microRISC

Andrew Schomber

December 10, 2024

Table of Contents

Architecture

Registers

General Purpose Registers

The general purpose registers are used to store data and perform arithmetic operations. They are named R0-R29 and are 32 bits wide.

Register	Binary Representation	Register	Binary Representation
R0	00000	R1	00001
R2	00010	R3	00011
R4	00100	R5	00101
R6	00110	R7	00111
R8	01000	R9	01001
R10	01010	R11	01011
R12	01100	R13	01101
R14	01110	R15	01111
R16	10000	R17	10001
R18	10010	R19	10011
R20	10100	R21	10101
R22	10110	R23	10111
R24	11000	R25	11001
R26	11010	R27	11011
R28	11100	R29	11101

Program Counter (PC)

The program counter register keeps track of the current instruction being executed. It is automatically incremented after each instruction is executed. It can not be directly accessed or modified by the programmer.

CMP Register

The CMP register is used to store the result of a comparison operation. It is set by the CMP instruction, which subtracts the second operand from the first operand and sets the CMP register based on the result. Reference it using the CMP keyword (its an operation and a register).

Register	Binary Representation
CMP	11110

Stack Pointer (SP)

The stack pointer register is used to keep track of the top of the stack. Similarly to ARM, you must manually manage the stack pointer. Reference it using the SP keyword.

Register	Binary Representation
SP	11111

Opcode Layout and Control Signals

Every opcode is 6 bits wide. Each bit will be a dedicated control signal:

Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5
Control Signal 0	Control Signal 1	Control Signal 2	Control Signal 3	Control Signal 4	Control Signal 5

There are also other control signals that are not part of the opcode. These are:

Memory Layout

This is the content of Subsection 1.3.

Instruction Set

Arithmetic and Logical

The following arithmetic and logical operations are supported:

Syntax	Opcode	Rd	Rn	Rm
ADD Rd, Rn, Rm	Item2.1	5 bits	5 bits	5 bits
SUB Rd, Rn, Rm	Item2.2	5 bits	5 bits	5 bits
MUL Rd, Rn, Rm	Item2.3	5 bits	5 bits	5 bits
SDIV Rd, Rn, Rm	Item2.4	5 bits	5 bits	5 bits
UDIV Rd, Rn, Rm	Item2.5	5 bits	5 bits	5 bits
LDR Rd, [Rn, Rm]	Item2.5	5 bits	5 bits	5 bits
STR Rd, [Rn, Rm]	Item2.6	5 bits	5 bits	5 bits
AND Rd, Rn, Rm	Item2.7	5 bits	5 bits	5 bits
ORR Rd, Rn, Rm	Item2.8	5 bits	5 bits	5 bits
XOR Rd, Rn, Rm	Item2.9	5 bits	5 bits	5 bits
LSL Rd, Rn, Rm	Item2.10	5 bits	5 bits	5 bits
LSR Rd, Rn, Rm	Item2.11	5 bits	5 bits	5 bits
ASR Rd, Rn, Rm	Item2.12	5 bits	5 bits	5 bits

Branching

The following branching instructions are supported:

Syntax	Opcode	Label	Unused
B Label	Item2.27	20 bits	5 bits
BL Label	Item2.28	20 bits	5 bits
BEQ Label	Item2.29	20 bits	5 bits
BNE Label	Item2.30	20 bits	5 bits
BGT Label	Item2.31	20 bits	5 bits
BLT Label	Item2.32	20 bits	5 bits
BGE Label	Item2.33	20 bits	5 bits
BLE Label	Item2.34	20 bits	5 bits

Syntax	Opcode	Rd	Rn	Unused
CMP Rd, Rn	Item2.26	5 bits	5 bits	16 bits

Syntax	Opcode	Rd	Label	Unused
CBZ Rd, Label	Item2.35	5 bits	20 bits	1 bit
CBNZ Rd, Label	Item2.36	5 bits	20 bits	1 bit

Syntax	Opcode	Unused
RET	Item2.37	26 bits

Other

These are other instructions that don’t fit under the existing categories:

Syntax	Opcode	Rd	Rn	Unused
MOV Rd, Rn	Item2.26	5 bits	5 bits	16 bits

Syntax	Opcode	Rd	Imm	
MOV Rd, Imm	Item2.27	5 bits	21 bits	

Syntax	Opcode	Rd	Label	Unused
ADR Rd, Label	Item2.28	5 bits	20 bits	1 bit

Syntax	Opcode	Rd	Unused
NEG Rd	Item2.29	5 bits	21 bits

Syntax	Opcode	Unused
NOP	000000	26 bits