8-ДӘРІС

Дисплейлер және деректерді өңдеу

OLED визуализациясы, өлшеуді сүзгілеу және журналдау

әл-Фараби атындағы Қазақ ұлттық университеті

Физика-техникалық факультет

Дәрістің мақсаты және күтілетін нәтижелер

- Негізгі ұғымдарды түсіндіру: OLED, Frame buffer, Жылжымалы орташа.
- Raspberry Pi Pico құрылғысында тиісті MicroPython API құралдарын қолдану.
- Аппараттық қосылымның қауіпсіздігін және бағдарламаның дұрыстығын бағалау.
- Өлшеу немесе басқару алгоритмін тәжірибелік міндетке бейімдеу.

ТЕОРИЯ

Негізгі ұғымдар және анықтамалар

OLED

Әр пикселі жарық шығаратын ықшам дисплей технологиясы.

Frame buffer

Экран пиксельдерінің күйін жадта сақтайтын буфер.

Жылжымалы орташа

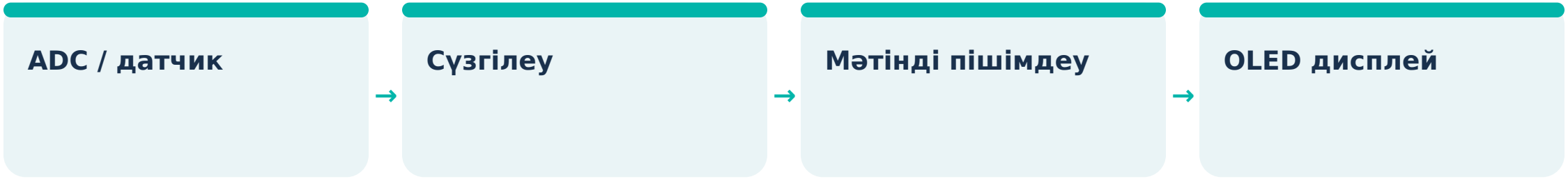
Соңғы N өлшемнің орташа мәні арқылы шуды азайту.

Теориялық негіздер: басты қағидалар

- SSD1306 OLED жиі I²C арқылы басқарылады; драйвер файлы құрылғыға жүктелуі тиіс.
- Экранды әр циклде жаңарту қажет емес: мән өзгергенде немесе тұрақты аралықпен жаңартуға болады.
- Flash жадына тым жиі жазу тозуды арттырады; журналдау жиілігін негіздеу қажет.

АРХИТЕКТУРА

Жүйенің функционалдық құрылымы



Блок-схема сигналдың немесе деректердің негізгі қозғалыс бағытын көрсетеді. Нақты қосылым пиндері және қоректендіру талаптары қолданылатын модульдің техникалық құжаттамасымен салыстырылады.

ТАЛДАУ

Аппараттық және бағдарламалық байланыс

Аппараттық қабат

GPIO, қоректендіру, датчик, шина және физикалық сигнал деңгейлері.

Бағдарламалық қабат

MicroPython кітапханалары, алгоритм, деректерді өңдеу және қате жағдайлары.

Инженерлік қағида: бағдарламаның дұрыс орындалуы электрлік қосылымның қауіпсіздігін автоматты түрде қамтамасыз етпейді.

OLED дисплейіне жазу

```
from machine import Pin, I2C
from ssd1306 import SSD1306_I2C
i2c = I2C(0, sda=Pin(0), scl=Pin(1))
oled = SSD1306_I2C(128, 64, i2c)
oled.fill(0)
oled.text("Raspberry Pi Pico", 0, 0)
oled.show()
```

Ескерту: кодты іске қоспас бұрын пиндер мен модуль түрін тексеріңіз.

Жылжымалы орташа

```
samples = []
def smooth(value, n=8):
    samples.append(value)
    if len(samples) > n:
        samples.pop(0)
    return sum(samples) / len(samples)
```

Ескерту: кодты іске қоспас бұрын пиндер мен модуль түрін тексеріңіз.

CSV форматында журнал

```
from machine import ADC, Pin
from time import sleep
adc = ADC(Pin(26))
with open("log.csv", "a") as f:
    for i in range(10):
        f.write(f"{i},{adc.read_u16()}\n")
        sleep(1)
```

Ескерту: кодты іске қоспас бұрын пиндер мен модуль түрін тексеріңіз.

Практикалық қолдану сценарийлері

- SSD1306 OLED-де ADC кернеуін көрсететін интерфейс құру.
- Шикі және орташаланған өлшемдерді салыстыру.
- Тәжірибе барысында алынған нәтижені кесте немесе қысқа техникалық есеп түрінде рәсімдеу.

Жиі кездесетін қателер және қауіпсіздік

- ssd1306.py драйверін құрылғыға көшірмеу.
- Экран өлшемін немесе I²C пиндерін қате көрсету.
- Құрылғыны жалғар алдында қорек көзін ажыратыңыз; 3,3 В логикалық деңгей шегін сақтаңыз.

Бақылау сұрақтары және талқылау

- OLED frame buffer қандай міндет атқарады?
- Неліктен өлшеулерді сүзгілеу қажет?
- Flash жадына жазу жиілігі неге маңызды?

Қорытынды және ұсынылатын әдебиеттер

Дәріс бойынша қорытынды: теориялық ұғымдар MicroPython коды және аппараттық жүйе жұмысы арқылы бекітіледі.

- Raspberry Pi Ltd. Raspberry Pi Pico-series documentation: raspberrypi.com/documentation/microcontrollers/
- Raspberry Pi Ltd. RP2040 Datasheet: datasheets.raspberrypi.com/rp2040/rp2040-datasheet.pdf
- MicroPython documentation: docs.micropython.org/en/latest/rp2/quickref.html

- Қосымша: пайдаланылатын датчиктер мен модульдердің өндіруші дерекпарақтары.