 0c 0c  #ivt\_tab.data  0x0000: 69 6e 76 61 6c 69 64 20 69 6e 73 74  0x000c: 72 75 63 74 69 6f 6e 0a 00  #.symtab  # name value type bind shndx ndx  0x0000 0x0000 NOTYPE LOCAL UND 0 #  0x0001 0x0000 SECTION LOCAL 1 1 # ivt\_tab  0x0009 0x0000 SECTION LOCAL 2 2 # ivt\_tab.text  0x0016 0x0000 SECTION LOCAL 3 3 # ivt\_tab.data  0x0023 0x001d NOTYPE LOCAL 2 4 # quit  0x0028 0x0005 NOTYPE LOCAL 2 5 # loop  0x002d 0x0000 NOTYPE LOCAL 2 6 # \_invalid\_instruction  0x0042 0x001e NOTYPE LOCAL 2 7 # \_timer  0x0049 0x0000 NOTYPE LOCAL 3 8 # msg  0x004d 0xff02 NOTYPE GLOBAL ABS 9 # data\_out  0x0056 0x0000 NOTYPE GLOBAL UND 10 # \_start  0x005d 0x001f NOTYPE LOCAL 2 11 # \_terminal  0x0067 0xff00 NOTYPE GLOBAL ABS 12 # data\_in  #.strtab  0x0000: ivt\_tabivt\_tab.textivt  0x0019: \_tab.dataquitloop\_inva  0x0032: lid\_instruction\_timerms  0x004b: gdata\_out\_start\_termin  0x0064: aldata\_in  #ivt\_tab.text.rel  #offset type stndx ndx  0x001a ERT\_16 2 0  0x0007 ERT\_16 3 1  0x0013 ERT\_16 3 2  #ivt\_tab.rel  #offset type stndx ndx  0x0002 ERT\_16 2 0  0x0004 ERT\_16 2 1  0x0006 ERT\_16 2 2  0x0000 ERT\_16 10 3  #.shstrtab  0x0000: ivt\_tabivt\_tab.textivt  0x0019: \_tab.data.symtab.strtab  0x0032: ivt\_tab.text.relivt\_tab  0x004b: .rel.shstrtab |

# **Prekidna rutina terminala**

|  |  |
| --- | --- |
| Ulazni fajl text.s | |
| .global \_start  .section text  \_start:  jmp \_start  .end | #./tests/interrupt\_terminal/text.o  #sh\_offs sh\_entries sym\_tab\_ndx str\_tab\_ndx sh\_str\_ndx  0x006e 6 2 3 5  #sections  # name type offs size link ndx  0x0000 UND 0x0016 0 0 0 #  0x0001 PROGBITS 0x0016 4 4 1 # text  0x0006 SYMTAB 0x001a 24 0 2 # .symtab  0x000e STRTAB 0x0032 13 0 3 # .strtab  0x0016 REL 0x003f 6 0 4 # text.rel  0x001f STRTAB 0x0045 41 0 5 # .shstrtab  #text  0x0000: 2c 00 00 00  #.symtab  # name value type bind shndx ndx  0x0000 0x0000 NOTYPE LOCAL UND 0 #  0x0001 0x0000 SECTION LOCAL 1 1 # text  0x0006 0x0000 NOTYPE GLOBAL 1 2 # \_start  #.strtab  0x0000: text\_start  #text.rel  #offset type stndx ndx  0x0002 ERT\_16 1 0  #.shstrtab  0x0000: text.symtab.strtabtex  0x0019: t.rel.shstrtab |

