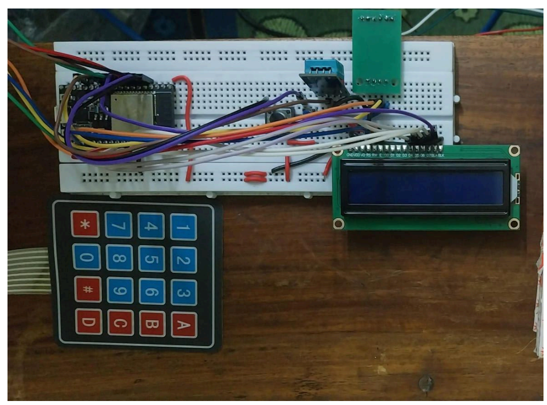
Smart Shelf.

Introduction

This project presents a Smart Shelf system designed to enhance inventory management and environmental monitoring by integrating a load cell and a temperature and humidity sensor. The load cell measures weight changes in real-time, enabling precise monitoring of shelf content usage and restocking needs. The temperature and humidity sensor tracks environmental conditions, ensuring optimal storage for sensitive items. By combining these functionalities, the Smart Shelf provides a comprehensive solution for smart storage applications, suitable for homes, warehouses, and retail environments. The system emphasizes automation, efficiency, and reliability, paving the way for advanced inventory and environmental management solutions.

Components required

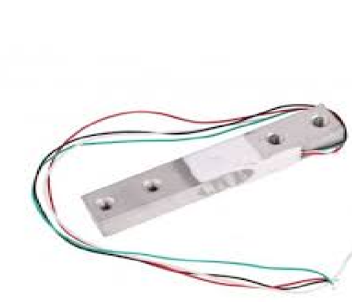
1] Load cell
2] HX 711
3] ESP 32 (microcontroller)
4] Keypad
5] Push button switch
6] Humidity and Temperature Sensor (DHT11)
7] LCD Display
8] Wires (jumper and simple wires)
9] Breadboard



Specifications of components:

Load cell :

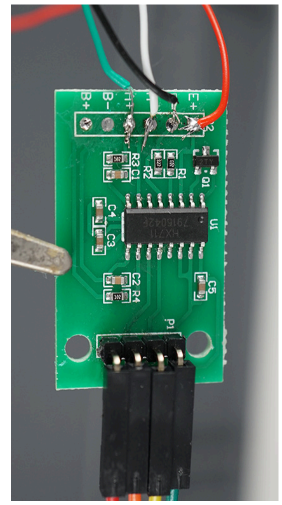
Ours is a single-point load cell. Single-point load cells are usually compact and consist of a single strain gauge (or sometimes a pair) mounted on a beam. This beam is designed to bend or flex when a load is applied, generating a strain that the strain gauge detects.
They measure the force applied at a single central point on a platform. They are typically mounted in a way that distributes the weight evenly across the cell.



HX711 IC

The HX711 communicates with the microcontroller using a two-wire interface (Clock and Data). You need to solder header pins on the GND(ground), DT, SCK, and VCC pins to connect to the ESP32.

About pins:



- **GND (Ground):**
  Connect this to the ground of your power supply. It's the reference voltage level for the chip and other components connected to it.
  DT (Data):
  This is the data output pin. The HX711 outputs data here, which can be read by a microcontroller (like an Arduino). The DT pin is often connected to a digital input pin on the microcontroller.

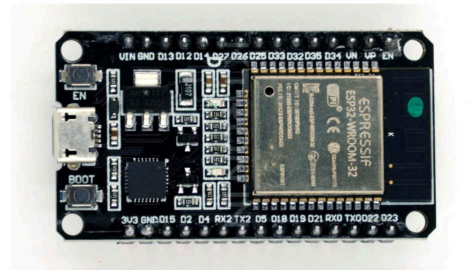
- **SCK (Clock):**
  This is the clock input pin. It's used to trigger the reading of data from the HX711. A microcontroller sends pulses to this pin to control the data transfer.
  The SCK (or Clock) pin in the HX711 works as a timing signal, controlling the rate at which data is read from the chip.

- **VCC (Voltage Common Collector or Power Supply):**
  This is the power supply pin. Typically, it's connected to 2.6V to 5.5V to power the chip.
  Working mechanism:
    - **Data Ready Signal:**
      The HX711 continuously samples data from the connected load cell (or analog sensor). Once it has a new 24-bit data value ready, it signals this by setting the DT (Data) pin low.

    - **Clock Pulses on SCK**:
      After seeing the DT pin go low, the microcontroller sends exactly 25 pulses on the SCK pin to retrieve the data. Each pulse shifts out one bit of data on the DT line, starting with the most significant bit (MSB) and ending with the least significant bit (LSB).

