CEC Continuing Education Center

DesignNews

Getting Started in Automation with Arduino

DAY 1: Introduction to Automation

Sponsored by

DigiKey

informa markets

Webinar Logistics

- Turn on your system sound to hear the streaming presentation.
- If you have technical problems, click “Help” or submit a question asking for assistance.
- Participate in ‘Attendee Chat’ by maximizing the chat widget in your dock.

Dr. Don Wilcher

Visit ‘Lecturer Profile’ in your console for more details.

Course Kit and Materials

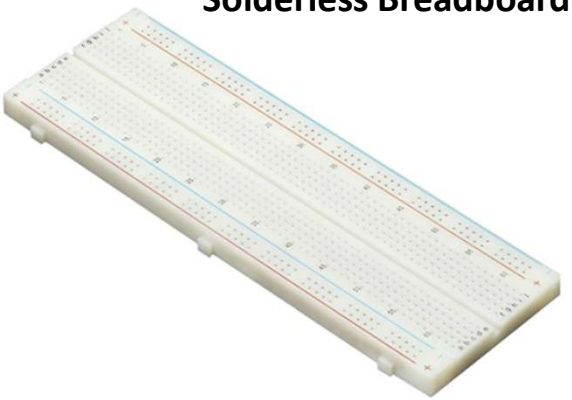
Arduino Opta

12VDC @ 500mA Wall Mount Power Supply

DC Motor: Medium Torque

Adafruit Parts Pal Kit

Solderless Breadboard



Jumper Wires: Male to Male

Solderless Breadboard Power Supply

Agenda:

- Automation Explained
  - a) Overview
  - b) Devices
  - c) Integrated Systems
- Programmable Logic Controllers
  - a) Description
  - b) Core Components
- Brief Overview of the Arduino Opta
  - a) Pin Out
  - b) Accessing LEDs, Output Relays, USR Button

Seminal Research Perspective

"Programmable Logic Controller (PLC) is the most important component in industrial automation, and it has become one of the three pillars (robots, PLC, and CAD/CAM) of the modern industrial control technology"(Liao, 2007).

Automation Explained: Overview

- The use of technology to perform tasks by minimizing human interaction is called Automation.
- Basic Automation considers these tasks.
  a) simple
  b) repetitive
- Digitizing work will help streamline and centralize industrial routine tasks like:
  a) sorting
  b) picking
  c) painting
- Performing such repetitive tasks can eliminate or minimize error. (The goal!)

Automation Explained: Overview. . .

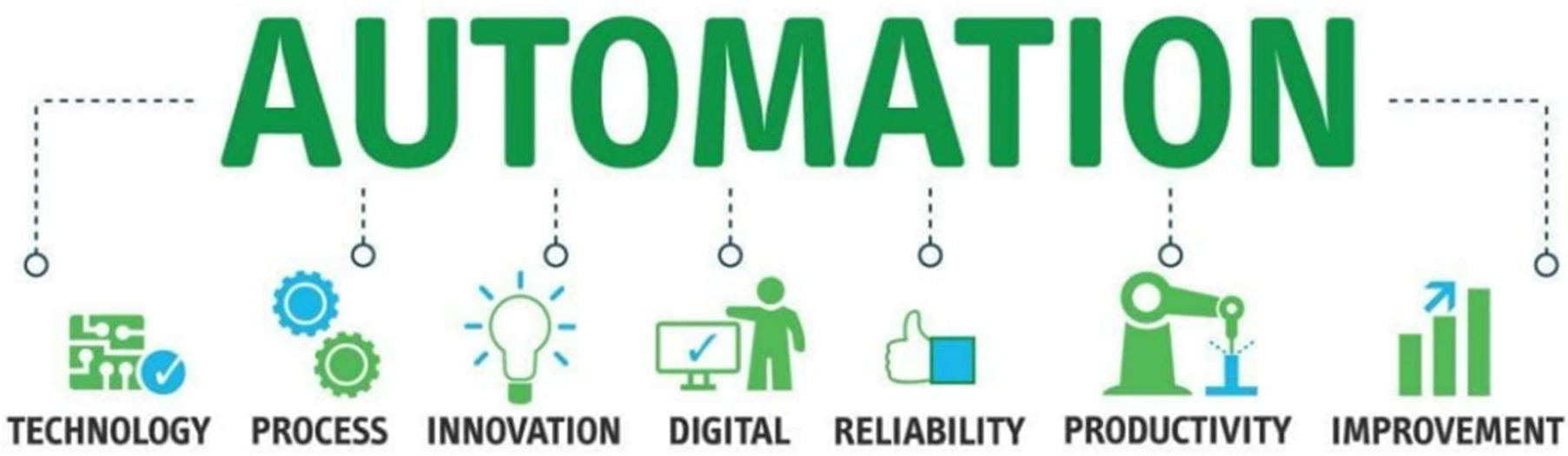


Image Source: Industrial Automation Asia

Automation Explained: Overview. . .

