

Ассемблер RISC-V

Ассемблер

- **Язык ассемблера (assembly language)** — система обозначений, используемая для представления в человеко-читаемой форме программ, записанных в машинном коде
- **Ассемблер (assembler)** — программа, транслирующая код на языке ассемблера в машинный код

Машинный код (hex)	Код на языке ассемблера RISC-V	Код на языке ассемблера RISC-V с псевдо-инструкциями
00b00293	<code>addi x5, x0, 11</code>	<code>li x5, 11</code>
00928393	<code>addi x7, x5, 9</code>	<code>addi x7, x5, 9</code>
00000013	<code>addi x0, x0, 0</code>	<code>nop</code>
00038593	<code>addi x11, x7, 0</code>	<code>mv x11, x7</code>

GNU binutils

- **as** — ассемблер
 - Превращает код на языке ассемблера в объектный файл
- **ld** — линкер
 - Соединяет объектные файлы в бинарный файл в формате ELF
- **objcopy** — манипулирует секциями бинарных и объектных файлов
- **objdump** — отображает код из бинарных и объектных файлов
- **readelf** — отображает информацию об ELF-файле

as

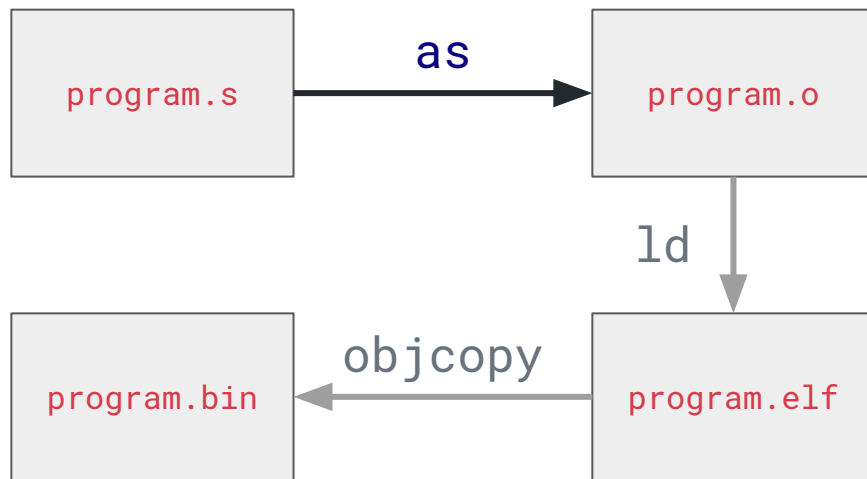
Файл addi.s

```
.text
.globl _start
.globl _finish

_start:
    li    x5, 11
    addi  x7, x5, 9
    nop
    mv    x11, x7

_finish:
    nop
```

as -c addi.s -o addi.o



ld

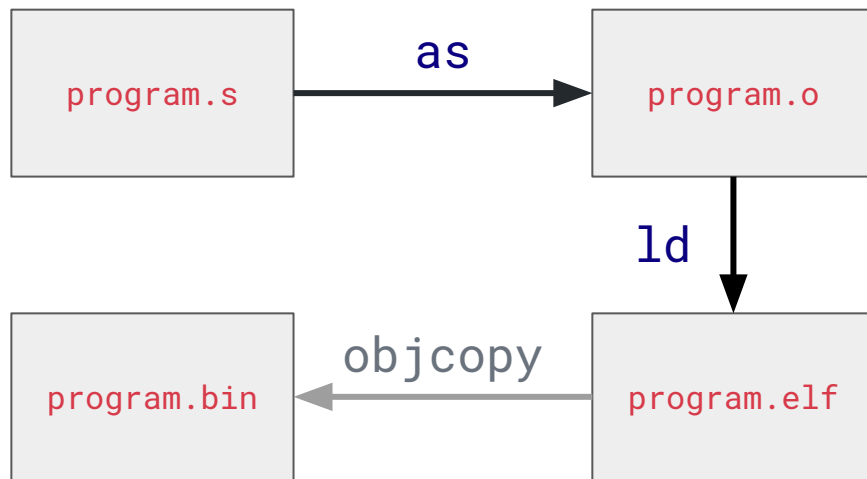
Файл addi.s

```
.text
.globl _start
.globl _finish

_start:
    li    x5, 11
    addi  x7, x5, 9
    nop
    mv    x11, x7

_finish:
    nop
```

