



HỆ THỐNG THỜI GIAN THỰC

Thiết kế shipping robot

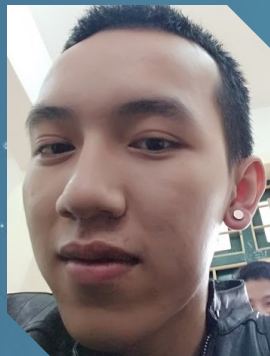
T1 TEAM



NGUYỄN HỮU SANG



LÊ HOÀI NAM



NGUYỄN ĐÌNH DŨNG



NGUYỄN VĂN HÙNG

YÊU CẦU BÀI TOÁN

Thiết kế shipping robot sử dụng
hệ điều hành thời gian thực

01

PHƯƠNG ÁN THIẾT KẾ

Cơ khí, phần cứng, phần mềm

02

NỘI DUNG

03

CHƯƠNG TRÌNH

Thuật toán điều khiển

04

BÀI TẬP

Bài tập 10.1 10.2 10.3



01

Yêu cầu bài toán

Thiết kế shipping robot sử dụng hệ điều hành thời gian thực

SHIPPING ROBOT

PHẦN CỨNG

- Thiết kế robot với cảm biến phù hợp, ứng dụng IoT
- Đầu vào: cảm biến / Đầu ra: motor, buzzer
- Trọng lượng $\leq 2\text{kg}$
- Giao tiếp không dây: RF, Zigbee, Wifi
- Sử dụng VĐK PIC

PHẦN MỀM

- Viết chương trình C dựa trên hệ điều hành thời gian thực (RTOS)
- Xây dựng thuật toán dựa trên yêu cầu thiết kế (đi thẳng, rẽ, tránh vật cản)





