Princess Sumaya University for Technology

King Abdullah II Faculty of Engineering

Princess Sumaya University for Technology
جامعة الأميرة سميّة للتكنولوجيا

EMBEDDED SYSTEMS PROJECT
SMART SANITARY SYSTEM

Authors:

Omar Muhanna 20210466
Ahmad Abu Hejleh 20210688
Kinan Radaideh 20220646

Supervisor:

Dr. Belal Sababha
Dr. Essam Qaralleh

Abstract
This project builds a touch-free trash can and a hands-free water dispenser. An infrared sensor on the trash can lid tells a small servo motor to open and close the bin. A second infrared sensor at the water spout turns on a small pump. An ultrasonic sensor keeps track of the tank's water level and lights an LED when it is low. A PIC16F877A microcontroller runs at 8 MHz and ties everything together. The lid opens in under 300 ms and the pump delivers the right amount of water depending on the potentiometer's setting. This low-cost, battery-powered design makes throwing away waste and washing hands more hygienic in homes, offices, or any public spot.

Introduction & Background

Maintaining good hygiene and avoiding contact with shared surfaces is important to help prevent the spread of germs. Common items like trash cans and water taps usually require touch, which can increase the risk of illness.

Our project is a *Smart Sanitary System* that includes two separate parts: a smart trash can and an automated water tank. Both are designed to work without needing physical contact, making hygiene easier and safer.

The 3D-printed smart trash can uses an infrared (IR) sensor to detect when a hand, or any object, is nearby. When triggered, a servo motor opens the lid automatically. This allows users to throw away trash without touching the bin.

The water tank system has an ultrasonic sensor placed at the top to measure the water level. Four LEDs light up to show how full the tank is. It also includes an IR sensor near the outlet that turns on a water pump when a hand is detected, making it easy to wash hands without pressing any buttons.

Together, these systems show how simple embedded technology can improve daily tasks, support better hygiene, and reduce waste. The design is affordable, practical, and can be used in both public and private spaces.

The Design

Mechanical Design
As the project consists of two interconnected subsystems, the water tank and trash can, it is best to study the mechanical design of each subsystem separately.

The trash can, designed in the Fusion software and 3D-printed, is comprised of a base (Object numbered 1 in Figure 1) and a lid (Object number 2 in Figure 1). The lid is attached to the base with a set of screws, which is attached to the servo motor by a crane (Object numbered 4). This attachment is shown, numbered 3 in Figure 1. Further figures show the trash can from different perspectives. Figure 2 shows the back view and Figure 3 shows the front view, with an infrared sensor attached by glue. This sensor detects the presence of the hand, activating the servo motor and opening the lid. The same components numbered in Figure 1 are visible in the other figures.