" Automation may be defined as the use of control systems and information technologies to reduce the need of human involvement in the production of goods and services" (Mandal et al., 2015).

Automation Explained: Devices

- Production equipment that can automatically perform tasks are known as industrial automation tasks.
- Some industrial automation devices include
    a) Industrial robots
    b) Automation cells
    c) Conveyors
    d) Lifters
    e) Pick and Place machines
    f) Programmable Logic Controllers (PLCs)
    g) Distributed Control Systems (DCS)

Automation Explained: Devices . . .

- Modern-day machinery consists of the following components.
  a) electrical
  b) mechanical
  c) electronics
  d) sensors
  d) instrumentation
- The combination of these components that align with their respective engineering fields falls under the multidisciplinary subject of Mechatronics.
- Automation is part of Mechatronics.

Mandal et al., 2015

Question 1

Automation is part of_____.

a) robotics
b) modern-day machinery
c) Mechatronics
d) None of the above

Automation Explained: Devices . . .

Robot

Automation Cell

Programmable Logic Controller

Automation Explained: Integrated Automation Systems . . .

- Manufacturing systems that integrate a computer-controlled, digitally process and coordinated system is known as an Integrated Automation System (IAS).
- An IAS include
  a) robots
  b) conveyor or motion-controlled transporter
  c) PLCs or Programmable Automation Controllers
  d) Sensors and Actuators

Automation Explained: Integrated Automation Systems

Parts-Sorting

Cookie – Sorting/Baking

Automotive Manufacturing

Automation Explained: Sensors and Actuators

- Sensors and actuators occasionally work in tandem.
- A sensor monitors conditions and signals when change occurs.
- An actuator receives a signal and acts.
  a) Performs an action
  b) The action performed is movement in a mechanical machine.
- Key differences between sensors and actuators:
  a) Sensors are placed at input points in automation systems
  b) Actuators are placed at output points of automation systems.
  c) PLCs or Programmable Automation Controllers (PACs) receive sensor data and provide control signals to actuators.

Image Source: ManagerPlus

Question 2

A key difference between sensors and actuators:

a) Actuators are placed at input points automation systems.

b) Sensors are placed at the output points of automation systems.

c) PLCs or Programmable Automation Controllers (PACs) receive sensor data and provide control signals to actuators.

d) None of the above

Automation Explained: Sensors and Actuators

Stepper Motor-
Threaded Rod

Proximity Sensor

Image Source: Electrical Technology

Automation Explained: Sensors and Actuators. . .

Fan Controller: Sensor and Actuator Functional Relationship

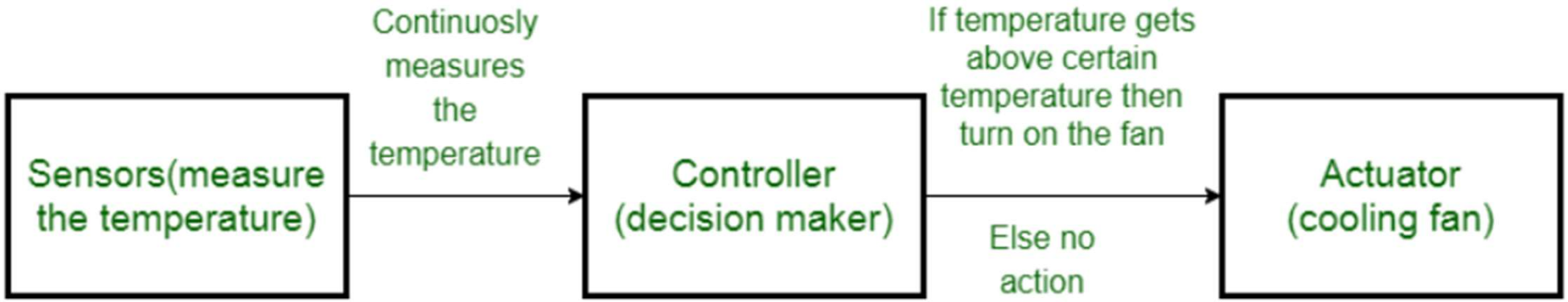


Image Source: GeeksForGeeks

Programmable Logic Controllers: Description

- The common choice for automation systems management and control is the Programmable Logic Controller (PLC).
- The PLC has its origins in electromechanical relay-based wiring controls.
  a) Electromechanical relays allow power to be switched to actuators.
  b) Low-level logic-based circuits were created with electromechanical relays.
- Programmed PLCs manage the processes and manufacturing operations of automation systems.
- Two parts of a PLC
  a) Hardware.
  b) Software