Zone 1

Zone 2

Zone 3

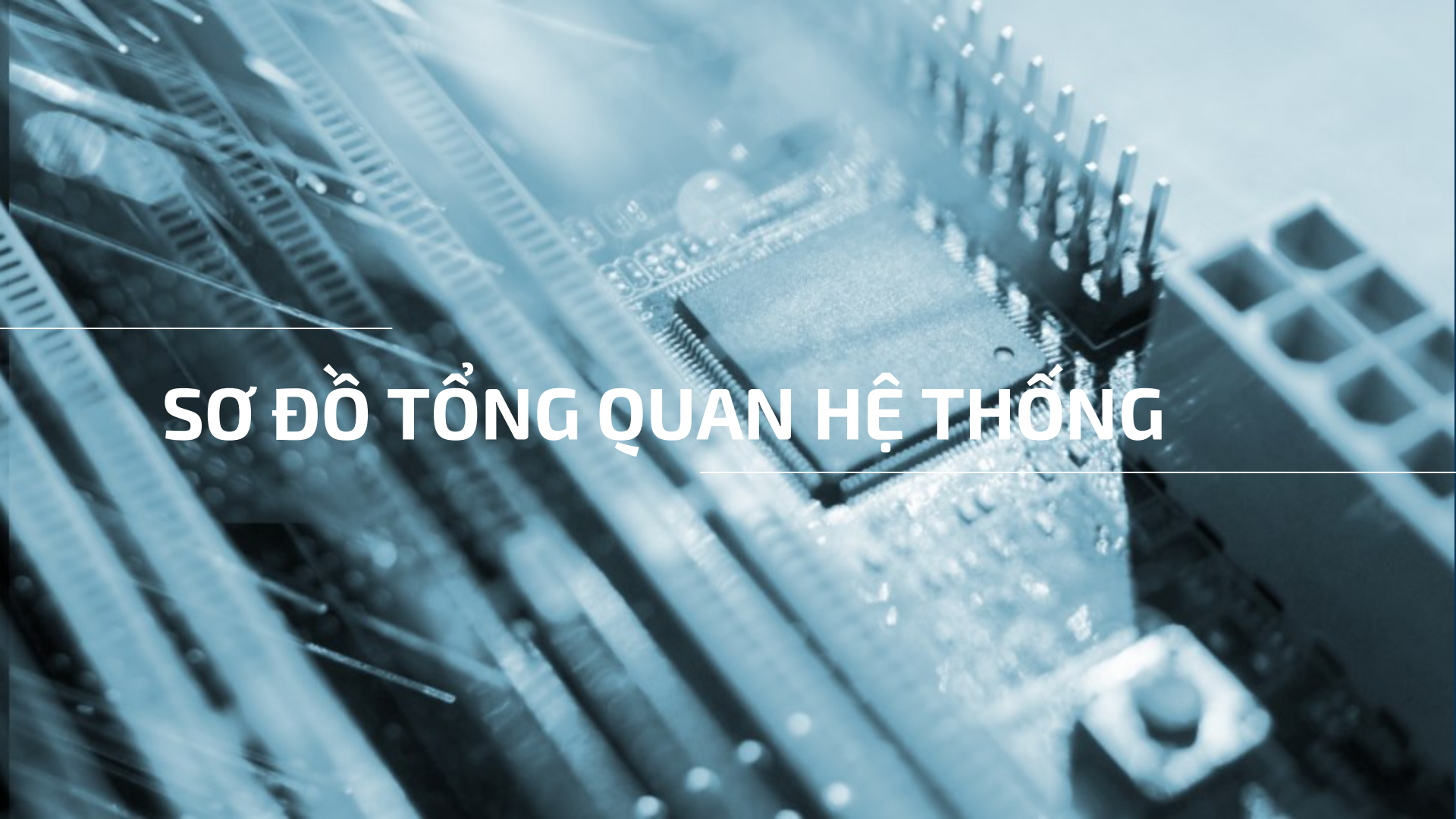


Zone
4



Zone 5
Victory

Sân thi đấu



SƠ ĐỒ TỔNG QUAN HỆ THỐNG



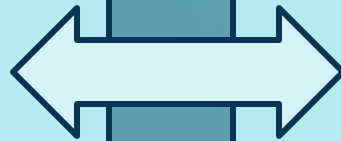
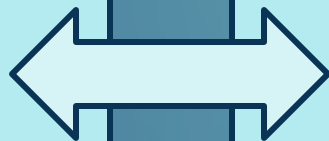
Giao diện
người-máy



Hệ thống
máy tính
thời gian
thực



Đối tượng
điều khiển





02

Phương án thiết kế

Cơ khí, phần cứng, phần mềm

Thiết kế cơ khí

- Xe có trọng lượng nhẹ $\leq 2\text{kg}$
 - Sử dụng khung xe bằng nhựa mica, Nguồn: khay pin 18650, điện áp ra $\sim 12\text{V}$
- Xe có thể đi thẳng, rẽ hướng, điều khiển tốc độ
 - Sử dụng 2 động cơ DC motor điều khiển độc lập
 - kết hợp driver điều khiển động cơ L298D



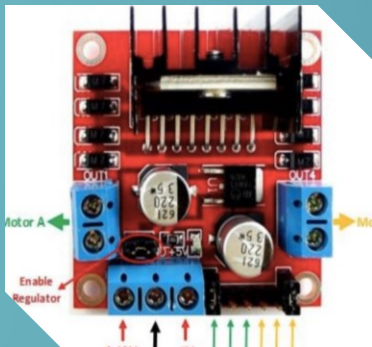


Khung xe

Chất liệu: nhựa mica
Mẫu sắc: trong suốt
Kích thước: 25 x 14.8 cm

Nguồn

Điện áp ra: ~12V
Dòng sạc: 1.25 A
Dòng xả: 20A



Module L298D

IC chính: L298 - Dual Full Bridge Driver
Điện áp đầu vào: 5~30VDC
Công suất tối đa: 25W 1 cầu
Dòng tối đa cho mỗi cầu H là: 2A
Mức điện áp logic: Low -0.3V~1.5V, High: 2.3V~Vss
Kích thước: 43x43x27mm

Bánh điều hướng Omni

Điện áp sử dụng: 3~6VDC.
Dòng điện tiêu thụ: 110-200mA
Tỉ số truyền: 1:48
Số vòng/1phút:
90 vòng/ 1 phút tại 3VDC.
200 vòng/ 1 phút tại 6VDC.





DC MOTOR+ Bánh xe

Điện áp sử dụng: 3~6VDC.