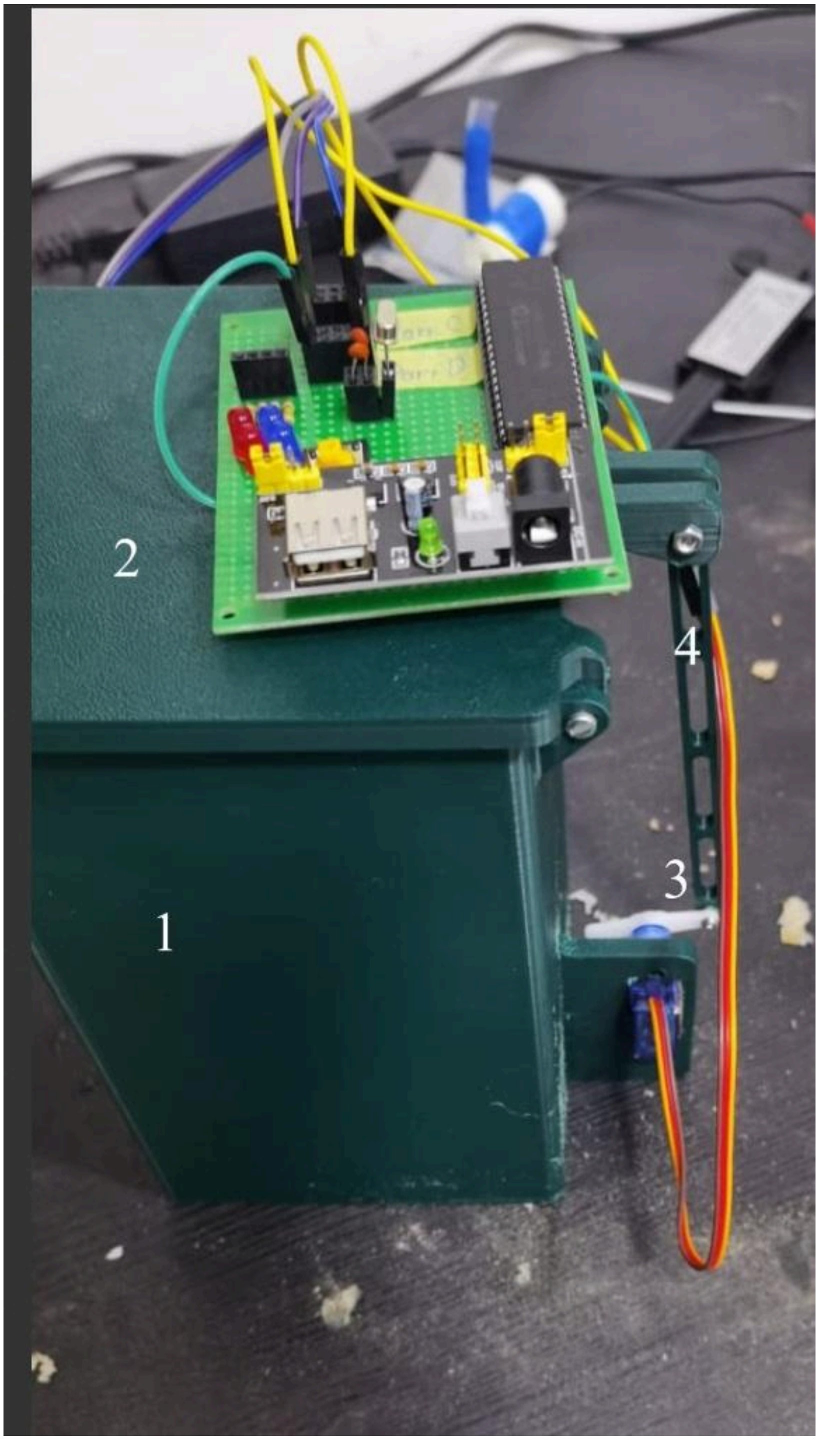

Figure 1

Figure 2 (back view)

Figure 3 (front view)

As for the water tank, a plastic jug is placed inside a wooden box that is solid only on one side, and is a frame on all three sides, as seen in Figure 4. At the bottom of the plastic jug is a water pump, numbered 1 in Figure 4. The water pump is also a motor, which begins pumping water when another IR sensor, visible in Figure 5 and numbered 1, detects motion. The front solid side is seen in Figure 6, through a hole in which protrudes a tube which begins at the water pump, which dispenses water for use. This tube is numbered 3 in Figure 4. At the top of the water tank is an ultrasonic sensor, numbered 4 in Figure 4, suspended by a wooden holder, which detects the water level and activates LEDs based on the level detected. Figure 7 shows the right side view, in which the tube is more clearly visible, starting at the water pump and going through a hole on the solid side, and Figure 8 shows the top view, where the ultrasonic sensor is clearer.