Mandal et al., 2015

Programmable Logic Controllers: Core Components

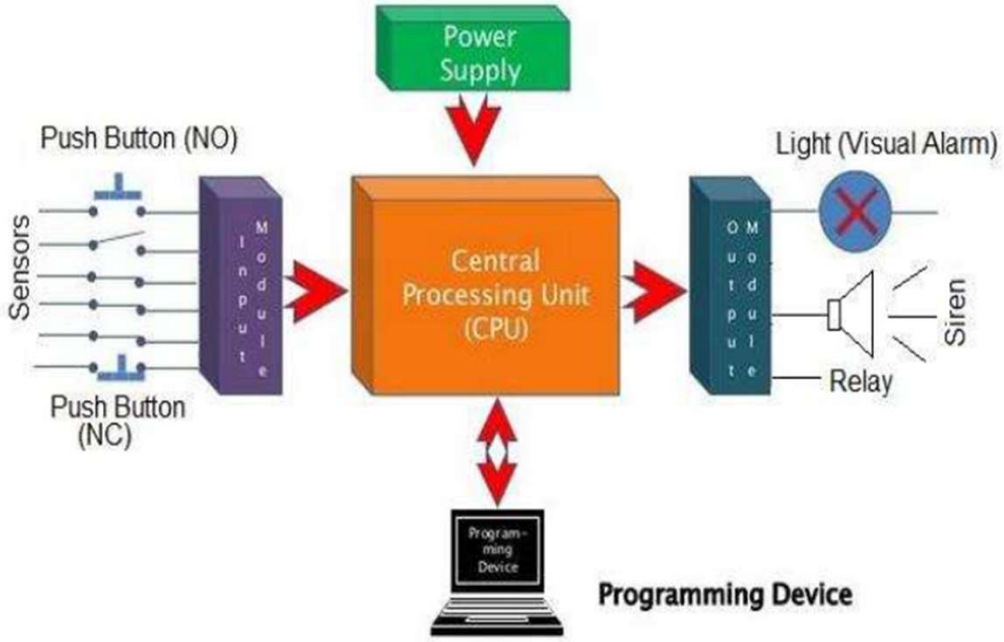


Fig. 2: Block diagram of PLC system

Mandal et al., 2015

Programmable Logic Controllers: Core Components . . .

