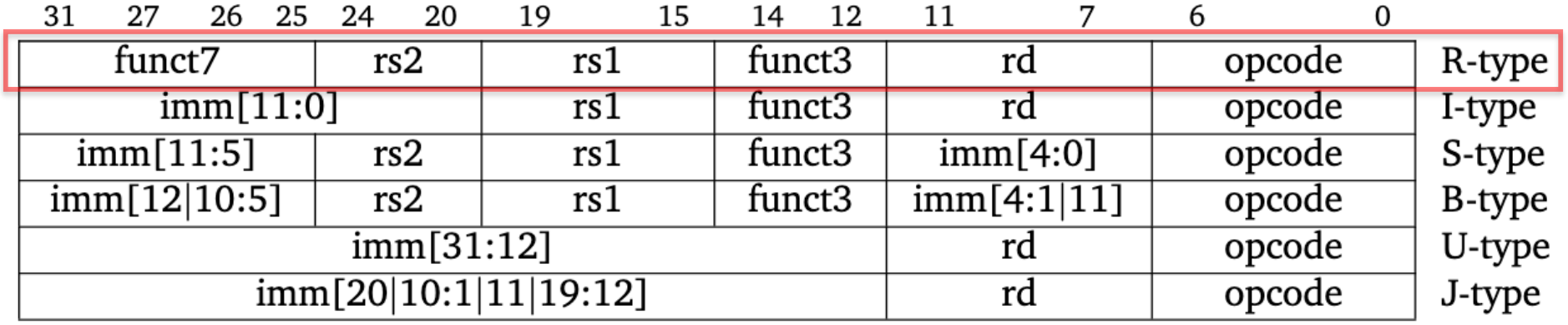
RISC-V ISA: riscv-uconn

- Total memory size limited to 16,384 addresses
- Each memory address stores a word (4 bytes)
- Instructions reside in addresses 0—255, followed by data in the remaining addresses
- Program counter initially points to address 0
- RISC-V registers: 32 x 32 register file
  - x0 is hardwired to 0, but it will be set to 1 to trigger program termination

Register Number	ABI Name	Description
x0	zero	hardwired 0x00000000
x1-4	ra, sp, gp, tp	return address, stack pointer, global pointer, thread pointer
x5-7	t0-2	temporary registers
x8-9	s0-1	saved registers
x10-11	a0-1	function arguments / return values
x12-17	a2-7	function arguments
x18-27	s2-11	saved registers
x28-31	t3-6	temporary registers

