

# SISTEM KOMUNIKASI NIRKABEL

## MODUL 6



# **PEMROGRAMAN API SINGLE BOARD COMPUTER (SBC) INTERNET OF THINGS BERBASIS PACKET TRACER**

---

Mochammad Zen Samsono Hadi, ST. MSc. Ph.D

# TOPIK BAHASAN

---

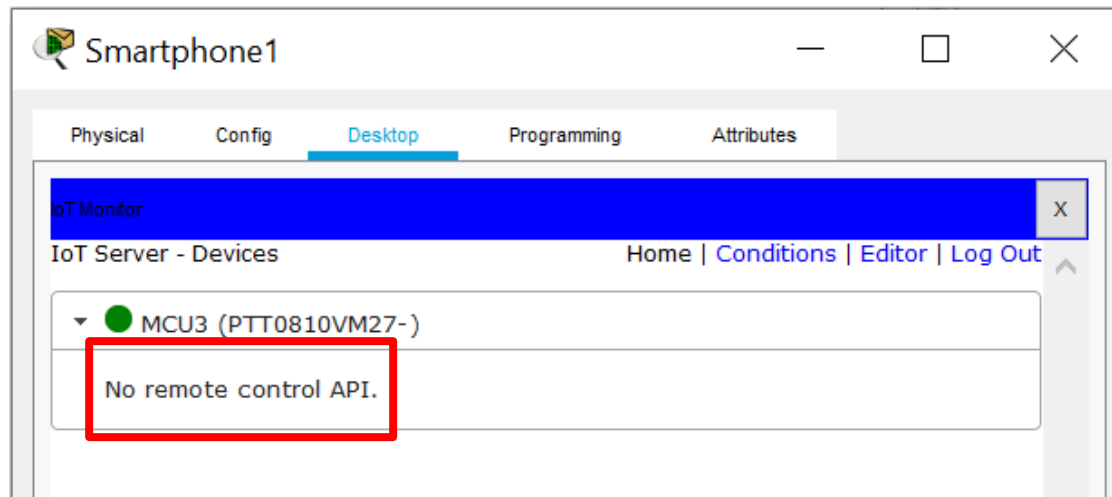
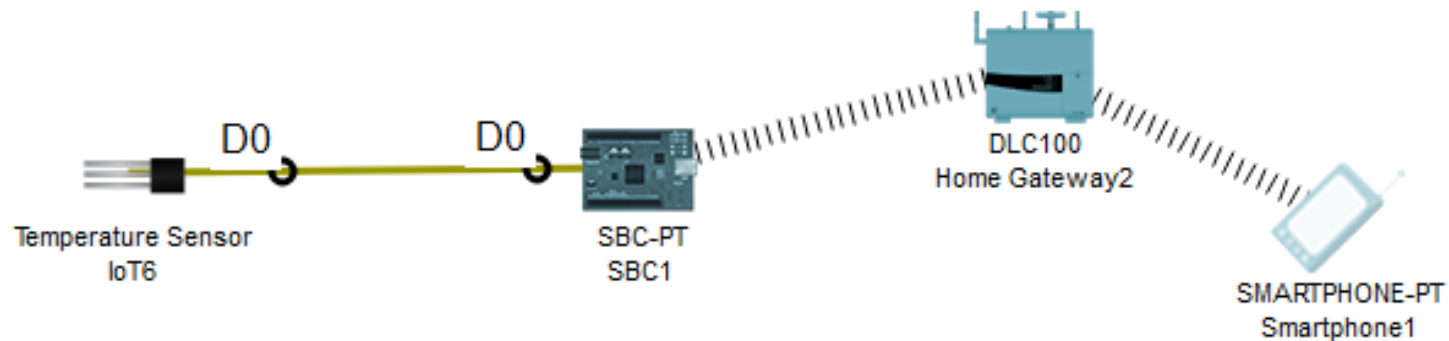
- SBC API PYTHON
- SETTING ENVIRONMENT

---

# SBC API PYTHON

# Topologi Jaringan

- Designlah jaringan seperti berikut:



