

Week0900 – Siemens Simatic IPC127	
ID:	Name:
ID:	Name:

- ทำการทดลองและเติมรายละเอียดให้เรียบร้อย
- เก็บผลการทดลองและทำรายงานแยกกันเป็นรายบุคคล
- แปลงไฟล์เป็น pdf
- กำหนดชื่อไฟล์ตามรูปแบบ “B3601234 - Wichai Srisuruk - Wk0900 IPC127.pdf”
- ส่งก่อน 20250902-0600 ที่ <https://forms.gle/iKuBiqZWXoMx3Y8x7>
- Update Web blog ของนักศึกษาเอง แปะลิงค์ == \_\_\_\_\_

#### Read More

- <https://www.mpen.co.th/17256294/siemens-simatic-ipc127>
- <https://circuitdigest.com/microcontroller-projects/controlling-leds-using-hand-gestures-with-esp32-and-python>

## Experiment 7/9: IPC127 (1/3) -- Getting Started

1. ติดตั้ง Win 10 IOT บน IPC127 >> กำหนด login=cpe28sut, password=adminZO1

ภาพการทำงาน 1/4

ภาพการทำงาน 2/4

ภาพการทำงาน 3/4

ภาพการทำงาน 4/4

2. เปิดใช้งาน Remote Desktop

ภาพการทำงาน 1/4

ภาพการทำงาน 2/4

ภาพการทำงาน 3/4

ภาพการทำงาน 4/4

## Experiment 8/9: IPC127 (1/3) – Image Processing

3. ติดตั้ง Thony Python และทดสอบการทำงาน
4. Add Library
  - Install OpenCV      `pip install opencv-python`
  - Install mediapipe      `pip install mediapipe`
  - Install requests      `pip install requests`
5. ทดสอบ mediapipe

ภาพการทำงาน 1/3

ภาพการทำงาน 2/3

ภาพการทำงาน 3/3

6. ทดสอบ mediapipe ส่งค่าไปแสดงที่ 5 LED

ภาพการทำงาน 1/3

ภาพการทำงาน 2/3

ภาพการทำงาน 3/3

7. ทดสอบ mediapipe ส่งค่าไปแสดงที่ 10 LED

ภาพการทำงาน 1/3

ภาพการทำงาน 2/3

ภาพการทำงาน 3/3

## Experiment 9/9: IPC127 (1/3) – Embedded System based on Windows Device

### 8. ปรับแต่งให้ โปรแกรมทำงานทันทีเมื่อเปิดเครื่อง IPC217

ภาพการทำงาน 1/4

ภาพการทำงาน 2/4

ภาพการทำงาน 3/4

ภาพการทำงาน 4/4

## 7.0 On PC Test

### 1. Install Python Lib

- MediaPipe แนะนำให้ทำงานกับ Python 3.12.10
- Upgrade Pip >> python.exe -m pip install --upgrade pip
- pip install opencv-python==4.11.0.86
- pip install numpy==1.26.4
- pip install mediapipe==0.10.21
- pip install requests

### 2. Test Code\_Version

```
1 import sys, cv2
2 import mediapipe
3 import numpy
4
5 print('Python = ',sys.version)
6 print('OpenCV =',cv2.__version__)
7 print('Numpy  =',numpy.__version__)
8 print('MediaPipe =',mediapipe.__version__)
9
```

```
Python = 3.12.9 | packaged by Anaconda, Inc. | (main, Feb 6 2025, 18:49:16) [MSC v.1929 64 bit (AMD64)]
OpenCV = 4.11.0
Numpy = 1.26.4
MediaPipe = 0.10.21
```

```
import sys, cv2
import mediapipe
import numpy

print('Python = ',sys.version)
print('OpenCV =',cv2.__version__)
print('Numpy =',numpy.__version__)
print('MediaPipe =',mediapipe.__version__)
```

