

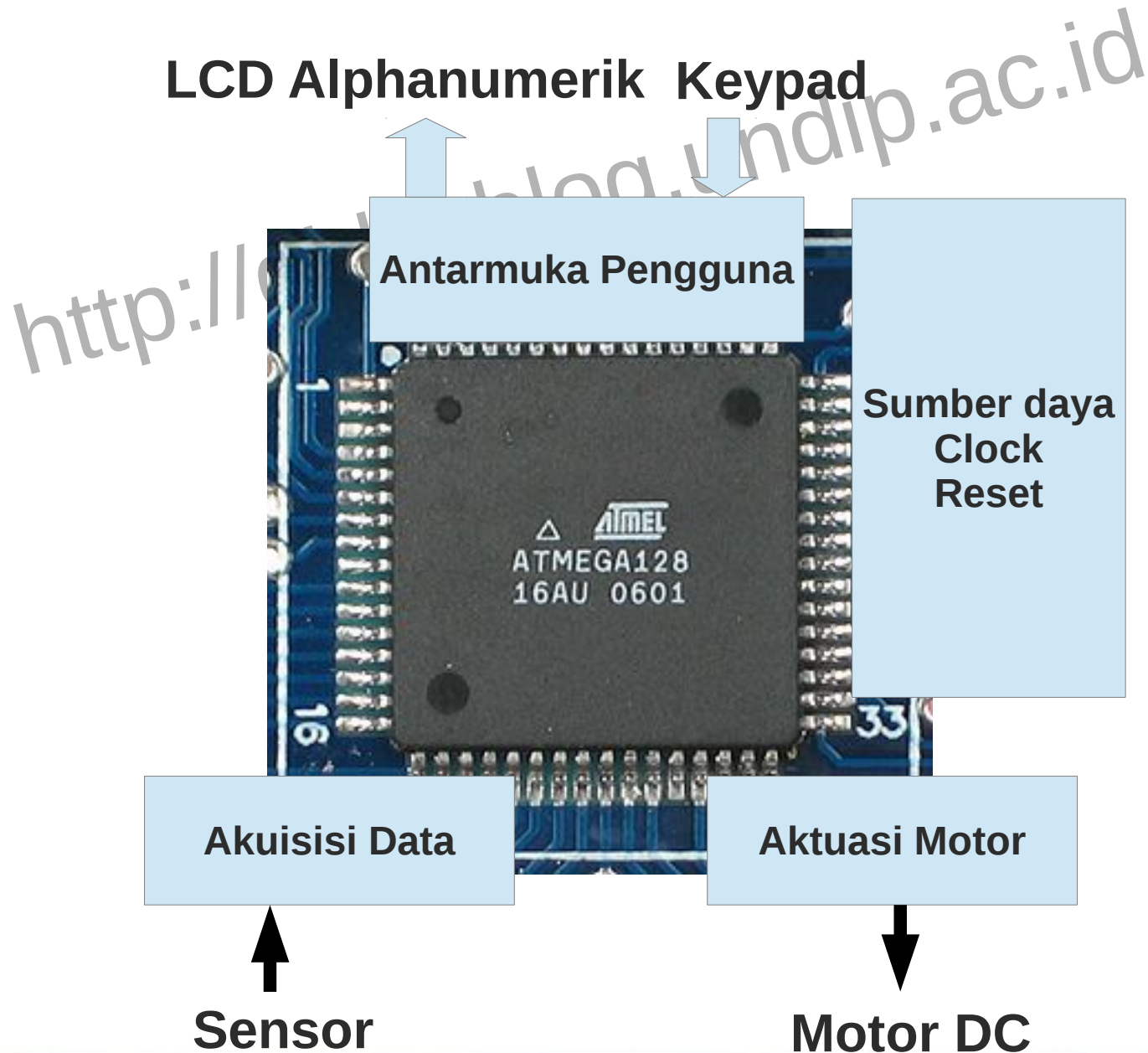
# Teknik Pemrograman Robot

<http://didik.blog.undip.ac.id>

TKC225 - Robotika  
PROGRAM STUDI SISTEM KOMPUTER  
UNIVERSITAS DIPONEGORO

# Arsitektur Robot Beroda

LCD Alphanumerik Keypad



# Topologi Pemrograman (AVR)

- IDE (Integrated Development Environment)
  - Eclipse, Keil, AVR Studio, editor lainnya
- Toolchain
  - Kompiler: avr-gcc, bascom
  - Assembler: avr-as
  - Linker/Archiver: avr-ld (dinamik), avr-ar (statik)
- Programmer
  - Avrdude (<http://download.savannah.gnu.org/releases/avrdude/>)
  - Bascom-avr (<http://www.mcselec.com/>)
  - Khazama AVR Programmer

