

# DT-SENSE

---

## THERMOPILE ARRAY SENSOR

### **Trademarks & Copyright**

AT, IBM, and PC are trademarks of International Business Machines Corp.

Windows is a registered trademark of Microsoft Corporation.

Pentium is a trademark of Intel Corporation.

CodeVisionAVR is copyright by Pavel Haiduc, HP InfoTech s.r.l.

BASCOM-51 and BASCOM-AVR are copyright by MCS Electronics.

°C is a registered trademark of Philips Semiconductors.

Delphi is a trademark of Inprise Corporation.

DT-51 is a trademark of Innovative Electronics.

# Daftar Isi

---

|                 |  |           |
|-----------------|--|-----------|
| <b>1</b>        | <b>Pendahuluan.....</b>                                      | <b>3</b>  |
| 1.1             | Spesifikasi DT-SENSE THERMOPILE ARRAY SENSOR.....            | 3         |
| 1.2             | Sistem yang Dianjurkan.....                                  | 3         |
| <b>2</b>        | <b>Perangkat Keras DT-SENSE THERMOPILE ARRAY SENSOR.....</b> | <b>4</b>  |
| 2.1             | Tata Letak Komponen DT-SENSE THERMOPILE ARRAY SENSOR....     | 4         |
| 2.2             | Konektor dan Pengaturan Jumper.....                          | 4         |
| <b>3</b>        | <b>Perangkat Lunak DT-SENSE THERMOPILE ARRAY SENSOR.....</b> | <b>6</b>  |
| 3.1             | Antarmuka UART TTL.....                                      | 6         |
| 3.2             | Antarmuka I <sup>2</sup> C.....                              | 7         |
| 3.3             | Command Set.....   | 8         |
| 3.3.1           | Status Check.....  | 8         |
| 3.3.2           | Autoscan Peak.....   | 9         |
| 3.3.3           | Autoscan Valley.....   | 10        |
| 3.3.4           | Autoscan Higher.....   | 12        |
| 3.3.5           | Autoscan Lower.....  | 13        |
| 3.3.6           | Read Autoscan.....   | 14        |
| 3.3.7           | Set Servo 1.....   | 15        |
| 3.3.8           | Set Servo 2.....   | 16        |
| 3.3.9           | Get Servo 1.....   | 16        |
| 3.3.10          | Get Servo 2.....   | 17        |
| 3.3.11          | Enable/Disable Servo 1.....                                  | 18        |
| 3.3.12          | Enable/Disable Servo 2.....                                  | 19        |
| 3.3.13          | Enable/Disable Channel.....                                  | 20        |
| 3.3.14          | Read All Channels.....                                       | 21        |
| 3.3.15          | Read Ambient.....  | 22        |
| 3.3.16          | Read Channel N-1.....  | 22        |
| <b>4</b>        | <b>Prosedur Pengujian.....</b>                               | <b>23</b> |
| <b>5</b>        | <b>Contoh Aplikasi dan Program.....</b>                      | <b>23</b> |
| <b>Lampiran</b> |  |           |
| A.              | Skematik DT-SENSE THERMOPILE ARRAY SENSOR.....               | 27        |

## 1. PENDAHULUAN

DT-SENSE THERMOPILE ARRAY SENSOR merupakan sebuah sistem sensor cerdas yang mampu mengukur temperatur obyek secara non-kontak berdasarkan radiasi inframerah yang dipancarkan obyek tersebut. Sistem ini terdiri dari sebuah sensor *thermopile array* dan sebuah modul berbasis mikrokontroler yang digunakan untuk mengambil data sensor dan mengolahnya menjadi data temperatur, serta mengatur antarmuka dengan sistem lain. Salah satu aplikasi DT-SENSE THERMOPILE ARRAY SENSOR antara lain untuk mendeteksi api lilin, mengukur temperatur bola lampu, kompor, solder, atau obyek-obyek lainnya secara non-kontak.

### 1.1. SPESIFIKASI DT-SENSE THERMOPILE ARRAY SENSOR

Spesifikasi DT-SENSE THERMOPILE ARRAY SENSOR sebagai berikut:

- Mengukur suhu obyek pada range 0°C - 100°C.
- Beroperasi pada suhu ruangan 0°C - 60°C.
- Akurasi sensor *thermopile array*  $\pm 2^\circ\text{C}$ .
- Terdapat 8 kanal sensor *thermopile* dan 1 *thermistor* untuk mengukur temperatur ruangan.
- Mampu mendeteksi api sebuah lilin atau panas bohlam 15W pada jarak maksimum 200 cm dari mata sensor pada suhu ruangan 20°C.
- Sumber catu daya menggunakan tegangan 9 - 12 VDC melalui regulator atau tegangan 4,8 - 5,4 VDC tanpa melalui regulator.
- Pin Input/Output kompatibel dengan level tegangan TTL dan CMOS.
- Dilengkapi dengan antarmuka UART TTL dan I<sup>2</sup>C.
- Tersedia 2 konektor untuk motor servo standar.
- Dilengkapi dengan *jumper* untuk pengaturan alamat, sehingga bisa di-cascade sampai 8 modul tanpa perangkat keras tambahan (untuk satu *master* menggunakan antarmuka I<sup>2</sup>C).

### 1.2. SISTEM YANG DIANJURKAN

Sistem yang dianjurkan untuk penggunaan DT-SENSE THERMOPILE ARRAY SENSOR adalah:

Perangkat keras:

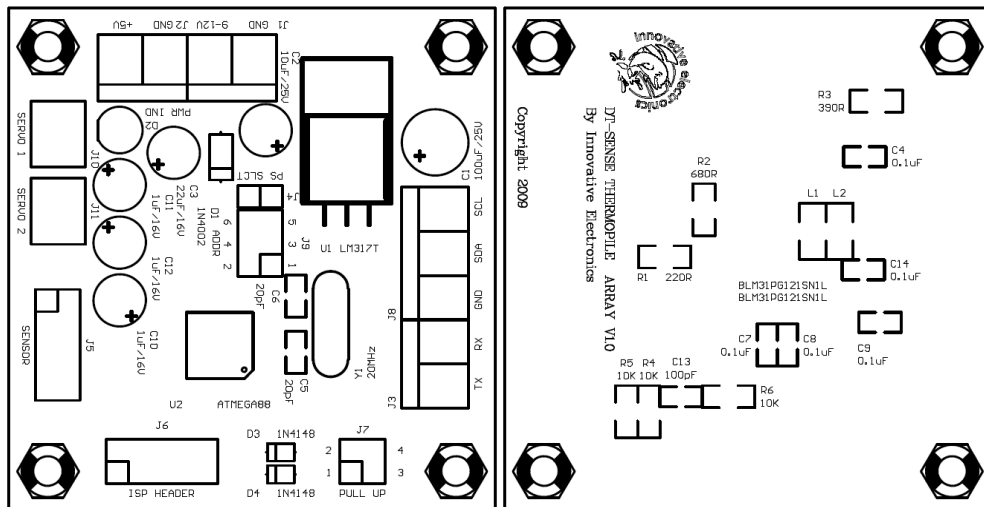
- PC™ AT™ Pentium® IBM™ Compatible dengan port Serial (COM1/COM2) dan Paralel (LPT) atau USB.
- DT-51 Minimum System, DT-51 Low Cost Series, atau DT-AVR Low Cost Series.
- CD-ROM Drive dan Hard disk.

Perangkat lunak:

- Sistem operasi Windows® 98 SE.
- BASCOM-8051®, BASCOM-AVR®, atau CodeVisionAVR®.
- File yang ada pada CD program: CONTOH\_I2C.PRJ, CONTOH\_I2C.C, MANUAL DT-SENSE THERMOPILE ARRAY SENSOR, dan QUICK START DT-SENSE THERMOPILE ARRAY SENSOR.