Dòng điện tiêu thụ: 110-200mA

Tỉ số truyền: 1:48

Số vòng/1phút:

90 vòng/ 1 phút tại 3VDC.

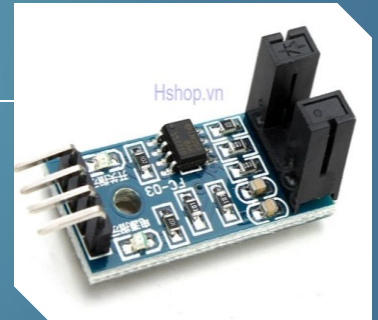
200 vòng/ 1 phút tại 6VDC.

Cảm Biến Tốc Độ Encoder V1

Điện áp sử dụng: 3.3~5VDC

Dòng sử dụng: 15mA

Ngõ ra: Analog và Digital.



Thiết kế phần cứng

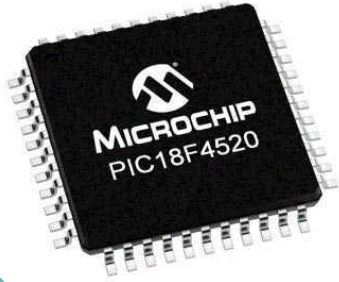
- Sử dụng VĐK PIC

→ Lựa chọn VĐK **PIC18F4520**, đã được sử dụng ở môn học trước

- Giao tiếp không dây

→ Giao tiếp qua wifi sử dụng module wifi **nodeMCU**, dễ sử dụng, truyền thông UART đơn giản với PIC





PIC18F4520

Lỗi: 8-bit

Tốc độ: 40MHz

Kết nối: I2C, SPI, UART

Thiết bị ngoại vi: HLVD, POR, PWM, WDT

Số chân I/O: 36

Điện áp: 4.2 ~ 5.5V

NodeMCU

IC chính: ESP8266 Wifi SoC.

Cấp nguồn: 5VDC MicroUSB hoặc Vin.

GPIO giao tiếp mức 3.3VDC

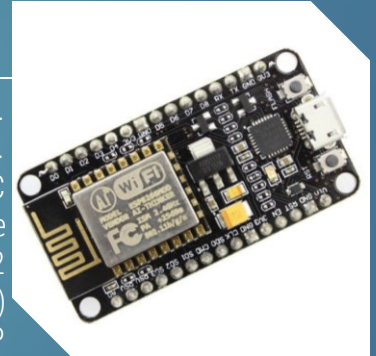
30 pins: 10 GPIO/PWM, I2C, 1-wire

WiFi @ 2.4 GHz, hỗ trợ bảo mật WPA / WPA2

Wi-Fi Connectivity (802.11 b/ g/ n)

modes: STA/AP/STA+AP

Kích thước: 25 x 50 mm



Lựa chọn cảm biến

- Xác định hướng di chuyển của xe

→ Lựa chọn Cảm Biến Góc Gia Tốc **MPU6050 GY-521**, đã được sử dụng ở môn học trước

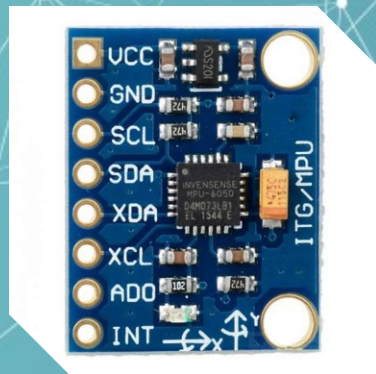
- Phát hiện vật cản

→ Vật cản ở phía trước xe nên sử dụng cảm biến siêu âm **HY-SRF05** đặt phía trước xe

- Phát hiện Zone

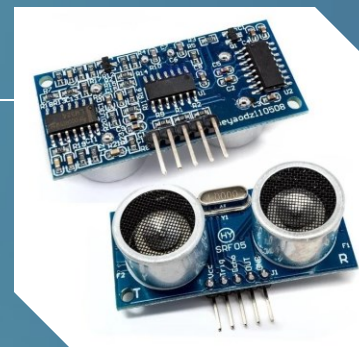
→ Zone được xác định bằng các ô line màu đen, sử dụng cảm biến dò line đơn **TCRT5000** đặt dưới gầm xe