# ISP Programmer

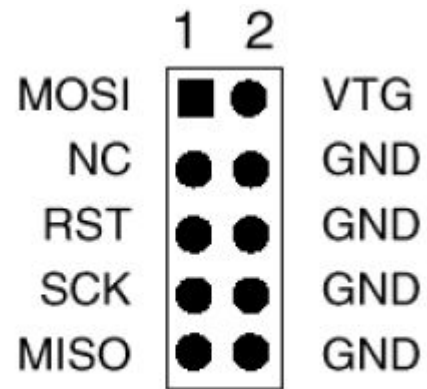
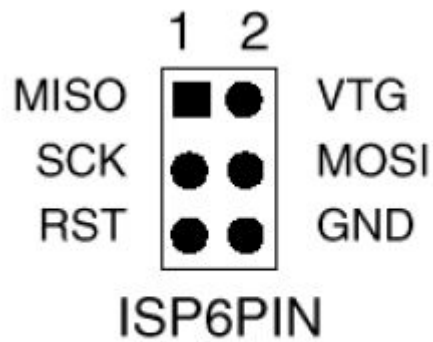


**AVRISP2**

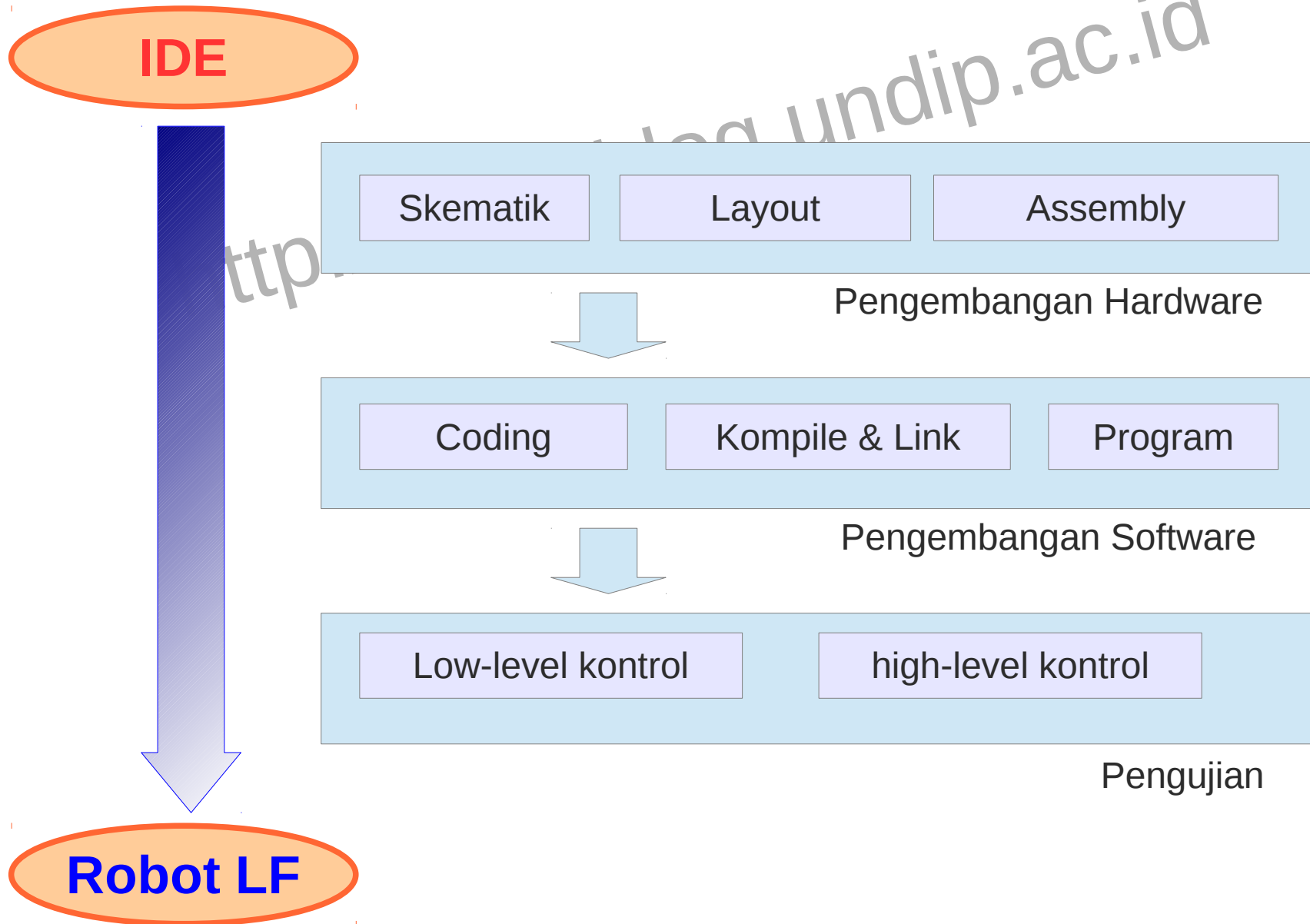


**USBASP**

<http://www.fischl.de/usbasp/>

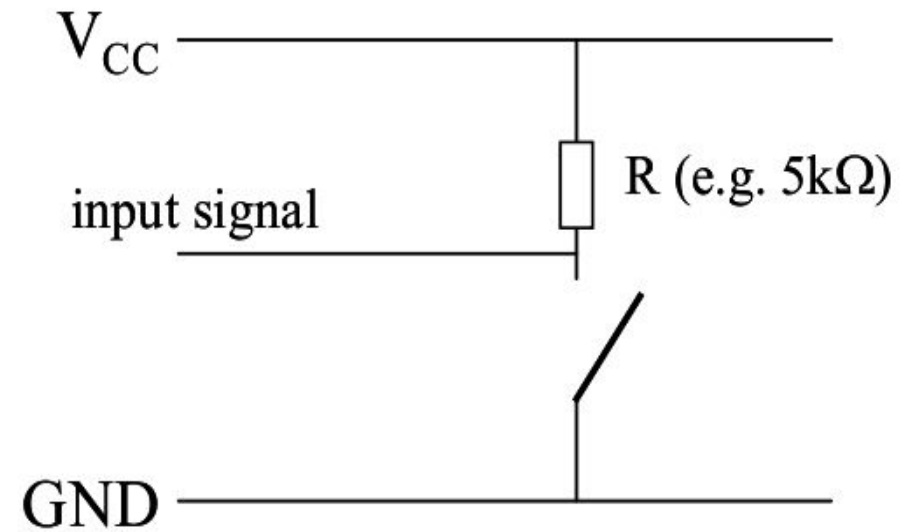


# Metodologi Pengembangan



# Akuisisi Data Sensor

- Taktil sensor (digital)