# Python Programming di SBC

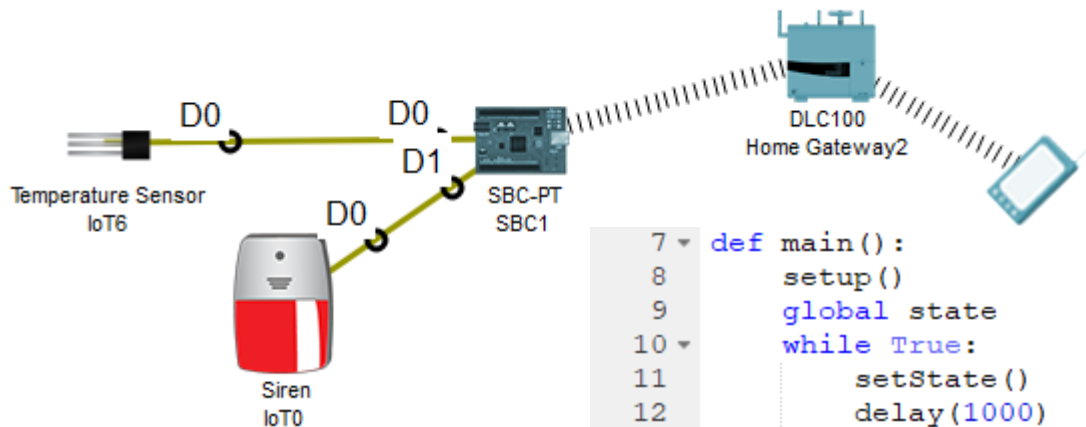
The image displays two windows from a development environment. The left window, titled 'SBC1', shows a Python script named 'main.py' with the following code:

```
1 from gpio import *
2 from time import *
3 from ioeclient import *
4
5 state = 0
6
7 def main():
8     setup()
9     global state
10    while True:
11        setState()
12        delay(1000)
13
14 def setup():
15     IoEClient.setup({
16         "type": "Microcontroller",
17         "states": [
18             {
19                 "name": "Temperature",
20                 "type": "number",
21                 "unit": "C"
22             }
23         ]
24     })
25
26 def setState():
27     temp = digitalRead(0);
28     print(temp);
29     IoEClient.reportStates(temp)
30
31 if __name__ == "__main__":
32     main()
```

The right window, titled 'Smartphone1', shows a web browser interface. The address bar contains 'http://192.168.25.1/home.html'. Below the browser, the 'IoT Server - Devices' section displays a table with the following data:

Device Name	Device Type	Temperature
SBC1 (PTT0810GK52-)	Microcontroller	582 C

# Dengan Aktuator



```
7 def main():
8     setup()
9     global state
10    while True:
11        setState()
12        delay(1000)
13
14    def setup():
15        IoEClient.setup({
16            "type": "Microcontroller",
17            "states": [
18                {
19                    "name": "Temperature",
20                    "type": "number",
21                    "unit": "C"
22                }
23            ]
24        })
25
26    def setState():
27        temp = digitalRead(0);
28        print(temp):
29        if (temp>660):
30            customWrite(1,1)
31        else:
32            customWrite(1,0)
33
34        IoEClient.reportStates(temp)
35
36 if __name__ == "__main__":
37     main()
```