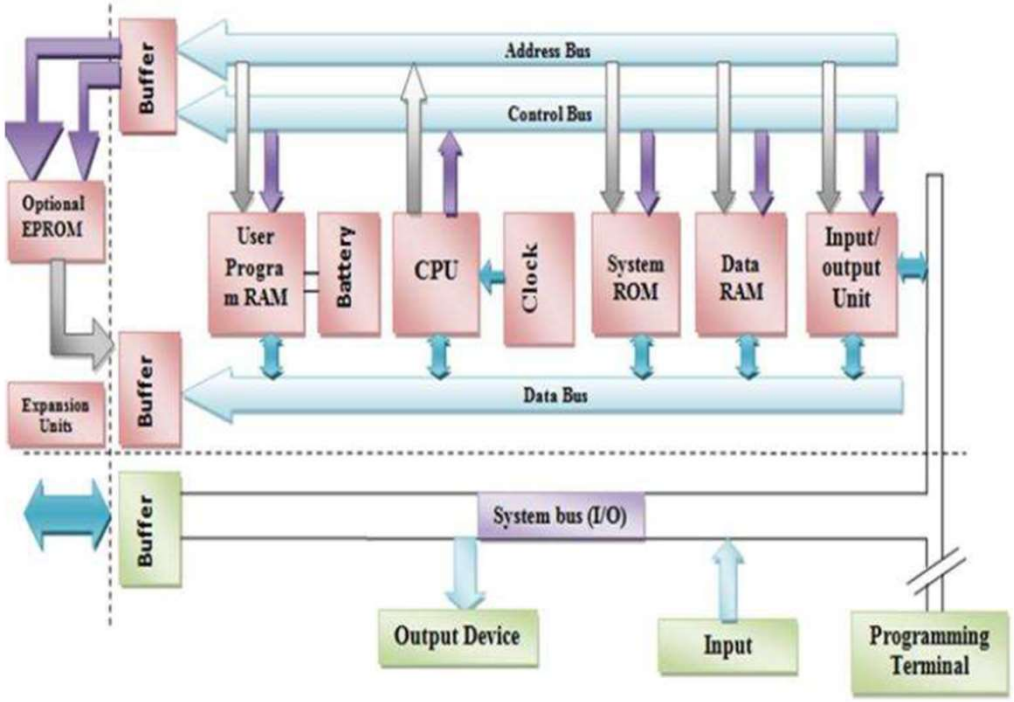


Fig. 3: Block diagram of PLC CPU architecture

Mandal et al., 2015

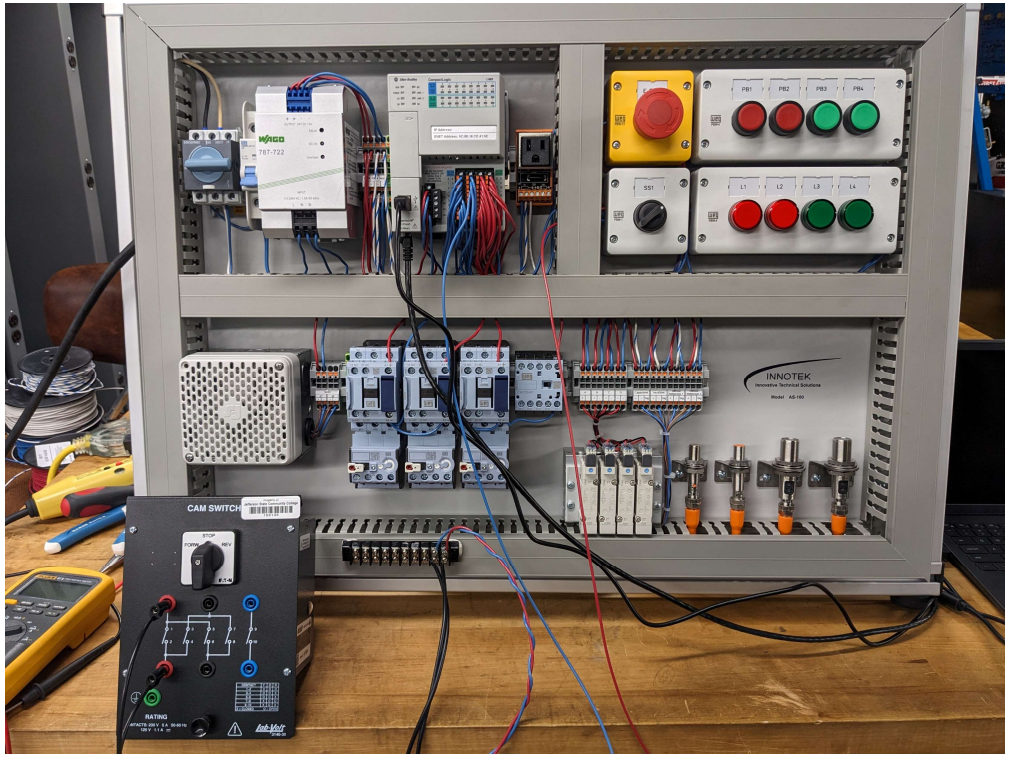
Question 3

In reviewing the Figure 3 block diagram, which subsystem block drives the CPU?

a) Battery
b) Buffer
c) Clock
d) None of the above

Programmable Logic Controllers: Core Components . . .

PLC-Based
Automation Trainer



Mandal et al., 2015

Programmable Logic Controllers: Core Components . . .

TECHNICAL ARTICLE

Turn a Raspberry Pi Into a PLC Using OpenPLC

February 21, 2024 by Dr. Don Wilcher

Using a Raspberry Pi and the OpenPLC software platform, create a simple PLC that can be programmed in ladder diagrams with remote access and I/O monitoring dashboards.

OpenPLC provides a control engineering development platform that transforms various microcontrollers into programmable logic controllers. OpenPLC is compatible with platforms including the Arduino Uno, ESP32, and RP2040, and even single-board computers like the Raspberry Pi can be used as a PLC with the editor, a runtime engine, and a web server.

This project article will explain the steps used to create a PLC with a Raspberry Pi using OpenPLC.

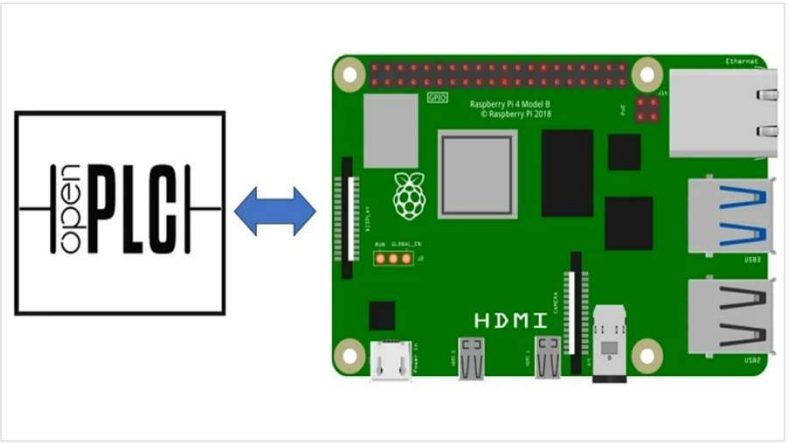
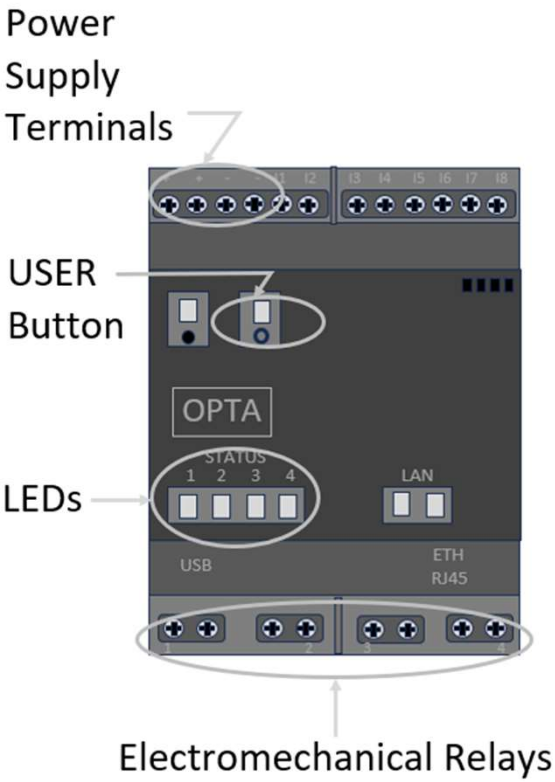