### 3. Test Code\_OpenCAM

```
1
2 import cv2
3 cap = cv2.VideoCapture(0)
4 if not cap.isOpened():
5     print("ไม่สามารถเปิดกล้องได้")
6     exit()
7
8 while True:
9     ret, frame = cap.read()
10    if ret:
11        cv2.imshow('Camera Capture', frame)
12        if cv2.waitKey(1) & 0xFF == ord('q'):
13            break
14    else:
15        print("ไม่สามารถอ่านภาพจากกล้องได้")
16        break
17
18 cap.release()
19 cv2.destroyAllWindows()
20
21
```



```
import cv2
cap = cv2.VideoCapture(0)
if not cap.isOpened():
    print("ไม่สามารถเปิดกล้องได้")
    exit()

while True:
    ret, frame = cap.read()
    if ret:
        cv2.imshow('Camera Capture', frame)
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break
    else:
        print("ไม่สามารถอ่านภาพจากกล้องได้")
        break

cap.release()
cv2.destroyAllWindows()
```

#### 4. Test Code\_MediaPipe

```
1 import cv2
2 import mediapipe as mp
3
4 # Initialize MediaPipe Hands
5 mp_hands = mp.solutions.hands
6 hands = mp_hands.Hands()
7 mp_drawing = mp.solutions.drawing_utils
8
9 # Function to detect the state of each finger
10 def count_fingers(hand_landmarks):
11     # Detect finger states (up or down)
12     thumb_up = hand_landmarks.landmark[mp_hands.HandLandmark.THUMB_TIP].x < hand_landmarks.landmark[mp_hands.HandLandmark.THUMB_BASE].x
13     index_up = hand_landmarks.landmark[mp_hands.HandLandmark.INDEX_FINGER_TIP].y < hand_landmarks.landmark[mp_hands.HandLandmark.INDEX_FINGER_BASE].y
14     middle_up = hand_landmarks.landmark[mp_hands.HandLandmark.MIDDLE_FINGER_TIP].y < hand_landmarks.landmark[mp_hands.HandLandmark.MIDDLE_FINGER_BASE].y
15     ring_up = hand_landmarks.landmark[mp_hands.HandLandmark.RING_FINGER_TIP].y < hand_landmarks.landmark[mp_hands.HandLandmark.RING_FINGER_BASE].y
16     pinky_up = hand_landmarks.landmark[mp_hands.HandLandmark.PINKY_TIP].y < hand_landmarks.landmark[mp_hands.HandLandmark.PINKY_BASE].y
17     # Combine finger statuses into a list
18     finger_status = [thumb_up, index_up, middle_up, ring_up, pinky_up]
19     return finger_status
20
21 # Initialize VideoCapture
22 cap = cv2.VideoCapture(0)
23 while cap.isOpened():
24     ret, frame = cap.read()
25     if not ret:
26         break
27     frame = cv2.flip(frame, 1)
28     frame_rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
29     # Detect hand landmarks
30     results = hands.process(frame_rgb)
31     if results.multi_hand_landmarks:
32         for hand_landmarks in results.multi_hand_landmarks:
