The TAL Instruction Set

Machine Code	Assembly Language Format	Effect
`1000 11bb bbbt tttt iiii iiii iiii iiii`	`lw` R_t, I(R_b)	$R_t \leftarrow M[[R_b] + (I_{15})^{16} \Vert I_{15..0}]$
`1000 00bb bbbt tttt iiii iiii iiii iiii`	`lb` R_t, I(R_b)	$R_t \leftarrow m[[R_b] + (I_{15})^{16} \Vert I_{15..0}]_7)^{24} \Vert$ $m[[R_b] + (I_{15})^{16} \Vert I_{15..0}]$
`1001 00bb bbbt tttt iiii iiii iiii iiii`	`lbu` R_t, I(R_b)	$R_t \leftarrow 0^{24} \Vert m[[R_b] + (I_{15})^{16} \Vert I_{15..0}]$
`1010 11bb bbbt tttt iiii iiii iiii iiii`	`sw` R_t, I(R_b)	$R_t \rightarrow M[[R_b] + (I_{15})^{16} \Vert I_{15..0}]$
`1010 00bb bbbt tttt iiii iiii iiii iiii`	`sb` R_t, I(R_b)	$[R_t]_{7..0} \rightarrow m[[R_b] + (I_{15})^{16} \Vert I_{15..0}]$
`0000 00ss ssst tttt dddd d000 0010 0000`	`add` R_d, R_s, R_t	$R_d \leftarrow [R_s] + [R_t]$
`0000 00ss ssst tttt dddd d000 0010 0010`	`sub` R_d, R_s, R_t	$R_d \leftarrow [R_s] - [R_t]$
`0000 00ss ssst tttt 0000 0000 0001 1000`	`mult` R_s, R_t	HI $\Vert$ LO $\leftarrow [R_s] * [R_t]$
`0000 00ss ssst tttt 0000 0000 0001 1010`	`div` R_s, R_t	LO $\leftarrow [R_s]\text{div}[R_t]$; HI $\leftarrow [R_s]\text{mod}[R_t]$
`0000 00ss ssst tttt dddd d000 0010 0001`	`addu` R_d, R_s, R_t	$R_d \leftarrow [R_s] + [R_t]$, (overflow ignored)
`0000 00ss ssst tttt dddd d000 0010 0011`	`subu` R_d, R_s, R_t	$R_d \leftarrow [R_s] - [R_t]$, (overflow ignored)
`0000 00ss ssst tttt 0000 0000 0001 1001`	`multu` R_s, R_t	HI $\Vert$ LO $\leftarrow [R_s] * [R_t]$, (overflow ignored)
`0000 00ss ssst tttt 0000 0000 0001 1011`	`divu` R_s, R_t	LO $\leftarrow [R_s]\text{div}[R_t]$;HI $\leftarrow [R_s]\text{mod}[R_t]$ (overflow ignored)
`0000 00ss ssst tttt dddd d000 0010 0100`	`and` R_d, R_s, R_t	$R_d \leftarrow [R_s]$ **AND** $[R_t]$
`0000 00ss ssst tttt dddd d000 0010 0111`	`nor` R_d, R_s, R_t	$R_d \leftarrow [R_s]$ **NOR** $[R_t]$
`0000 00ss ssst tttt dddd d000 0010 0101`	`or` R_d, R_s, R_t	$R_d \leftarrow [R_s]$ **OR** $[R_t]$
`0000 00ss ssst tttt dddd d000 0010 0110`	`xor` R_d, R_s, R_t	$R_d \leftarrow [R_s]$ **XOR** $[R_t]$

The TAL Instruction Set (cont.)