Figure 1. *OpenPLC can be implemented on a Raspberry Pi*

Control.com

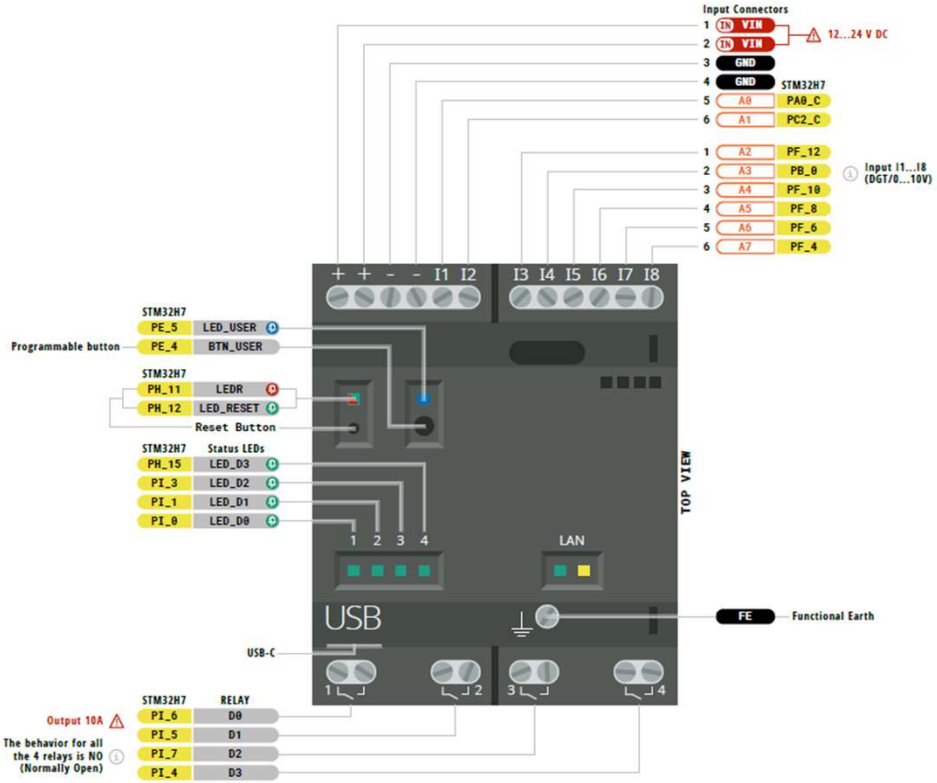
Brief Overview of the Arduino Opta:

The Arduino Opta Basic I/O and Power Supply Devices and Physical Connections



Brief Overview of the Arduino Opta: Pin Out

Pinout for the Arduino Opta



Brief Overview of the Arduino Opta: Pin Out. . .

Pinout Table for the Arduino Opta

Terminal/Device	Input/Output Pin Designator
I1	A0
I2	A1
I3	A2
I4	A3
I5	A4
I6	A5
I7	A6
I8	A7
Programmable Button	BTN_USER
Button	LED_USER
LEDR	LEDR
LED_RESET	LED_RESET
Reset Button	Reset_Button
Status LED1	LED_D0
Status LED2	LED_D1
Status LED3	LED_D2
Status LED4	LED_D3
Relay 1	D0
Relay 2	D1
Relay 3	D2
Relay 4	D3