33             mp_drawing.draw_landmarks(frame, hand_landmarks, mp_hands.HAND_CONNECTIONS)
34             fingers = count_fingers(hand_landmarks)
35             cv2.putText(frame, f"Fingers: {fingers}", (10, 50), cv2.FONT_HERSHEY_SIMPLEX, 0.8, (0, 255, 0))
36     cv2.imshow('Hand Gesture Recognition', frame)
37     if cv2.waitKey(5) & 0xFF == 27: # Exit on pressing 'Esc'
38         break
39 cap.release()
40 cv2.destroyAllWindows()
41
```



```

import cv2
import mediapipe as mp

# Initialize MediaPipe Hands
mp_hands = mp.solutions.hands
hands = mp_hands.Hands()
mp_drawing = mp.solutions.drawing_utils

# Function to detect the state of each finger
def count_fingers(hand_landmarks):
    # Detect finger states (up or down)
    thumb_up = hand_landmarks.landmark[mp_hands.HandLandmark.THUMB_TIP].x < hand_landmarks.landmark[mp_hands.HandLandmark.THUMB_IP].x
    index_up = hand_landmarks.landmark[mp_hands.HandLandmark.INDEX_FINGER_TIP].y < hand_landmarks.landmark[mp_hands.HandLandmark.INDEX_FINGER_PIP].y
    middle_up = hand_landmarks.landmark[mp_hands.HandLandmark.MIDDLE_FINGER_TIP].y < hand_landmarks.landmark[mp_hands.HandLandmark.MIDDLE_FINGER_PIP].y
    ring_up = hand_landmarks.landmark[mp_hands.HandLandmark.RING_FINGER_TIP].y < hand_landmarks.landmark[mp_hands.HandLandmark.RING_FINGER_PIP].y
    pinky_up = hand_landmarks.landmark[mp_hands.HandLandmark.PINKY_TIP].y < hand_landmarks.landmark[mp_hands.HandLandmark.PINKY_PIP].y
    # Combine finger statuses into a list
    finger_status = [thumb_up, index_up, middle_up, ring_up, pinky_up]
    return finger_status

# Initialize VideoCapture
cap = cv2.VideoCapture(0)
while cap.isOpened():
    ret, frame = cap.read()
    if not ret:
        break
    frame = cv2.flip(frame, 1)
    frame_rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
    # Detect hand landmarks
    results = hands.process(frame_rgb)
    if results.multi_hand_landmarks:
        for hand_landmarks in results.multi_hand_landmarks:
            mp_drawing.draw_landmarks(frame, hand_landmarks, mp_hands.HAND_CONNECTIONS)
            fingers = count_fingers(hand_landmarks)
            cv2.putText(frame, f"Fingers: {fingers}", (10, 50), cv2.FONT_HERSHEY_SIMPLEX, 0.8, (0, 255, 0), 2)
    cv2.imshow('Hand Gesture Recognition', frame)
    if cv2.waitKey(5) & 0xFF == 27: # Exit on pressing 'Esc'
        break
cap.release()
cv2.destroyAllWindows()

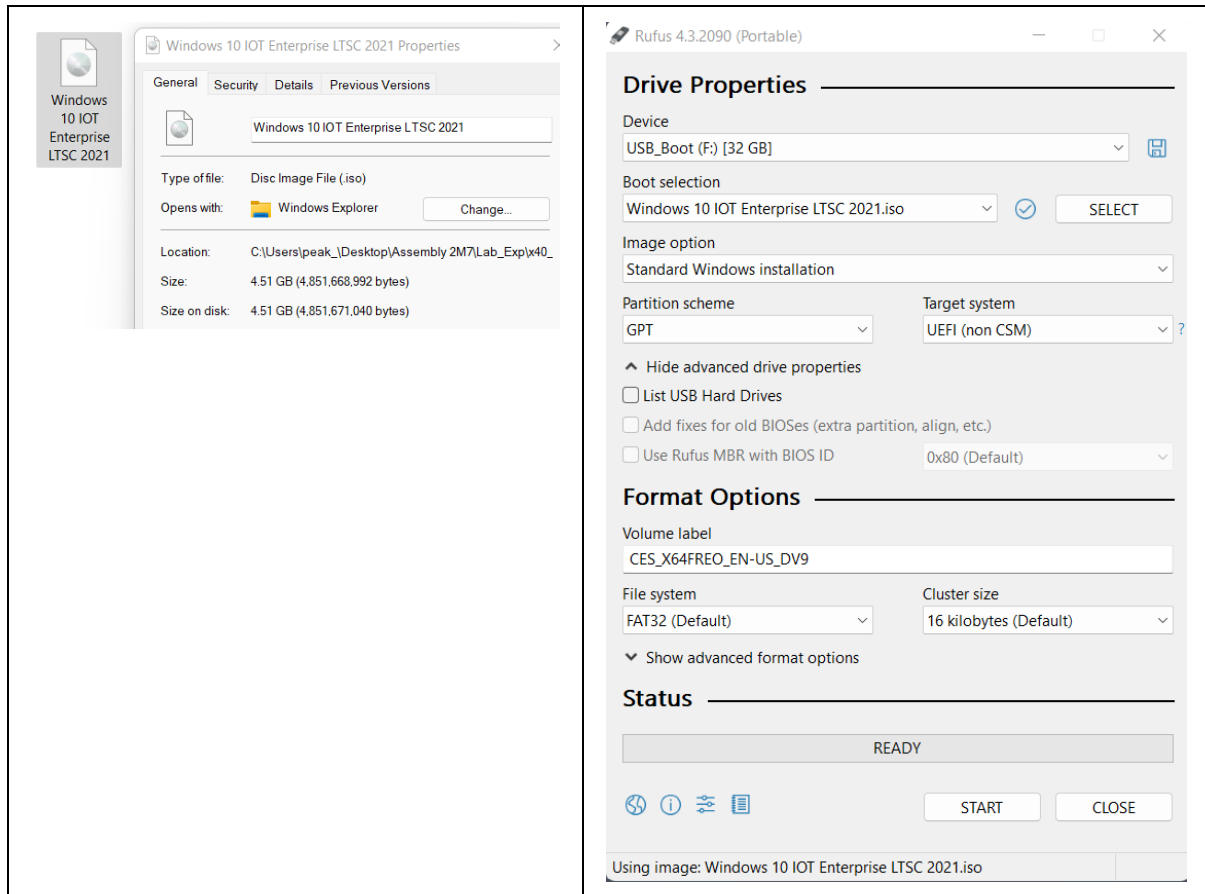
```



