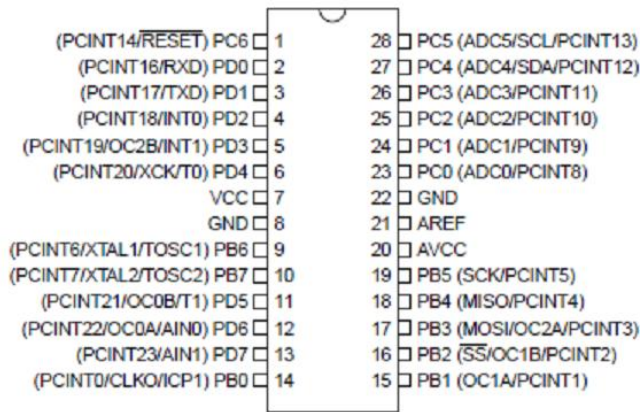
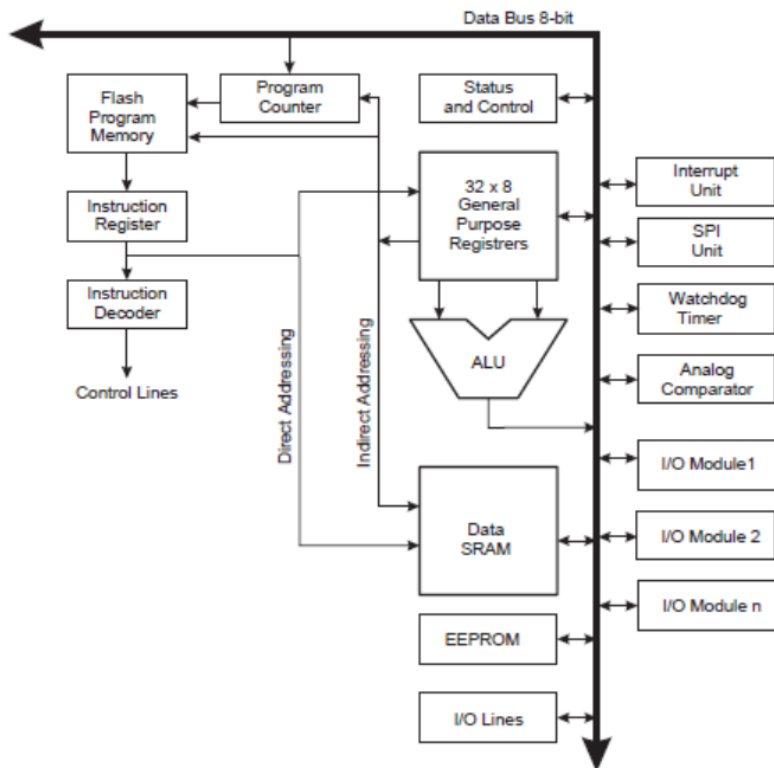


ATmega88 Microcontroller

28-Pin PDIP Package ja 32-pin TQFP package.



Block Diagram of the AVR Architecture



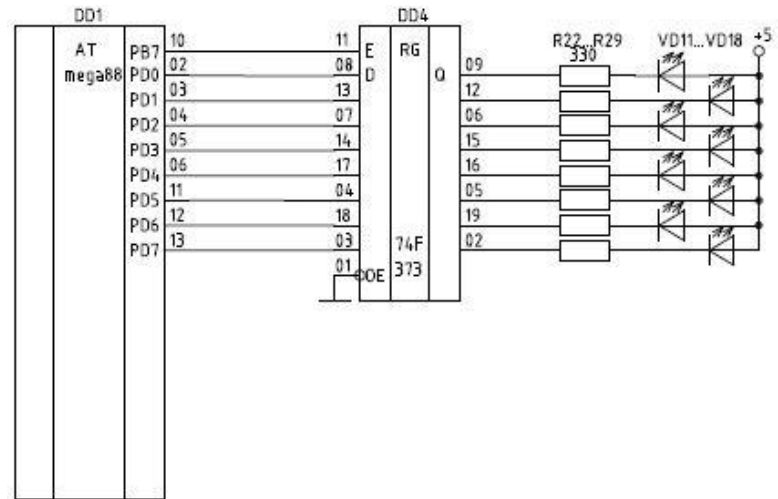
Atmel - ATmega88

8-bitine mikrokontroller 8KB sisemiselt programmeeritava Flash mälu.