## 2. PERANGKAT KERAS DT-SENSE THERMOPILE ARRAY SENSOR

### 2.1. TATA LETAK KOMPONEN DT-SENSE THERMOPILE ARRAY SENSOR



### 2.2. KONEKTOR DAN PENGATURAN JUMPER

Konektor POWER (J1 dan J2) berfungsi sebagai konektor untuk catu daya modul. Catu daya bisa melalui regulator tegangan (menggunakan J1) atau tanpa melalui regulator tegangan (menggunakan J2).

| Konektor | Pin | Nama    | Fungsi   |
|----------|-----|---------|--|
| J1       | 1   | GND     | Titik referensi untuk catu daya input                            |
|          | 2   | 9-12VDC | Terhubung ke catu daya untuk input (9 - 12 V)                    |
| J2       | 1   | GND     | Titik referensi untuk catu daya input                            |
|          | 2   | 5VDC    | Terhubung ke catu daya untuk input tanpa regulator (4,8 - 5,4 V) |

Jumper PS SLCT (J4) berfungsi untuk memilih sumber catu daya yang digunakan oleh DT-SENSE THERMOPILE ARRAY SENSOR. Berikut deskripsi bagaimana memilih sumber catu daya yang digunakan oleh DT-SENSE THERMOPILE ARRAY SENSOR melalui jumper PS Select:

| Jumper PS Select J4 | Sumber Catu Daya                   |
|---------------------|------------------------------------|
| 1 2<br>             | J1 Dengan regulator (9 - 12 V)     |
| 1 2<br>             | J2 Tanpa regulator (4,8 V - 5,4 V) |

#### Penting !



Jika J1 digunakan sebagai jalur catu daya input (J4 terpasang), maka J2 tidak boleh digunakan sebagai jalur catu daya input.

Jika J2 digunakan sebagai jalur catu daya input, maka J4 harus dilepas dan J1 tidak boleh digunakan sebagai jalur catu daya input.

Konektor INTERFACE (J8 dan J3) masing-masing berfungsi sebagai konektor antarmuka I<sup>2</sup>C dan antarmuka UART TTL.

| Konektor | Pin | Nama | Fungsi                                   |
|----------|-----|------|--|
| J8       | 1   | GND  | Titik referensi                          |
|          | 2   | SDA  | I <sup>2</sup> C-bus data input / output |
|          | 3   | SCL  | I <sup>2</sup> C-bus clock input         |
| J3       | 1   | TXD  | Output serial level TTL dari modul       |
|          | 2   | RXD  | Input serial level TTL ke modul          |

Jumper PULL UP (J7) berfungsi untuk mengaktifkan resistor *pull-up* untuk pin SDA dan SCL pada antarmuka I<sup>2</sup>C.

| Jumper PULL UP J7  | Fungsi  |
|--|---|
|   | <i>Pull-up</i> tidak aktif<br>( <i>jumper</i> terlepas) |
|  | <i>Pull-up</i> aktif<br>( <i>jumper</i> terpasang)      |

**Penting !**

Apabila lebih dari satu modul dihubungkan pada I<sup>2</sup>C-bus maka *jumper* J7 (SCL/SDA) salah satu modul saja yang perlu dipasang.

Jumper ADDR (J9) berfungsi untuk mengatur alamat I<sup>2</sup>C dari modul DT-SENSE THERMOPILE ARRAY SENSOR.

| J9 (A2)<br>Pin 5-6 | J9 (A1)<br>Pin 3-4 | J9(A0)<br>Pin 1-2 | Alamat I <sup>2</sup> C       |                              |
|--------------------|--------------------|-------------------|-------------------------------|------------------------------|
|                    |                    |                   | Alamat Tulis I <sup>2</sup> C | Alamat Baca I <sup>2</sup> C |
| ■                  | ■                  | ■                 | E0H                           | E1H                          |
| ■                  | ■                  |                   | E2H                           | E3H                          |
| ■                  |                    | ■                 | E4H                           | E5H                          |
| ■                  |                    |                   | E6H                           | E7H                          |
|                    | ■                  | ■                 | E8H                           | E9H                          |
|                    | ■                  |                   | EAH                           | EBH                          |
|                    |                    | ■                 | ECH                           | EDH                          |
|                    |                    |                   | EEH                           | EFH                          |

Keterangan:

■ : *jumper* terpasang

Konektor SENSOR (J5) sebagai konektor untuk sensor *thermopile array sensor*.

| Pin     | Nama    | Fungsi                               |
|---------|---------|--------------------------------------|
| 2       | GND     | Titik referensi catu daya ke sensor  |
| 4       | VAmb    | Sinyal output dari <i>thermistor</i> |
| 5       | VSensor | Sinyal output dari <i>thermopile</i> |
| 7       | C       | Selektor input ke-3                  |
| 8       | AVCC    | Output tegangan catu daya ke sensor  |
| 9       | B       | Selektor input ke-2                  |
| 10      | A       | Selektor input ke-1                  |
| 1, 3, 6 | NC      | Tidak terhubung kemana-mana          |

Konektor SERVO1 (J10) sebagai konektor untuk motor servo standar 1. Motor servo digunakan sebagaiudukan sensor sehingga mata sensor dapat menyapu sudut 180° secara horisontal atau vertikal pada proses *autoscan*.

| Pin | Nama  | Fungsi                                   |
|-----|-------|--|
| 1   | GND   | Titik referensi catu daya ke motor servo |
| 2   | VCC   | Tegangan catu daya ke motor servo        |
| 3   | Servo | Output pulsa (1 - 2 ms) ke motor servo   |

Konektor SERVO2 (J11) sebagai konektor untuk motor servo standar 2. Motor servo ini tidak digunakan pada proses *autoscan* sehingga fungsinya lebih bebas. Motor servo ini dapat digunakan sebagaiudukan motor servo ke-1 sehingga mata sensor dapat menyapu sudut 180° untuk arah yang berbeda.

| Pin | Nama  | Fungsi                                   |
|-----|-------|--|
| 1   | GND   | Titik referensi catu daya ke motor servo |
| 2   | VCC   | Tegangan catu daya ke motor servo        |
| 3   | Servo | Output pulsa (1 - 2 ms) ke motor servo   |

### 3. PERANGKAT LUNAK DT-SENSE THERMOPILE ARRAY SENSOR

DT-SENSE THERMOPILE ARRAY SENSOR memiliki antarmuka UART TTL dan I<sup>2</sup>C yang dapat digunakan untuk menerima perintah atau mengirim data.

#### 3.1. ANTARMUKA UART TTL

Parameter komunikasi UART TTL adalah sebagai berikut:

- 38400 bps

- 8 data bit
- 1 stop bit
- tanpa *parity* bit
- tanpa *flow control*

Semua perintah yang dikirim melalui antarmuka UART TTL dimulai dengan mengirim 1 byte data yang berisi <nomor perintah> dan (jika diperlukan) 1 byte data parameter perintah.

Jika perintah yang telah dikirimkan merupakan perintah yang meminta data dari modul DT-SENSE THERMOPILE ARRAY SENSOR, maka DT-SENSE THERMOPILE ARRAY SENSOR akan mengirimkan data melalui jalur TX TTL.

Sebuah data parameter yang memiliki *range* lebih besar dari 255 desimal (lebih besar dari 1 byte) dikirim secara dua tahap. Satu byte data MSB dikirim lebih dahulu kemudian diikuti dengan data LSB. Misalnya parameter <PosisiServo> yang memiliki *range* 500 - 2500. Jika <PosisiServo> bernilai 1500 maka byte MSB yang dikirim/diterima adalah 5 dan byte LSB yang dikirim/diterima adalah 220 ((5x256)+220=1500).

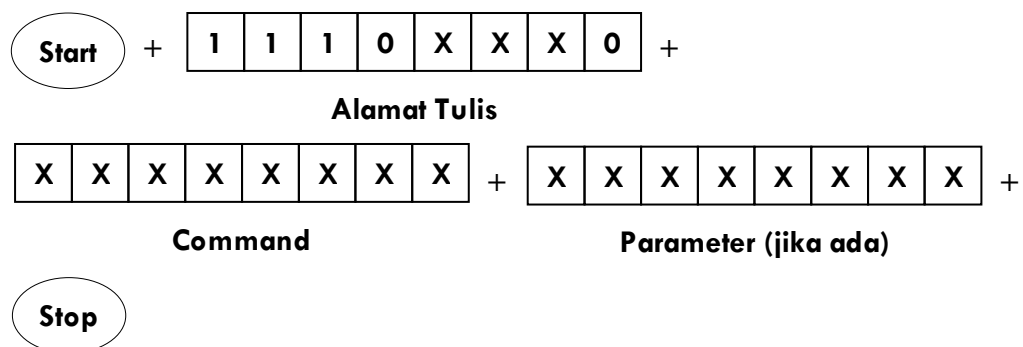
Perintah dan parameter yang bisa digunakan dapat dilihat pada **bagian 3.3**.

### 3.2. ANTARMUKA I<sup>2</sup>C

Modul DT-SENSE THERMOPILE ARRAY SENSOR memiliki antarmuka I<sup>2</sup>C. Pada antarmuka I<sup>2</sup>C ini, modul DT-SENSE THERMOPILE ARRAY SENSOR bertindak sebagai *slave* dengan alamat sesuai dengan telah ditentukan sebelumnya melalui pengaturan *jumper* (lihat **bagian 2.2**). Antarmuka I<sup>2</sup>C pada modul DT-SENSE THERMOPILE ARRAY SENSOR mendukung *bit rate* sampai dengan maksimum 100 kHz.