## 7.1 Install OS

### Install Win-10 IOT OS

1. Download “Windows 10 IOT Enterprise LTSC 2021.iso”
2. สร้าง USB Boot Drive ด้วยโปรแกรม “rufus-4.3p.exe”



3. IPC127 เข้า Boot Manu ด้วยการกด ESC Key
4. เลือก USB Boot และติดตั้ง Win10 IOT

Home

Find a setting

System

Display

Sound

Notifications & actions

Focus assist

Power & sleep

Storage

Tablet

Multitasking

Projecting to this PC

About

Device specifications

Device name	CPESUT01-IPC127E
Processor	Intel(R) Atom(TM) Processor E3940 @ 1.60GHz 1.60 GHz
Installed RAM	4.00 GB (3.82 GB usable)
Device ID	903B3F56-209B-478A-A707-E5E637902D22
Product ID	00436-50013-42754-AAOEM
System type	64-bit operating system, x64-based processor
Pen and touch	No pen or touch input is available for this display

Copy

Rename this PC

Windows specifications

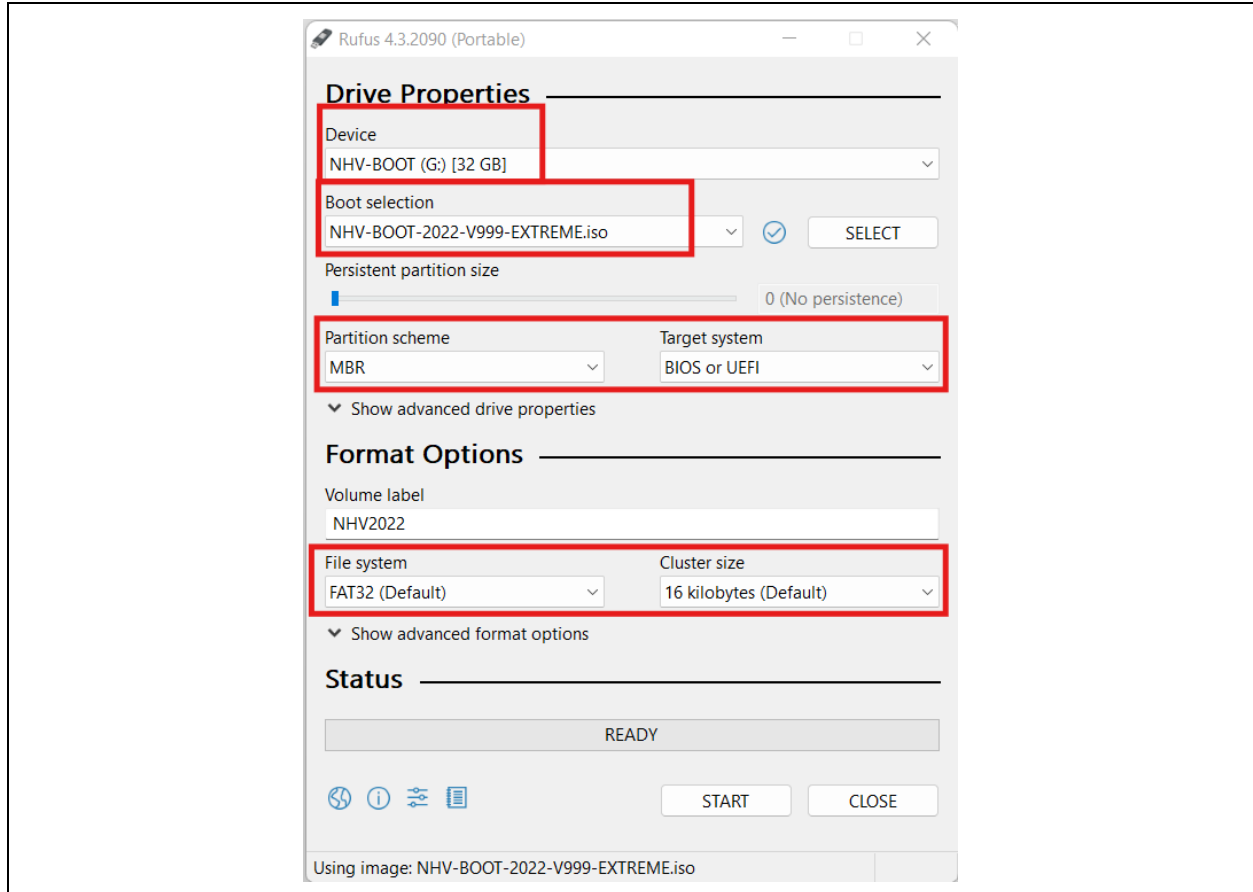
Edition	Windows 10 IoT Enterprise LTSC
Version	21H2
Installed on	6/27/2025
OS build	19044.5965
Experience	Windows Feature Experience Pack 1000.19061.1000.0

