Harvard Architecture Processor

Ports:

CLK	INPUT (1 BIT)
RESET	INPUT (1 BIT)
INSTRUCTION ADD BUS	OUTPUT (8 BIT)
INSTRUCTION BUS	INPUT (16 BIT)
DATA BUS	INOUT (8 BIT)
DATA ADD BUS	OUTPUT (8 BIT)

8-bit Registers:

AX	000
BX	001
CX	010
DX	011
EX	100
FX	101
GX	110
HX	111

Instructions:

ADD	Addition
AND	Logical AND
BE	Branch if Equal
BER	Branch to Address in Register if Equal
BNE	Branch if Not Equal
BNER	Branch to Address in Register if Not Equal
DIV	Division
EQ	Equal
GT	Greater Than
GTE	Greater Than or Equal
J	Jump
JR	Jump to Address in Register
LD	Load
LI	Load Immediate
LT	Less Than
LTE	Less Than or Equal
MULT	Multiplication
NE	Not Equal
NOT	Logical NOT
OR	Logical OR
ROL	Rotate Left
ROR	Rotate Right
SL	Shift Left
SRA	Shift Right Arithmetic
SRL	Shift Right Logical
STORE	Store to Memory
SUB	Subtraction
XOR	Logical XOR

Legend:

(2:0)	Bits 2 to 0 (3-bit)
(3:0)	Bits 3 to 0 (4-bit)
111 or R1C	3-bit value for register 1 (R1)
222 or R2C	3-bit value for register 2 (R2)
DDD or RDC	3-bit value for destination register (RD)
MMMM-MMMM	8-bit Memory Address
IIII-IIII	8-bit Immediate Data
OOOOO	5-bit Opcode
x	Don't care Conditions
R1	Register 1
R2	Register 2
RD	Destination Register
M	Memory Address
I	Immediate Data
PC	Program Counter
NPC	Next PC or Increment PC

Instruction Set

Hex Code	Opcode	Instruction	Format	Action
Shift Instructions				
0-0RDC-xR1C-xR2C	00000	SRL	OOOO-ODDD-x111-x222	RD <= R1 srl R2(2:0)
0-1RDC-xR1C-xR2C	00001	SRA	OOOO-ODDD-x111-x222	RD <= R1 sra R2(2:0)
1-0RDC-xR1C-xR2C	00010	SL	OOOO-ODDD-x111-x222	RD <= R1 sl R2(2:0)
1-1RDC-xR1C-xR2C	00011	ROL	OOOO-ODDD-x111-x222	RD <= R1 rol R2(2:0)
2-0RDC-xR1C-xR2C	00100	ROR	OOOO-ODDD-x111-x222	RD <= R1 ror R2(2:0)
Logic Instructions				
2-1RDC-xR1C-xR2C	00101	AND	OOOO-ODDD-x111-x222	RD <= R1 and R2
3-0RDC-xR1C-xR2C	00110	OR	OOOO-ODDD-x111-x222	RD <= R1 or R2
3-1RDC-xR1C-xxxx	00111	NOT	OOOO-ODDD-x111-xxxx	RD <= not (R1)
4-0RDC-xR1C-xxxx	01000	XOR	OOOO-ODDD-x111-x222	RD <= R1 xor R2
Addition				
4-1RDC-xR1C-xR2C	01001	ADD	OOOO-ODDD-x111-x222	RD <= R1 + R2
Subtraction				
5-0RDC-xR1C-xR2C	01010	SUB	OOOO-ODDD-x111-x222	RD <= R1 - R2
Compare Instructions				
5-1RDC-xR1C-xR2C	01011	LT	OOOO-ODDD-x111-x222	RD <= 1 if R1<R2, else 0
6-0RDC-xR1C-xR2C	01100	GT	OOOO-ODDD-x111-x222	RD <= 1 if R1>R2, else 0
6-1RDC-xR1C-xR2C	01101	EQ	OOOO-ODDD-x111-x222	RD <= 1 if R1=R2, else 0
7-0RDC-xR1C-xR2C	01110	GTE	OOOO-ODDD-x111-x222	RD<=1 if R1>=R2,else 0
7-1RDC-xR1C-xR2C	01111	LTE	OOOO-ODDD-x111-x222	RD<=1 if R1<=R2,else 0
8-0RDC-xR1C-xR2C	10000	NE	OOOO-ODDD-x111-x222	RD <=1 if R1/=R2,else 0
Multiplication				
8-1RDC-xR1C-xR2C	10001	MULT	OOOO-ODDD-x111-x222	RD <= R1(3:0) * R2(3:0)
Division				
9-0RDC-xR1C-xR2C	10010	DIV	OOOO-ODDD-x111-x222	RD <= R1 / R2(3:0)
Branch Instructions				
9-1R1C-MMMM-MMMM	10011	BNE	OOOO-O111-MMMM-MMMM	If R1 /= 0 PC <= M, else PC <= NPC (next PC)
A-0R1C-MMMM-MMMM	10100	BE	OOOO-O111-MMMM-MMMM	If R1 = 0 PC <= M, else PC <= NPC (next PC)
A-1R1C-xxxx-x R2C	10101	BNER	OOOO-O111-xxxx-x222	If R1 /= 0 PC <= R2, else PC <= NPC (next PC)
B-0R1C-xxxx-xR2C	10110	BER	OOOO-O111-xxxx-x222	If R1 = 0 PC <= R2, else PC <= NPC (next PC)
B-1xxx-MMMM-MMMM	10111	J	OOOO-Oxxx-MMMM-MMMM	PC <= M
C-0xxx-xxxx-xR1C	11000	JR	OOOO-Oxxx-xxxx-x111	PC <= R1
Load Instructions				
C-1RDC-IIII-IIII	11001	LI	OOOO-ODDD-IIII-IIII	RD <= I
D-0RDC-MMMM-MMMM	11010	LD	OOOO-ODDD-MMMM-MMMM	RD <= M
D-1R1C-MMMM-MMMM	11011	STORE	OOOO-O111-MMMM-MMMM	M <= R1