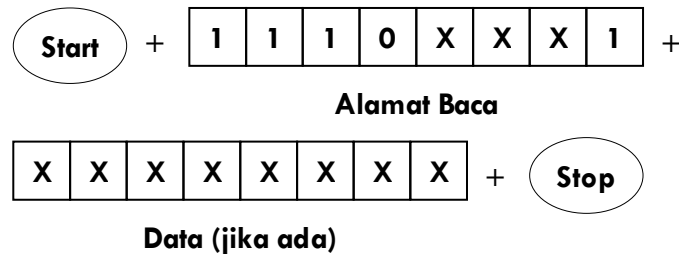
Semua perintah yang dikirim melalui antarmuka I<sup>2</sup>C diawali dengan **start condition** dan kemudian diikuti dengan pengiriman 1 byte alamat modul DT-SENSE THERMOPILE ARRAY SENSOR. Setelah pengiriman alamat, selanjutnya *master* harus mengirim 1 byte data yang berisi <nomor perintah> dan (jika diperlukan) 1 byte data parameter perintah. Selanjutnya, setelah seluruh parameter perintah telah dikirim, urutan perintah diakhiri dengan **stop condition**.

Berikut urutan yang harus dilakukan untuk mengirimkan perintah melalui antarmuka I<sup>2</sup>C.



Jika perintah yang telah dikirimkan merupakan perintah yang meminta data dari modul DT-SENSE THERMOPILE ARRAY SENSOR, maka data-data tersebut

dapat dibaca dengan menggunakan urutan perintah baca. Berikut urutan yang harus dilakukan untuk membaca data dari DT-SENSE THERMOPILE ARRAY SENSOR.



Sebuah data parameter yang memiliki *range* lebih besar dari 255 desimal (lebih besar dari 1 byte) dikirim secara dua tahap. Satu byte data MSB dikirim lebih dahulu kemudian diikuti dengan data LSB. Misalnya parameter <PosisiServo> yang memiliki *range* 500 - 2500. Jika <PosisiServo> bernilai 1500 maka byte MSB yang dikirim/diterima adalah 5 dan byte LSB yang dikirim/diterima adalah 220 ((5x256)+220=1500).

Perintah dan parameter yang bisa digunakan dapat dilihat pada **bagian 3.3**.

### 3.3. COMMAND SET

Berikut ini daftar lengkap perintah-perintah dalam antarmuka UART dan I<sup>2</sup>C.

#### 3.3.1. STATUS CHECK

|  |   |
|--|---|
| <b>Fungsi</b>                          | Untuk memeriksa status DT-SENSE THERMOPILE ARRAY SENSOR   |
| <b>Command</b>                         | 00H   |
| <b>Parameter</b>                       | -   |
| <b>Respon</b>                          | <status><br>06H → modul dalam posisi <i>idle</i> dan siap menerima perintah<br>15H → modul sedang sibuk dan tidak siap menerima perintah                          |
| <b>Delay antara Command dan Respon</b> | 10 ms   |
| <b>Keterangan</b>                      | <ul style="list-style-type: none"> <li>● Khusus untuk jalur komunikasi UART, modul tidak akan memberikan respon balasan (15H) saat modul sedang sibuk.</li> </ul> |

Contoh dengan antarmuka UART:

```
User       : 00H
DT-SENSE  : <status>
```

Berikut ini contoh *pseudo code* C untuk menggunakan perintah ini dengan antarmuka I<sup>2</sup>C (misalkan alamat I<sup>2</sup>C = E0H):

```

i2c_start();           // Start Condition
i2c_write(0xE0);      // Tulis ke modul THERMOPILE ARRAY
i2c_write(0x00);      // Perintah "Status Check"
i2c_stop();           // Stop Condition
```

```

delay_ms(10);           // delay 10 ms

i2c_start();           // Start Condition
i2c_write(0xE1);       // Baca ke modul THERMOPILE ARRAY
status = i2c_read(0);  // Data Status
i2c_stop();            // Stop Condition

```

**3.3.2. AUTOSCAN PEAK**

|  |  |
|--|--|
| <b>Fungsi</b>                          | Untuk menentukan letak temperatur paling tinggi  |
| <b>Command</b>                         | <b>01H</b>   |
| <b>Parameter</b>                       | <p>&lt;Posisi Awal Scan&gt;<br/> 50 - 250 → Posisi awal motor servo yang diinginkan sebelum memulai proses <i>autoscan</i>, dalam satuan 10ms.<br/> Range yang diperbolehkan 500 - 2500ms.</p> <p>&lt;Posisi Akhir Scan&gt;<br/> 50 - 250 → Posisi akhir motor servo yang diinginkan setelah melalui proses <i>autoscan</i>, dalam satuan 10ms.<br/> Range yang diperbolehkan 500 - 2500ms.</p> <p>&lt;Interval Posisi&gt;<br/> 25 - 200 → Interval perubahan posisi motor servo pada proses <i>autoscan</i>, dalam satuan ms.<br/> Range yang diperbolehkan 25 - 200ms.</p> <p>&lt;Tunda&gt;<br/> 0 - 255 → Waktu jeda sebelum melakukan <i>sampling</i> data dan mengubah posisi motor servo pada proses <i>autoscan</i>, dalam satuan 10ms.<br/> Range yang diperbolehkan 0 - 2550ms.</p> |
| <b>Respon</b>                          | -  |
| <b>Delay antara Command dan Respon</b> | -  |
| <b>Keterangan</b>                      | <ul style="list-style-type: none"> <li>● Pada proses ini, konektor servo yang digunakan adalah konektor servo ke-1.</li> <li>● Arah putaran motor servo yang terhubung pada konektor servo ke-1 (CW/CCW) pada saat proses <i>autoscan</i> ditentukan melalui variabel &lt;Posisi Awal Scan&gt; dan &lt;Posisi Akhir Scan&gt;.</li> <li>● Setelah selesai melakukan <i>scanning</i>, motor servo akan bergerak ke posisi data temperatur paling tinggi (puncak).</li> </ul>   |

Contoh dengan antarmuka UART untuk menentukan letak temperatur tertinggi antara posisi motor servo 1000 - 2000ms (atau 0° - 180° pada motor servo standar) dengan interval sekitar 4,5° (25ms) dan waktu tunda setiap perubahan posisi 100ms:

User : 01H 64H C8H 19H 0AH  
DT-SENSE : -

Berikut ini contoh *pseudo code* C untuk menggunakan perintah ini dengan antarmuka I<sup>2</sup>C (misalkan alamat I<sup>2</sup>C = E0H):

```

i2c_start();           // Start Condition
i2c_write(0xE0);      // Tulis ke modul THERMOPILE ARRAY
i2c_write(0x01);      // Perintah "Autoscan Peak"
i2c_write(0x64);      // Posisi awal scan (1000 ms)
i2c_write(0xC8);      // Posisi akhir scan (2000 ms)
i2c_write(0x19);      // increment/decrement (25 ms)
i2c_write(0x0A);      // waktu tunda (100 ms)
i2c_stop();           // Stop Condition

```

### 3.3.3. AUTOSCAN VALLEY

|  |  |
|--|--|
| <b>Fungsi</b>                          | Untuk menentukan letak temperatur paling rendah  |
| <b>Command</b>                         | <b>02H</b>   |
| <b>Parameter</b>                       | <p>&lt;Posisi Awal Scan&gt;<br/> 50 - 250 → Posisi awal motor servo yang diinginkan sebelum memulai proses <i>autoscan</i>, dalam satuan 10ms.<br/> Range yang diperbolehkan 500 - 2500ms.</p> <p>&lt;Posisi Akhir Scan&gt;<br/> 50 - 250 → Posisi akhir motor servo yang diinginkan setelah melalui proses <i>autoscan</i>, dalam satuan 10ms.<br/> Range yang diperbolehkan 500 - 2500ms.</p> <p>&lt;Interval Posisi&gt;<br/> 25 - 200 → Interval perubahan posisi motor servo pada proses <i>autoscan</i>, dalam satuan ms.<br/> Range yang diperbolehkan 25 - 200ms.</p> <p>&lt;Tunda&gt;<br/> 0 - 255 → Waktu jeda sebelum melakukan <i>sampling</i> data dan mengubah posisi motor servo pada proses <i>autoscan</i>, dalam satuan 10ms.<br/> Range yang diperbolehkan 0 - 2550ms.</p> |
| <b>Respon</b>                          | -  |
| <b>Delay antara Command dan Respon</b> | -  |
| <b>Keterangan</b>                      | <ul style="list-style-type: none"> <li>● Pada proses ini, konektor servo yang digunakan adalah konektor servo ke-1.</li> <li>● Arah putaran motor servo yang terhubung pada konektor servo ke-1 (CW/CCW) pada saat proses <i>autoscan</i> ditentukan melalui variabel &lt;Posisi Awal Scan&gt; dan &lt;Posisi Akhir Scan&gt;.</li> <li>● Setelah selesai melakukan <i>scanning</i>, motor servo akan bergerak ke posisi data temperatur paling rendah (lembah).</li> </ul>   |