GY-521 6DOF MPU6050

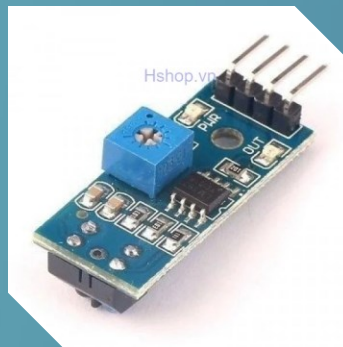


Điện áp sử dụng: 3~5VDC
Điện áp giao tiếp: 3~5VDC
Chuẩn giao tiếp: I2C
Tín hiệu ra:
Gyroscopes : +/- 250 degree/sec
Acceleration: +/- 2g

UltraSonic HY-SRF05



Điện áp hoạt động: 5VDC
Dòng tiêu thụ: 10~40mA
Tín hiệu giao tiếp: TTL
Góc quét: <15 độ
Khoảng cách đo được: 2~450cm.
Kích thước: 43mmx20mmx17mm



TCRT5000

Nguồn cung cấp: 5VDC
Có thể điều chỉnh độ nhạy bằng biến trở
Dòng điện tiêu thụ: <10mA.
Ngõ giao tiếp: 4 dây VCC, GND, DO, AO.
Mức tín hiệu ngõ ra: Digital TTL & Analog
Kích thước: 3.2 x 1.4mm

Thiết kế phần mềm

- Viết chương trình C dựa trên hệ điều hành thời gian thực (RTOS)
 - Sử dụng **CSS 5** là trình biên dịch ngôn ngữ C, hỗ trợ họ VDK PIC, tích hợp sẵn thư viện RTOS
- Giao diện người – máy
 - Sử dụng ứng dụng viết bằng **Blynk** để cấu hình robot, nhận thông báo gửi lên
 - Kết nối qua wifi, dễ kết nối, lập trình, giao diện trực quan, sử dụng được trên điện thoại di động



CCS5 Compiler

Lập trình RTOS cho VĐK PIC18F4520

Arduino IDE

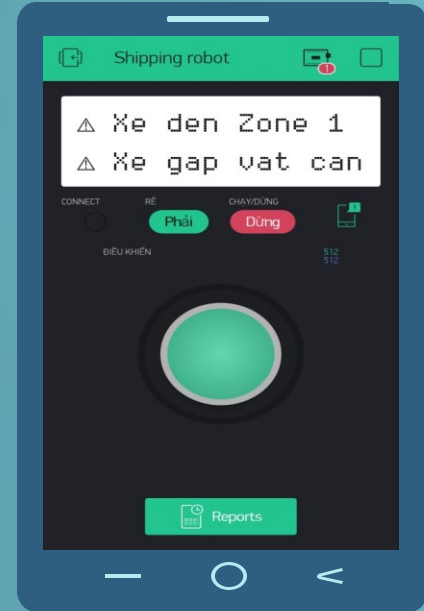
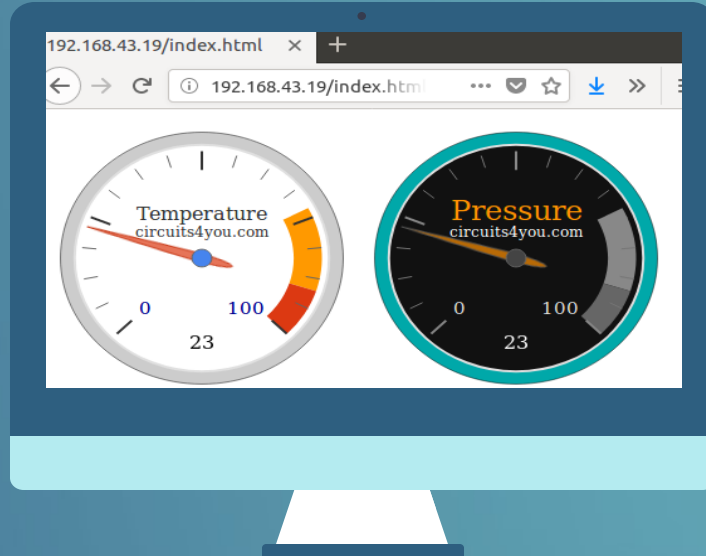
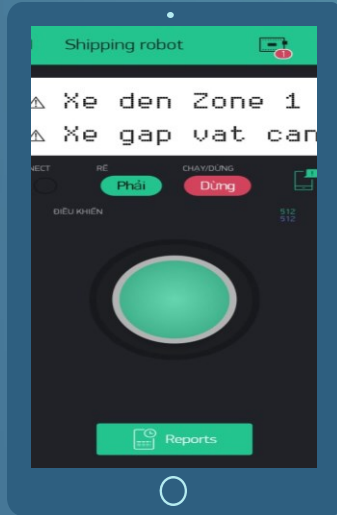
Lập trình cho NodeMCU



