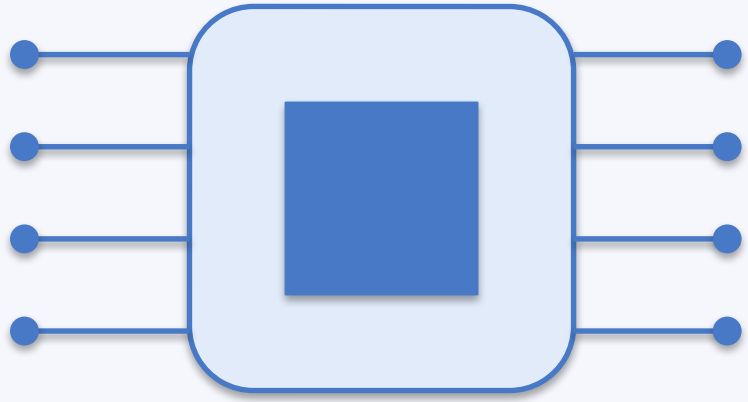
EMBEDDED / CONTROL

РОБОТТАРДЫҢ ЭЛЕКТРОНДЫҚ ЖӘНЕ МИКРОПРОЦЕССОРЛЫҚ ЖҮЙЕЛЕРІ

Электрондық тораптар • микроконтроллер • басқару архитектурасы



CONTROL • COMPUTE • CONNECT

Балжан Ибрагимқызы
PhD, ассистент-профессор

Дәрістің мақсаты және жүйенің орны

Электрондық және микропроцессорлық жүйелер роботтың сенсорларын, жетектерін және байланысын біріктіреді.

Мақсаты

Роботтың электрондық құрылымын, процессорлық модульдерін және сигнал алмасуын түсіндіру.

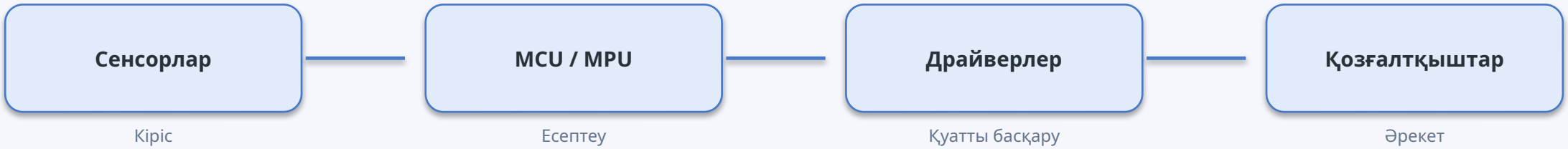
Міндеттері

Қоректендіру, есептеу, енгізу/шығару және драйвер тораптарын қарастыру.

Нәтиже

Қарапайым роботтың басқару архитектурасын блок-схема түрінде құрастыру.

Роботтың электрондық архитектурасы



1. Роботтың электрондық жүйесінің құрамы

Электрондық жүйе қорек, есептеу, сигналды түрлендіру және қуат коммутациясы тораптарынан тұрады.

Қоректендіру

Батарея, DC/DC түрлендіргіш

Есептеу модулі

MCU, MPU немесе SBC

Сигнал интерфейсі

ADC, GPIO, PWM

Қуат драйвері

Мотор, сервожетек, реле

Байланыс

UART, I^2C, SPI, CAN

Қорғаныс

Сақтандырғыш, TVS, ток шектеу

2. Микроконтроллер мен микропроцессор

Екі құрылғы да есептеу орындайды, бірақ интеграция деңгейі мен қолданылу міндеттері өзгеше.

Микроконтроллер (MCU)

CPU, жад және периферия бір чипте орналасады. Нақты уақыттағы басқаруға, төмен қуат тұтынуға ыңғайлы.
Мысал: ATmega328P, STM32, ESP32.

Микропроцессор (MPU)

Негізінен есептеу ядросы; жедел жад пен қосымша контроллерлер сыртқы болуы мүмкін. Күрделі ОЖ мен жоғары өнімді есептерге арналған.
Мысал: ARM Cortex-A негізіндегі SoC.

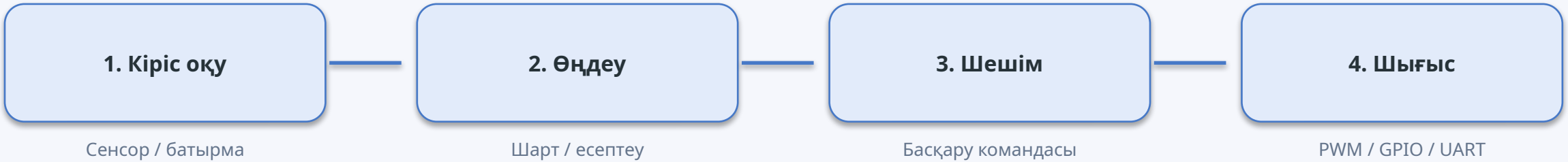
Қолдану мысалы

MCU → моторды басқару, сенсорды оқу

MPU / SBC → камера, карталау, компьютерлік көру

3. Микроконтроллер қалай жұмыс істейді?

Контроллер бағдарламадағы командаларды цикл бойынша орындайды және перифериялық құрылғылармен алмасады.



Негізгі цикл

loop() ішінде датчиктер оқылады, мәндер тексеріледі және моторға команда беріледі.

