

# Compiler Construction

## MIPS

University of Freiburg



UNI  
FREIBURG

Konrad Anton, Manuel Geffken, *Matthias Keil*

University of Freiburg

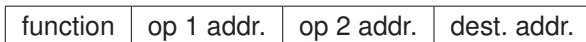
15. Dezember 2016

- **Microprocessor without interlocked pipeline stage**
- Simple but serious RISC architecture
- Used in workstations (SGI), PDAs, routers, . . .
- Simulator `SPIM` available
  - easier to debug assembly code than on host
  - supports MIPS32 architecture
- *Today:* Just a taste – more in exercise 6



- Load-Store architecture
- Three-address instructions
- Word size: 32 bits (in MIPS32)
- 32 registers
- one addressing mode: register+immediate

## ■ 3-address instruction format (MIPS, ARM)

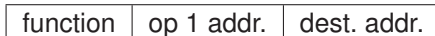


## ■ Other instruction formats

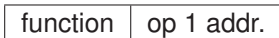
### ■ 4-address instruction format



### ■ 2-address instruction format (used by the Thumb instruction set of ARM, x86)



### ■ 1-address instruction format (used in MU0 and some 8-bit microcontrollers such as MC6811)



## ■ Arithmetic

```
add    rd,rs,rt    # [rd] := [rs] + [rt]
addi   rt,rs,imm   # [rt] := [rs] + imm
```

## ■ Comparison

```
slt    rd,rs,rt    # [rd] := 1 if [rs]<[rt]
                               # [rd] := 0 otherwise
sltu   rd,rs,rt    # (same in unsigned)
```

## ■ Jump and Branch

```
beq    rs,rt,L     # ->label if [rs]=[rt]
j      L           # unconditional jump
jal    L           # saving next addr in $ra
```

Registername	Number	Usage
\$zero	0	constant 0
\$at	1	assembler temporary (reserved)
\$v0– \$v1	2,3	function result
\$a0– \$a3	4–7	function arguments
\$t0– \$t9	8–15,24,25	temporary (caller-save)
\$s0 – \$s7	16–23	saved temporary (callee-save)
\$k0, \$k1	26,27	OS kernel (reserved)
\$gp	28	pointer to global area
\$sp	29	stack pointer
\$fp	30	frame pointer
\$ra	31	return address