UCONN

riscv-uconn: RISC-V Integer Instruction Set

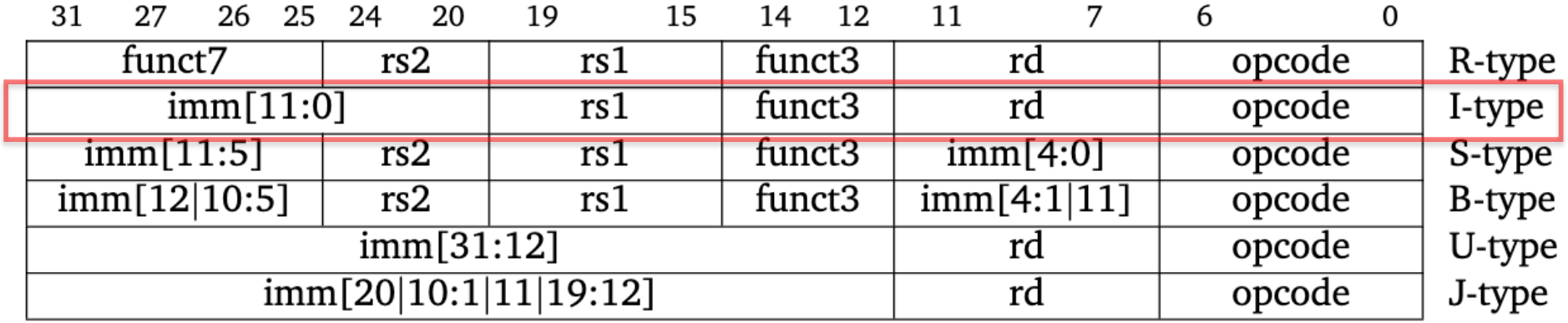


31 27	26 25	24 20	19 15	14 12	11 7	6 0	
funct7		rs2	rs1	funct3	rd	opcode	R-type
imm[11:0]			rs1	funct3	rd	opcode	I-type
imm[11:5]		rs2	rs1	funct3	imm[4:0]	opcode	S-type
imm[12|10:5]		rs2	rs1	funct3	imm[4:1|11]	opcode	B-type
imm[31:12]					rd	opcode	U-type
imm[20|10:1|11|19:12]					rd	opcode	J-type

Inst.	Opcode	funct3	funct7	Description	Name
add	0110011	0x0	0x00	rd=rs1+rs2	ADD
sub	0110011	0x0	0x20	rd=rs1-rs2	SUB
and	0110011	0x7	0x00	rd=rs1&rs2	AND
or	0110011	0x6	0x00	rd=rs1|rs2	OR
xor	0110011	0x4	0x00	rd=rs1^rs2	XOR
slt	0110011	0x2	0x00	rd = (rs1 < rs2)?1:0	Set Less Than
sll	0110011	0x1	0x00	rd=rs1<<rs2	Shift Left Logical
srl	0110011	0x5	0x00	rd=rs1>>rs2	Shift Right Logical

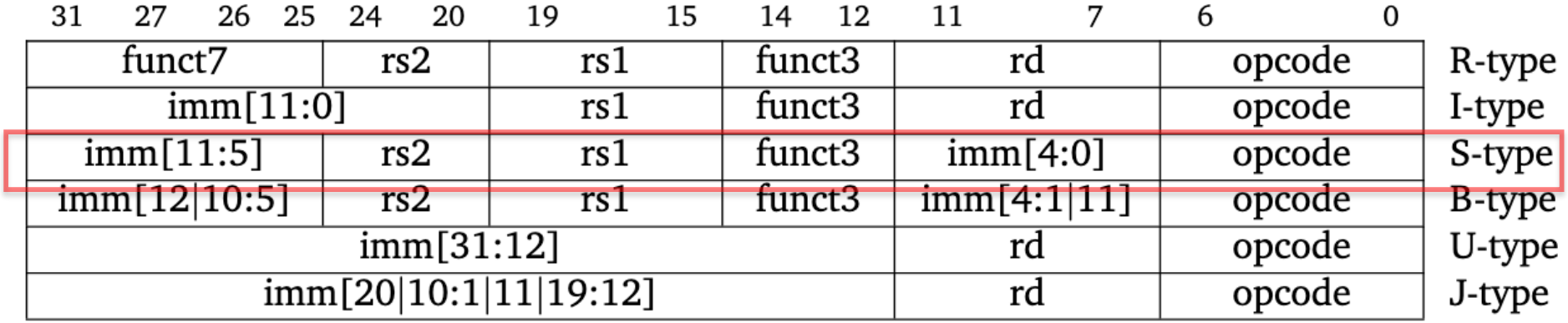
UCONN

riscv-uconn: RISC-V Integer Instruction Set



Inst.	Opcode	funct3	funct7	Description	Name
addi	0010011	0x0		rd=rs1+imm	ADDI
xori	0010011	0x4		rd=rs1^rs2	XORI
ori	0010011	0x6		rd=rs1|imm	ORI
andi	0010011	0x7		rd=rs1&imm	ANDI
slti	0010011	0x2		rd = (rs1 < imm)?1:0	Set Less Than Imm
slli	0010011	0x1	Imm[5:11]=0x00	rd = rs1 << imm[0:4]	Shift L Logical Imm
srli	0010011	0x5	Imm[5:11]=0x00	rd = rs1 >> imm[0:4]	Shift R Logical Imm
lw	0000011	0x2		rd =M[rs1+imm][0:31]	Load Word
jalr	1100111	0x0		rd=pc+4;pc=rs1+imm	Jump & Link Reg

