

Projekt: Mikrorechner

Ideensammlung 1

WS 2013/2014

1 Ideensammlung - Instruction Set Architecture

1.1 General

- Wortbreite 32 bit
- Adressbreite 32 bit
- 3-Operanden (1 Ziel-, 2 Quell-Operanden)
- 13 General-Purpose Register
- \$0-Register: Null
- \$14-Register: Stack Pointer
- \$15-Register: Return Address
- 4 Statusflags
 - Z Zero - 1 wenn Ergebnis == 0
 - N Negative - 1 wenn MSB == 1
 - C Carry - 1 wenn ein (signed) Over- oder Underflow stattgefunden hat
 - V Overflow - 1 wenn ein (unsigned) Over- oder Underflow stattgefunden hat
- Big Endian
- Befehle können Statusflags setzen
- 16-Bit immediates

1.2 Arithmetik / Logik

- ADD, ADC
- SUB, SBC, RSB, RSC
- MUL, DIV
- LSL, ADR, LSR, ROR
- AND, ORR, XOR, NOT
- SWP (Wenn Felix ganz lieb fragt evtl soch)

1.3 Controlflow

- JMP
- Jxx (Flag dependent JMP)
- CALL
- RET (== JMP \$15)
- SWI (Software Interrupt)

1.4 Memory

- LD
- STR

1.5 Befehlsstruktur

$ALU - 00 : Opcode - (4bit) : SF - (1bit) : R_{dest} - (4bit) : R_{s1} - (4bit) : OP2 - (17bit)$

$MEM - 10 : LD/ST - (1bit) : R_{dest} - (4bit) : OP2 - (25bit)$

$Jxx - 110 : Condition - (4bit) : OP2 - (25bit)$

$JMP - 1110000 : OP2 - (25bit)$

$CALL - 1110001 : OP2 - (25bit)$

$ADR - 010 : R_{dest} - (4bit) : OP2 - (25bit)$

$SWI - 0110010 : OP2 - (25bit)$

$PUSH - 0110100 : 0* : R_{src} - (4bit)$

$POP - 0110101 : 0* : R_{dest} - (4bit)$