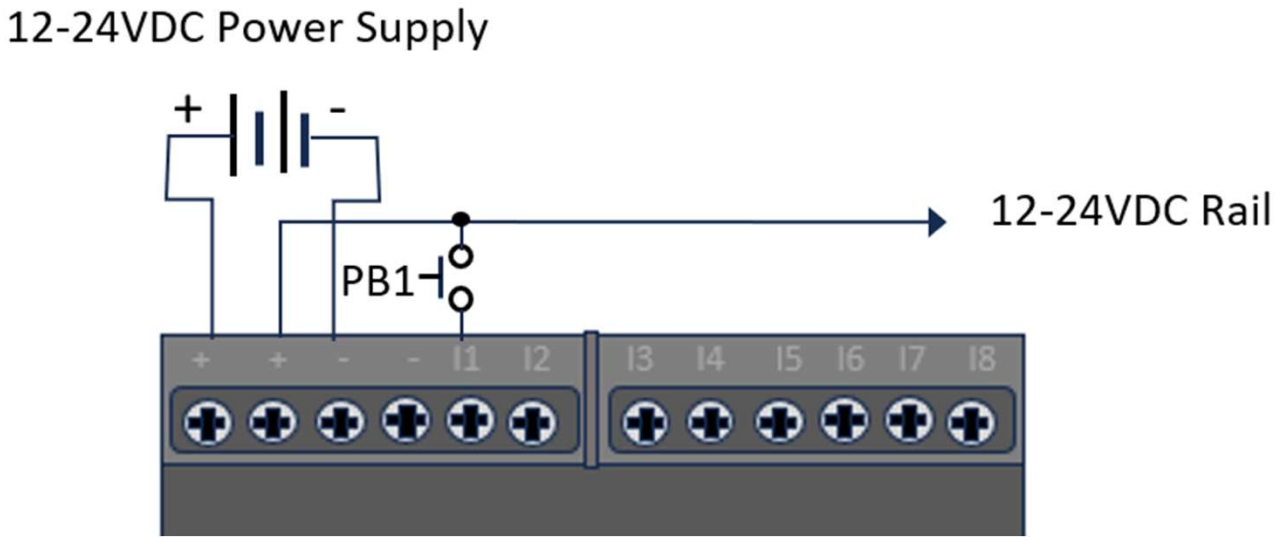
Question 4

In reviewing slide 29, Pin Designator A7 aligns with what Terminal?

a) I1
b) I2
c) I4
d) I8

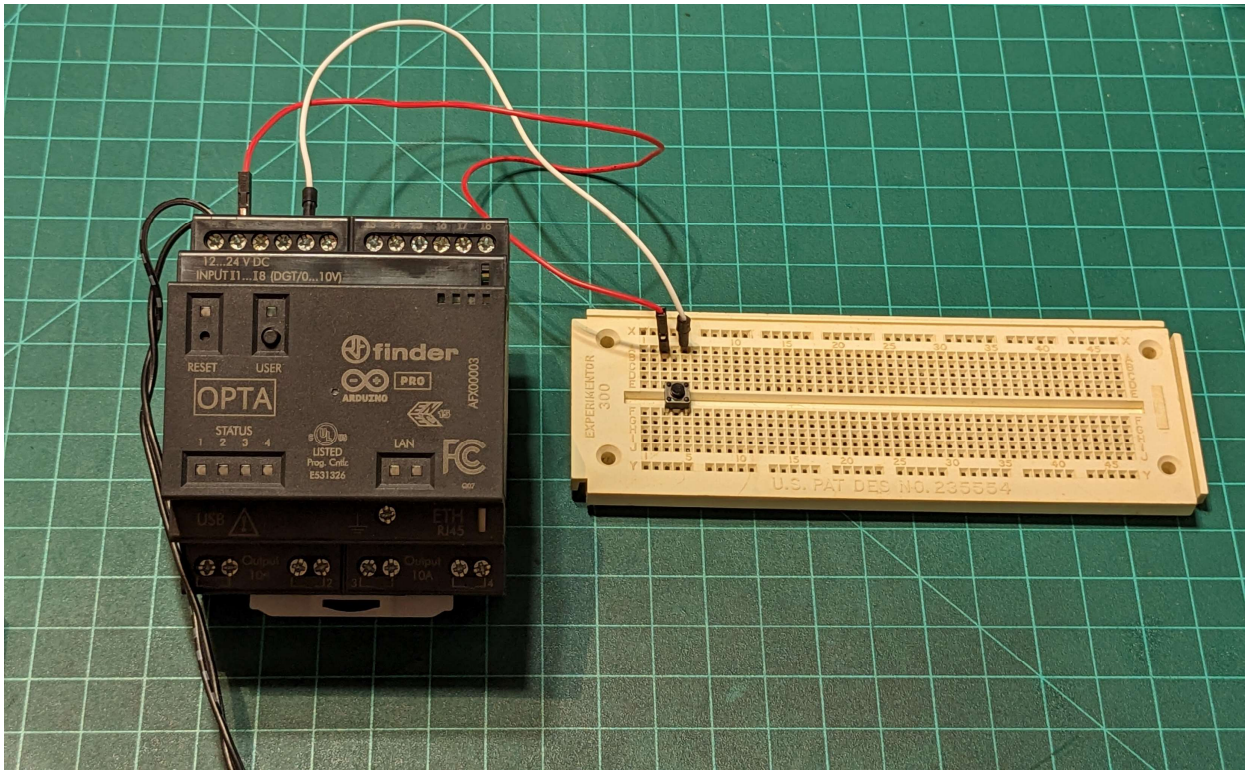
Brief Overview of the Arduino Opta: Accessing LEDs, Relays, and USR Button

Electrical Wiring Diagram of DC Power Supply and Pushbutton (PB1) switch to the Arduino Opta



Brief Overview of the Arduino Opta: Accessing LEDs, Relays, and USR Button. . .

Actual Electrical Wiring of DC Power Supply and Pushbutton (PB1) switch to the Arduino Opta



Brief Overview of the Arduino Opta: Accessing LEDs, Relays, and USR Button. . .