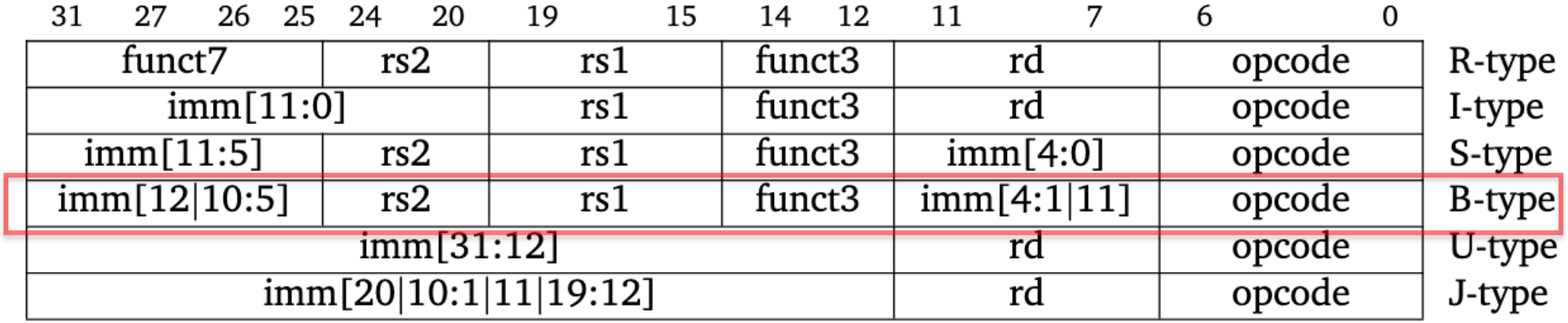
riscv-uconn: RISC-V Integer Instruction Set



31 27	26 25	24 20	19 15	14 12	11 7	6 0	
funct7		rs2	rs1	funct3	rd	opcode	R-type
imm[11:0]			rs1	funct3	rd	opcode	I-type
imm[11:5]		rs2	rs1	funct3	imm[4:0]	opcode	S-type
imm[12|10:5]		rs2	rs1	funct3	imm[4:1|11]	opcode	B-type
imm[31:12]					rd	opcode	U-type
imm[20|10:1|11|19:12]					rd	opcode	J-type

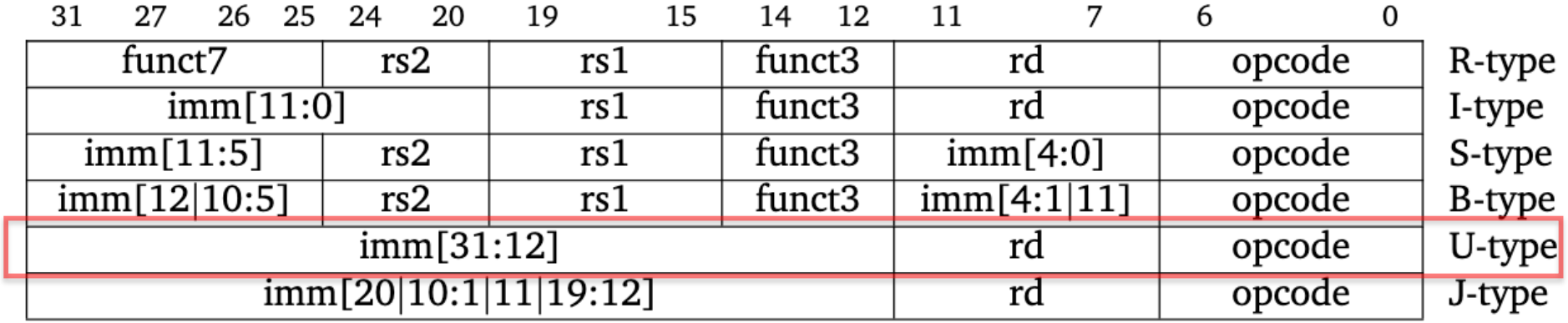
Inst.	Opcode	funct3	funct7	Description	Name
sw	0100011	0x2		M[rs1+imm][0:31] = rs2[0:31]	Store Word