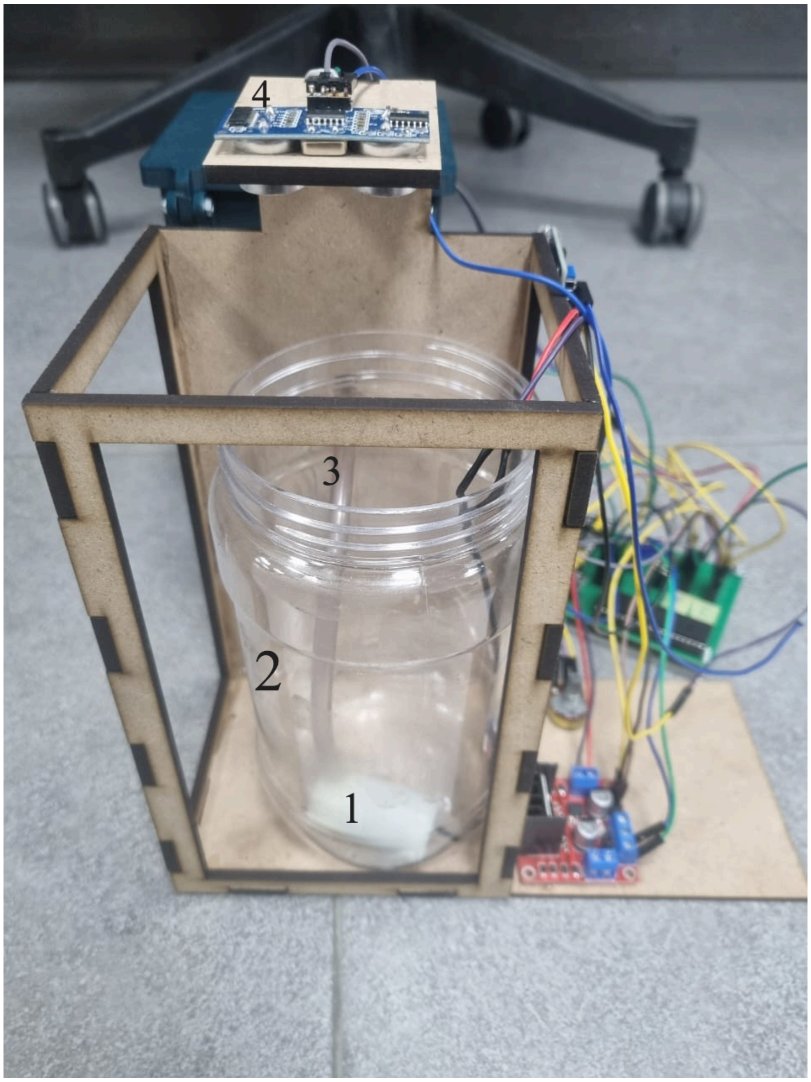

Figure 4 (back view)

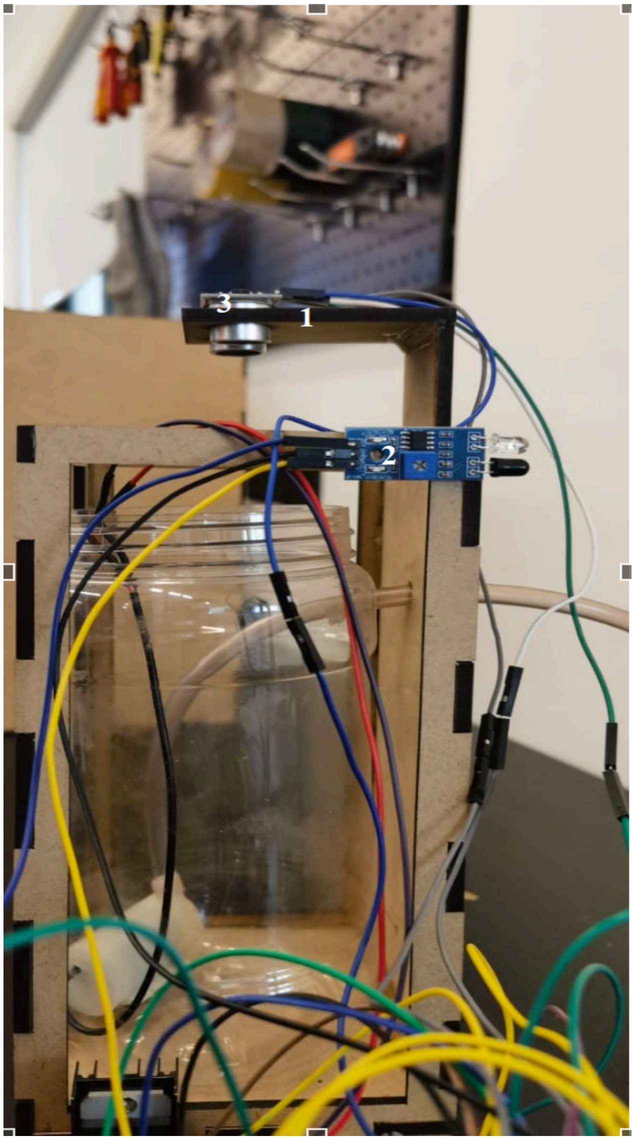

Figure 5 (left side view)