Tehnilised parameetrid:

- **Laiendatud RISC arhitektuur:**
 - 131 tegusat ja võimast käsku – Enamus käske täidetakse ühe tsükli ajal;
 - 32 x 8 tööregistrit, moodustades registrite faili;
 - kiirus kuni 20 MIPS, taksagedus kuni 20 MHz.
- **Hea püsi- ja müütmälude valik ja info säilivus:**
 - 8KB sisemiselt programmeeritav Flash mälu programmi jaoks;
 - 512 Baiti EEPROM (Elektriliselt programmeeritav püsimälu);
 - 1KB sisemine SRAM;
 - püsimälude ümberprogrammeerimise tsükleid 10 000 tsükli programmi Flash ja 100 000 tsükli EEPROM.
- **Sisemine programmälu programmeerimine:**
 - kasutatakse sisemist programmi (ISP - In System Programming);
 - ja algaadimise programm (*Boot Code Section*) koos sõltukatute kaitsebitidega (*Lock Bits*).
- **Sisemised ja välised sisend/väljundseadmed:**
 - kaks 8-bitist timerit ja loendurit koos eelneva jagauriga ja võrdlemisega;
 - üks 16-bitine taimer ja loendur koos eelkeva jagauriga ja väärdlemisega ning välise lüüsi kasutamisega (*Compare Mode, and Capture Mode*);
 - reaalaaja loendur koos välise kvartsiga (*Real Time Counter with Separate Oscillator*);
 - kuus laius-pulsimodulatsiooni kanalit (PWM - *Pulse Width Modulation*);
 - kuus 10-bitist analoog-digitaalmuundurit (ADC - *Analog-to-Digital Converter*);
 - programmeeritav täisdupleks järjestikliides (Serial USART - *Universal Synchronous Receiver Transmitter*);
 - sünkroonne järjestikliides Master/Slave (SPI - *Serial Peripheral Interface*);
 - bitt-orjenteeritud kahejuhtmeline järjestikliides (Philips I2C liidesega ühilduv);
 - tõrgete korral taaskäivitamiseks on programmeeritav vahikoer/taimer (WDT - *Watchdog Timer with Separate On-chip Oscillator*);
 - Analoog komparaator (*On-chip Analog Comparator*);
 - katkestuse kotroller ja „uinuvast olekust“ äratamine (*Wake-up on Pin Change*);
- **Spetsiaalsed mikrokontrolleri omadused:**
 - toite sisselülitamise ja toitepinge alanemise taimese kontrollimaks õiget käivitamist (*Power-on Reset and Programmable BOD - Brown-out Detection*);
 - sisemine kalibreeritav generaator (Internal Calibrated Oscillator);
 - sisemine ja väliline katkestus (external and Internal Interrupt Sources);
 - viis „uinuvat“ režiimi, mis tarbivad vähem voolu: Idle, ADC Noise Reduction, Power-save, Power-down, and Standby;
- **Sisend/väljund ja korpused:**
 - 23 Programmeeritavat kahesuunalist väljaviiku;
 - 28-jalaga PDIP korpust, 32-viiguga TQFP korpust, 28-jalaga QFN/MLF või 32-jalaga QFN/MLF
- **Toitepinged:** – 1.8 - 5.5V ATmega88V jaoks ja – 2.7 - 5.5V ATmega88 jaoks
- **Töökiirused:**
 - ATmega88V: 0 - 4 MHz @ 1.8 - 5.5V, 0 - 10 MHz @ 2.7 - 5.5V
 - ATmega88: 0 - 10 MHz @ 2.7 - 5.5V, 0 - 20 MHz @ 4.5 - 5.5V
- **Voolutarvudus:** – 250µA töörežiimis; – 0,1µA „uinuvast“ režiimis.

