

RISC-V Instruction Overview					
op-code	funct3	funct7	Instruction	Type	Description
0000011	000		LB	I - Type	Load Byte to rd register (signed extended)
0000011	001		LH	I - Type	Load 2 Bytes to rd register (signed extended)
0000011	010		LW	I - Type	Load Word to rd register (signed extended)
0000011	100		LBU	I - Type	Load Byte to rd register (zero extended)
0000011	101		LHU	I - Type	Load 2 Bytes to rd register (zero extended)
0001111	000		FENCE		Fence on all memory and I/O
0010011	000		ADDI	I - Type	ADD immediate value and value of rs1 and put result to rd register
0010011	010		SLTI	I - Type	if immediate value is less than value of rs1 put 1 to rd register otherwise put 0
0010011	011		SLTIU	I - Type	if immediate value is less than value of rs1 put 1 to rd register otherwise put 0 (unsigned)
0010011	100		XORI	I - Type	XOR immediate value and value of rs1 and put result to rd register
0010011	110		ORI	I - Type	OR immediate value and value of rs1 and put result to rd register
0010011	111		ANDI	I - Type	AND immediate value and value of rs1 and put result to rd register
0010011	001		SLLI	I - Type	Shift Left Logical with Immediate (shift rs1 value by immediate amount)
0010011	101		SRLI	I - Type	Shift Right Logical with Immediate (shift rs1 value by immediate amount)
0010011	101		SRAI	I - Type	Shift Right Arithmetic Immediate (shift rs1 value by immediate amount)
0010111			AUIPC	U -Type	Add upper immediate to PC and put to rd register
0100011	000		SB	S-Type	Store Byte rs2 reg value, base is in rs1 address and the offset taken from the immediate value
0100011	001		SH	S-Type	rs2 reg value, base is in rs1 address and the offset taken from the immediate value
0100011	010		SW	S-Type	rs2 reg value, base is in rs1 address and the offset taken from the immediate value
0110011	000		ADD	R-Type	ADD the value in rs1 and rs2 value and put it into rd
0110011	000		SUB	R-Type	SUBTRACT the value in rs1 and rs2 value and put it into rd
0110011	001		SLL	R-Type	Shift Left Logical (rs1 value by rs2 amount and put it to rd)
0110011	010		SLT	R-Type	Set less than (if rs1 value less than rs2 value then put 1 to the rd register else 0 to the rd register)
0110011	011		SLTU	R-Type	Set less than Unsigned (if rs1 value less than rs2 value then put 1 to the rd register else 0 to the rd register)
0110011	100		XOR	R-Type	XOR the value in rs1 and rs2 value and put it into rd
0110011	101		SRL	R-Type	Shift Right Logical (rs1 value by rs2 amount and put it to rd)
0110011	101		SRA	R-Type	Shift Right Arithmetic (rs1 value by rs2 amount and put it to rd)
0110011	110		OR	R-Type	OR the value in rs1 and rs2 value and put it into rd
0110011	111		AND	R-Type	AND the value in rs1 and rs2 value and put it into rd
0110011	000	0111011	MUL	R-Type	Multiplication
0110011	001	0111011	MULH	R-Type	Returns upper 32-bits of signed x signed (use rs1 value and the rs2 value and put the answer to the rd register)
0110011	010	0111011	MULHSU	R-Type	Returns upper 32-bits of signed x unsigned (use rs1 value and the rs2 value and put the answer to the rd register)
0110011	011	0111011	MULHU	R-Type	Returns upper 32-bits of unsigned x unsigned (use rs1 value and the rs2 value and put the answer to the rd register)
0110011	100	0111011	DIV	R-Type	Signed Integer division (use rs1 value and the rs2 value and put the answer to the rd register)
0110011	101	0111011	REM	R-Type	Signed remainder of integer division (use rs1 value and the rs2 value and put the answer to the rd register)
0110011	111	0111011	REMU	R-Type	Unsigned remainder of interger division (use rs1 value and the rs2 value and put the answer to the rd register)
0110111			LUI	U -Type	Load Upper Immediate (puts the immediate value with 12 zeros at the end and put it into rd register)
1101111			JAL	J-Type	Jump and Link (Jumps to the address in rs1 and put the current PC to the rd register)
1100111	000		JALR	I - Type	Jump and Link register (Jumps to the address in rs1 and put the current PC to the link register)
1100011	000		BEQ	B-Type	Branch if equal (if values in rs1 and rs2 are equal jump to the offset)
1100011	001		BNE	B-Type	Branch if not equal (if values in rs1 and rs2 are not equal jump to the offset)
1100011	100		BLT	B-Type	Branch if lower than (if values in rs1 < rs2 jump to the offset)
1100011	101		BGE	B-Type	Branch if grater than (if values in rs1 > rs2 jump to the offset)
1100011	110		BLTU	B-Type	Branch if lower than, unsigned (if values in rs1 < rs2 jump to the offset)
1100011	111		BGEU	B-Type	Branch greater than or equa, unsigned (if values in rs1 > rs2 jump to the offset)
1110011	000		ECALL		used to make a service request to the execution environment
1110011	000		EBREAK		Brake to debugger