Machine Code	Format	Effect
`0010 00ss ssst tttt iiii iiii iiii iiii`	`addi` R_t, R_s, I	$R_t \leftarrow [R_s] + ([I_{15}]^{16} \Vert I_{15..0})$
`0010 01ss ssst tttt iiii iiii iiii iiii`	`addiu` R_t, R_s, I	$R_t \leftarrow [R_s] + ([I_{15}]^{16} \Vert I_{15..0})$ (overflow ignored)
`0011 00ss ssst tttt iiii iiii iiii iiii`	`andi` R_t, R_s, I	$R_t \leftarrow 0^{16} \Vert ([R_s]_{15..0}$ **AND** $I_{15..0})$
`0011 1100 000t tttt iiii iiii iiii iiii`	`lui` R_t, I	$R_t \leftarrow I_{15..0} \Vert 0^{16}$
`0011 01ss ssst tttt iiii iiii iiii iiii`	`ori` R_t, R_s, I	$R_t \leftarrow [R_s]_{31..16} \Vert ([R_s]_{15..0}$ **OR** $I_{15..0})$
`0011 10ss ssst tttt iiii iiii iiii iiii`	`xori` R_t, R_s, I	$R_t \leftarrow [R_s]_{31..16} \Vert ([R_s]_{15..0}$ **XOR** $I_{15..0})$
`0000 0000 0000 0000 dddd d000 0001 0000`	`mfhi` R_d	$R_d \leftarrow$ [HI]
`0000 00ss sss0 0000 0000 0000 0001 0001`	`mthi` R_s	HI $\leftarrow [R_s]$
`0000 0000 0000 0000 dddd d000 0001 0010`	`mflo` R_d	$R_d \leftarrow$ [LO]
`0000 00ss sss0 0000 0000 0000 0001 0011`	`mtlo` R_s	LO $\leftarrow [R_s]$
`0000 0000 000t tttt dddd diii ii00 0000`	`sll` R_d, R_t, I	$R_d \leftarrow [R_t]_{31-(I_{10..6})..0} \Vert 0^{(I_{10..6})}$
`0000 0000 000t tttt dddd diii ii00 0010`	`srl` R_d, R_t, I	$R_d \leftarrow 0^{I_{10..6}} \Vert [R_t]_{31..(I_{10..6})}$
`0000 0000 000t tttt dddd diii ii00 0011`	`sra` R_d, R_t, I	$R_d \leftarrow ([R_t]_{31})^{(I_{10..6})} \Vert [R_t]_{31..(I_{10..6})}$
`0000 00ss ssst tttt dddd d000 0000 0100`	`sllv` R_d, R_t, R_s	$R_d \leftarrow [R_t]_{(31-[R_s]_{4..0})..0} \Vert 0^{[R_s]_{4..0}}$
`0000 00ss ssst tttt dddd d000 0000 0110`	`srlv` R_d, R_t, R_s	$R_d \leftarrow 0^{[R_s]_{4..0}} \Vert [R_t]_{31..([R_s]_{4..0})}$
`0000 00ss ssst tttt dddd d000 0000 0111`	`srav` R_d, R_t, R_s	$R_d \leftarrow ([R_t]_{31})^{[R_s]_{4..0}} \Vert [R_t]_{31..([R_s]_{4..0})}$
`1100 01bb bbbt tttt iiii iiii iiii iiii`	`lwc1` F_t, I(R_b)	$F_t \leftarrow M[[R_b] + ([I_{15}])^{16} \Vert [I]_{15..0})]$
`1110 01bb bbbt tttt iiii iiii iiii iiii`	`swc1` F_t, I(R_b)	$F_t \rightarrow M[[R_b] + ([I_{15}])^{16} \Vert [I]_{15..0})]$
`0100 0110 000t tttt ssss sddd dd00 0000`	`add.s` F_d, F_s, F_t	$F_d \leftarrow [F_s] + [F_t]$
`0100 0110 000t tttt ssss sddd dd00 0001`	`sub.s` F_d, F_s, F_t	$F_d \leftarrow [F_s] - [F_t]$
`0100 0110 000t tttt ssss sddd dd00 0010`	`mul.s` F_d, F_s, F_t	$F_d \leftarrow [F_s] * [F_t]$
`0100 0110 000t tttt ssss sddd dd00 0011`	`div.s` F_d, F_s, F_t	$F_d \leftarrow [F_s]/[F_t]$
`0100 0110 1000 0000 ssss sddd dd10 0000`	`cvt.s.w` G_d, W_s	$G_d \leftarrow [W_s]$
`0100 0110 0000 0000 ssss sddd dd10 0100`	`cvt.w.s` W_d, G_s	$W_d \leftarrow [G_s]$
`0000 01ss sss0 0000 iiii iiii iiii iiii`	`bltz` R_s, I	if($[R_s] < 0$), then PC $\leftarrow [PC] + 4 + ([I_{15}]^{14} \Vert [I]_{15..0} \Vert 0^2)$
`0000 01ss sss0 0001 iiii iiii iiii iiii`	`bgez` R_s, I	if($[R_s] \geq 0$), then PC $\leftarrow [PC] + 4 + ([I_{15}]^{14} \Vert [I]_{15..0} \Vert 0^2)$
`0001 10ss sss0 0000 iiii iiii iiii iiii`	`blez` R_s, I	if($[R_s] \leq 0$), then PC $\leftarrow [PC] + 4 + ([I_{15}]^{14} \Vert [I]_{15..0} \Vert 0^2)$
`0001 11ss sss0 0000 iiii iiii iiii iiii`	`bgtz` R_s, I	if($[R_s] > 0$), then PC $\leftarrow [PC] + 4 + ([I_{15}]^{14} \Vert [I]_{15..0} \Vert 0^2)$
`0001 00ss ssst tttt iiii iiii iiii iiii`	`beq` R_s, R_t, I	if($[R_s] = [R_t]$), then PC $\leftarrow [PC] + 4 + ([I_{15}]^{14} \Vert [I]_{15..0} \Vert 0^2)$
`0001 01ss ssst tttt iiii iiii iiii iiii`	`bne` R_s, R_t, I	if($[R_s] \neq [R_t]$), then PC $\leftarrow [PC] + 4 + ([I_{15}]^{14} \Vert [I]_{15..0} \Vert 0^2)$