LED_Relay_PB_Test Code: Pressing PB1 switch will turn ON Status LED1 and Relay 1.

File Edit Sketch Tools Help

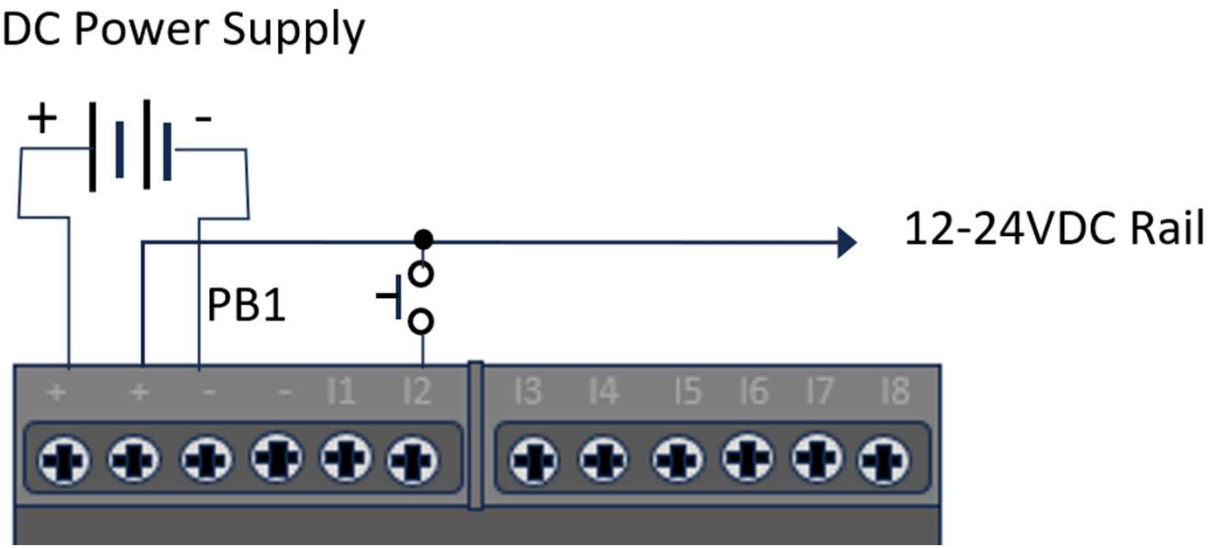
Opta

LED_Relay_PB_Test.ino

```
int buttonstatus = 0;   // variable to read pushbutton switch data
int Button = A0;        // A0 (Arduino Opta Core pin) which is physically wired to I1 terminal

// setup of Arduino Opta I/O
void setup() {
  pinMode(Button, INPUT);
  pinMode(D0, OUTPUT);
  pinMode(LED_D0, OUTPUT);
  pinMode(LED_D3, OUTPUT);

}

void loop() {
buttonstatus = digitalRead(Button);  // read Button data an store it into buttonstatus variable
  if (buttonstatus== HIGH) {        // if buttonstatus is HIGH, turn on LED1 and relay 1
    digitalWrite(LED_D0, HIGH);
    digitalWrite(LED_D3, HIGH);
    digitalWrite(D0, HIGH);
  }
  else{
    digitalWrite(LED_D0, LOW);    // if buttonstatus is LOW, turn OFF LED1 and relay 1
    digitalWrite(LED_D3, LOW);
    digitalWrite(D0, LOW);
  }

}

```

Brief Overview of the Arduino Opta: Accessing LEDs, Relays, and USR Button. . .

PB1 switch wired to Terminal I2 on Arduino Opta.



Brief Overview of the Arduino Opta: Accessing LEDs, Relays, and USR Button. . .

LED_Relay_PB1_Test_v2 Code:
Pressing PB1 switch will turn ON Status LED1, LED 3, and Relay 1.

