



# ATMEL AVR

## LISHA/UFSC

Prof. Dr. Antônio Augusto Fröhlich

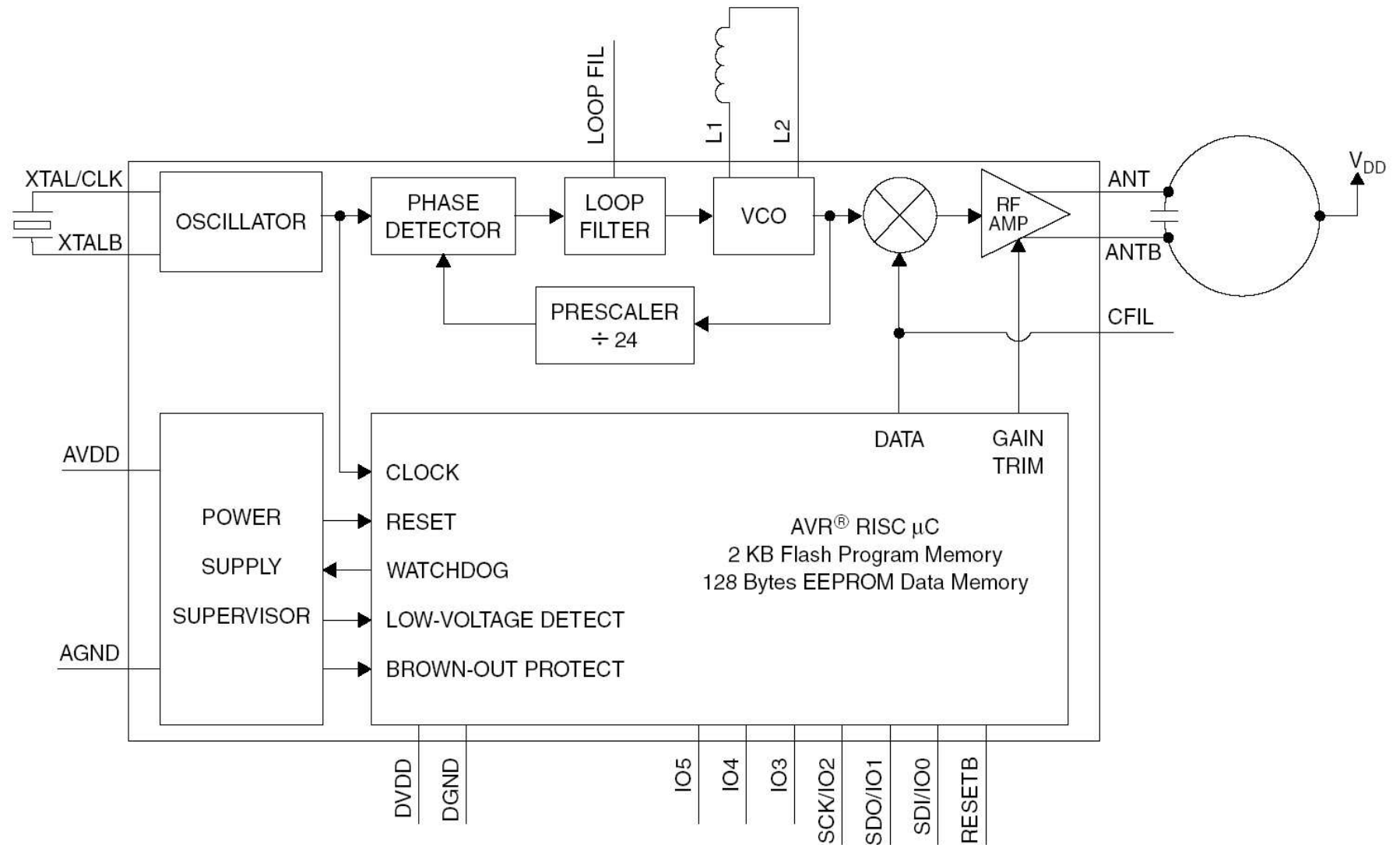
Lucas Francisco Wanner

<http://www.lisha.ufsc.br/~guto>

March 2004



# AVR AT86RF401 WSD

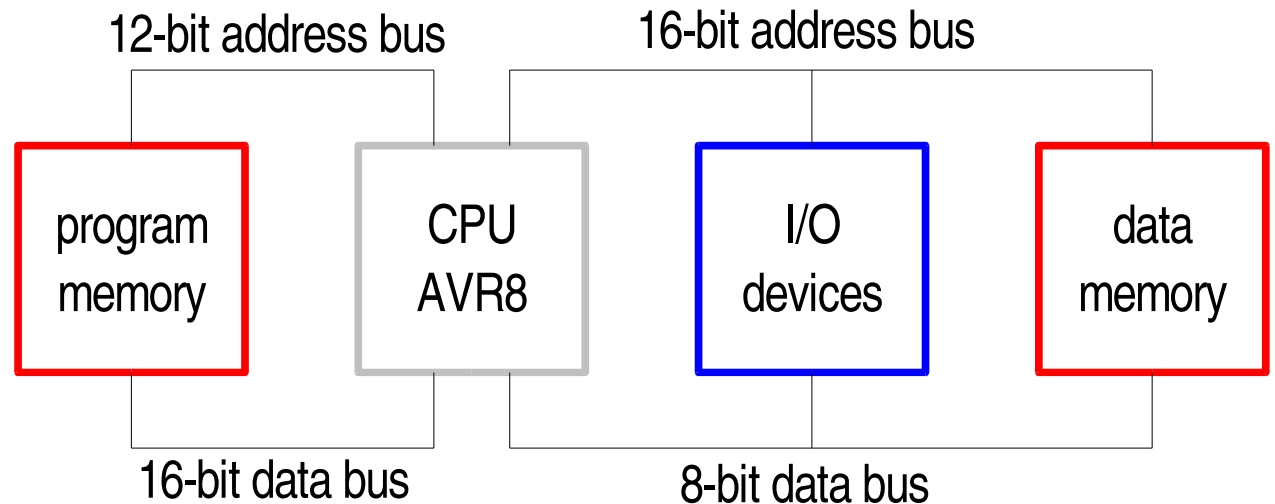




# AVR AT90S Microcontroller

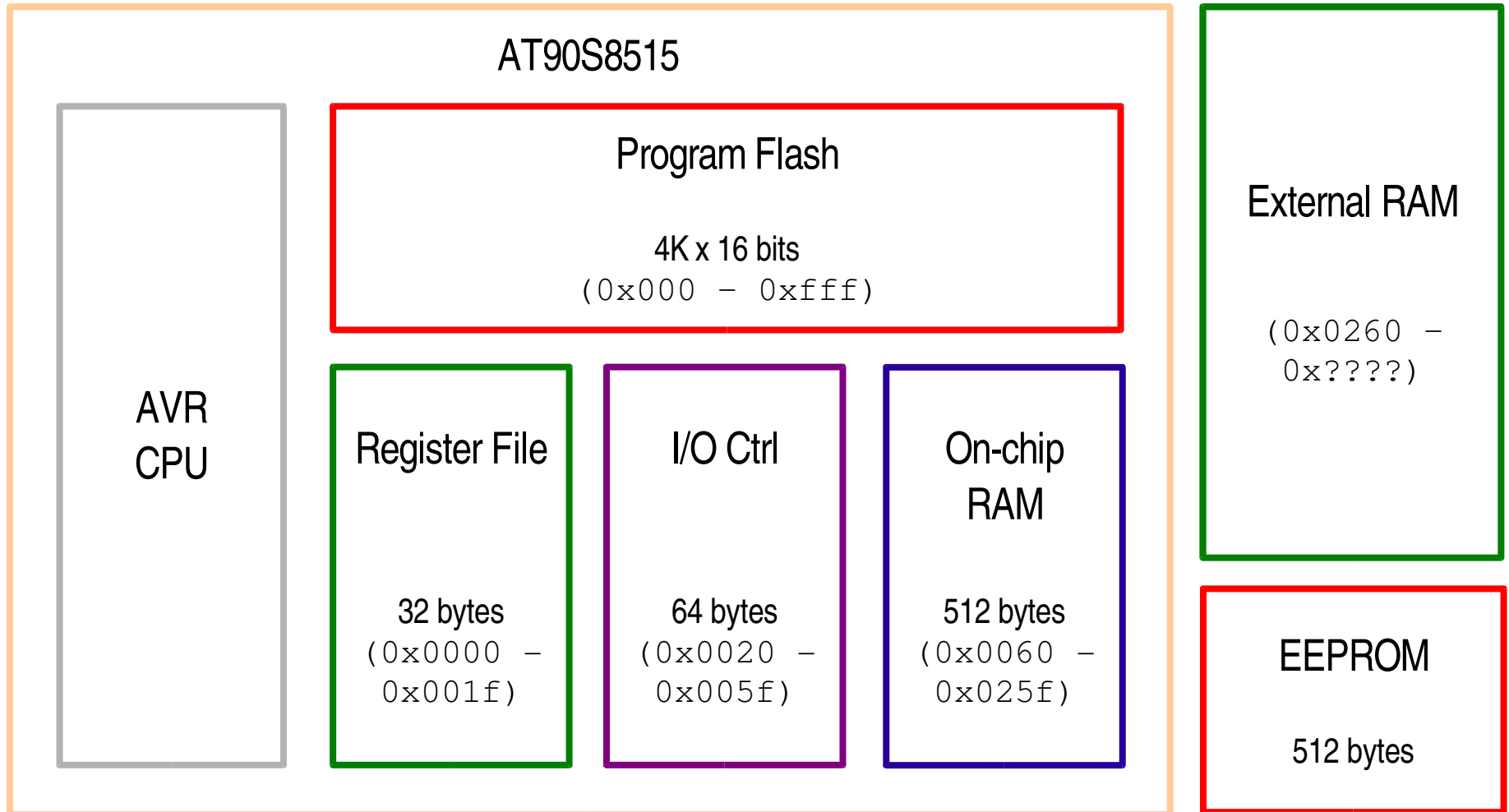
## ■ AVR CPU

- RISC
- 8-bit data
- Harvard architecture
  - 16-bit data address space
  - 12-bit program address space
- MCU
  - UART
  - SPI
  - A/D
  - timers
- 32 x 8-bit GP registers (r0 – r31)
  - 3 16-bit pointers: X (r26:27), Y (r28:29), Z (r30:31)
- Memory-mapped I/O Registers
- 8 MHz clock





# AVR AT90S Organization





# AVR Instruction Set

- Instructions are 16 or 32 bits wide
  - Program memory is word-addressed
- Instructions take 1-4 clock cycles
  - Most take one
- ALU instructions
  - `add, subi, and, inc, mul, ...`
- Branch instructions
  - `rjmp, ljmp, rcall, ret, reti, brge, ...`
- Data transfer instructions
  - `mov, ld, st, lpm, push, pop`
- Bit and bit-test instructions
  - `lsl, swap, cli, ...`



# AVR Data Types

- Data Types
  - Bytes (8 bits)
  - Words (16 bits)
- Little-endian





# AVR Registers

0x00	r0	general purpose registers
0x01	r1	
	:	
0x19	r25	
0x1a	r26	x (lsb)
0x1b	r27	x (msb)
0x1c	r28	y (lsb)
0x1d	r29	y (msb)
0x1e	r30	z (lsb)
0x1f	r31	z (msb)