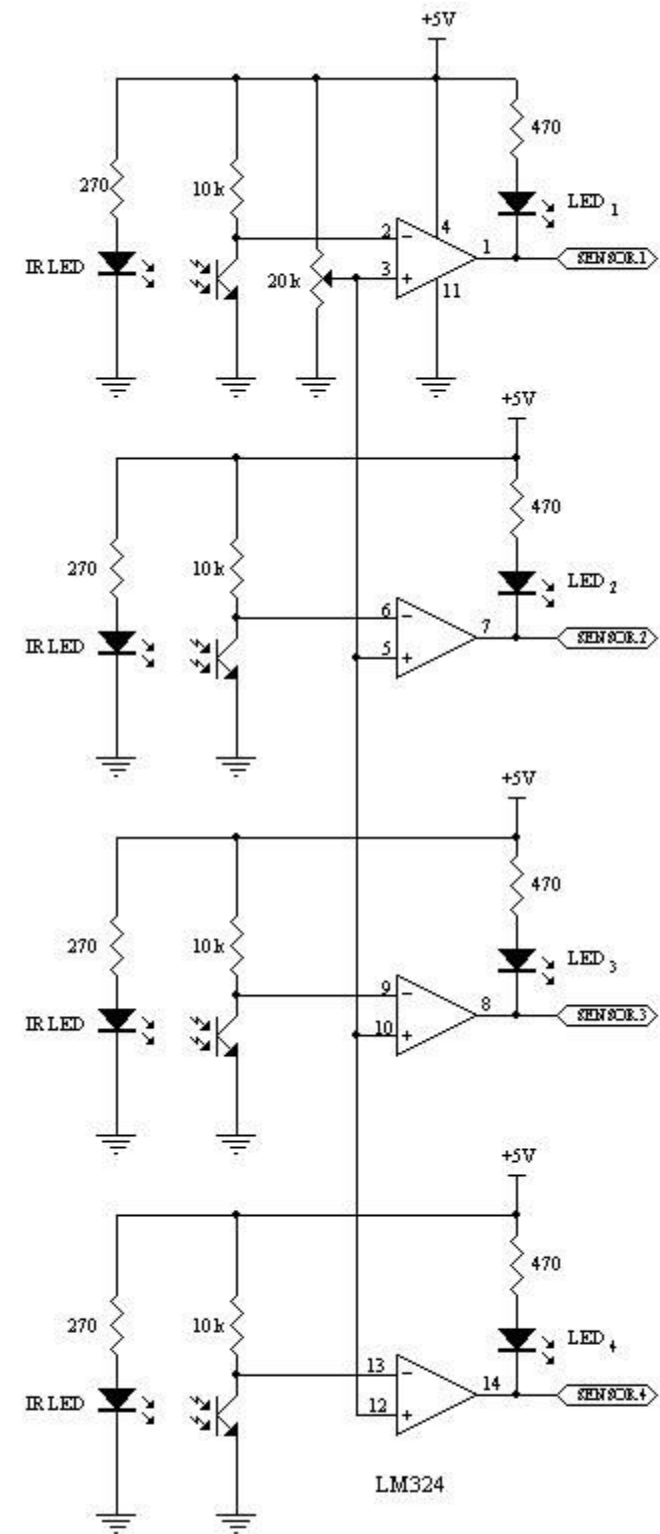


- Pembacaan sensor bernilai 0 atau 1 dari DIO
  - Pada dasarnya sebuah switch

```
int button_is_pressed()
{
    /* the button is pressed when BUTTON_BIT is clear */
    if (bit_is_clear(BUTTON_PIN, BUTTON_BIT))
