ISA 24bits

Instruction Formats

R (register):

unused	rs2	rs1	funct3	rd	opcode
3b	5b	5b	3b	5b	3b

I (immediate):

imm[7:0]		rs1	funct3	rd	opcode
8b		5b	3b	5b	3b

V (void):

imm[7:5]	rs2	rs1	funct3	imm[4:0]	opcode
3b	5b	5b	3b	5b	3b

J (jump):

imm[12:0]			funct3	rd	opcode
13b			3b	5b	3b

L (load):

imm[15:0]				rd	opcode
16b				5b	3b

LU (load upper):

imm[31:16]				rd	opcode
16b				5b	3b

Instructions

load / store — opcode 000

```
I [000 000] lw – load word
I [000 001] lh – load half
I [000 010] lb – load byte
V [000 011] sw – store word
V [000 100] sh – store half
V [000 101] sb – store byte

[u] lbu – load byte unsigned
[u] lhu – load half unsigned
```

arithmetic — opcode 001

```
R [001 000] add – addition
I [001 001] addi – add immediate
R [001 010] sub – subtraction
R [001 011] div – division
R [001 100] mul – multiplication
R [001 101] mod – remainder
I [001 110] modi – remainder immediate

[?] auipc – add upper immediate to PC
```

logical — opcode 010

```
R [010 000] and – logical and
R [010 001] or – logical or
R [010 010] xor – logical xor
I [010 011] andi – logical and immediate
I [010 100] ori – logical or immediate
I [010 101] xori – logical xor immediate
R [010 110] slt – set less than
I [010 111] slti – set less than immediate

[u] sltu – set less than unsigned
[u] sltiu – set less than immediate unsigned
```

shift — opcode 011

```
I [011 000] slli – shift left logical immediate
I [011 001] srli – shift right logical immediate
I [011 010] srai – shift right arithmetic immediate
R [011 011] sll – shift left logical
R [011 100] srl – shift right logical
R [011 101] sra – shift right arithmetic
```

branch / jump —opcode 100

```
V [100 000] beq – branch if equal
V [100 001] bne – branch if not equal
V [100 010] bge – branch if greater or equal
V [100 011] blt – branch if less than
J [100 100] jal – jump and link
I [100 101] jalr – jump and link register

[u] bltu – branch less than unsigned
[u] bgeu – branch greater or equal unsigned
```

system — opcode 101

```
I [101 000] halt – halt the program execution
```

load immediate — opcode 11x

```
L  [110] li – load immediate
```

LU [111] lui – load upper immediate