---

# SETTING ENVIRONMENT

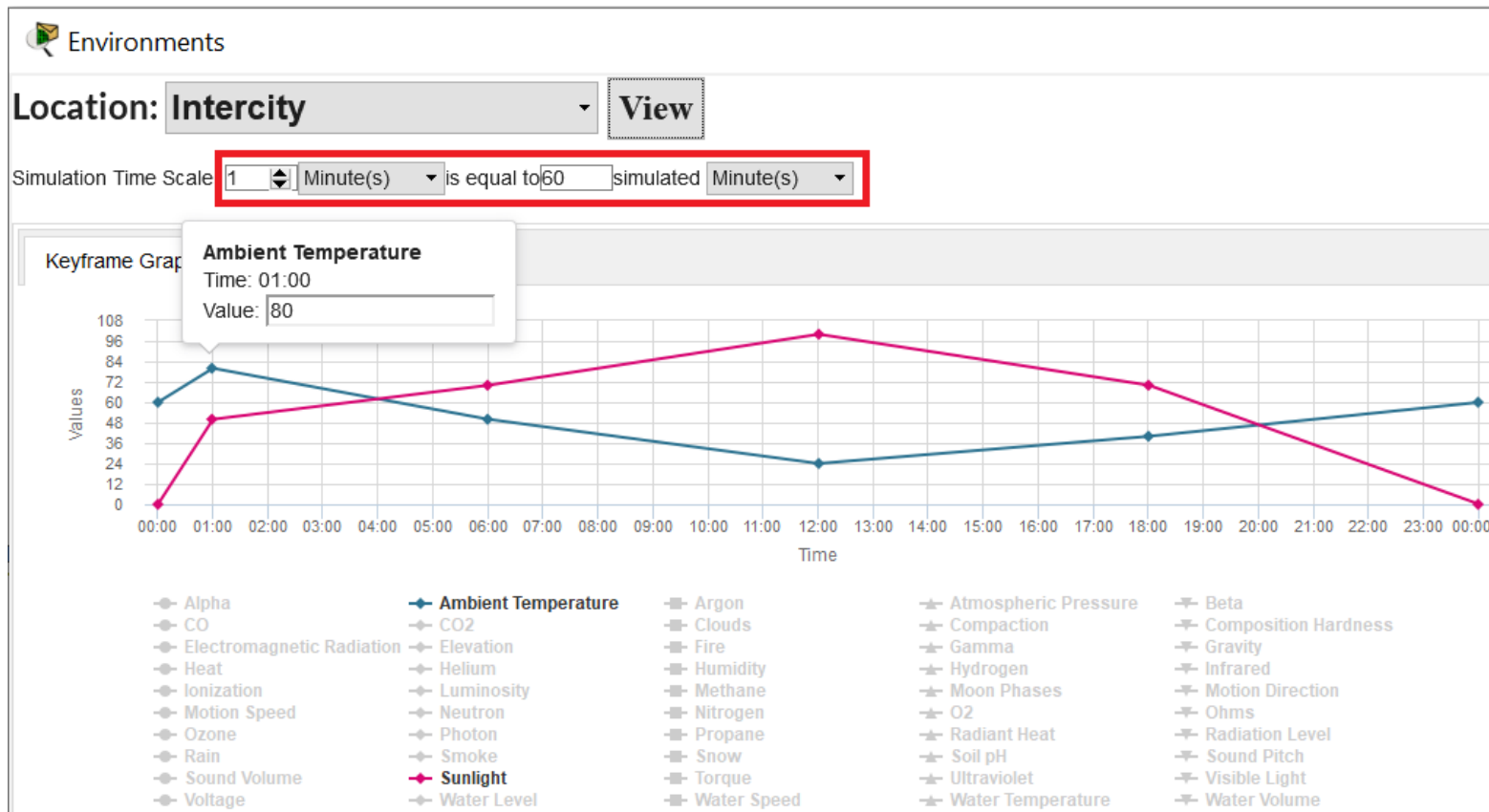
# Setting Environment

The screenshot shows the 'Environments' application interface. At the top right, there is a blue header bar with the text '[Root]' and a clock showing '00:25:00'. Below the header, the main content area is titled 'Environments' and features a close button (X) in the top right corner. The 'Location' is set to 'Intercity', with an 'Edit' button next to it. Below the location, the 'Current Time' is displayed as '00:24:00', also with an 'Edit' button and a 'Pause' button. A section titled 'Select an environment to show its chart.' includes a 'Filter...' input field, a 'Search' button, and a 'Reset' button. The main data area is organized into four columns of environmental parameters, each with a collapse/expand arrow (▾) to its left:

- Earth Physical Features**
  - Elevation: 22.00 m
  - Soil pH: 7.00 pH
- Gases**
  - Argon: 0.9340 %
  - CO: 0.00 %
  - CO<sub>2</sub>: 0.0360 %
  - He: 0.0005240 %
  - H: 0.00050 %
  - Methane: 0.000150 %
  - Nitrogen: 78.0840 %
  - O<sub>2</sub>: 20.9460 %
- Gravity**
  - Gravity: 9.80 m/s<sup>2</sup>
- Light (Sun)**
  - Electromagnetic Radiation: 0.00 %
  - Infrared: 0.00 %
  - Radiant Heat: 0.00 %
  - Sunlight: 20.00 %
  - Ultraviolet: 0.00 %
  - Visible: 0.00 %
- Other**
  - Atmospheric Pressure: 101.3250 kPa
- Radiation**
  - Level: 0.00 mrem
- Temperature**
  - Ambient Temperature: 68.00 C
- Water**
  - Clouds: 5.17 %
  - Humidity: 79.67 %
  - Rain: 0.03 cm
  - Snow: 0.00 cm
  - Water Level: 0.00 cm
- Wind**
  - Direction: 7.83 degree
  - Variance (gusts): 0.67 %
  - Speed: 0.00 kph



# Setting Temperature dan simulation time



# TUGAS

---

- Project IoT:
  - Tidak boleh Smart Home
  - Smart Industri, Smart Agriculture, dll
  - Harus mencakup: Sensor, Aktuator, RFID/BLE dan SBC
  - Terintegrasi dgn celluler network
  - Buat IoT Server sendiri dan DNS untuk mengaksesnya
- Buatlah laporan dengan melampirkan:
  - Desain dan penjelasannya di file word
  - Desain di packet tracer
  - Terakhir pengumpulan: 6 November 2020, hari Jumát jam 23.59
- Demo setelah UTS
- Upload di google drive