    - **Data Transmission:**
      With each pulse:
        - The HX711 outputs one bit of the 24-bit data on the **DT** pin.
        - The microcontroller reads this bit and builds the 24-bit result.
    - **Connections:**
    - The HX711 amplifier communicates via a two-wire interface. You can connect it to any GPIOs of your chosen microcontroller. You can use any other suitable pins.

ESP 32



Dual-Core Processor

The ESP32 has a dual-core 32-bit LX6 processor based on the Xtensa architecture, running at up to 240 MHz, making it quite powerful for a microcontroller.

1. Key Features of GPIO in ESP32:

1. **Input Mode:**
    - GPIO pins can read digital signals (high or low). When set as input, they can detect the voltage level on the pin, typically either 0V (low) or 3.3V (high).
    - Some pins also support **analog input**, allowing the measurement of voltages in the range from 0V to 3.3V.

2. **Output Mode:**
    - GPIO pins can output digital signals (high or low) to control devices such as LEDs, relays, or transistors.
    - Pins can also provide **PWM output** (Pulse Width Modulation) to control the brightness of LEDs or the speed of motors.

3. **Multiple Functions:**
   Many of the GPIO pins on the ESP32 are multifunctional, meaning they can be used for various purposes. This includes digital I/O, analog input, PWM, I2C, SPI, UART, and more.
    - For example, GPIO pins can be configured for **I2C communication** (SCL, SDA), **SPI communication** (MISO, MOSI, SCK, CS), **UART** (TX, RX), or **Touch sensor** functionality.
4. **Input Pull-Up/Pull-Down:**
    - GPIOs can be configured with internal pull-up or pull-down resistors, which is useful when connecting a GPIO pin to a button or other digital input device. This ensures a stable signal when no active voltage is present.

5. **Interrupts:**
    - GPIOs can be set up to trigger interrupts. For example, a pin can be configured to trigger an interrupt when the input signal changes from low to high (rising edge) or from high to low (falling edge), enabling more responsive applications.

   Note:
   Not all GPIO pins have the same functions. Some pins may be reserved for certain purposes (e.g., flash memory, boot mode). It's important to consult the ESP32 datasheet or pinout diagram to determine which GPIO pins are available for general use in a given application.

6. **Analog and Digital Peripherals**
    - **ADC**: Two 12-bit analog-to-digital converters (ADCs), which support up to 18 channels, allowing it to read analog signals from sensors.

   - **DAC**: Two 8-bit digital-to-analog converters (DACs) for generating analog output. DAC outputs may also be used for controlling the output (brightness) of the sensor.
   - **PWM**: Capable of generating PWM signals, useful for controlling motors, LEDs, etc

7. **Reading Sensor Data**

   Many sensors output analog voltages that vary based on certain physical parameters. For example:

   - **Light sensors (LDR)** change resistance based on light intensity, which can be converted to a voltage.
   - **Gas sensors** or **humidity sensors** also often provide analog outputs.

   The ADC allows the ESP32 to read these voltages and convert them into digital values for processing in your application.

2. Wi-Fi and Bluetooth Connectivity

- **Wi-Fi**: Supports 802.11 b/g/n standards, allowing it to connect to Wi-Fi networks or act as an access point.
- **Bluetooth**: Includes Bluetooth 4.2 and BLE (Bluetooth Low Energy), allowing easy integration with smartphones and other BLE devices.

Keypad

A 4x4 keypad matrix is a grid of buttons arranged in 4 rows and 4 columns, commonly used in embedded systems for input devices like calculators, locks, and other keypad-based interfaces. The keypad typically has 16 buttons, where each button represents a number, letter, or symbol, and pressing a button connects a row with a column.

Working Details

1. Structure and Layout

- A 4x4 keypad has 16 keys organized in 4 rows and 4 columns, with each button at the intersection of a specific row and column.
- The rows (R1, R2, R3, R4) and columns (C1, C2, C3, C4) are typically connected to an external microcontroller via GPIO pins.

2. Wiring and Connections

- The rows and columns of the keypad are connected to the microcontroller. For a simple setup:
   - **Rows** might be connected to the microcontroller as input pins.
   - **Columns** are connected as output pins, or vice versa, depending on the desired scanning technique.

3. Scanning Method

- To detect which button is pressed, the microcontroller follows a **scanning process**:
    - **Set one column to LOW** (or HIGH) at a time and keep other columns in the opposite state.
    - **Read each row** to check for a signal change:
        - If a button in that column is pressed, it will connect the row and column, changing the signal on that row pin.
    - Repeat this for each column, checking rows each time.
    - By identifying the active row and column, the microcontroller pinpoints which button was pressed.