|  |  |
| --- | --- |
| Ulazni fajl ivt\_tab.s | |
| .extern \_start  .section ivt\_tab  .word \_start  .word \_invalid\_instruction  .word \_timer  .word \_terminal  .section ivt\_tab.text  .equ data\_in, 0xFF00  .equ data\_out, 0xFF02  \_invalid\_instruction:  iret  \_timer:  iret  .equ diff, 'a' - 'A'  \_terminal:  mov data\_in, %r0  mov %r0, data\_out  cmp %r0, $'a'  jgt exit  cmp $'z', %r0  jgt exit  cmp $'q', %r0  jeq quit  sub $diff, data\_out  exit:  iret  quit:  halt  .end | #./tests/interrupt\_terminal/ivt\_tab.o  #sh\_offs sh\_entries sym\_tab\_ndx str\_tab\_ndx sh\_str\_ndx  0x0188 8 3 4 7  #sections  # name type offs size link ndx  0x0000 UND 0x0016 0 0 0 #  0x0001 PROGBITS 0x0016 8 6 1 # ivt\_tab  0x0009 PROGBITS 0x001e 48 5 2 # ivt\_tab.text  0x0016 SYMTAB 0x004e 96 0 3 # .symtab  0x001e STRTAB 0x00ae 99 0 4 # .strtab  0x0026 REL 0x0111 18 0 5 # ivt\_tab.text.rel  0x0037 REL 0x0123 24 0 6 # ivt\_tab.rel  0x0043 STRTAB 0x013b 77 0 7 # .shstrtab  #ivt\_tab  0x0000: 00 00 00 00 01 00 02 00  #ivt\_tab.text  0x0000: 0c 0c 64 80 00 ff 20 64 20 80 02 ff  0x000c: 8c 20 00 61 00 44 00 2e 00 8c 00 7a  0x0018: 00 20 44 00 2e 00 8c 00 71 00 20 34  0x0024: 00 2f 00 74 00 20 00 80 02 ff 0c 04  #.symtab  # name value type bind shndx ndx  0x0000 0x0000 NOTYPE LOCAL UND 0 #  0x0001 0x0000 SECTION LOCAL 1 1 # ivt\_tab  0x0009 0x0000 SECTION LOCAL 2 2 # ivt\_tab.text  0x0016 0x002f NOTYPE LOCAL 2 3 # quit  0x001b 0x002e NOTYPE LOCAL 2 4 # exit  0x0020 0x0000 NOTYPE LOCAL 2 5 # \_invalid\_instruction  0x0035 0x0020 NOTYPE LOCAL ABS 6 # diff  0x003a 0x0001 NOTYPE LOCAL 2 7 # \_timer  0x0041 0xff02 NOTYPE LOCAL ABS 8 # data\_out  0x004a 0x0000 NOTYPE GLOBAL UND 9 # \_start  0x0051 0x0002 NOTYPE LOCAL 2 10 # \_terminal  0x005b 0xff00 NOTYPE LOCAL ABS 11 # data\_in  #.strtab  0x0000: ivt\_tabivt\_tab.textqui  0x0019: texit\_invalid\_instructi  0x0032: ondiff\_timerdata\_out\_  0x004b: start\_terminaldata\_in  #ivt\_tab.text.rel  #offset type stndx ndx  0x0013 ERT\_16 2 0  0x001c ERT\_16 2 1  0x0025 ERT\_16 2 2  #ivt\_tab.rel  #offset type stndx ndx  0x0002 ERT\_16 2 0  0x0004 ERT\_16 2 1  0x0006 ERT\_16 2 2  0x0000 ERT\_16 9 3  #.shstrtab  0x0000: ivt\_tabivt\_tab.text.sy  0x0019: mtab.strtabivt\_tab.text  0x0032: .relivt\_tab.rel.shstrta  0x004b: b |

# **Prekidna rutina tajmera**

|  |  |
| --- | --- |
| Ulazni fajl text.s | |
| .global \_start  .section text  \_start:  jmp \_start  .end | #./tests/interrupt\_timer/text.o  #sh\_offs sh\_entries sym\_tab\_ndx str\_tab\_ndx sh\_str\_ndx  0x006e 6 2 3 5  #sections  # name type offs size link ndx  0x0000 UND 0x0016 0 0 0 #  0x0001 PROGBITS 0x0016 4 4 1 # text  0x0006 SYMTAB 0x001a 24 0 2 # .symtab  0x000e STRTAB 0x0032 13 0 3 # .strtab  0x0016 REL 0x003f 6 0 4 # text.rel  0x001f STRTAB 0x0045 41 0 5 # .shstrtab  #text  0x0000: 2c 00 00 00  #.symtab  # name value type bind shndx ndx  0x0000 0x0000 NOTYPE LOCAL UND 0 #  0x0001 0x0000 SECTION LOCAL 1 1 # text  0x0006 0x0000 NOTYPE GLOBAL 1 2 # \_start  #.strtab  0x0000: text\_start  #text.rel  #offset type stndx ndx  0x0002 ERT\_16 1 0  #.shstrtab  0x0000: text.symtab.strtabtex  0x0019: t.rel.shstrtab |