Үзіліс (Interrupt)

Маңызды оқиғаға негізгі циклді күтпей жауап беруге мүмкіндік береді.

4. Жад түрлері және перифериялық модульдер

Бағдарлама мен деректер әртүрлі жад аймақтарында сақталады; периферия сыртқы құрылғылармен байланысады.

Flash

Бағдарлама коды мен тұрақты деректерді қуат өшкенде де сақтайды.

SRAM / RAM

Бағдарлама орындалып жатқанда айнымалылар мен уақытша деректерді сақтайды.

EEPROM

Қуат өшкенде сақталуы қажет шағын параметрлерге қолданылады.

Кең таралған интерфейстер

GPIO

Сандық кіріс/шығыс

PWM

Импульс енін басқару

ADC

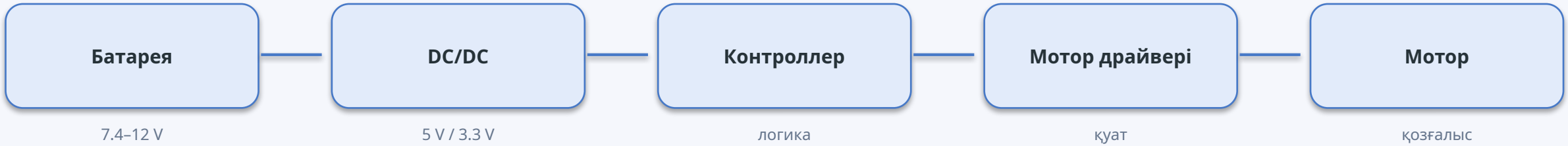
Аналогты цифрлау

UART / I^2C / SPI

Дерек алмасу

5. Қоректендіру және қозғалтқыш драйверлері

Контроллердің логикалық шығысы моторды тікелей қоректендіруге арналмаған; қуат драйвері қажет.



Логикалық қорек

MCU көбіне 3.3 V немесе 5 V деңгейінде жұмыс істейді. Нақты мән модульге байланысты.

Мотор қорегі

Мотор кернеуі мен тогы бөлек есептеледі; драйвердің ток рейтингі жүктемеге сай болуы керек.

6. Байланыс интерфейстері

Интерфейс сенсор, контроллер, драйвер және компьютер арасында дерек алмасуды ұйымдастырады.

UART

Асинхронды тізбекті байланыс. TX және RX желілері арқылы екі құрылғы дерек алмасады.

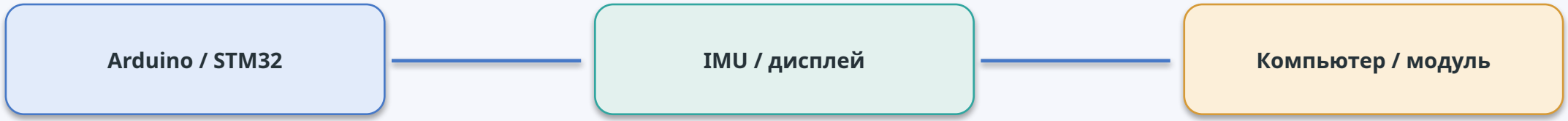
I²C

Екі желі: SDA және SCL. Бір шинада бірнеше адрестелетін құрылғы болуы мүмкін.

SPI

SCLK, MOSI, MISO және CS сигналдарын пайдаланады. Жылдам периферияға ыңғайлы.

Интерфейстерді қолдану мысалдары



7. Робототехникадағы есептеу платформалары

Платформа таңдау кезінде өнімділік, порттар, қуат тұтыну, бағдарламалық орта және баға ескеріледі.

Arduino Uno

Оқу жобалары мен қарапайым басқару жүйелеріне ыңғайлы. ATmega328P микроконтроллері.

STM32 / ESP32

Таймерлер, PWM, ADC және байланыс интерфейстері бар; күрделірек ендірілген жүйелерге арналған.

Raspberry Pi

Linux негізіндегі бірплаталы компьютер; камера, желі және жоғары деңгейлі алгоритмдерге ыңғайлы.

Шартты архитектуралық бөлу

Жоғары деңгей: камера / карта / жоспарлау

Төмен деңгей: PWM / энкодер / мотор

Күрделі роботта SBC мен MCU бірлесіп жұмыс істей алады.

Қорытынды және бақылау сұрақтары

Электрондық архитектура сенсорлардан алынған ақпаратты өңдеп, жетектерге қауіпсіз басқару сигналын береді.

Негізгі қорытынды

MCU нақты уақыттағы басқаруды орындайды; MPU/SBC күрделі есептерді орындай алады; драйверлер қуатты басқарады.

Инженерлік қағида

Қорек кернеуі, ток рейтингі, логикалық деңгейлер және ортақ GND дұрыс жобалануы тиіс.

Бақылау сұрақтары

1. Электрондық жүйенің негізгі блоктары қандай?
2. MCU мен MPU арасындағы айырмашылық неде?
3. Flash, RAM және EEPROM қандай міндет атқарады?
4. UART, I^2C және SPI интерфейстері қалай ерекшеленеді?
5. Неліктен моторды микроконтроллер пиніне тікелей қосуға болмайды?

Әдебиет: M. Barr & A. Massa, Programming Embedded Systems; M. Margolis, Arduino Cookbook.