ld addi.o -o addi.elf



objcopy

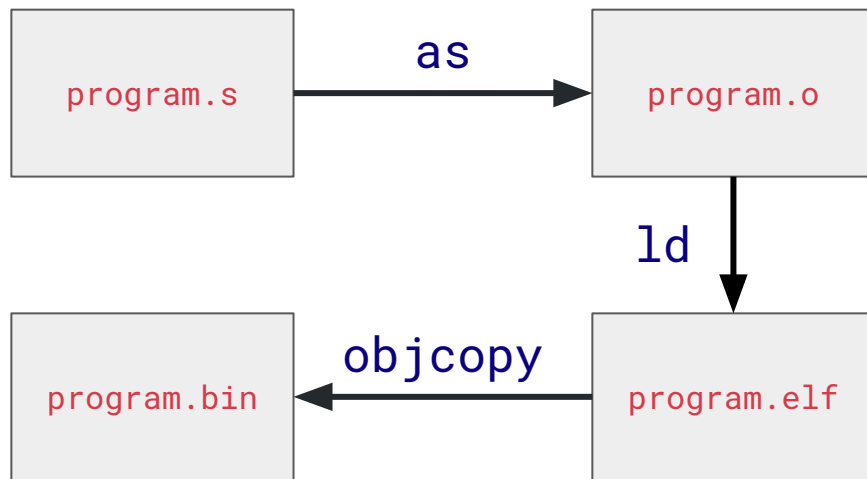
Файл addi.s

```
.text
.globl _start
.globl _finish

_start:
    li    x5, 11
    addi  x7, x5, 9
    nop
    mv    x11, x7

_finish:
    nop
```

objcopy -O binary addi.elf addi.bin



hexdump

Файл addi.s

```
.text
.globl _start
.globl _finish

_start:
    li    x5, 11
    addi  x7, x5, 9
    nop
    mv    x11, x7

_finish:
    nop
```

```
hexdump -v -e '"%08x\n"' addi.bin
```

00b00293

00928393

00000013

00038593

00000013

readelf и objdump

```
riscv64-linux-gnu-objdump -M numeric -d addi.elf
```

```
a.out:      file format elf64-littleriscv
```

Disassembly of section .text:

```
00000000000010078 <_start>:
```

```
10078:      00b00293      li      x5,11
1007c:      00928393      addi    x7,x5,9
10080:      00000013      nop
10084:      00038593      mv      x11,x7
```

```
00000000000010088 <_finish>:
```

```
10088:      00000013      nop
```

```
riscv64-linux-gnu-readelf -a addi.elf
```

ELF Header:

```
.....
Class:                                ELF64
.....
Machine:                              RISC-V
.....
Entry point address:                  0x10078
```

Section Headers:

[Nr]	Name	Type	Address	Offset
	Size	EntSize	Flags Link Info Align	
[1]	.text	PROGBITS	0000000010078	00000078
	0000000000000014	0000000000000000	AX 0 0	4

Symbol table '.symtab' contains 8 entries:

Num:	Value	Size	Type	Bind	Vis	Ndx	Name
3:	00000000000010078	0	NOTYPE	GLOBAL	DEFAULT	1	_start
4:	00000000000010088	0	NOTYPE	GLOBAL	DEFAULT	1	_finish

GitHub

github.com/viktor-prutyanov/drec-fpga-intro