|  |  |
| --- | --- |
| Ulazni fajl ivt\_tab.s | |
| .extern \_start  .section ivt\_tab  .word \_start  .word \_invalid\_instruction  .word \_timer  .word \_terminal  .section ivt\_tab.text  .equ data\_in, 0xFF00  .equ data\_out, 0xFF02  \_invalid\_instruction:  iret  \_timer:  add $1, cnt  mov cnt, %r0  add $'0', %r0  mov %r0, data\_out  cmp cnt, $9  jne exit  mov $0, cnt  exit:  iret  cnt:  .word 0  \_terminal:  cmp data\_in, $'q'  jeq quit  iret  quit:  halt  .end | #./tests/interrupt\_timer/ivt\_tab.o  #sh\_offs sh\_entries sym\_tab\_ndx str\_tab\_ndx sh\_str\_ndx  0x01a2 8 3 4 7  #sections  # name type offs size link ndx  0x0000 UND 0x0016 0 0 0 #  0x0001 PROGBITS 0x0016 8 6 1 # ivt\_tab  0x0009 PROGBITS 0x001e 57 5 2 # ivt\_tab.text  0x0016 SYMTAB 0x0057 96 0 3 # .symtab  0x001e STRTAB 0x00b7 98 0 4 # .strtab  0x0026 REL 0x0119 36 0 5 # ivt\_tab.text.rel  0x0037 REL 0x013d 24 0 6 # ivt\_tab.rel  0x0043 STRTAB 0x0155 77 0 7 # .shstrtab  #ivt\_tab  0x0000: 00 00 00 00 01 00 2c 00  #ivt\_tab.text  0x0000: 0c 6c 00 01 00 80 2a 00 64 80 2a 00  0x000c: 20 6c 00 30 00 20 64 20 80 02 ff 8c  0x0018: 80 2a 00 00 09 00 3c 00 29 00 64 00  0x0024: 00 00 80 2a 00 0c 00 00 8c 80 00 ff  0x0030: 00 71 00 34 00 38 00 0c 04  #.symtab  # name value type bind shndx ndx  0x0000 0x0000 NOTYPE LOCAL UND 0 #  0x0001 0x0000 SECTION LOCAL 1 1 # ivt\_tab  0x0009 0x0000 SECTION LOCAL 2 2 # ivt\_tab.text  0x0016 0x0038 NOTYPE LOCAL 2 3 # quit  0x001b 0x0029 NOTYPE LOCAL 2 4 # exit  0x0020 0x0000 NOTYPE LOCAL 2 5 # \_invalid\_instruction  0x0035 0x0001 NOTYPE LOCAL 2 6 # \_timer  0x003c 0xff02 NOTYPE LOCAL ABS 7 # data\_out  0x0045 0x0000 NOTYPE GLOBAL UND 8 # \_start  0x004c 0x002c NOTYPE LOCAL 2 9 # \_terminal  0x0056 0x002a NOTYPE LOCAL 2 10 # cnt  0x005a 0xff00 NOTYPE LOCAL ABS 11 # data\_in  #.strtab  0x0000: ivt\_tabivt\_tab.textqui  0x0019: texit\_invalid\_instructi  0x0032: on\_timerdata\_out\_start  0x004b: \_terminalcntdata\_in  #ivt\_tab.text.rel  #offset type stndx ndx  0x0020 ERT\_16 2 0  0x0006 ERT\_16 2 1  0x000a ERT\_16 2 2  0x0019 ERT\_16 2 3  0x0027 ERT\_16 2 4  0x0035 ERT\_16 2 5  #ivt\_tab.rel  #offset type stndx ndx  0x0002 ERT\_16 2 0  0x0004 ERT\_16 2 1  0x0006 ERT\_16 2 2  0x0000 ERT\_16 8 3  #.shstrtab  0x0000: ivt\_tabivt\_tab.text.sy  0x0019: mtab.strtabivt\_tab.text  0x0032: .relivt\_tab.rel.shstrta  0x004b: b |