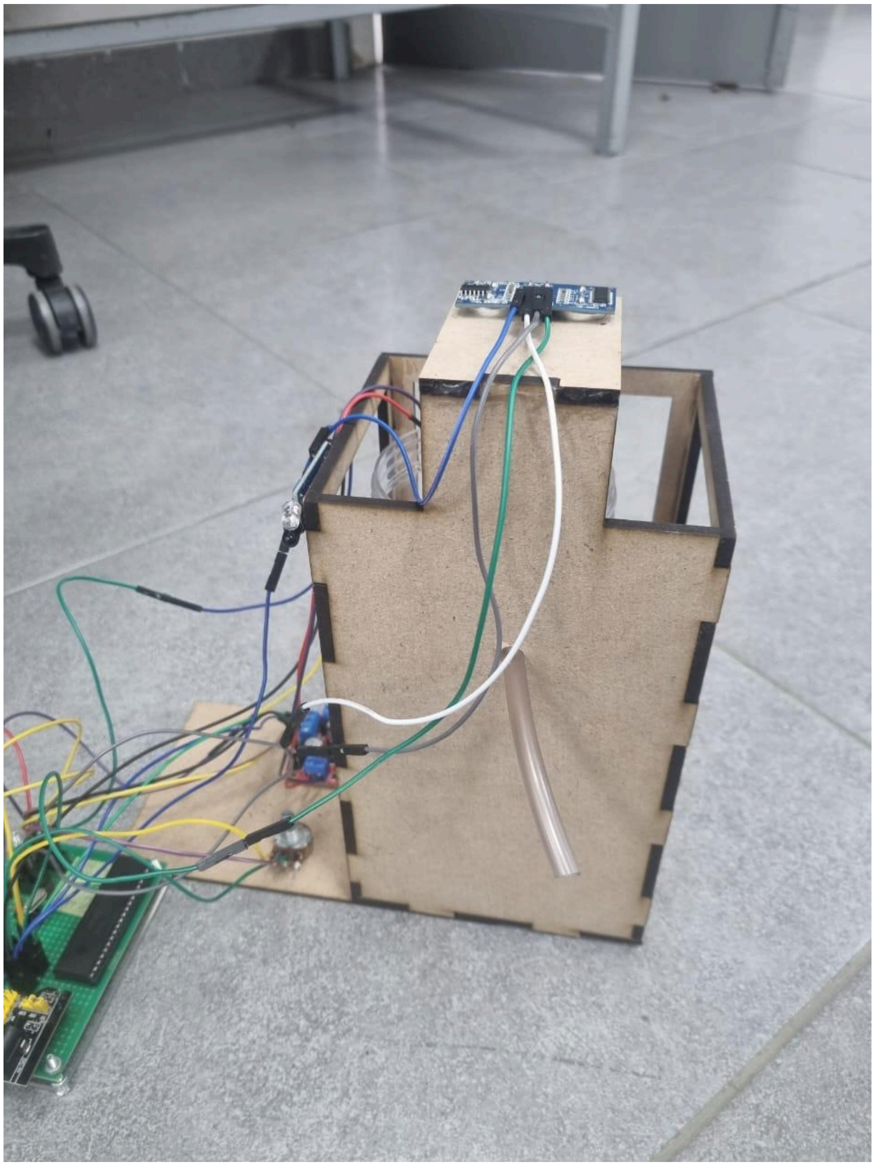

Figure 6 (front view)

Figure 7 (right side view)

