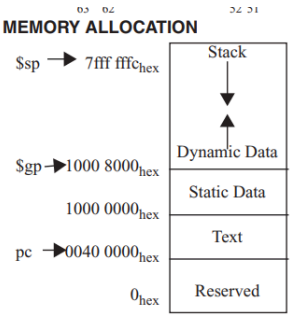
This is the ISA for a single cycle processor with 32-bit ALU operands, address bus size, 32 different registers each holding 32-bits, and is byte addressable. As a result, the program counter typically increments by 4 bytes, there are 4 bytes per word (both instruction and data), and as a result, the program counter progresses by 4 every instruction unless specified otherwise by a jump or branch instruction.

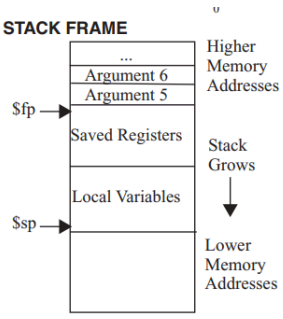
Core Instruction Set:

ALU controls			
ALU Control	ALU OP code	Funct	ALU Control Value
addi/lw/sw/jr	11	xxxxxx	010
beq/subi	10	xxxxxx	100
slti	01	xxxxxx	101
and	00	100100	000
or	00	100101	001
add	00	100000	010
nor	00	100111	011
sub	00	100010	100
slt	00	101010	101
sll	00	000000	110
slr	00	000010	111

REGISTER NAME, NUMBER, USE, CALL CONVENTION

NAME	NUMBER	USE	PRESERVED ACROSS A CALL?
$zero	0	The Constant Value 0	N.A.
$at	1	Assembler Temporary	No
$v0-$v1	2-3	Values for Function Results and Expression Evaluation	No
$a0-$a3	4-7	Arguments	No
$t0-$t7	8-15	Temporaries	No
$s0-$s7	16-23	Saved Temporaries	Yes
$t8-$t9	24-25	Temporaries	No
$k0-$k1	26-27	Reserved for OS Kernel	No
$gp	28	Global Pointer	Yes
$sp	29	Stack Pointer	Yes
$fp	30	Frame Pointer	Yes
$ra	31	Return Address	No





Basic Instruction Formats:

R type:

opcode	rs	rt	rd	shamt	funct
31-26	25-21	20-16	15-11	10-6	5-0

I type:

opcode	rs	rt	immediate
31-26	25-21	20-16	15-0

J type:

opcode	address
31-26	25-0

Operations and OPCodes	
Operation	Opcode
R-type instruction	000000
I: Load Word	000001
I: Store Word	000010
I: addi	000011
I: branch equal	000100
I: set if less than immediate	000101
J: jump	000110
J: jump and link	000111

Control Values									
Instruction	Reg Write	Reg Dest	ALU Src	Branch	Mem Write	Mem to Reg	Jump	Jump to Reg	ALU op
R-type	1	1	0	0	0	0	0	0	00
LW	1	0	1	0	0	1	0	0	11
SW	0	0	1	0	1	0	0	0	11
Addi	1	0	1	0	0	0	0	0	11
BEQ	0	0	0	1	0	0	0	0	10
STLI	1	0	1	0	0	0	0	0	01
jump	0	0	0	0	0	0	1	0	00
JAL	1	1	0	0	0	1	1	0	00
jr	0	0	1	0	0	0	1	1	11