Blynk App

Thiết kế giao diện người – máy
Cấu hình, nhận thông báo từ robot
Sử dụng trên điện thoại di động

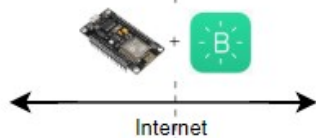
Giao diện người-máy



Online: Sử dụng Blynk App
Offline: AP mode Webserver

Sơ đồ hệ thống

Man-machine interface



Realtime Computer System



Tín hiệu đk động cơ

Hướng

Zone

Vật cần

Tốc độ

Object





03

Chương trình

Chương trình, thuật toán điều khiển

Thuật toán điều khiển

(Thuật toán đặt lịch ưu tiên)





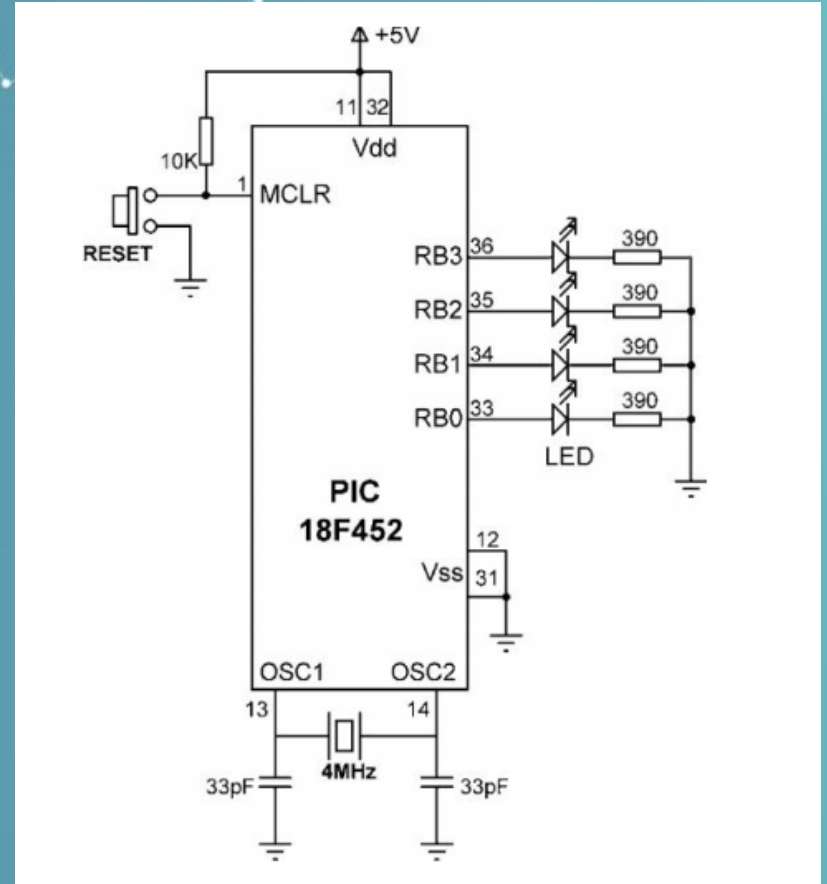
04

BÀI TẬP

Bài tập 10.1 10.2 10.3

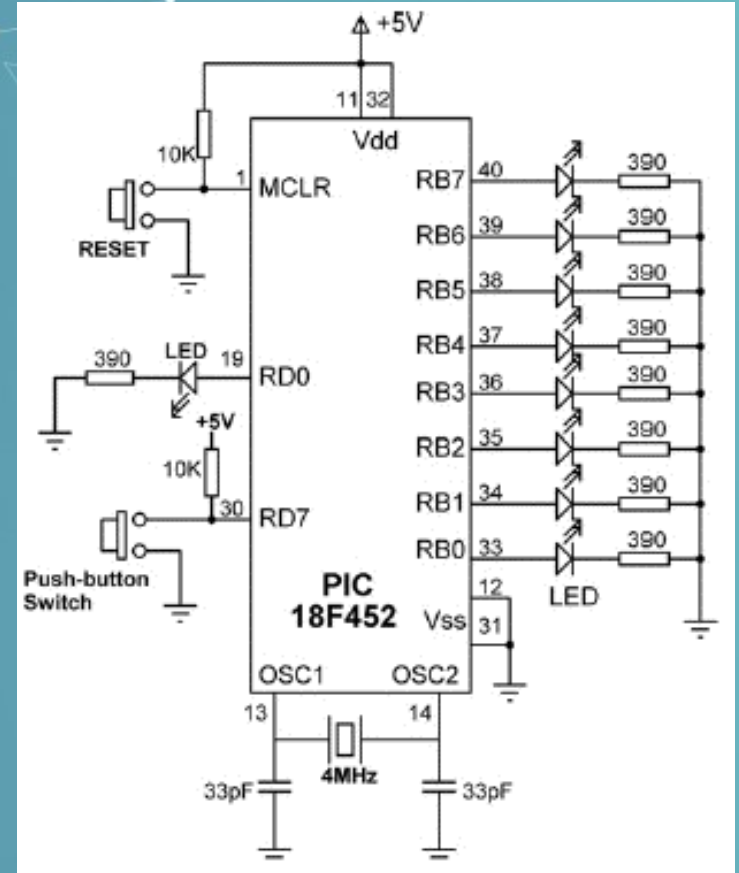
10.1

- **Tác vụ 1:** chạy mỗi 250ms, đổi trạng thái chân RB0
- **Tác vụ 2:** chạy mỗi 500ms, đổi trạng thái chân RB1
- **Tác vụ 3:** chạy mỗi 1s, đổi trạng thái chân RB2