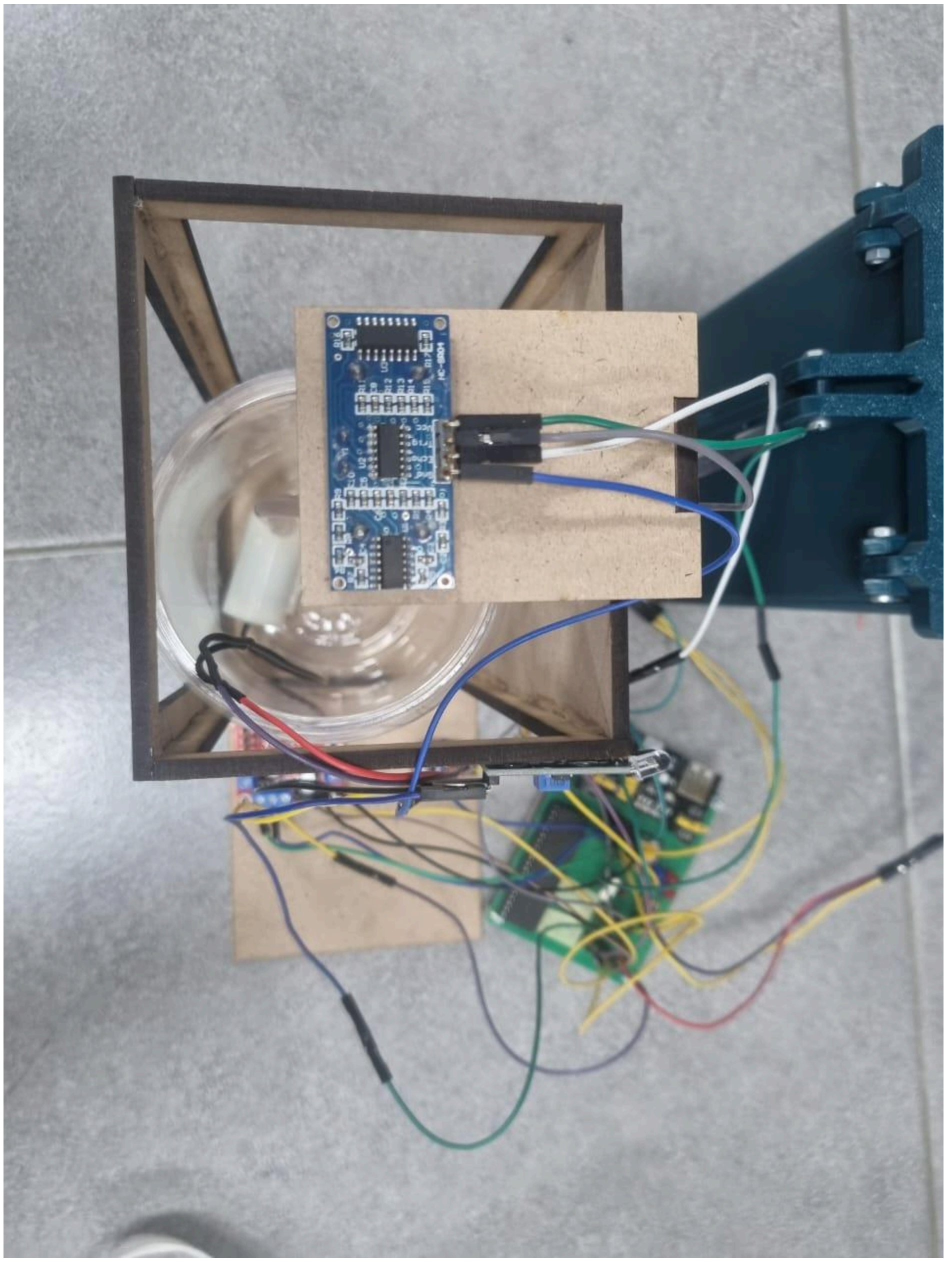

Figure 8 (top view)

Finally, Figures 9 & 10 shows both subsystems side-to-side as a single larger system. It shows the front view of the project as a whole and Figure 10 shows the back view.