The TAL Instruction Set (cont.)

Machine Code	Format	Effect

`0000 00ss ssst tttt dddd d000 0010 1010`	`slt` R_d, R_s, R_t	if($[R_s] < [R_t]$), then $R_d \leftarrow 0^{31} \parallel 1$
		else $R_d \leftarrow 0^{32}$

`0010 10ss ssst tttt iiii iiii iiii iiii`	`slti` R_t, R_s, I	if$[R_s]) < ([I_{15}]^{16} \parallel [I]_{15..0})$, then $R_t \leftarrow 0^{31} \parallel 1$
		else $R_t \leftarrow 0^{32}$

`0000 10ii iiii iiii iiii iiii iiii iiii`	`j` I	PC $\leftarrow [PC]_{31..28} \parallel [I]_{25..0} \parallel 0^2$
`0000 00ss sss0 0000 0000 0000 0000 1000`	`jr` R_s	PC $\leftarrow [R_s]$
`0000 11ii iiii iiii iiii iiii iiii iiii`	`jal` I	$R_{31} \leftarrow [PC] + 4$; PC $\leftarrow [PC]_{31..28} \parallel [I]_{25..0} \parallel 0^2$
`0000 00ss sss0 0000 dddd d000 0000 1001`	`jalr` R_d, R_s	$R_d \leftarrow [PC] + 4$; PC $\leftarrow [R_s]$
`0000 0000 0000 0000 0000 0000 0000 1100`	`syscall`	PC ← ExceptionHandler
`0000 00xx xxxx xxxx xxxx 0xxx xx00 1101`	`break`	PC ← ExceptionHandler
`0100 0010 0000 0000 0000 0000 0001 0000`	`rfe`	restore state information
`0100 0000 000t tttt dddd d000 0000 0000`	`mfc0` R_t, C_d	$R_t \leftarrow [C_{d(CP0)}]$
`0100 0000 100t tttt dddd d000 0000 0000`	`mtc0` R_t, C_d	$C_{d(CP0)} \leftarrow [R_t]$
`0100 0100 000t tttt dddd d000 0000 0000`	`mfc1` R_t, F_d	$R_t \leftarrow [F_{d(CP1)}]$
`0100 0100 100t tttt dddd d000 0000 0000`	`mtc1` R_t, F_d	$F_{d(CP1)} \leftarrow [R_t]$

General Notes

(1) I specifies part of the instruction.

(2) M[i] is the value of the (aligned) word of memory beginning at location i.

(3) m[i] is the value of the byte of memory at location i.

(4) R_b, R_d, R_s, and R_t specify general registers. R_{31} specifies register 31.

(5) C_d specifies a control register (co-processor 0).

(6) F_d specifies a floating point register (co-processor 1). G_d and G_s specify a floating point register in single-precision floating point format. W_d and W_s specify a floating point register in two's complement format.

(7) || indicates concatenation of bit fields.

(8) Superscripts indicate repetitions of a binary value.

(9) Subscripts indicate bit positions (Little-Endian) of sub-field.

(10) Square brackets ([]) indicate "the contents of."