- **4. Debouncing**
    - When pressing a button, mechanical switches may produce a small amount of noise, causing multiple signals in a short time. To avoid detecting false presses, a **debouncing algorithm** (either in hardware or software) ensures stable input before registering a button press.

Push-button switch:

Connections:

If the button does not come with its resistor, you can connect it as follows:
GPIO's pins < - - ->Button < - - - > GND.

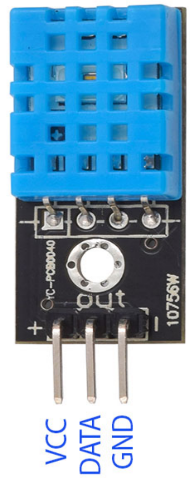
Humidity and temp sensor

DHT11 is a combined humidity and temperature sensor. Sensors like the DHT11 typically provide both:

1. Relative Humidity (RH):
    - Measures the amount of water vapour in the air relative to the maximum amount the air can hold at a specific temperature.
    - Output is typically in percentage (%RH).
2. Temperature:
    - Measures the ambient temperature.
    - Output can be in Celsius (°C) or Fahrenheit (°F), depending on configuration.

The DHT11 humidity sensor is typically used to measure relative humidity in various industrial and environmental settings. Here's an overview of its general characteristics and applications:

1. **Humidity Measurement**: It usually measures relative humidity in a range from around 0 to 100%, with good accuracy and stability.
2. **Temperature Compensation**: These sensors often have temperature compensation, ensuring that humidity readings are not affected by changes in ambient temperature.



Communication Protocols: It may use protocols such as I2C or SPI, allowing easy integration with microcontrollers or PLCs (Programmable Logic Controllers) for real-time monitoring and data logging.**Features of DHT11**

1. **Humidity Measurement:**
    - Range: 20% to 90% Relative Humidity (RH)
    - Accuracy: ±5% RH
2. **Temperature Measurement:**
    - Range: 0°C to 50°C
    - Accuracy: ±2°C
3. **Signal Output:**
    - Digital data is transmitted using a proprietary one-wire communication protocol.
4. **Power Requirements:**
    - Voltage: 3.3V to 5V
    - Low power consumption
5. **Sampling Rate:**
    - One reading per second (1 Hz)
6. **Size:**
    - Compact design, typically in a plastic casing for environmental protection.

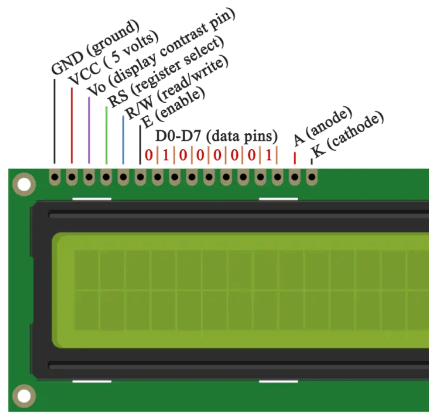
Pin Configuration:

1. **VCC:** Connect to 3.3V or 5V.
2. **GND:** Connect to ground.
3. **DATA:** Connect to a digital input pin on the microcontroller, typically with a pull-up resistor (10kΩ)

LCD DISPLAY

An LCD screen is an electronic display module that uses liquid crystal to produce a visible image. The 16×2 LCD display is a very basic module commonly used in circuits. The 16×2 translates a display of 16 characters per line into 2 such lines. In this LCD, each character is displayed in a 5×7 pixel matrix.

Input/Output Interface Pins



- **Power Supply Pins:** VCC and GND to power the LCD.
- **Data Pins (D0–D7):** 8-bit parallel data pins for sending data/commands (D4–D7 used in 4-bit mode).
- **RS (Register Select) Pin:** Selects between command and data register.
- **RW (Read/Write) Pin:** Specifies read or write mode.
- **Enable Pin:** Acts as a trigger for reading/writing data.

Conclusion:

Smart Shelf system integrates real-time weight monitoring and environmental condition tracking to deliver an innovative solution for efficient inventory and storage management. By using a load cell and a temperature and humidity sensor, this system ensures precise monitoring of usage patterns, timely restocking, and optimal storage conditions for sensitive items.

Applications:

The Smart Shelf system has a wide range of applications across various domains, including:

Home Automation: Monitoring pantry or refrigerator stock levels.
Ensuring ideal storage conditions for perishable or sensitive items like medicines or electronics.
Retail Environments: Real-time tracking of product availability on shelves.
Automated restocking alerts to reduce out-of-stock situations.
Industrial Applications: Monitoring raw materials or finished product stocks.
Ensuring storage conditions for temperature-sensitive equipment or chemicals.
Libraries and Archives: Monitoring the weight of stacks to assess book lending or detect unbalanced loads. Ensuring the right environmental conditions for preserving old books or documents.