riscv-uconn: RISC-V Integer Instruction Set



Inst.	Opcode	funct3	funct7	Description	Name
beq	1100011	0x0		if (rs1 == rs2) pc += imm	Branch ==
bne	1100011	0x1		if (rs1 != rs2) pc += imm	Branch !=
blt	1100011	0x4		if (rs1 < rs2) pc += imm	Branch <
bge	1100011	0x5		If (rs1 >= rs2) pc += imm	Branch ≥

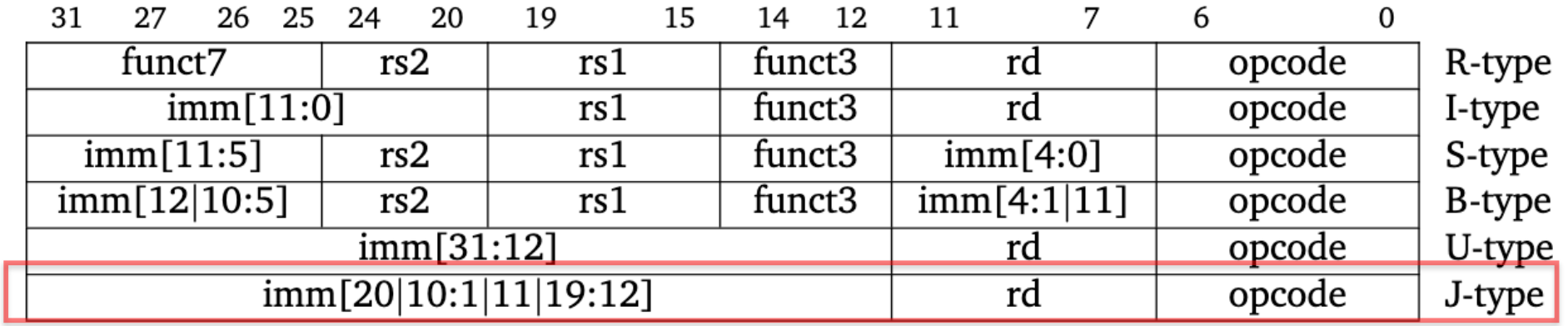
riscv-uconn: RISC-V Integer Instruction Set



31 27	26 25	24 20	19 15	14 12	11 7	6 0	
funct7		rs2	rs1	funct3	rd	opcode	R-type
imm[11:0]			rs1	funct3	rd	opcode	I-type
imm[11:5]		rs2	rs1	funct3	imm[4:0]	opcode	S-type
imm[12|10:5]		rs2	rs1	funct3	imm[4:1|11]	opcode	B-type
imm[31:12]					rd	opcode	U-type
imm[20|10:1|11|19:12]					rd	opcode	J-type

Inst.	Opcode	funct3	funct7	Description	Name
lui	0110111			rd = imm << 12	Load Upper Imm

riscv-uconn: RISC-V Integer Instruction Set



31 27	26 25	24 20	19 15	14 12	11 7	6 0	
funct7		rs2	rs1	funct3	rd	opcode	R-type
imm[11:0]			rs1	funct3	rd	opcode	I-type
imm[11:5]		rs2	rs1	funct3	imm[4:0]	opcode	S-type
imm[12|10:5]		rs2	rs1	funct3	imm[4:1|11]	opcode	B-type
imm[31:12]					rd	opcode	U-type
imm[20|10:1|11|19:12]					rd	opcode	J-type

Inst.	Opcode	funct3	funct7	Description	Name
jal	1101111			rd = pc+4; pc += imm	Jump and Link