# AVR Data Addressing Modes

## ■ Register direct

```
clr  r0          ; r0 is cleared  
add  r0, r1      ; r0 = r0 + r1
```

## ■ I/O direct

```
in   r16, mcucr  ; r16 = mcucr (I/O 0x35)
```

## ■ Memory direct

```
lds  r0, 1234    ; r0 = *((unsigned char *)0x1234)
```

## ■ Memory indirect with displacement

```
ldd  r0, y + $3F ; r0 = y[$3F]
```

## ■ Memory indirect (pre-decrement or post-increment)

```
ld   r0, x+      ; r0 = *x++
```

## ■ Constant addressing using LPM (program memory)

```
lpm          ; r0 = Program_Memory[z]
```





# AVR Program Addressing Modes

## ■ Indirect

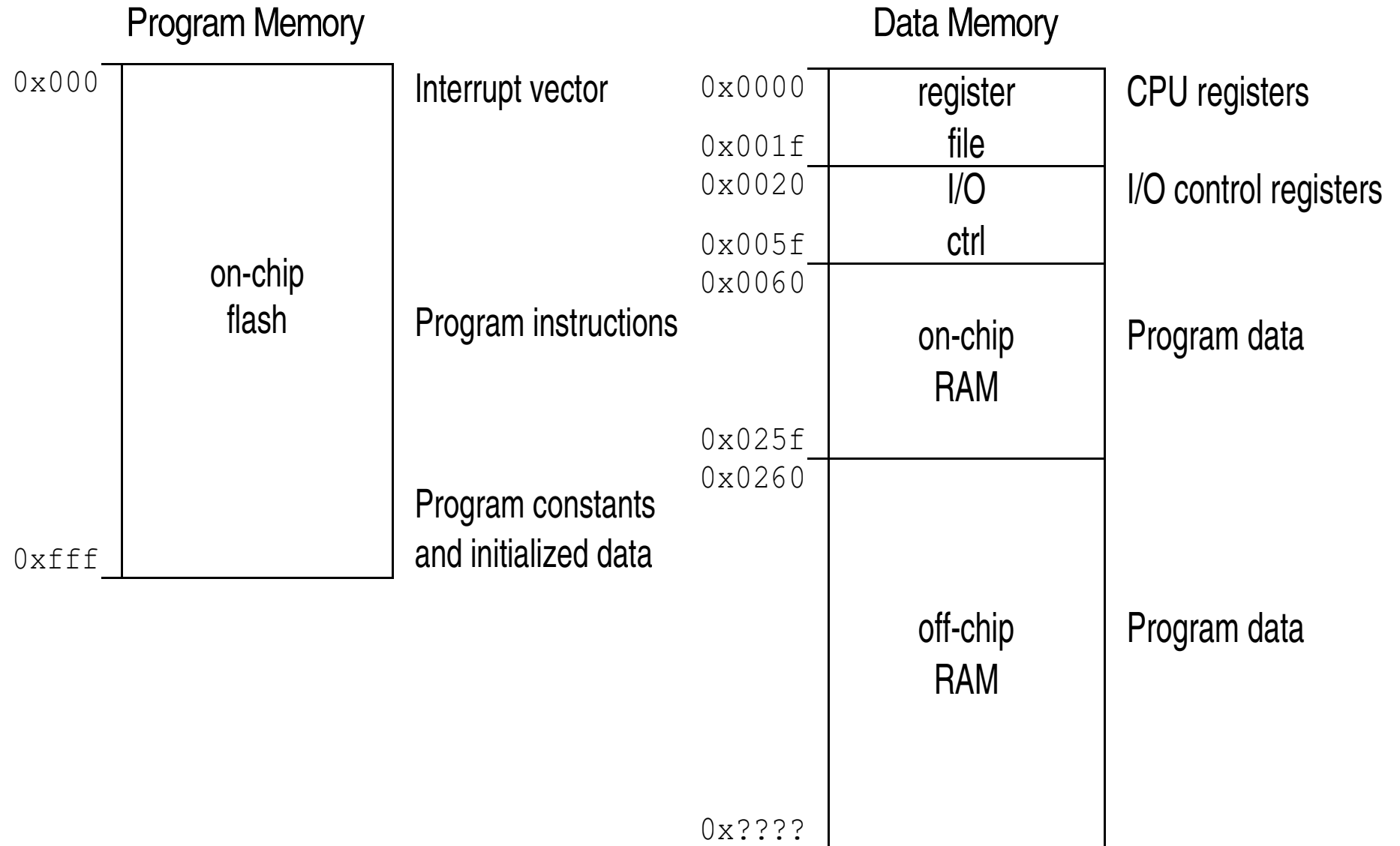
`ijmp           ; PC = z`

## ■ Relative

`rjmp $20   ; PC = PC + 20 + 1`



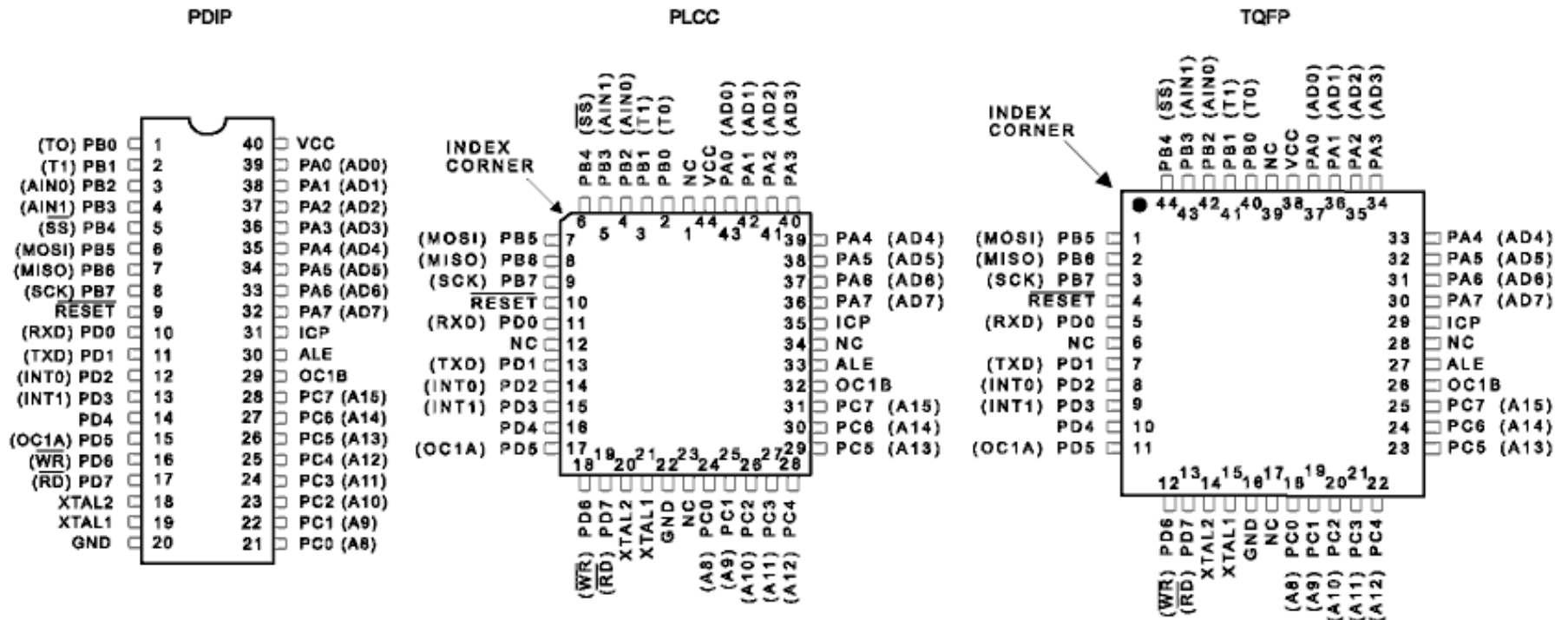
# AVR AT90S Memory Layout





# AVR AT90S8515 I/O

- 40-44 pins
- 4 GPIO ports
- VCC, GND, RESET, XTAL, ...





# AVR I/O and Peripherals

- Parallel ports
  - CPU pins / GPIO
- Timers / counters
  - Generic (control sequences, OS ticks, ...)
  - Watchdog
- Serial ports
  - UART
- Serial Peripheral Interface (SPI)
- I2C bus
- DMA controllers
- A/D and D/A converters