    {
        delay_ms(DEBOUNCE_TIME);
        if (bit_is_clear(BUTTON_PIN, BUTTON_BIT)) return 1;
    }
    return 0;
}
```

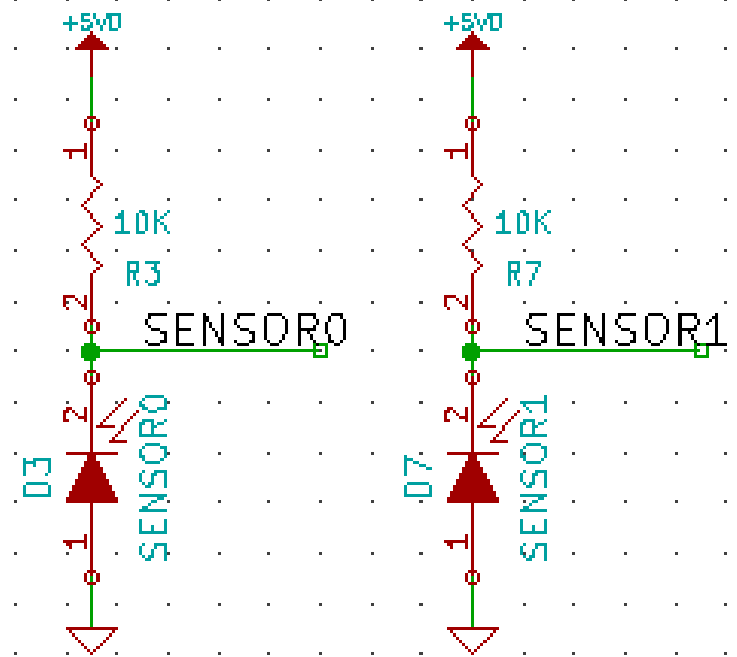
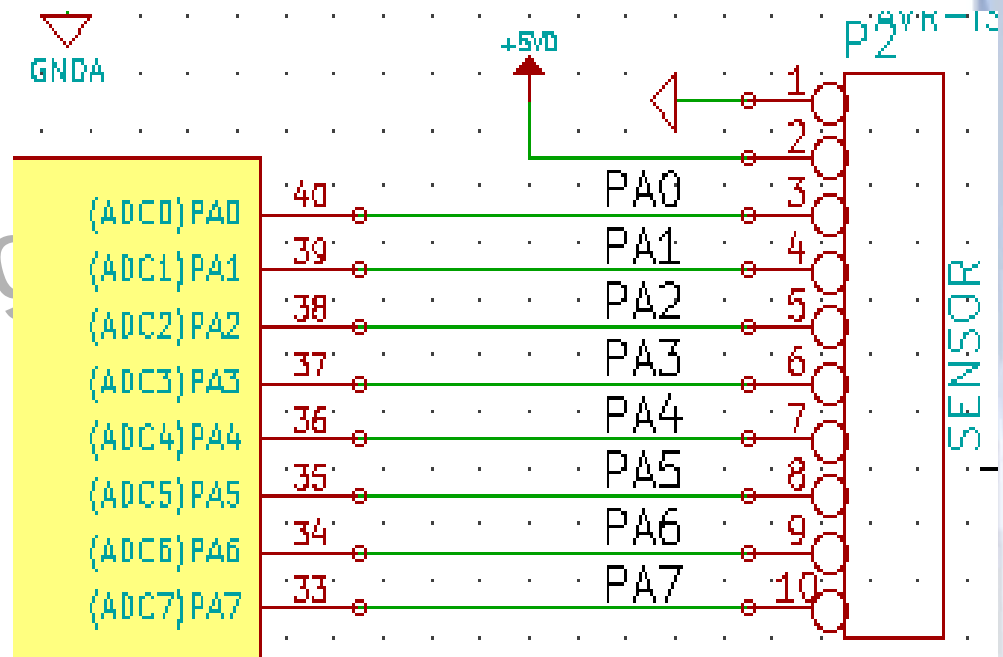
# Akuisisi Data Sensor Garis