Contoh dengan antarmuka UART untuk menentukan letak temperatur terendah antara posisi motor servo 1000 - 2000ms (atau 0° - 180° pada motor servo

standar) dengan interval sekitar 4,5° (25ms) dan waktu tunda setiap perubahan posisi 100ms:

User : 02H 64H C8H 19H 0AH  
DT-SENSE : -

Berikut ini contoh *pseudo code* C untuk menggunakan perintah ini dengan antarmuka I<sup>2</sup>C (misalkan alamat I<sup>2</sup>C = E0H):

```
i2c_start();           // Start Condition
i2c_write(0xE0);       // Tulis ke modul THERMOPILE ARRAY
i2c_write(0x02);       // Perintah "Autoscan Valley"
i2c_write(0x64);       // Posisi awal scan (1000 ms)
i2c_write(0xC8);       // Posisi akhir scan (2000 ms)
i2c_write(0x19);       // increment/decrement (25 ms)
i2c_write(0x0A);       // waktu tunda (100 ms)
i2c_stop();            // Stop Condition
```

### 3.3.4. AUTOSCAN HIGHER

|  |   |
|--|---|
| <b>Fungsi</b>                          | Untuk menentukan letak temperatur yang lebih tinggi dari batas  |
| <b>Command</b>                         | <b>03H</b>  |
| <b>Parameter</b>                       | <p>&lt;Posisi Awal Scan&gt;<br/>           50 - 250 → Posisi awal motor servo yang diinginkan sebelum memulai proses <i>autoscan</i>, dalam satuan 10ms.<br/>           Range yang diperbolehkan 500 - 2500ms.</p> <p>&lt;Posisi Akhir Scan&gt;<br/>           50 - 250 → Posisi akhir motor servo yang diinginkan setelah melalui proses <i>autoscan</i>, dalam satuan 10ms.<br/>           Range yang diperbolehkan 500 - 2500ms.</p> <p>&lt;Interval Posisi&gt;<br/>           25 - 200 → Interval perubahan posisi motor servo pada proses <i>autoscan</i>, dalam satuan ms.<br/>           Range yang diperbolehkan 25 - 200ms.</p> <p>&lt;Tunda&gt;<br/>           0 - 255 → Waktu jeda sebelum melakukan <i>sampling</i> data dan mengubah posisi motor servo pada proses <i>autoscan</i>, dalam satuan 10ms.<br/>           Range yang diperbolehkan 0 - 2550ms.</p> <p>&lt;Batas&gt;<br/>           0 - 255 → Batas temperatur yang digunakan sebagai syarat pembanding, dalam satuan °C.<br/>           Range yang diperbolehkan 0 - 255°C.</p> |
| <b>Respon</b>                          | -   |
| <b>Delay antara Command dan Respon</b> | -   |
| <b>Keterangan</b>                      | <ul style="list-style-type: none"> <li>● Pada proses ini, konektor servo yang digunakan adalah konektor servo ke-1.</li> <li>● Arah putaran motor servo yang terhubung pada konektor servo ke-1 (CW/CCW) pada saat proses <i>autoscan</i> ditentukan melalui variabel &lt;Posisi Awal Scan&gt; dan &lt;Posisi Akhir Scan&gt;.</li> <li>● Begitu menemukan temperatur yang lebih tinggi daripada temperatur batas, motor servo akan berhenti dan proses <i>autoscan</i> selesai.</li> </ul>  |

Contoh dengan antarmuka UART untuk menentukan letak temperatur lebih tinggi dari 50°C antara posisi motor servo 1000 - 2000ms (atau 0° - 180° pada motor servo standar) dengan interval sekitar 45ms dan waktu tunda setiap perubahan posisi 0ms:

```
User       : 03H 64H C8H 2DH 00H 32H
DT-SENSE  : -
```

Berikut ini contoh *pseudo code* C untuk menggunakan perintah ini dengan antarmuka I2C (misalkan alamat I2C = E0H):

```

i2c_start();           // Start Condition
i2c_write(0xE0);      // Tulis ke modul THERMOPILE ARRAY
i2c_write(0x03);      // Perintah "Autoscan Higher"
i2c_write(0x64);      // Posisi awal scan (1000 ms)
i2c_write(0xC8);      // Posisi akhir scan (2000 ms)
i2c_write(0x2D);      // increment/decrement (45 ms)
i2c_write(0x00);      // waktu tunda (0 ms)
i2c_write(0x32);      // threshold temperatur (50 °Celcius)
i2c_stop();           // Stop Condition

```

### 3.3.5. AUTOSCAN LOWER

|  |   |
|--|---|
| <b>Fungsi</b>                          | Untuk menentukan letak temperatur yang lebih rendah dari batas  |
| <b>Command</b>                         | <b>04H</b>  |
| <b>Parameter</b>                       | <p>&lt;Posisi Awal Scan&gt;<br/> 50 - 250 → Posisi awal motor servo yang diinginkan sebelum memulai proses <i>autoscan</i>, dalam satuan 10ms.<br/> Range yang diperbolehkan 500 - 2500ms.</p> <p>&lt;Posisi Akhir Scan&gt;<br/> 50 - 250 → Posisi akhir motor servo yang diinginkan setelah melalui proses <i>autoscan</i>, dalam satuan 10ms.<br/> Range yang diperbolehkan 500 - 2500ms.</p> <p>&lt;Interval Posisi&gt;<br/> 25 - 200 → Interval perubahan posisi motor servo pada proses <i>autoscan</i>, dalam satuan ms.<br/> Range yang diperbolehkan 25 - 200ms.</p> <p>&lt;Tunda&gt;<br/> 0 - 255 → Waktu jeda sebelum melakukan <i>sampling</i> data dan mengubah posisi motor servo pada proses <i>autoscan</i>, dalam satuan 10ms.<br/> Range yang diperbolehkan 0 - 2550ms.</p> <p>&lt;Batas&gt;<br/> 0 - 255 → Batas temperatur yang digunakan sebagai syarat pembanding, dalam satuan °C.<br/> Range yang diperbolehkan 0 - 255°C.</p> |
| <b>Respon</b>                          | -   |
| <b>Delay antara Command dan Respon</b> | -   |
| <b>Keterangan</b>                      | <ul style="list-style-type: none"> <li>● Pada proses ini, konektor servo yang digunakan adalah konektor servo ke-1.</li> <li>● Arah putaran motor servo yang terhubung pada konektor servo ke-1 (CW/CCW) pada saat proses <i>autoscan</i> ditentukan melalui variabel &lt;Posisi Awal Scan&gt; dan &lt;Posisi Akhir Scan&gt;.</li> <li>● Begitu menemukan temperatur yang lebih rendah daripada temperatur batas, motor servo akan berhenti dan proses <i>autoscan</i> selesai.</li> </ul>  |

Contoh dengan antarmuka UART untuk menentukan letak temperatur lebih rendah dari 15°C antara posisi motor servo 1000 - 2000ms (atau 0° - 180° pada motor servo standar) dengan interval sekitar 45ms dan waktu tunda setiap perubahan posisi Oms:

User : 04H 64H C8H 2DH 00H 0FH  
DT-SENSE : -

Berikut ini contoh *pseudo code* C untuk menggunakan perintah ini dengan antarmuka I<sup>2</sup>C (misalkan alamat I<sup>2</sup>C = E0H):

```

i2c_start();           // Start Condition
i2c_write(0xE0);      // Tulis ke modul THERMOPILE ARRAY
i2c_write(0x04);      // Perintah "Autoscan Lower"
i2c_write(0x64);      // Posisi awal scan (1000 ms)
i2c_write(0xC8);      // Posisi akhir scan (2000 ms)
i2c_write(0x2D);      // increment/decrement (45 ms)
i2c_write(0x00);      // waktu tunda (0 ms)
i2c_write(0x0F);      // threshold temperatur (15 °Celcius)
i2c_stop();           // Stop Condition

