Registers

- DI (32bit)
- RI (32bit)
- BI (32bit)
- S (32bit)
- C1 (32bit)
- C2 (32bit)
- PC (32bit)
- AR (32bit)
- IR (32bit)

Instruction Set Architecture

Opcode	Task	decimal
LDAC Γ	AC <= Γ	
CLAC	AC <= 0, Z <= 1	
INCAC	AC <= AC + 1; if(AC+1=0) Z <= 1 else Z <= 0	
DECAC	AC <= AC - 1; if(AC+1=0) Z <= 1 else Z <= 0	
MVAC_DI	DI <= AC	
MVAC_RI	RI <= AC	
MVAC_BI	BI <= AC	
MVAC_S	S <= AC	
MVAC_C1	C1 <= AC	
MVAC_C2	C2 <= AC	
MVAC_AR	AR <= AC	
MVDI	AC <= DI	
MVRI	AC <= RI	
MVBI	AC <= BI	
MVS	AC <= S	
MVC1	AC <= C1	
MVC2	AC <= C2	
MVAR	AC <= AR	
READ	AC <= M[AR]	
WRITE	M[AR] <= AC	
ADD_DI	AC <= AC + DI; if(AC+DI=0) Z <= 1 else Z <= 0	
ADD_RI	AC <= AC + RI; if(AC+RI=0) Z <= 1 else Z <= 0	
ADD_BI	AC <= AC + BI; if(AC+BI=0) Z <= 1 else Z <= 0	

ADD_S	AC <= AC + S; if(AC+S=0) Z <= 1 else Z <= 0	
ADD_C1	AC <= AC + C1; if(AC+C1=0) Z <= 1 else Z <= 0	
ADD_C2	AC <= AC + C1; if(AC+C2=0) Z <= 1 else Z <= 0	
ADD_AR	AC <= AC + AR; if(AC+AR=0) Z <= 1 else Z <= 0	
SUB_DI	AC <= AC - DI; if(AC-DI=0) Z <= 1 else Z <= 0	
SUB_RI	AC <= AC - RI; if(AC-RI=0) Z <= 1 else Z <= 0	
SUB_BI	AC <= AC - BI; if(AC-BI=0) Z <= 1 else Z <= 0	
SUB_S	AC <= AC - S; if(AC-S=0) Z <= 1 else Z <= 0	
SUB_C1	AC <= AC – C1; if(AC-C1=0) Z <= 1 else Z <= 0	
SUB_C2	AC <= AC – C2; if(AC-C2=0) Z <= 1 else Z <= 0	
SUB_AR	AC <= AC - AR; if(AC-AR=0) Z <= 1 else Z <= 0	
MUL2	AC <= AC << 1; if(AC << 1=0) Z <= 1 else Z <= 0	
MUL4	AC <= AC << 2; if(AC << 2=0) Z <= 1 else Z <= 0	
DIV16	AC <= AC >> 4; if(AC >> 4=0) Z <= 1 else Z <= 0	
JMPZ Γ	If(z=0) goto Γ	
JMPNZ Γ	If(z!=0) goto Γ	
END	PC <= 0; FINISH <= 1	
NOP	Do nothing	