- Dengan komparator
  - Output: digital
  - Tidak programmable



# Akuisisi Data Sensor Garis

- Tanpa komparator
  - Output: Analog
  - Perlu ADC





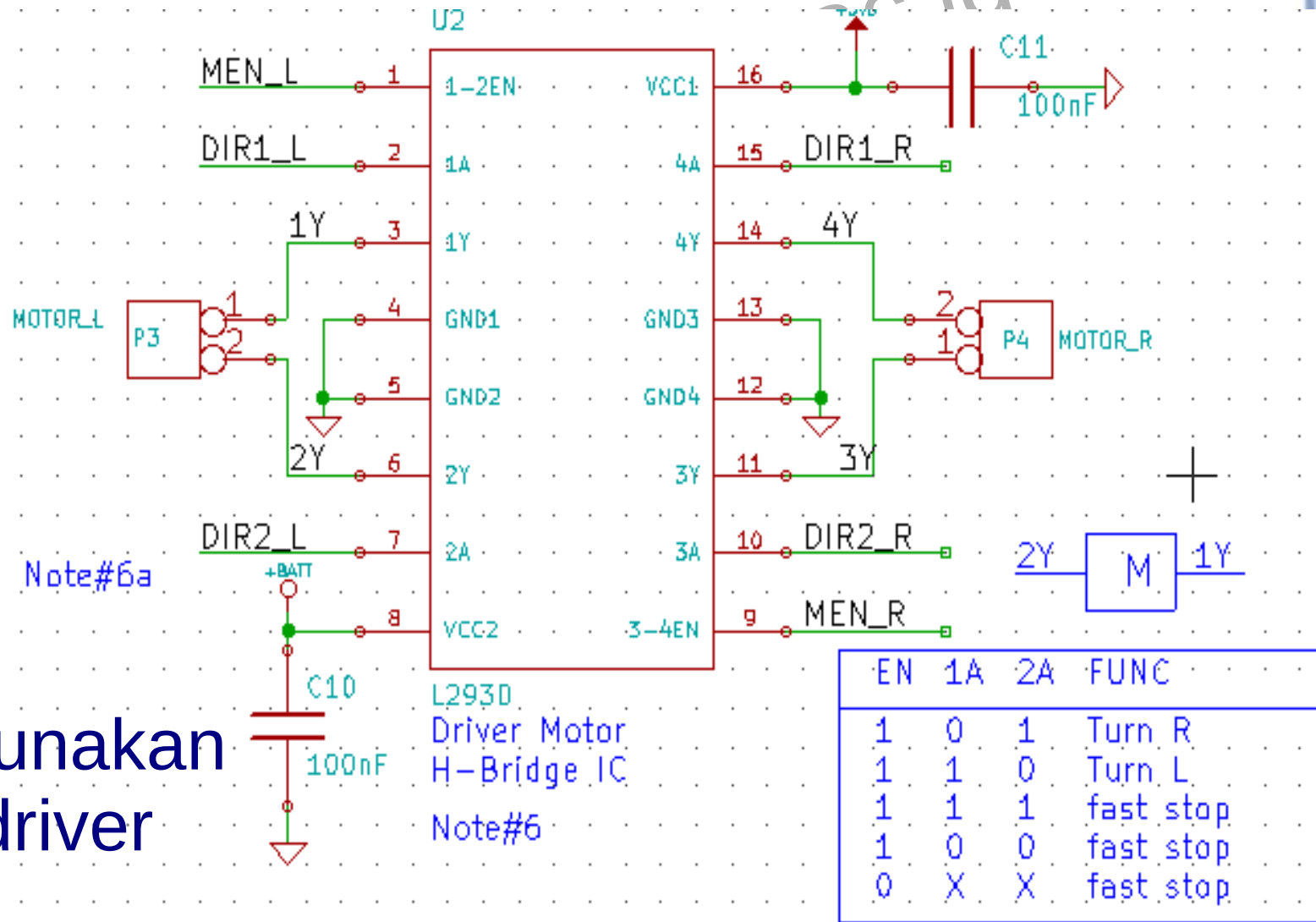
# Akuisisi Data ADC

- Mode:
  - Polling
    - Akuisisi dilakukan di loop main()
  - Interupsi
    - Akuisisi dilakukan di ISR
    - Timer atau ADC completion
- Sumber: <http://www.avrfreaks.net/index.php?name=PNphpBB2&file=printview&t=56429&start=0>

# Bagaimana Menentukan Threshold

- Salah satu problem menggunakan ADC adalah menentukan threshold untuk garis dan lantai
  - Kalibrasi otomatis?