```

### 3.3.6. READ AUTOSCAN

|  |   |
|--|---|
| <b>Fungsi</b>                          | Untuk membaca hasil <i>autoscan</i>   |
| <b>Command</b>                         | <b>05H</b>  |
| <b>Parameter</b>                       | -   |
| <b>Respon</b>                          | <p>&lt;maxVal&gt;<br/>0 - 255 → Nilai temperatur hasil proses <i>autoscan</i> terakhir (Peak, Valley, Higher, Lower), dalam satuan °C.<br/>Range yang diperbolehkan 0 - 255°C.</p> <p>&lt;maxX&gt;<br/>0 - 255 → Jumlah <i>increment/decrement</i> yang telah dilakukan dari mulai &lt;Posisi Awal Scan&gt; sampai saat menemukan posisi akhir <i>autoscan</i>.<br/>Range yang diperbolehkan 0 - 255 kali.</p> <p>&lt;maxY&gt;<br/>0 - 7 → Kanal sensor <i>thermopile</i> yang menangkap temperatur tertinggi/terendah/lebih tinggi dari batas/lebih rendah dari batas pada proses <i>autoscan</i>.<br/>Range yang diperbolehkan kanal 0 - 7.</p> |
| <b>Delay antara Command dan Respon</b> | 10 ms   |
| <b>Keterangan</b>                      | -   |

Contoh dengan antarmuka UART:

User : 05H  
DT-SENSE : <maxVal> <maxX> <maxY>

Berikut ini contoh *pseudo code* C untuk menggunakan perintah ini dengan antarmuka I<sup>2</sup>C (misalkan alamat I<sup>2</sup>C = E0H):

```

i2c_start();           // Start Condition
i2c_write(0xE0);      // Tulis ke modul THERMOPILE ARRAY
i2c_write(0x05);      // Perintah "Read Autoscan"
i2c_stop();           // Stop Condition

delay_ms(10);         // delay 10 ms

i2c_start();           // Start Condition
i2c_write(0xE1);      // Baca ke modul THERMOPILE ARRAY
maxVal = i2c_read(1); // nilai temperatur
maxX    = i2c_read(1); // jumlah increment/decrement
maxY    = i2c_read(0); // kanal sensor
i2c_stop();           // Stop Condition

```

**3.3.7. SET SERVO 1**

|  |   |
|--|---|
| <b>Fungsi</b>                          | Untuk mengatur posisi motor servo 1   |
| <b>Command</b>                         | <b>06H</b>  |
| <b>Parameter</b>                       | <Posisi><br>500 - 2500 → Posisi motor servo yang diinginkan, dalam satuan ms.<br>Range yang diperbolehkan 500 - 2500ms. |
| <b>Respon</b>                          | -   |
| <b>Delay antara Command dan Respon</b> | -   |
| <b>Keterangan</b>                      | ● Perintah ini digunakan untuk mengatur posisi servo yang terhubung pada konektor servo ke-1.                           |

Contoh dengan antarmuka UART untuk menggerakkan motor servo 1 ke posisi 1500ms (atau posisi tengah pada motor servo standar):

User : 06H 05H DCH  
DT-SENSE : -

Berikut ini contoh *pseudo code* C untuk menggunakan perintah ini dengan antarmuka I<sup>2</sup>C (misalkan alamat I<sup>2</sup>C = E0H):

```

i2c_start();           // Start Condition
i2c_write(0xE0);      // Tulis ke modul THERMOPILE ARRAY
i2c_write(0x06);      // Perintah "Set Servo 1"
i2c_write(0x05);      // MSB dari 1500 ms
i2c_write(0xDC);      // LSB dari 1500 ms
i2c_stop();           // Stop Condition

```

### 3.3.8. SET SERVO 2

|  |   |
|--|---|
| <b>Fungsi</b>                          | Untuk mengatur posisi motor servo 2   |
| <b>Command</b>                         | <b>07H</b>  |
| <b>Parameter</b>                       | <Posisi><br>500 - 2500 → Posisi motor servo yang diinginkan, dalam satuan ms.<br>Range yang diperbolehkan 500 - 2500ms. |
| <b>Respon</b>                          | -   |
| <b>Delay antara Command dan Respon</b> | -   |
| <b>Keterangan</b>                      | ● Perintah ini digunakan untuk mengatur posisi servo yang terhubung pada konektor servo ke-2.                           |

Contoh dengan antarmuka UART untuk menggerakkan motor servo 2 ke posisi 1500ms (atau posisi tengah pada motor servo standar):

User : 07H 05H DCH  
DT-SENSE : -

Berikut ini contoh *pseudo code* C untuk menggunakan perintah ini dengan antarmuka I<sup>2</sup>C (misalkan alamat I<sup>2</sup>C = E0H):

```

i2c_start();           // Start Condition
i2c_write(0xE0);      // Tulis ke modul THERMOPILE ARRAY
i2c_write(0x07);      // Perintah "Set Servo 2"
i2c_write(0x05);      // MSB dari 1500 ms
i2c_write(0xDC);      // LSB dari 1500 ms
i2c_stop();           // Stop Condition

```

### 3.3.9. GET SERVO 1

|  |   |
|--|---|
| <b>Fungsi</b>                          | Untuk membaca posisi motor servo 1  |
| <b>Command</b>                         | <b>08H</b>  |
| <b>Parameter</b>                       | -   |
| <b>Respon</b>                          | <PosisiServo><br>500 - 2500 → Posisi motor servo sekarang, dalam satuan ms.<br>Range yang diperbolehkan 500 - 2500ms.                     |
| <b>Delay antara Command dan Respon</b> | 15 ms   |
| <b>Keterangan</b>                      | ● Perintah ini digunakan untuk mengetahui posisi servo yang terhubung pada konektor servo ke-1, misalkan setelah proses <i>autoscan</i> . |

Contoh dengan antarmuka UART:

User : 08H  
DT-SENSE : <PosisiServoMSB> <PosisiServoLSB>

Berikut ini contoh *pseudo code* C untuk menggunakan perintah ini dengan antarmuka I<sup>2</sup>C (misalkan alamat I<sup>2</sup>C = E0H):

```

i2c_start();           // Start Condition
i2c_write(0xE0);      // Tulis ke modul THERMOPILE ARRAY
i2c_write(0x08);      // Perintah "Get Servo 1"
i2c_stop();           // Stop Condition

delay_ms(15);         // delay 15 ms

i2c_start();           // Start Condition
i2c_write(0xE1);      // Baca ke modul THERMOPILE ARRAY
posMSB = i2c_read(1); // Data posisi servo MSB
posLSB = i2c_read(0); // Data posisi servo LSB
i2c_stop();           // Stop Condition

```

### 3.3.10. GET SERVO 2

|  |   |
|--|---|
| <b>Fungsi</b>                          | Untuk membaca posisi motor servo 2  |
| <b>Command</b>                         | <b>09H</b>  |
| <b>Parameter</b>                       | -   |
| <b>Respon</b>                          | <PosisiServo><br>500 - 2500 → Posisi motor servo sekarang, dalam satuan ms.<br>Range yang diperbolehkan 500 - 2500ms.                     |
| <b>Delay antara Command dan Respon</b> | 15 ms   |
| <b>Keterangan</b>                      | ● Perintah ini digunakan untuk mengetahui posisi servo yang terhubung pada konektor servo ke-2, misalkan setelah proses <i>autoscan</i> . |

Contoh dengan antarmuka UART:

User : 09H  
DT-SENSE : <PosisiServoMSB> <PosisiServoLSB>

Berikut ini contoh *pseudo code* C untuk menggunakan perintah ini dengan antarmuka I<sup>2</sup>C (misalkan alamat I<sup>2</sup>C = E0H):

```

i2c_start();           // Start Condition
i2c_write(0xE0);      // Tulis ke modul THERMOPILE ARRAY
i2c_write(0x09);      // Perintah "Get Servo 2"
i2c_stop();           // Stop Condition

delay_ms(15);         // delay 15 ms

i2c_start();           // Start Condition
i2c_write(0xE1);      // Baca ke modul THERMOPILE ARRAY
posMSB = i2c_read(1); // Data posisi servo MSB
posLSB = i2c_read(0); // Data posisi servo LSB
i2c_stop();           // Stop Condition