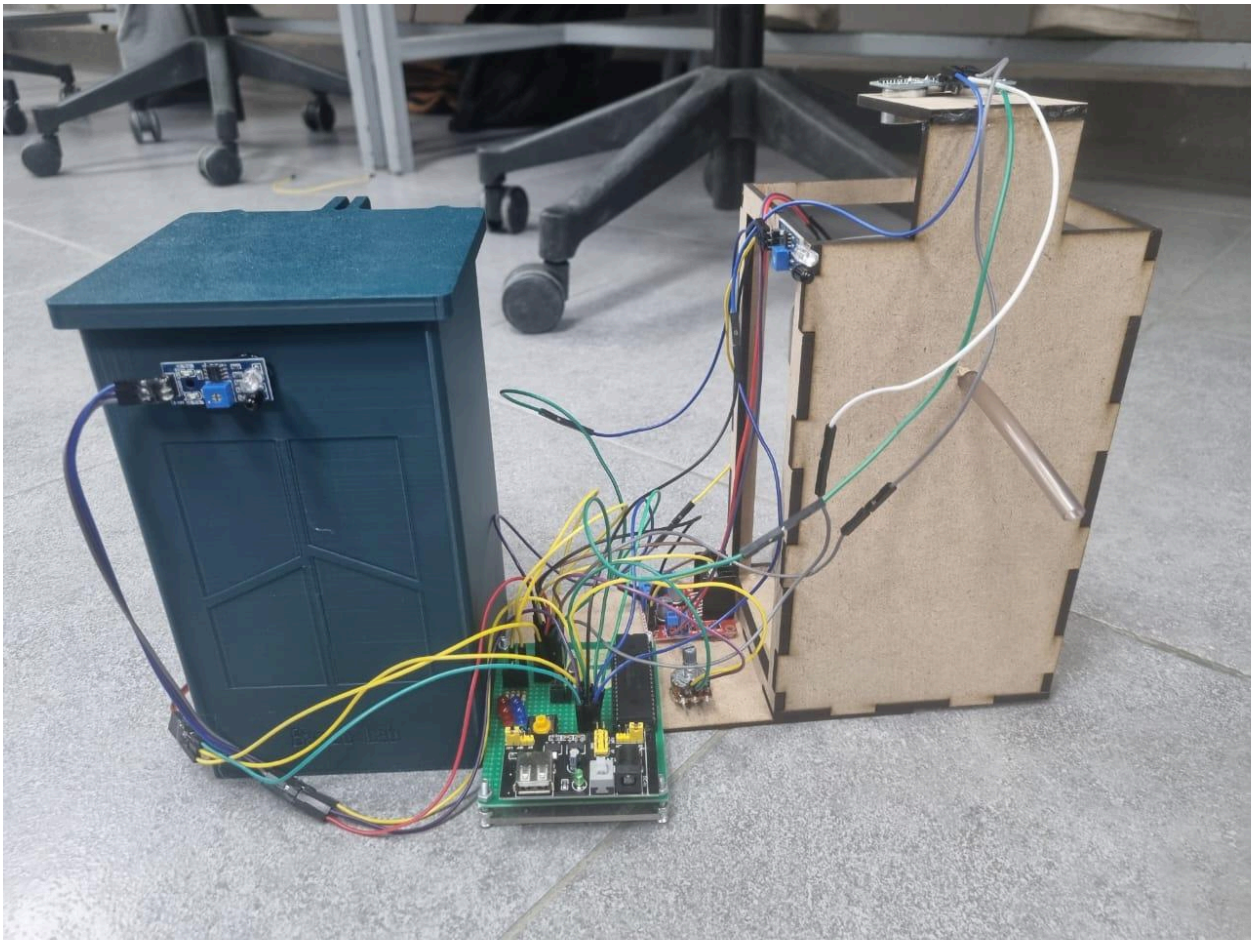

Figure 9 (front view)