# Aktuasi Motor Differensial



- Menggunakan motor driver L293D

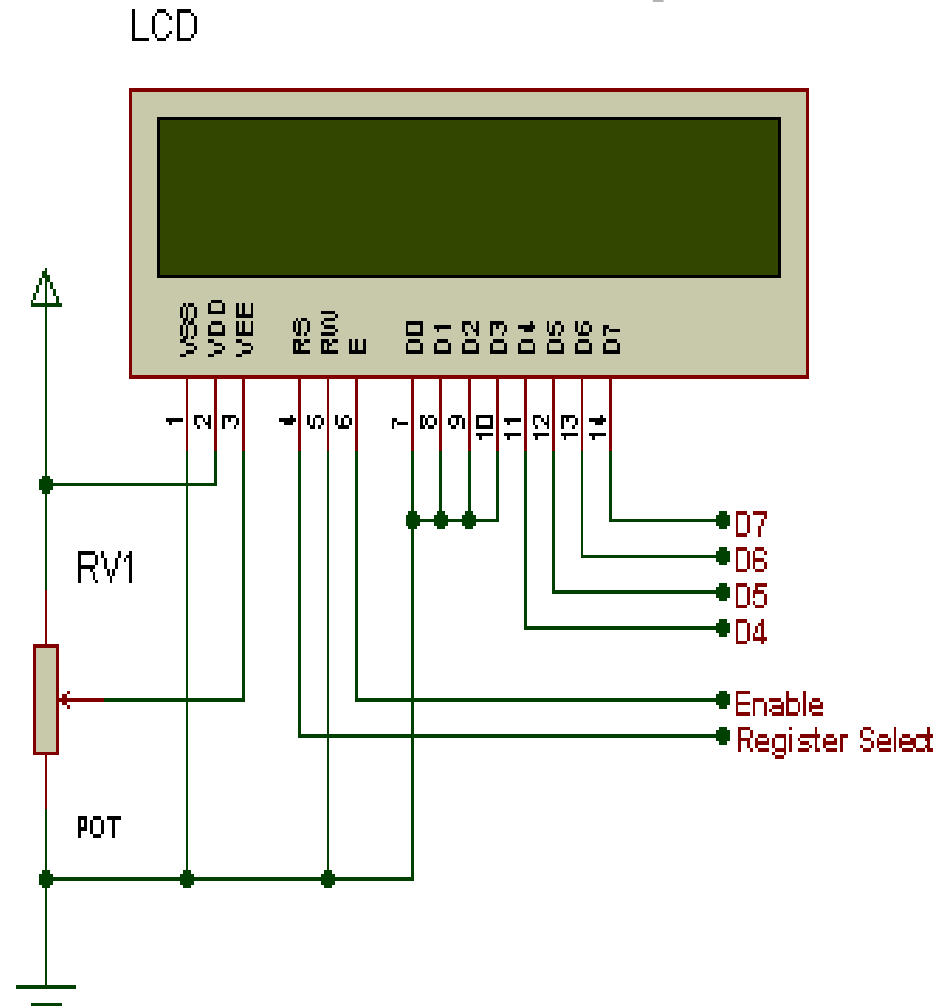
# Contoh Kode Aktuasi Motor

```
#define MOTOR_EN_L B, 3
#define MOTOR_DIR1_L D, 5
#define MOTOR_DIR2_L B, 2
#define MOTOR_EN_R B, 1
#define MOTOR_DIR1_R D, 4
#define MOTOR_DIR2_R B, 0

void forward_full_speed(unsigned char idx) {
    if (idx == MOTOR_RIGHT_CTL) {
        SET (PORT, MOTOR_EN_R);
        CLR (PORT, MOTOR_DIR1_R);
        SET (PORT, MOTOR_DIR2_R);
    } else if (idx == MOTOR_LEFT_CTL) {
        SET (PORT, MOTOR_EN_L);
        CLR (PORT, MOTOR_DIR1_L);
        SET (PORT, MOTOR_DIR2_L);
    }
}
```