```

### 3.3.11. ENABLE/DISABLE SERVO 1

|  |   |
|--|---|
| <b>Fungsi</b>                          | Untuk mengaktifkan atau menon-aktifkan motor servo 1  |
| <b>Command</b>                         | <b>0AH</b>  |
| <b>Parameter</b>                       | <Status><br>0 → Menon-aktifkan motor servo 1<br>1 → Mengaktifkan motor servo 1  |
| <b>Respon</b>                          | -   |
| <b>Delay antara Command dan Respon</b> | -   |
| <b>Keterangan</b>                      | <ul style="list-style-type: none"><li>● Jika motor servo diaktifkan, maka motor servo akan selalu berusaha mempertahankan posisi sudutnya sesuai dengan yang diperintahkan.</li><li>● Status motor servo harus aktif untuk perintah-perintah yang menggunakan motor servo (perintah: Set Servo atau Autoscan).</li><li>● Jika motor servo dinon-aktifkan, maka motor servo tidak berfungsi dan dapat digerakkan dengan tangan.</li><li>● Perintah menon-aktifkan motor servo dapat digunakan untuk menghemat penggunaan catu daya.</li><li>● Status terakhir motor servo tidak disimpan pada EEPROM dan akan kembali menjadi aktif saat modul baru dinyalakan atau setelah reset.</li></ul> |

Contoh dengan antarmuka UART untuk mengaktifkan motor servo 1:

User : 0AH 01H  
DT-SENSE : -

Berikut ini contoh pseudo code C untuk menggunakan perintah ini dengan antarmuka I<sup>2</sup>C (misalkan alamat I<sup>2</sup>C = E0H):

```
i2c_start(); // Start Condition
i2c_write(0xE0); // Tulis ke modul THERMOPILE ARRAY
i2c_write(0x0A); // Perintah "Enable/Disable Servo 1"
i2c_write(0x01); // Status servo diset menjadi "enable"
i2c_stop(); // Stop Condition
```

### 3.3.12. ENABLE/DISABLE SERVO 2

|  |   |
|--|---|
| <b>Fungsi</b>                          | Untuk mengaktifkan atau menon-aktifkan motor servo 2  |
| <b>Command</b>                         | <b>0BH</b>  |
| <b>Parameter</b>                       | <Status><br>0 → Menon-aktifkan motor servo 2<br>1 → Mengaktifkan motor servo 2  |
| <b>Respon</b>                          | -   |
| <b>Delay antara Command dan Respon</b> | -   |
| <b>Keterangan</b>                      | <ul style="list-style-type: none"><li>● Jika motor servo diaktifkan, maka motor servo akan selalu berusaha mempertahankan posisi sudutnya sesuai dengan yang diperintahkan.</li><li>● Status motor servo harus aktif untuk perintah-perintah yang menggunakan motor servo (perintah: Set Servo).</li><li>● Jika motor servo dinon-aktifkan, maka motor servo tidak berfungsi dan dapat digerakkan dengan tangan.</li><li>● Perintah menon-aktifkan motor servo dapat digunakan untuk menghemat penggunaan catu daya.</li><li>● Status terakhir motor servo tidak disimpan pada EEPROM dan akan kembali menjadi aktif saat modul baru dinyalakan atau setelah reset.</li></ul> |

Contoh dengan antarmuka UART untuk menon-aktifkan motor servo 2:

User : 0BH 00H  
DT-SENSE : -

Berikut ini contoh *pseudo code* C untuk menggunakan perintah ini dengan antarmuka I<sup>2</sup>C (misalkan alamat I<sup>2</sup>C = E0H):

```
i2c_start(); // Start Condition
i2c_write(0xE0); // Tulis ke modul THERMOPILE ARRAY
i2c_write(0x0B); // Perintah "Enable/Disable Servo 2"
i2c_write(0x00); // Status servo diset menjadi "disable"
i2c_stop(); // Stop Condition
```

### 3.3.13. ENABLE/DISABLE CHANNEL

|  |  |
|--|--|
| <b>Fungsi</b>                          | Untuk mengaktifkan atau menon-aktifkan kanal sensor <i>thermopile</i>  |
| <b>Command</b>                         | <b>OCH</b>   |
| <b>Parameter</b>                       | <StatusKanalSensor><br>bit 0 → Status kanal sensor 0 (LSB)<br>bit 1 → Status kanal sensor 1<br>bit 2 → Status kanal sensor 2<br>bit 3 → Status kanal sensor 3<br>bit 4 → Status kanal sensor 4<br>bit 5 → Status kanal sensor 5<br>bit 6 → Status kanal sensor 6<br>bit 7 → Status kanal sensor 7 (MSB)  |
| <b>Respon</b>                          | -  |
| <b>Delay antara Command dan Respon</b> | -  |
| <b>Keterangan</b>                      | <ul style="list-style-type: none"> <li>● Jika sebuah bit yang mewakili status kanal sensor diberi logika 1, maka kanal sensor menjadi aktif dan dapat mendeteksi temperatur di depannya.</li> <li>● Jika sebuah bit yang mewakili status kanal sensor diberi logika 0, maka kanal sensor menjadi non-aktif dan hasil pembacaan sensor akan selalu bernilai 0.</li> <li>● Pada proses <i>autoscan</i>, jika sebuah kanal sensor tidak aktif, maka proses <i>autoscan</i> akan mengabaikan nilai temperatur yang dihasilkan kanal sensor tersebut.</li> <li>● Status terakhir kanal sensor <i>thermopile</i> tidak disimpan pada EEPROM dan seluruh kanal akan kembali menjadi aktif saat modul baru dinyalakan atau setelah reset.</li> </ul> |

Contoh dengan antarmuka UART untuk mengaktifkan kanal sensor 0 hingga 3 serta menon-aktifkan kanal sensor 4 hingga 7:

```
User          :      OCH 0FH
DT-SENSE     :      -
```

Berikut ini contoh *pseudo code* C untuk menggunakan perintah ini dengan antarmuka I<sup>2</sup>C (misalkan alamat I<sup>2</sup>C = E0H):

```
i2c_start();           // Start Condition
i2c_write(0xE0);      // Tulis ke modul THERMOPILE ARRAY
i2c_write(0x0C);      // Perintah "Enable/Disable Channel"
i2c_write(0x0F);      // Status kanal sensor yg diinginkan
i2c_stop();           // Stop Condition
```

### 3.3.14. READ ALL CHANNELS

|  |   |
|--|---|
| <b>Fungsi</b>                          | Untuk membaca data temperatur semua kanal sensor  |
| <b>Command</b>                         | <b>ODH</b>  |
| <b>Parameter</b>                       | -   |
| <b>Respon</b>                          | 8 byte <AllTempData><br>0 - 255 → Nilai temperatur data kanal, dalam satuan °C.<br>Range yang diperbolehkan 0 - 255°C.  |
| <b>Delay antara Command dan Respon</b> | 100 ms  |
| <b>Keterangan</b>                      | <ul style="list-style-type: none"><li>● Nilai temperatur kanal sensor ke-0 dikirimkan pertama, kemudian data kanal sensor ke-1, dan seterusnya sampai kanal sensor ke-7.</li><li>● Jika sebuah kanal tidak diaktifkan, maka hasil pembacaannya akan bernilai 0.</li></ul> |

Contoh dengan antarmuka UART:

User : ODH  
DT-SENSE : <AllTempData>

Berikut ini contoh *pseudo code* C untuk menggunakan perintah ini dengan antarmuka I<sup>2</sup>C (misalkan alamat I<sup>2</sup>C = E0H):

```
i2c_start(); // Start Condition
i2c_write(0xE0); // Tulis ke modul THERMOPILE ARRAY
i2c_write(0x0D); // Perintah "Read All Channel"
i2c_stop(); // Stop Condition

delay_ms(100); // delay 100 ms

i2c_start(); // Start Condition
i2c_write(0xE1); // Baca ke modul THERMOPILE ARRAY
kanal1 = i2c_read(1); // Data temperatur kanal ke-0
kanal2 = i2c_read(1); // Data temperatur kanal ke-1
kanal3 = i2c_read(1); // Data temperatur kanal ke-2
kanal4 = i2c_read(1); // Data temperatur kanal ke-3
kanal5 = i2c_read(1); // Data temperatur kanal ke-4
kanal6 = i2c_read(1); // Data temperatur kanal ke-5
kanal7 = i2c_read(1); // Data temperatur kanal ke-6
kanal8 = i2c_read(0); // Data temperatur kanal ke-7
i2c_stop(); // Stop Condition
```

### 3.3.15.READ AMBIENT

|  |  |
|--|--|
| <b>Fungsi</b>                          | Untuk membaca data temperatur <i>ambient</i> /sekitar  |
| <b>Command</b>                         | <b>30H</b>   |
| <b>Parameter</b>                       | -  |
| <b>Respon</b>                          | <TemperatureData><br>0 - 255 → Nilai temperatur <i>ambient</i> , dalam satuan °C.<br>Range yang diperbolehkan 0 - 255°C. |
| <b>Delay antara Command dan Respon</b> | 100 ms   |
| <b>Keterangan</b>                      | -  |