# **Uvezivanje više fajlova**

|  |  |
| --- | --- |
| Ulazni fajl text0.s | |
| .extern data\_out, addr3  .global addr2  .section text  out2:  mov $'2', data\_out  jmp \*addr3(%pc)  .section addrsec  addr2:  .word out2  .end | #./tests/linking\_multiple\_files\_rel\_pc/text0.o  #sh\_offs sh\_entries sym\_tab\_ndx str\_tab\_ndx sh\_str\_ndx  0x00d2 8 3 4 7  #sections  # name type offs size link ndx  0x0000 UND 0x0016 0 0 0 #  0x0001 PROGBITS 0x0016 11 5 1 # text  0x0006 PROGBITS 0x0021 2 6 2 # addrsec  0x000e SYMTAB 0x0023 56 0 3 # .symtab  0x0016 STRTAB 0x005b 40 0 4 # .strtab  0x001e REL 0x0083 12 0 5 # text.rel  0x0027 REL 0x008f 6 0 6 # addrsec.rel  0x0033 STRTAB 0x0095 61 0 7 # .shstrtab  #text  0x0000: 64 00 32 00 80 00 00 2c 6e fe ff  #addrsec  0x0000: 00 00  #.symtab  # name value type bind shndx ndx  0x0000 0x0000 NOTYPE LOCAL UND 0 #  0x0001 0x0000 SECTION LOCAL 1 1 # text  0x0006 0x0000 SECTION LOCAL 2 2 # addrsec  0x000e 0x0000 NOTYPE GLOBAL 2 3 # addr2  0x0014 0x0000 NOTYPE GLOBAL UND 4 # addr3  0x001a 0x0000 NOTYPE LOCAL 1 5 # out2  0x001f 0x0000 NOTYPE GLOBAL UND 6 # data\_out  #.strtab  0x0000: textaddrsecaddr2addr3  0x0019: out2data\_out  #text.rel  #offset type stndx ndx  0x0005 ERT\_16 6 0  0x0009 ERT\_PC16 4 1  #addrsec.rel  #offset type stndx ndx  0x0000 ERT\_16 1 0  #.shstrtab  0x0000: textaddrsec.symtab.st  0x0019: rtabtext.reladdrsec.rel  0x0032: .shstrtab |

|  |  |
| --- | --- |
| Ulazni fajl text1.s | |
| .global \_start  .extern data\_out, addr2  .section text  addr0:  .word out0  out0:  mov $'0', data\_out  jmp \*addr1(%pc)  \_start:  mov $'s', data\_out  jmp \*addr0(%pc)  out1:  mov $'1', data\_out  jmp \*addr2(%pc)  addr1:  .word out1  .end | #./tests/linking\_multiple\_files\_rel\_pc/text1.o  #sh\_offs sh\_entries sym\_tab\_ndx str\_tab\_ndx sh\_str\_ndx  0x0102 6 2 3 5  #sections  # name type offs size link ndx  0x0000 UND 0x0016 0 0 0 #  0x0001 PROGBITS 0x0016 37 4 1 # text  0x0006 SYMTAB 0x003b 72 0 2 # .symtab  0x000e STRTAB 0x0083 50 0 3 # .strtab  0x0016 REL 0x00b5 36 0 4 # text.rel  0x001f STRTAB 0x00d9 41 0 5 # .shstrtab  #text  0x0000: 02 00 64 00 30 00 80 00 00 2c 6e 16  0x000c: 00 64 00 73 00 80 00 00 2c 6e e8 ff  0x0018: 64 00 31 00 80 00 00 2c 6e fe ff 18  0x0024: 00  #.symtab  # name value type bind shndx ndx  0x0000 0x0000 NOTYPE LOCAL UND 0 #  0x0001 0x0000 SECTION LOCAL 1 1 # text  0x0006 0x0002 NOTYPE LOCAL 1 2 # out0  0x000b 0x0000 NOTYPE LOCAL 1 3 # addr0  0x0011 0x000d NOTYPE GLOBAL 1 4 # \_start  0x0018 0x0000 NOTYPE GLOBAL UND 5 # addr2  0x001e 0x0000 NOTYPE GLOBAL UND 6 # data\_out  0x0027 0x0018 NOTYPE LOCAL 1 7 # out1  0x002c 0x0023 NOTYPE LOCAL 1 8 # addr1  #.strtab  0x0000: textout0addr0\_starta  0x0019: ddr2data\_outout1addr1  #text.rel  #offset type stndx ndx  0x0000 ERT\_16 1 0  0x0023 ERT\_16 1 1  0x0021 ERT\_PC16 5 2  0x0007 ERT\_16 6 3  0x0012 ERT\_16 6 4  0x001d ERT\_16 6 5  #.shstrtab  0x0000: text.symtab.strtabtex  0x0019: t.rel.shstrtab |