Mn. Kood	Operats.	Kirjeldus	Tegevus	Lipud
ADD	Rd, Rr	Add two Registers Liida kaks registrit	$Rd < Rd + Rr$	Z,C,N,V,H
ADC	Rd, Rr	Add with Carry two Registers Liida kaks registrit arvestades ülekannet	$Rd < Rd + Rr + C$	Z,C,N,V,H
SUB	Rd, Rr	Subtract two Registers Lahuta kaks registrit	$Rd < Rd - Rr$	Z,C,N,V,H
SUBI	Rd, K	Subtract Constant from Register Lahuta konstant registrist	$Rd < Rd - K$	Z,C,N,V,H
SBC	Rd, Rr	Subtract with Carry two Registers Lahuta kaks registrit arvestades ülekannet	$Rd < Rd - Rr - C$	Z,C,N,V,H
SBCI	Rd, K	Subtract with Carry Constant from Reg. Lahuta konstant registris ja arvesta ülekannet	$Rd < Rd - K - C$	Z,C,N,V,H
AND	Rd, Rr	Logical AND Registers Loogiline JA registrite vahel	$Rd < Rd \cdot Rr$	Z,N,V
ANDI	Rd, K	Logical AND Register and Constant Loogiline JA registry ja konstandi vahel	$Rd < Rd \cdot K$	Z,N,V
INC	Rd	Increment Suurenda	$Rd < Rd + 1$	Z,N,V
DEC	Rd	Decrement Vähenda	$Rd < Rd - 1$	Z,N,V
SEC		Set Carry Pane ülekande lipp üheks	$C < 1$	C
CLC		Clear Carry Pane ülekande lipp nulliks	$C < 0$	C
TST	Rd	Test for Zero or Minus Kontrolli nulli või miinust	$Rd < Rd \cdot Rd$	Z,N,V
SBI	P,b	Set Bit in I/O Register Pane bitt I/O registris üheks	$I/O(P,b) < 1$	None
CBI	P,b	Clear Bit in I/O Register Pane bitt I/O registris nulliks	$I/O(P,b) < 0$	None
LSL	Rd	Logical Shift Left Loogiline nihutamine vasakule	$Rd(n+1) < Rd(n), Rd(0) < 0$	Z,C,N,V
ROL	Rd	Rotate Left Through Carry Ringnihe vasakule läbi ülekande	$Rd(0) < C, Rd(n+1) < Rd(n), C < Rd(7)$	Z,C,N,V
ROR	Rd	Rotate Right Through Carry Ringnihe paremale läbi ülekande	$Rd(7) < C, Rd(n) < Rd(n+1), C < Rd(0)$	Z,C,N,V
MOV	Rd, Rr	Move Between Registers Liiguta ühest registrist teise	Rd, Rr	None
MOVW	Rd, Rr	Copy Register Word Liiguta 16 bitiline sõna ühest registrist teise	$Rd+1:Rd < Rr+1:Rr$	None
LDI	Rd, K	Load Immediate Lae konstant otse registrist	$Rd < K$	None
IN	Rd, P	In Port Loe sisendpordist	Rd, P	None
OUT	P, Rr	Out Port Kirjuta väljundporti	P, Rr	None
RJMP	k	Relative Jump Suhteline hüppamine	$PC < PC + k + 1$	None
RCALL	k	Relative Subroutine Call Suhteline alamprogrammi hüppamine	$PC < PC + k + 1$	None
RET		Subroutine Return Alamprogrammist tagasi pöördumine	$PC < STACK$	None
BRNE	s, k	Branch if Not Equal Hüppa kui pole võrdsed	if (Z = 0) then $PC < PC + k + 1$	None
BREQ	s, k	Branch if Equal Hüppa kui on võrdsed	if (Z = 1) then $pc < PC + k + 1$	None



Input/Output Ports

PORTB	0x05(0x25)	0x00	<input type="checkbox"/>
DDRB	0x04(0x24)		<input type="checkbox"/>
PINB	0x03(0x23)		<input type="checkbox"/>
PORTC	0x08(0x28)	0x00	<input type="checkbox"/>
DDRC	0x07(0x27)		<input type="checkbox"/>
PINC	0x06(0x26)		<input type="checkbox"/>
PORTD	0x0B(0x2B)		<input type="checkbox"/>
PORTD	0x0B(0x2B)		<input type="checkbox"/>
DDRD	0x0A(0x2A)		<input type="checkbox"/>
PIND	0x09(0x29)		<input type="checkbox"/>

```

ldi r16,0x80 ; 1000 0000
out 0x04,r16 ; PB7=1 Väljund - Enable pulss.
                ; ülejäänud sisendid

ldi r16,0xFF ; 1111 1111
out 0x0A,r16 ; PD kõik järgud välja
ldi r16,0xfe ; registriste r16 = 0xFE ehk 0b11111110
out 0x0B,r16 ; registri r16 sisu saadetakse portD 0x0B
                ; Kui registri järk on = 0 siis LED «põleb»
;--- registle 74F373 E strobi andmine -----
sbi 0x05,7 ; pordi B seitsmes bitt pannakse üheks
cbi 0x05,7 ; pordi B seitsmes bitt pannakse nulliks
Loop: rjmp Loop ;STOP (lõputu tsükkel)

```