

Pemrograman Perangkat Keras

Pemrograman Sensor Analog



Fakultas Informatika
Telkom University

Sensor Analog

Data Sensor Analog

Pemrograman Sensor Analog

Gaya Arduino

Gaya AVR

References

Apa itu sensor ?

- ▶ Sensor == Panca indra
 - Merasakan keadaan lingkungan
- ▶ Mengubah data lingkungan menjadi data digital
- ▶ Sensor yang baik memenuhi
 - sensitif terhadap properti yang diukur
 - tidak peka terhadap properti lain yang mungkin ditemui dalam aplikasinya
 - tidak mempengaruhi properti yang diukur

Digital Sensors vs Analog

Digital

- Less prone to noise
- One line can transmit multiple sensors
- Need to interpret the 1's and 0's
- Need to make sure clocks are synced up
- An agreed upon language must be set up in order to communicate

Analog

- Can be directly measured
- Usually simpler setups
- No compatibility issues
- More prone to noise
- Needs an ADC for computer interpretation

Figure 1: Sensor Digital dan Analog

Sensor Analog

- ▶ Memiliki data yang beragam
- ▶ Mentransmisikan data analog / gelombang
- ▶ Kecepatan transmisi relatif rendah
- ▶ Jangkauan data yang luas
- ▶ Kompatibilitas terhadap perangkat terdahulu sangat tinggi
- ▶ Memerlukan peripheral konverter

Analog and Digital Circuits

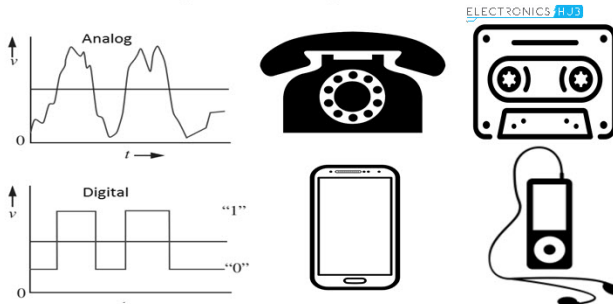


Figure 2: Sensor Digital dan Analog

Analog signals

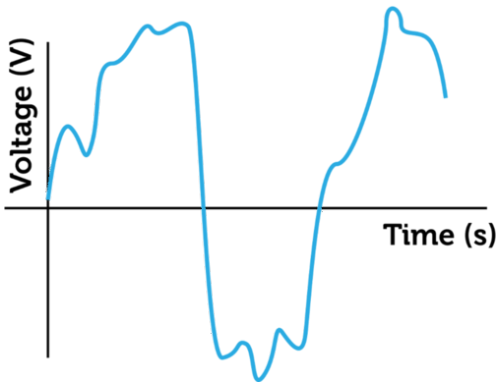


Figure 3: Data Sensor Analog

Data sensor analog

- ▶ Data integer
 - memiliki rentang nilai yang sangat banyak
 - kaya dalam merepresentasikan kumpulan data
- ▶ Merepresentasikan voltase
- ▶ Kepadatan sinyal dipengaruhi oleh panjang ADC
- ▶ Karakteristik transmisi
 - transmisi data lambat
 - berpengaruh pada media transmisi yang digunakan

Arduino IDE memiliki struktur tersendiri

- ▶ IDE menyediakan antarmuka yang simpel
- ▶ Memiliki tools yang dasar dan simpel
- ▶ Memiliki kompatibilitas terhadap produk arduino
- ▶ Struktur dasar kode
 - Setup ← dijalankan sekali
 - Loop ← dijalankan selamanya

Inisiasi penggunaan pin

- ▶ `pinMode(inst1,inst2)`
 - `inst1` $\leftarrow$ nomor pin
 - `inst2` $\leftarrow$ mode yang digunakan :: INPUT, OUTPUT, INPUT-PULLUP,
- ▶ Inisiasi pin hanya dilakukan pada setup
- ▶ Tidak memberikan balikan apapun

Kontrol pin

1. Digital

- `digitalWrite(inst1,inst2)`
 - ▶ `inst1` $\leftarrow$ pin yang dituju
 - ▶ `inst2` $\leftarrow$ data yang ditulis :: HIGH, LOW
- `digitalRead(inst1)`
 - ▶ `inst1` $\leftarrow$ pin yang akan dibaca

2. Analog

- `analogWrite(inst1,inst2)`
 - ▶ `inst1` $\leftarrow$ pin analog yang akan ditulis
 - ▶ `inst2` $\leftarrow$ Data analog yang akan ditulis :: duty cycle 0 - 1024 / 0 - 5 V
- `analogRead(inst1)`
 - ▶ `inst1` $\leftarrow$ pin analog yang akan dibaca

AVR memiliki struktur umum

- ▶ Menggunakan IDE yang memiliki kompiler GCC
 - Arduino IDE
 - Eclipse IDE
 - Atmel Studio
- ▶ Rata-rata tools yang disediakan sangat lengkap
 - Kompiler Dry and Run
 - Debugger
 - Uploader
- ▶ Kompatibilitas hampir semua chip mikrokontroler
- ▶ Struktur dasar kode
 - `main` ← untuk badan kode :: badan `main` dijalankan sekali
 - `while(1)` ← dijalankan selamanya

Pengetahuan tentang register AVR

- ▶ Data Register (PORT_x)
- ▶ Data Direction Register (DDR_x)
- ▶ Port Input Pin (PIN_x)

Data Direction Register

- ▶ Jika diset 1, maka didefinisikan sebagai output
- ▶ Jika diset 0, maka didefinisikan sebagai input

Contoh

- ▶ $DDRB = 0x08$
 - Port nomor 3 diset sebagai output $\leftarrow 00001000$
 - Port yang lainnya diset sebagai input $\leftarrow 00001000$

Data Register

- ▶ Menuliskan menuju port yang sudah diset output
- ▶ Hanya dapat menuliskan data 1 dan 0

Contoh

- ▶ `PORTB = 0x08`
 - Port nomor 3 ditulis data 1 $\leftarrow$ 00001000
 - Port selian nomor 3 ditulis data $\leftarrow$ 00001000

Port Input Pin

- ▶ Membaca port pin yang sudah diset sebagai input
- ▶ Hanya dapat membaca data digital dalam satu waktu

Contoh

- ▶ $A = \text{PINB} \ \& \ 0x08$
 - ▶ Nilai register PINB dioperasikan dengan AND dengan 00001000
 - ▶ Membaca port nomor 3

Thank You!