- **Tác vụ 4:** chạy mỗi 2s, đổi trạng thái chân RB3



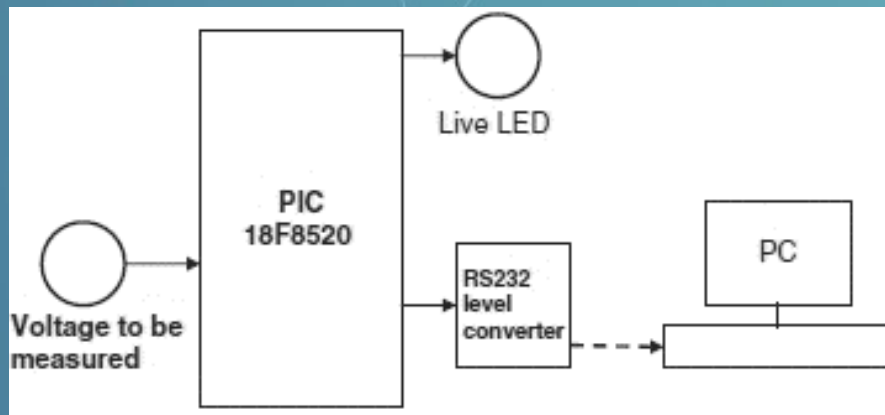
10.2

- **Task Live:** chạy sau mỗi 200ms, đổi trạng thái chân RD0
- **Task Generator:** đếm từ 0 -> 255,
Khi nhấn nút, gửi giá trị đếm sang task Display
- **Task Display:** đọc số từ hàng đợi thông điệp,
hiện thị số qua 8 led



10.3

- **Task Live:** chạy sau mỗi 20ms, đổi trạng thái chân RD7
- **Task Generator:** đọc giá trị analog từ chân AN0, chuyển đổi sang mV
- **Task Display:** gửi giá trị analog chuyển đổi sang RS232 để truyền qua máy tính



VIDEO





THANKS

CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, and infographics & images by **Freepik**.

Please keep this slide for attribution.