4. ADC In ATmega328p

4.1 ATmega328p ADC Specifications

The ATmega328p (used in the Arduino Uno) has a built-in ADC with the following specifications:

Specification	Value
Resolution	10-bit (produces values from 0 – 1023)
Conversion Method	Successive Approximation
Input Channels	8 analog channels (A0 – A7), multiplexed
Reference Voltage	AVcc, Internal 2.56V, or external AREF pin
Conversion Speed	50 kHz – 200 kHz (depending on prescaler)
Result Registers	ADCL (low-byte) + ADCH (high-byte)

4.2 Successive Approximation Method

image

The ATmega328p uses the Successive Approximation Register (SAR) method. In this method, the ADC works by performing a binary search for the Vin value.

At each step:

- The SAR sets a trial bit
- The DAC generates a voltage (Vdac)
- A comparator compares Vin with Vdac
- The bit is either kept or changed based on the comparison result

This process is repeated N times (N = ADC resolution, which is 10-bit for the ATmega328p).

An example of SAR implementation can be seen in this image:

image

1. **Started from the middle value (1000)** This is the representation of **½ of Vref** (MSB = 1).
2. **First comparison (Comparator)**
   - If **Vdac > Vin** → move down (red arrow)
   - If **Vdac < Vin** → move up (green arrow)
3. **Determining the next bit** Each step determines one additional bit. Example:

- `1000 → 1100` (if Vin is larger)
- `1000 → 0100` (if Vin is smaller)

4. **Repeat process (Binary Search)** The value range is continuously narrowed until all bits (**MSB → LSB**) are determined.
5. **Final result** The rightmost nodes show the **final binary code**. Example results: `1011`, `0110`, etc.

Revision #2
Created 2026-04-12 06:44:12 UTC by DS
Updated 2026-04-14 00:34:46 UTC by DS