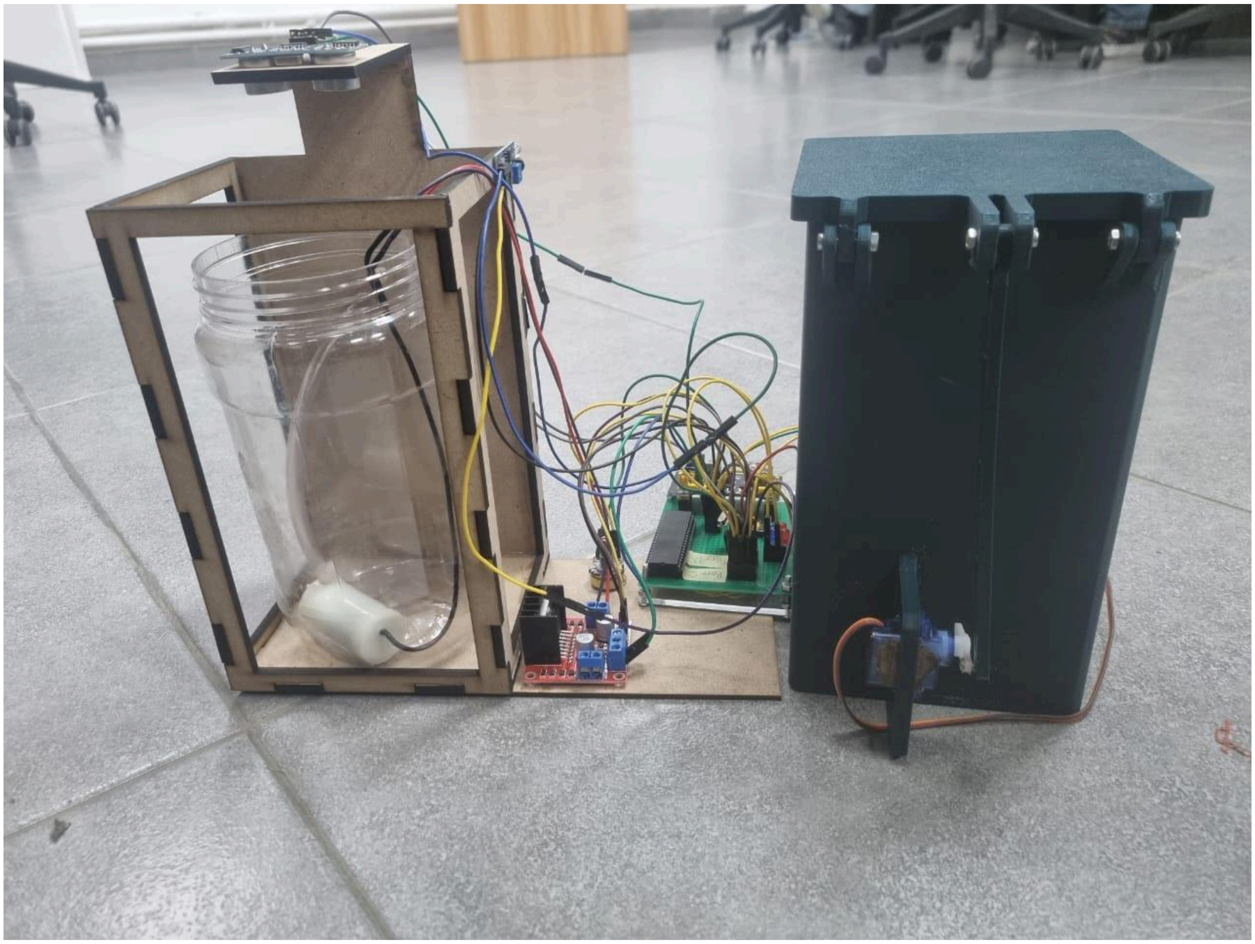

Figure 10 (back view)

Electrical Design

Figure 11 shows the circuit diagram, including sensors, actuators, ground and Vcc connections.

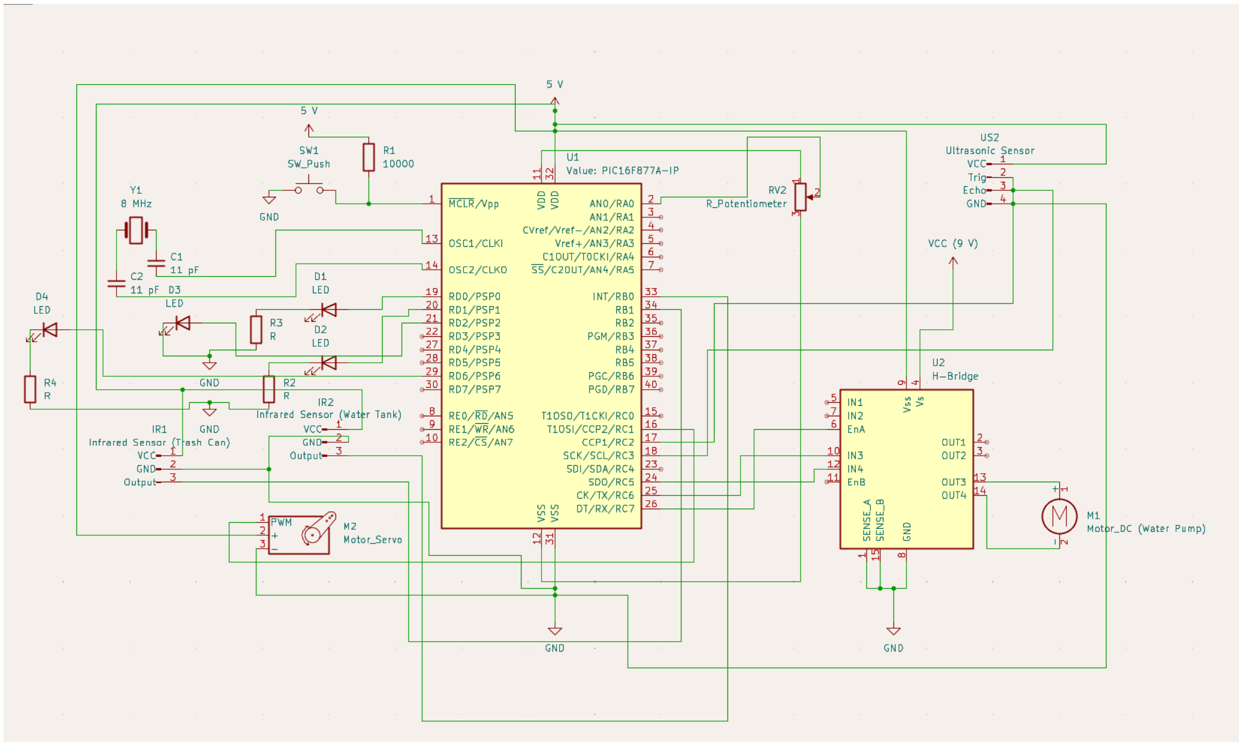

Figure 11 (Circuit Diagram)