# LCD (Data 4 bit, Write-Only)

- [http://winavr.scienc  
eprog.com/example  
-avr-projects/avr-  
gcc-4-bit-and-8-bit-  
lcd-library.html](http://winavr.scienc<br/>eprog.com/example<br/>-avr-projects/avr-<br/>gcc-4-bit-and-8-bit-<br/>lcd-library.html)

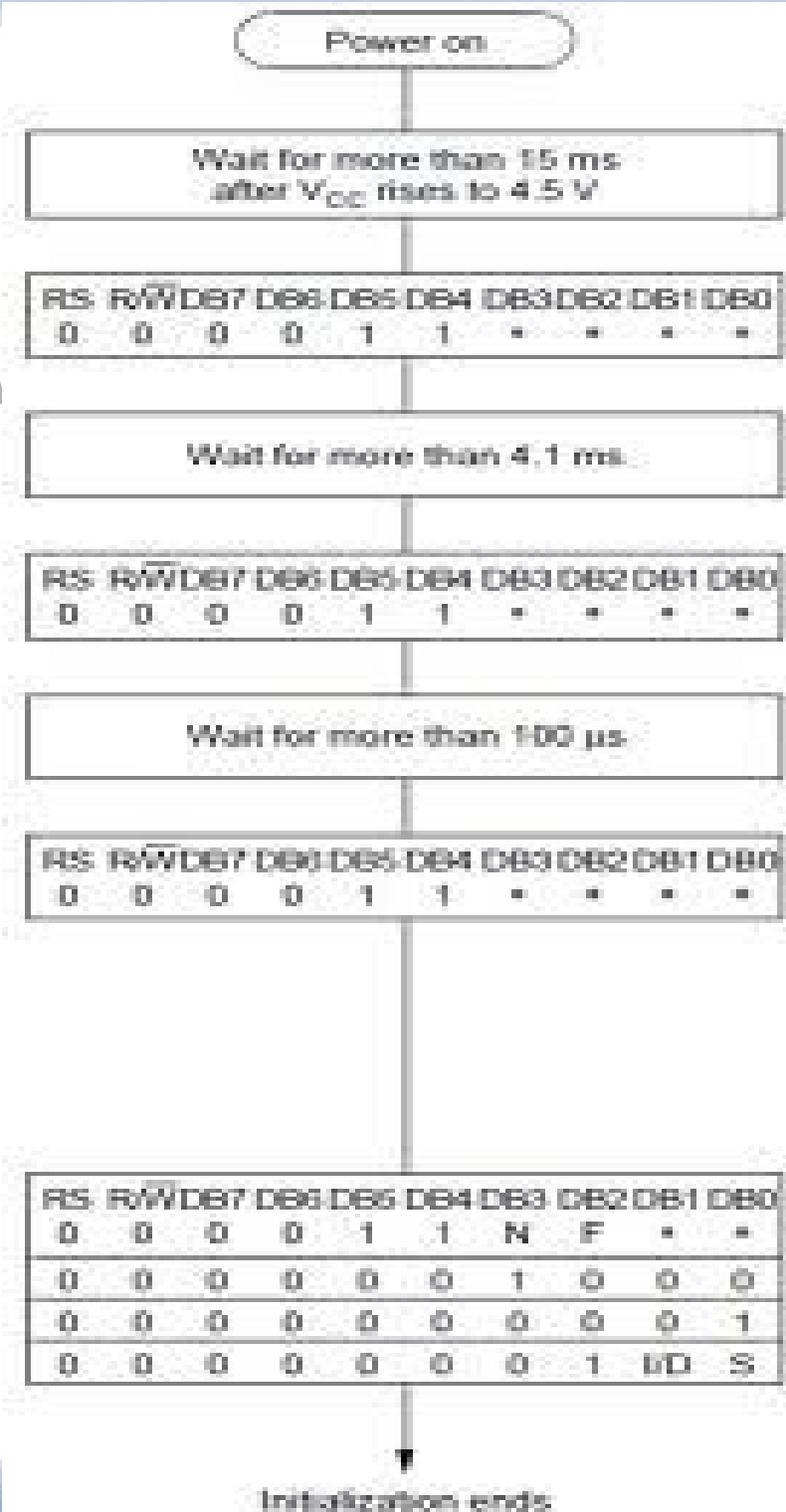


# Kode LCD

```

• void lcd_cmd (char cmd)
• {
•     lcd_port = ((cmd >> 4) & 0x0F)|LCD_EN;
•     lcd_port = ((cmd >> 4) & 0x0F);
•
•     lcd_port = (cmd & 0x0F)|LCD_EN;
•     lcd_port = (cmd & 0x0F);
•
•     delayus(200);
•     delayus(200);
• }
•
• void lcd_data (unsigned char dat)
• {
•     lcd_port = (((dat >> 4) & 0x0F)|LCD_EN|LCD_RS);
•     lcd_port = (((dat >> 4) & 0x0F)|LCD_RS);
•
•     lcd_port = ((dat & 0x0F)|LCD_EN|LCD_RS);
•     lcd_port = ((dat & 0x0F)|LCD_RS);
•
•     delayus(200);
•     delayus(200);
• }