Contoh dengan antarmuka UART:

User : 30H  
DT-SENSE : <TemperatureData>

Berikut ini contoh *pseudo code* C untuk menggunakan perintah ini dengan antarmuka I<sup>2</sup>C (misalkan alamat I<sup>2</sup>C = E0H):

```

i2c_start();           // Start Condition
i2c_write(0xE0);       // Tulis ke modul THERMOPILE ARRAY
i2c_write(0x30);       // Perintah "Read Ambient"
i2c_stop();           // Stop Condition

delay_ms(100);        // delay 100 ms

i2c_start();           // Start Condition
i2c_write(0xE1);       // Baca ke modul THERMOPILE ARRAY
tAmb = i2c_read(0);    // Data temperatur
i2c_stop();           // Stop Condition

```

### 3.3.16.READ CHANNEL N-1

|  |   |
|--|---|
| <b>Fungsi</b>                          | Untuk membaca data temperatur kanal sensor n-1  |
| <b>Command</b>                         | <b>3nH</b>  |
| <b>Parameter</b>                       | -   |
| <b>Respon</b>                          | <TempDataN><br>0 - 255 → Nilai temperatur dari kanal n-1, dalam satuan °C.<br>Range yang diperbolehkan 0 - 255°C.   |
| <b>Delay antara Command dan Respon</b> | 100 ms  |
| <b>Keterangan</b>                      | <ul style="list-style-type: none"> <li>● n diganti dengan angka 1 untuk kanal ke-0, angka 2 untuk kanal ke-1, dan seterusnya sampai dengan angka 8 untuk kanal ke-7.</li> <li>● Jika sebuah kanal tidak diaktifkan, maka hasil pembacaannya akan bernilai 0.</li> </ul> |

Contoh dengan antarmuka UART untuk membaca data temperatur kanal 2:

User : 33H  
DT-SENSE : <TempDataN>

Berikut ini contoh *pseudo code* C untuk menggunakan perintah ini dengan antarmuka I<sup>2</sup>C (misalkan alamat I<sup>2</sup>C = E0H):

```

i2c_start();           // Start Condition
i2c_write(0xE0);      // Tulis ke modul THERMOPILE ARRAY
i2c_write(0x33);      // Perintah "Read Channel 2"
i2c_stop();           // Stop Condition

delay_ms(100);        // delay 100 ms

i2c_start();           // Start Condition
i2c_write(0xE1);      // Baca ke modul THERMOPILE ARRAY
kanal2 = i2c_read(0); // Data temperatur kanal ke-2
i2c_stop();           // Stop Condition

```

#### 4. PROSEDUR PENGUJIAN

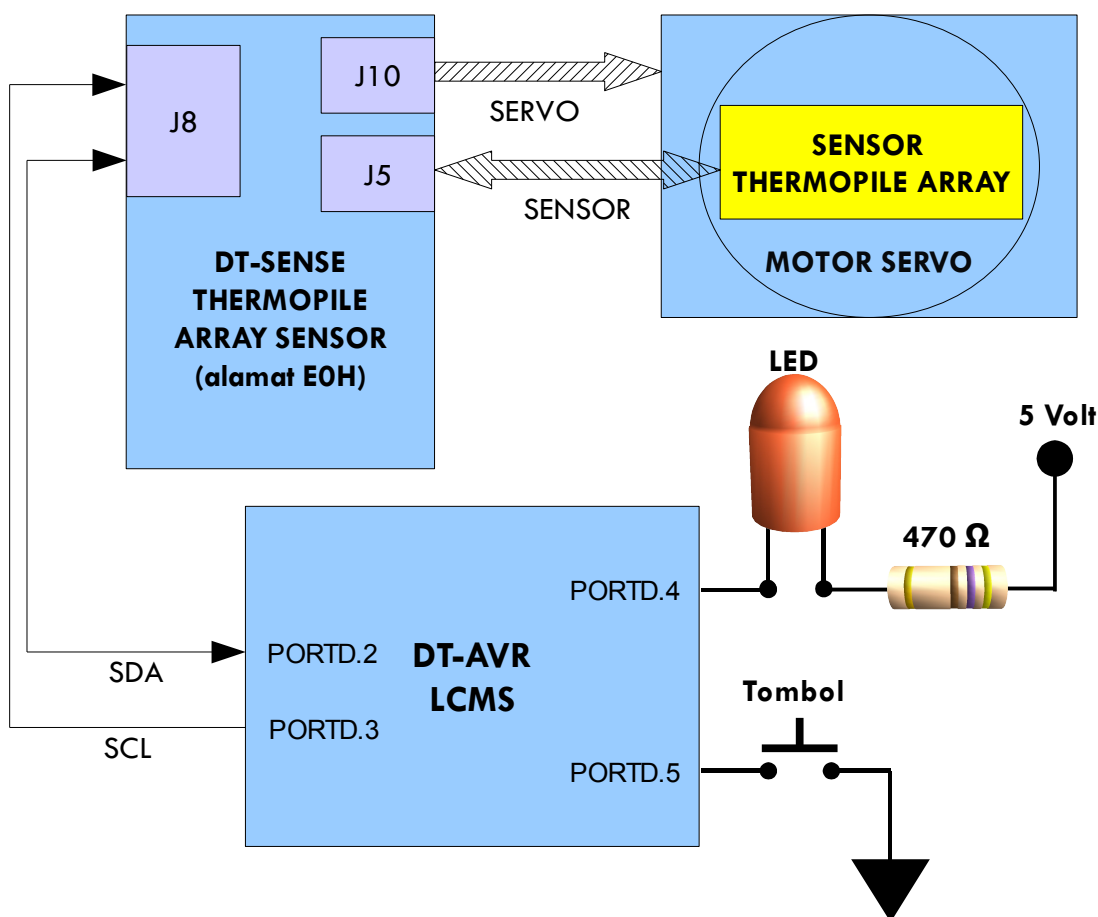
1. Pasang sensor ke motor servo sehingga sapuan sensor mencakup 180°.
2. Lepas *jumper* pada J4 sehingga sumber catu daya yang akan digunakan adalah 5 VDC.
3. Hubungkan servo pada konektor SERVO1 (J10).
4. Hubungkan sensor *thermopile* pada konektor SENSOR (J5).
5. Hubungkan sumber catu daya 5 VDC ke modul DT-SENSE THERMOPILE ARRAY SENSOR.
6. Jika modul DT-SENSE THERMOPILE ARRAY SENSOR berhasil melakukan *power-on reset* dengan baik, maka motor servo akan bergerak ke posisi *default*-nya.
7. Letakkan obyek yang memiliki temperatur cukup tinggi bila dibandingkan temperatur ruangan (misalkan solder, lilin, bohlam yang menyala, dsb.) dengan tinggi sesuai dengan tinggi mata sensor, dan letakkan pada jarak antara 5 - 40 cm dari sensor dan masih dalam jangkauan sapuan motor servo dan sensor.
8. Kirimkan perintah AUTOSCAN PEAK melalui antarmuka UART TTL.
9. Modul DT-SENSE THERMOPILE ARRAY SENSOR akan melakukan proses *autoscanning* dan jika berhasil mendeteksi obyek tersebut, maka mata sensor akan terarah menuju ke posisi obyek.

