Number	0		1 2		3	XX	4	. 5	6	7		9	10	11	12	13	14	15	DD	EE	FF	GG
Component	MARS	instruction memory	instruction decoder	reg dst m	nux	register file	sign extend	alu src mux	alu	shift left	adder 1	\$RA	adder2	branch and	branch mux	concatenate	jump mux	JR mux	PC	LSW mux	data memory	mem to reg mux
.dat file/ address	.dat file	address (0)				ab = rt (1)					[top = 4]											
ALL CONTROLS			ALL CONTROLS	REG DST		REG WR (2)		ALU SRC (2)	ALLI OR (2)			RdEn = JAL (2)		left = BRANCH (2)			JUMP (2)	JR (2)		LSW (2)	WrEn = MEM WF	R MEM TO REG (2)
				left = rd I	ight =		instruction	ALO SIC (2)	7120 07 (2)			NULII - UAL (2)		Divitori (2)		top = instruction [25:0] (1)		JN (2)		2517 (2)	(2)	MEM TO REG (2)
instruction		instruction	n instruction (1)	[15:11] (1) aw	t (1)	aa = instr(1) aw = aw(3)	[15:0] (1)									[25:0] (1)						
da				aw		da da			top = da(XX)													
																					din = db	
db						db	immediate	top = db (XX)													(register file, XX)	
							(not really an input															
(immediate)							anymore)															
sign extend out							sign extend out	bottom = sign ext out (4)		input = sign extend out (4)										left = sign extend out (4)		
_									bottom = alu	,												
alu src out								alu src out	src out (5)					right - alu						right = alu out		
alu out									alu out					right = alu out (6)						(6)		top = alu out (6)
zero flag									zero flag													
shift out										shift out			bottom = shift out (7)	t								
adder 1 out											adder 1 out	bottom = adder	top = adder 1 out (8)		top = adder 1 out (8)							
											uuuci i out		out (o)		001 (0)			bottom = \$RA				
\$RA out												\$RA out			., ,, ,,			out (9)				
adder 2 out													adder 2 out		side = adder 2 out (10)							
branch and out														branch and out	bottom = branch and out (11)							
																	jump side = branch mux out					
branch mux out															branch mux out		branch mux out (12)					
concatenate																concetonato	jump bottom = concatenate out					
out																out	(13)					
jump mux out																	jump mux out	top = jump mux out (14)				
																			pc in = JR mux			
JR mux out											bottom = PC					bottom = PC		JR mux out	out (15)			
PC out											out (DD)					[31:28] (DD)			PC out			
LSW mux out																				LSW mux out	address = LSW mux out (EE)	
data memory																						bottom = data memory out (FF)
out						dw = mem to															out	memory out (FF)
mem to reg						reg mux out (GG)																mem to reg mux out