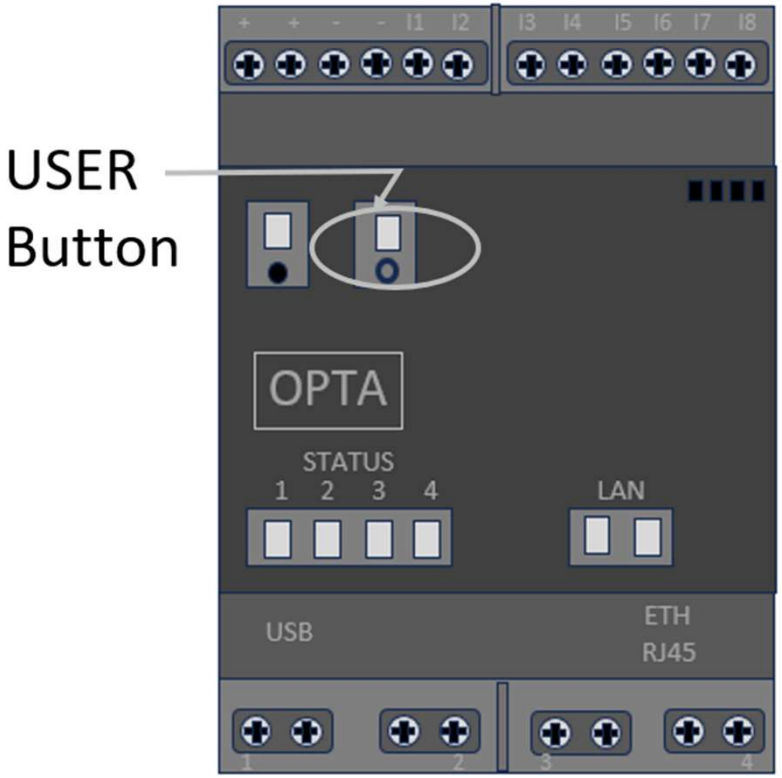
Opta

LED_Relay_PB_Test_v2.ino

```
1   int buttonstatus = 0;   // variable to read pushbutton switch data
2   int Button = A1;       // A1 (Arduino Opta Core pin) which is physically wired to I2 terminal
3
4   // setup of Arduino Opta I/O
5   void setup() {
6     //pinMode(BTN_USER, INPUT);
7     pinMode(Button, INPUT);
8     pinMode(D0, OUTPUT);
9     pinMode(LED_D0, OUTPUT);
10    pinMode(LED_D3, OUTPUT);
11
12  }
13
14  void loop() {
15  buttonstatus = digitalRead(Button);  // read Button data an store it into buttonstatus variable
16    if (buttonstatus== HIGH) {        // if buttonstatus is HIGH, turn on LED1, LED3, and relay 1
17      digitalWrite(LED_D0, HIGH);
18      digitalWrite(LED_D3, HIGH);
19      digitalWrite(D0, HIGH);
20    }
21    else{
22      digitalWrite(LED_D0, LOW);    // if buttonstatus is LOW, turn OFF LED1, LED3, and relay 1
23      digitalWrite(LED_D3, LOW);
24      digitalWrite(D0, LOW);
25    }
26
27  }
```

Brief Overview of the Arduino Opta: Accessing LEDs, Relays, and USR Button. . .

The USER Button will turn ON Status LED1, LED 3, and Relay 1.



Brief Overview of the Arduino Opta: Accessing LEDs, Relays, and USR Button. . .

LED_Relay_BTN_USER Test Code:
Pressing USER PB switch will turn ON Status LED1, LED 3, and Relay 1.

LED_Relay_BTN_USER__Test.ino

```
1   int buttonstatus = 0;   // variable to read pushbutton switch data
2   //int Button = A1;       // A1 (Arduino Opta Core pin) which is physically wired to I2 terminal
3
4   // setup of Arduino Opta I/O
5   void setup() {
6     pinMode(BTN_USER, INPUT);
7     //pinMode(Button, INPUT);
8     pinMode(D0, OUTPUT);
9     pinMode(LED_D0, OUTPUT);
10    pinMode(LED_D3, OUTPUT);
11
12  }
13
14  void loop() {
15  buttonstatus = digitalRead(BTN_USER);  // read BTN_USER data an store it into buttonstatus variable
16    if (buttonstatus== LOW) {        // if buttonstatus is LOW, turn on LED1, LED3, and relay 1
17      digitalWrite(LED_D0, HIGH);
18      digitalWrite(LED_D3, HIGH);
19      digitalWrite(D0, HIGH);
20    }
21    else{
22      digitalWrite(LED_D0, LOW);    // if buttonstatus is LOW, turn OFF LED1, LED3, and relay 1
23      digitalWrite(LED_D3, LOW);
24      digitalWrite(D0, LOW);
25    }
```

Question 5

Which line of code allows for the USER Button status to be read?

a) buttonstatus = digitalWrite(USER_Button);
b) buttonstatus = digitalWrite(USER_BTN);
c) buttonstatus = digitalRead(USER_Button);
d) buttonstatus = digitalRead(BTN_USER);

Thank you for attending

Please consider the resources below:

Liao, C.C. (2007). *Programming and application of S7-200 plc* (3rd ed.). Mechanical Industry Press.

Mandal. R, Maity, T., Prasad, G.M., & Verma, R. P. (2015). Automation of underground coal mines using plc. *Journal of Mines, Metals, and Fuels*, 174 – 181. https://www.researchgate.net/publication/317038146_Automation_of_underground_coal_mines_using_PLC#:~:text=This%20paper%20presents%20applications%20of,flammable%20gases%20exceeds%20permissible%20limit

Wilcher. D. (2024, February 21). *Turn a raspberry pi into a plc using openplc*. https://control.com/technical-articles/turn-a-raspberry-pi-into-a-plc-using-openplc/

Course_Lab_project_code.zip folder: Github Repository: https://github.com/DWilcher/DesignNews-WebinarCode

CEC Continuing Education Center

DesignNews

Thank You

Sponsored by

DigiKey

informa markets