#### 5. CONTOH APLIKASI DAN PROGRAM

Sebagai contoh aplikasi, dimisalkan modul DT-SENSE THERMOPILE ARRAY SENSOR digunakan untuk mencari posisi obyek yang memiliki temperatur di atas 50°C dengan antarmuka I<sup>2</sup>C. Modul DT-AVR Low Cost Micro System (LCMS) digunakan sebagai *master*. DT-AVR LCMS bertugas untuk mengirimkan perintah AUTOSCAN HIGHER dan membaca hasilnya serta menyalakan LED jika obyek ditemukan. Berikut koneksi antara modul-modul yang digunakan:

| DT-AVR LCNS | Terhubung ke  |
|-------------|---|
| PORTD.2     | SDA DT-SENSE THERMOPILE ARRAY SENSOR                                |
| PORTD.3     | SCL DT-SENSE THERMOPILE ARRAY SENSOR                                |
| PORTD.4     | Terhubung ke rangkaian LED <i>active low</i>                        |
| PORTD.5     | Tombol <i>active low</i> (salah satu kakinya dihubungkan ke ground) |

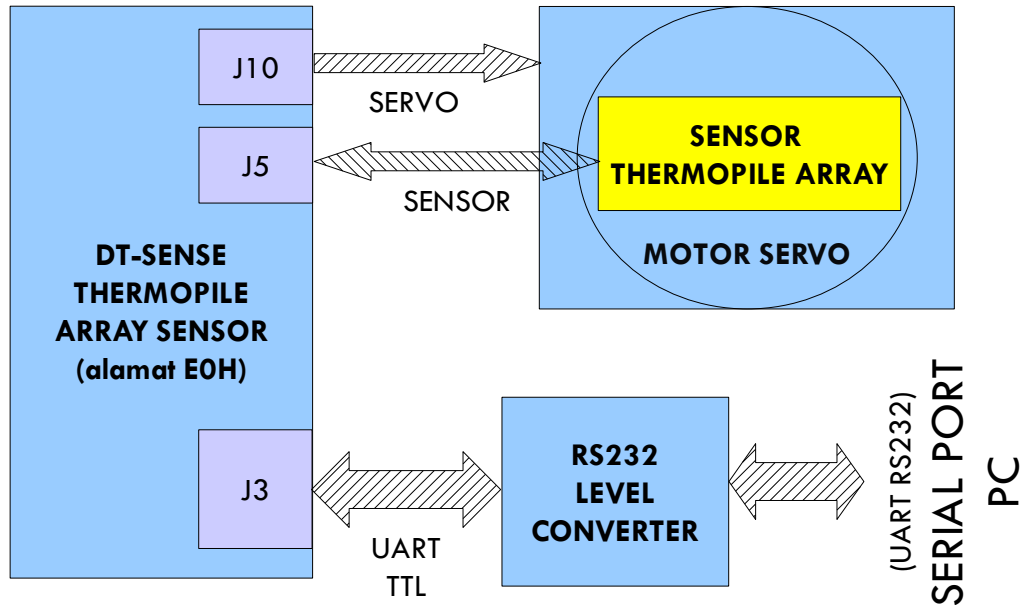
| DT-SENSE THERMOPILE ARRAY SENSOR | Terhubung ke |
|----------------------------------|--------------|
| J5                               | Sensor       |
| J10                              | Motor Servo  |



Sebagai contoh program untuk aplikasi di atas, pada CD yang disertakan pada saat pembelian modul DT-SENSE THERMOPILE ARRAY SENSOR disertakan program contoh\_i2c.c yang ditulis dengan menggunakan CodeVisionAVR 1.25.2 versi evaluasi.

Pada program tersebut, DT-AVR LCMS akan membaca kondisi tombol. Jika tombol ditekan, maka DT-AVR LCMS mengirimkan perintah AUTOSCAN HIGHER ke DT-SENSE THERMOPILE ARRAY SENSOR (alamat E0H). Setelah proses *autoscan* selesai dan ditemukan obyek yang memiliki temperatur lebih tinggi dari 50°C, maka DT-AVR LCMS akan mengirimkan perintah READ AUTOSCAN dan menyalakan LED. Setelah LED menyala selama 2 detik, DT-AVR LCMS akan mematikan LED dan kembali menunggu penekanan tombol lagi.

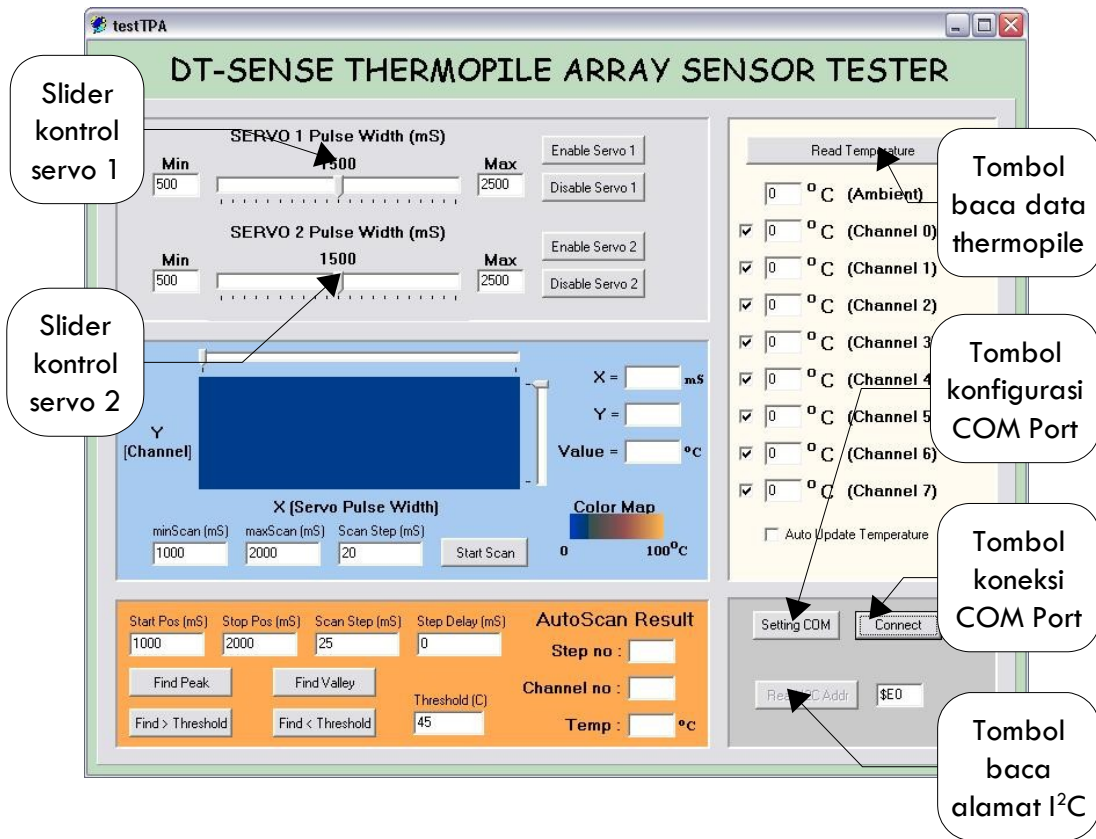
Sebagai contoh aplikasi lain, dimisalkan modul DT-SENSE THERMOPILE ARRAY SENSOR terhubung ke *Personal Computer* (PC) dan akan dikendalikan menggunakan perangkat lunak. Jalur komunikasi yang digunakan adalah jalur komunikasi UART. Oleh karena level tegangan UART pada modul DT-SENSE THERMOPILE ARRAY SENSOR dan PC berbeda, maka diperlukan perangkat UART Level Conveter. Berikut koneksi antara modul-modul yang digunakan:



Sebagai contoh program untuk aplikasi di atas, pada CD yang disertakan pada saat pembelian modul DT-SENSE THERMOPILE ARRAY SENSOR disertakan program “testTPA.exe” yang ditulis dengan menggunakan Delphi 7. Setelah modul DT-SENSE THERMOPILE ARRAY SENSOR terhubung ke PC serta catu daya telah menyala, jalankan perangkat lunak “testTPA.exe” pada PC. Pilih “**Setting COM**” untuk mengatur konfigurasi komunikasi serial yang akan digunakan. Pilih *port* sesuai dengan *port* yang digunakan pada PC dan atur konfigurasi lainnya menjadi **38400 bps, 8 data bits, 1 stop bits, no parity, no flow control**. Untuk mulai menguji fungsi-fungsi DT-SENSE THERMOPILE ARRAY SENSOR pilih tombol “**Connect**”.

Untuk menguji fungsi pengaturan servo pada konektor SERVO1, geser *slider* kontrol SERVO 1 ke kiri atau ke kanan. Demikian juga untuk menguji fungsi pengaturan servo pada konektor SERVO2, geser *slider* kontrol SERVO 2 ke kiri atau ke kanan.

Untuk menguji sensor *thermopile*, tekan tombol “**Read Temperature**”. Jika sensor dan koneksinya bekerja dengan baik, maka pada tampilan temperatur *ambient* akan ditampilkan temperatur sekitar dan pada tampilan temperatur tiap kanal akan ditampilkan data temperatur yang diperoleh oleh sensor (jika tidak ada obyek panas di depan sensor, maka nilai data masing-masing kanal akan mendekati temperatur sekitar).



◆ Terima Kasih atas kepercayaan Anda menggunakan produk kami, bila ada kesulitan, pertanyaan atau saran mengenai produk ini silakan menghubungi technical support kami :

[support@innovativeelectronics.com](mailto:support@innovativeelectronics.com)

**LAMPIRAN A.**  
**Skematik DT-SENSE THERMOPILE ARRAY SENSOR**