เพื่อความสะดวก กำหนดให้ตั้งค่า login=cpe28sut, password=adminZ01

## Clone OS

### 1. เรียกโปรแกรม rufus-4.3p.exe

- เขียนไฟล์ USBBOOT “NHV-BOOT-2022-V999-EXTREME.iso” ไปยัง USB Drive



### 2. BOOT SIEMENS IPC127 จาก USB\_Drive <IPC127 เข้า BOOT Manu ด้วยการกด ESC Key>

### 3. เรียกใช้โปรแกรม Symantec Ghost

- กรณีเก็บ HDD : Ghost >> Disk to Image
- กรณีเขียน HDD : Ghost >> Image to Disk

## 7.2 เปิดใช้งาน Remote Desktop

## 7.3 ติดตั้ง Python 3.12.10 และทดสอบการทำงาน

- MediaPipe แนะนำให้ทำงานกับ Python 3.12.10
- Upgrade Pip >> python.exe -m pip install --upgrade pip

## 7.4 Add Library < opencv-python, mediapipe, requests >

- pip install opencv-python==4.11.0.86
- pip install numpy==1.26.4
- pip install mediapipe==0.10.21
- pip install requests

## 7.5 Install OS ทดสอบ mediapipe

## 7.6 ทดสอบ mediapipe ส่งค่าไปแสดงที่ 5 LED

## 7.7 ทดสอบ mediapipe ส่งค่าไปแสดงที่ 10 LED

## 7.8 ปรับแต่งให้ โปรแกรมทำงานทันทีเมื่อเปิดเครื่อง IPC217