```

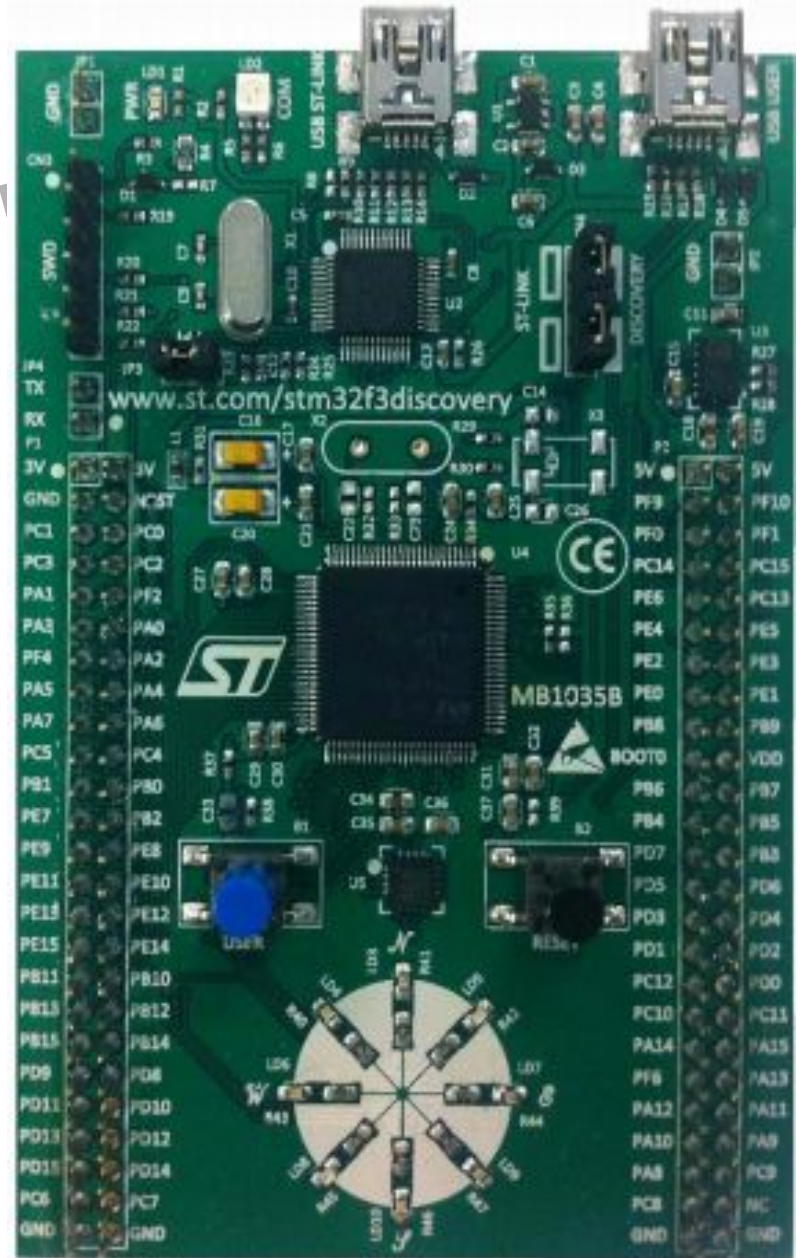


# LCD Menu

- <http://ee.cleversoul.com/lcd-project-source.html>
  - Menu rekursif
  - Sekuensial

# Robot Berbasis ARM 32-bit

- L3GD20, ST MEMs motion sensor, 3-axis digital output gyroscope
- LSM303DLHC, ST MEMs system-in-package featuring a 3D digital linear acceleration sensor and a 3D digital magnetic sensor



STM32F3DISCOVERY



# Tantangan

- IDE dan Toolchain?
- RTOS
  - Free RTOS
  - Ecos
  - Etc

<https://sites.google.com/site/stm32discovery/stm32-resources-and-links/open-source-real-time-operating-systems-for-the-stm32-and-cortex-m3-mpus>