|  |  |
| --- | --- |
| Ulazni fajl text2.s | |
| .extern data\_out  .global addr3  .section text  out3:  mov $'3', data\_out  halt  .section addrsec  addr3:  .word out3  .end | #./tests/linking\_multiple\_files\_rel\_pc/text2.o  #sh\_offs sh\_entries sym\_tab\_ndx str\_tab\_ndx sh\_str\_ndx  0x00bb 8 3 4 7  #sections  # name type offs size link ndx  0x0000 UND 0x0016 0 0 0 #  0x0001 PROGBITS 0x0016 8 5 1 # text  0x0006 PROGBITS 0x001e 2 6 2 # addrsec  0x000e SYMTAB 0x0020 48 0 3 # .symtab  0x0016 STRTAB 0x0050 34 0 4 # .strtab  0x001e REL 0x0072 6 0 5 # text.rel  0x0027 REL 0x0078 6 0 6 # addrsec.rel  0x0033 STRTAB 0x007e 61 0 7 # .shstrtab  #text  0x0000: 64 00 33 00 80 00 00 04  #addrsec  0x0000: 00 00  #.symtab  # name value type bind shndx ndx  0x0000 0x0000 NOTYPE LOCAL UND 0 #  0x0001 0x0000 SECTION LOCAL 1 1 # text  0x0006 0x0000 SECTION LOCAL 2 2 # addrsec  0x000e 0x0000 NOTYPE GLOBAL 2 3 # addr3  0x0014 0x0000 NOTYPE LOCAL 1 4 # out3  0x0019 0x0000 NOTYPE GLOBAL UND 5 # data\_out  #.strtab  0x0000: textaddrsecaddr3out3  0x0019: data\_out  #text.rel  #offset type stndx ndx  0x0005 ERT\_16 5 0  #addrsec.rel  #offset type stndx ndx  0x0000 ERT\_16 1 0  #.shstrtab  0x0000: textaddrsec.symtab.st  0x0019: rtabtext.reladdrsec.rel  0x0032: .shstrtab |

|  |  |
| --- | --- |
| Ulazni fajl ivt\_tab.s | |
| .extern \_start  .global data\_in, data\_out  .section ivt\_tab  .word \_start  .word \_invalid\_instruction  .word \_timer  .word \_terminal  .section ivt\_tab.text  .equ data\_in, 0xFF00  .equ data\_out, 0xFF02  \_invalid\_instruction:  iret  \_timer:  iret  \_terminal:  iret  .end | #./tests/linking\_multiple\_files\_rel\_pc/ivt\_tab.o  #sh\_offs sh\_entries sym\_tab\_ndx str\_tab\_ndx sh\_str\_ndx  0x0111 7 3 4 6  #sections  # name type offs size link ndx  0x0000 UND 0x0016 0 0 0 #  0x0001 PROGBITS 0x0016 8 5 1 # ivt\_tab  0x0009 PROGBITS 0x001e 3 0 2 # ivt\_tab.text  0x0016 SYMTAB 0x0021 72 0 3 # .symtab  0x001e STRTAB 0x0069 84 0 4 # .strtab  0x0026 REL 0x00bd 24 0 5 # ivt\_tab.rel  0x0032 STRTAB 0x00d5 60 0 6 # .shstrtab  #ivt\_tab  0x0000: 00 00 00 00 01 00 02 00  #ivt\_tab.text  0x0000: 0c 0c 0c  #.symtab  # name value type bind shndx ndx  0x0000 0x0000 NOTYPE LOCAL UND 0 #  0x0001 0x0000 SECTION LOCAL 1 1 # ivt\_tab  0x0009 0x0000 SECTION LOCAL 2 2 # ivt\_tab.text  0x0016 0xff00 NOTYPE GLOBAL ABS 3 # data\_in  0x001e 0x0002 NOTYPE LOCAL 2 4 # \_terminal  0x0028 0xff02 NOTYPE GLOBAL ABS 5 # data\_out  0x0031 0x0001 NOTYPE LOCAL 2 6 # \_timer  0x0038 0x0000 NOTYPE GLOBAL UND 7 # \_start  0x003f 0x0000 NOTYPE LOCAL 2 8 # \_invalid\_instruction  #.strtab  0x0000: ivt\_tabivt\_tab.textdat  0x0019: a\_in\_terminaldata\_out\_  0x0032: timer\_start\_invalid\_ins  0x004b: truction  #ivt\_tab.rel  #offset type stndx ndx  0x0002 ERT\_16 2 0  0x0004 ERT\_16 2 1  0x0006 ERT\_16 2 2  0x0000 ERT\_16 7 3  #.shstrtab  0x0000: ivt\_tabivt\_tab.text.sy  0x0019: mtab.strtabivt\_tab.rel  0x0032: .shstrtab |

Miloš Ćurčić 379/17