Figure 11 shows the complete electrical schematic for the smart trash can and water‑tank subsystems. At its core is a PIC16F877A microcontroller (U1). Its two VDD pins (11, 32) connect to +5 V, and its two VSS pins (12, 31) go to ground. The MCLR/VPP pin (1) is held high through a 10 kΩ pull-up (R1) and can be pulled low by a push-button for reset. An 8 MHz crystal (Y1) on OSC1/CLKI (13) and OSC2/CLKOUT (14), each side loaded by an 11 pF capacitor (C1, C2), provides the system clock.

An infrared sensor (IR1) ties to RB2 and detects when a hand or object approaches. Its output feeds the PIC microcontroller. When IR1 triggers, CCP2 on RC1 (pin 16) generates a PWM signal that drives a servo motor (M2) mounted on PWM (pin 1), opening the lid without touch. As with all devices that need a power and ground, the servo's power and ground share the microcontroller's +5 V and GND rails, seen above as Vcc and below as Vss respectively.

A four-pin connector is shown here as an ultrasonic sensor (US2). The TRIG line comes from RC2 (pin 17), and the ECHO line returns to RC3 (pin 18), letting the microcontroller time echo pulses on Timer 1. Four LEDs display the water level, as seen in the previous section. A second IR sensor (IR2) on RBO watches for a hand under the spout.

When IR2 sees a hand, the microcontroller drives a DC pump (M1) powered by a separate 9 V rail. An H-bridge (U2) provides a 5 V power supply to enable the water pump to work. IN4/IN3 connect to

RC5/RC6 respectively, and ENA comes from RC7. Since SENSE_A and SENSE_B are not used, they go to ground together with the GND terminal.

There are four LEDs, each connected to a resistor and ground: one in each of the pins RD0, RD1, RD2, and RD6. These indicate the water level in the tank.

A potentiometer (RV2) on AN0/RA0 (pin 2) allows for the control of the speed and intensity of the water pump.

Together, these circuits let the system open the trash lid hands-free and dispense water only when a hand is present—offering simple, contact-free hygiene and waste handling.

Software Design

We have two fundamental pieces of code: one for the trash can and one for the water tank.

As for the trash can**,** our code runs on a PIC16F877A at 8 MHz and uses CCP1 (Capture/Compare/PWM) plus Timer 1 to generate a 50 Hz PWM signal on RC2. We drive a hobby servo between 0° and 180° by changing the high‑pulse width from 0.5 ms (1,000 timer ticks) to 2.5 ms (5,000 ticks), with a fixed 20 ms frame length. An interrupt service routine (ISR) fires on every compare match:

On the first match, it pulls RC2 low after the programmed pulse width (high phase).

On the second match, it pulls RC2 high for the rest of the 20 ms frame (low phase).

This ISR toggles a flag (highPhase) and reloads Timer 1 so the cycle repeats automatically.

In the main() function, we:

- Enable PORTB pull-ups so RB0 floats high when unpressed.
- Configure RB0 as input (IR sensor), RC2 as PWM output (servo), and RC6 as an LED indicator.
- Set up CCP1 for compare mode and turn on Timer 1 with no prescaler.
- Enable the CCP1 interrupt and global interrupts.

Inside the infinite loop, we simply read RB0:

- If RB0 is LOW (IR sensor sees a hand), we set pulseTicks to 5,000 (180°) and light the LED on RC6.
- If RB0 is HIGH, we set pulseTicks back to 1,000 (0°) and turn the LED off.

Because the ISR always uses the current pulseTicks value on each 20 ms cycle, the servo moves smoothly to the commanded angle without any extra work in the main loop. This approach keeps our main code simple—just a pin read, a variable update, and an LED toggle—while the hardware‑driven ISR handles all the precise timing needed for clean PWM generation.

As for the water tank system, we have an ultrasonic sensor and LEDs that light depending on the water level between two particular levels in the tank. Our pump and LED control runs on the same PIC16F877A at 8 MHz. The code does three jobs in a tight loop:

- **Read the speed knob**
  We use the on-chip ADC on RA0 to convert the potentiometer voltage into a 0–1023 value.
- **Map knob to PWM**
  That 10-bit ADC reading is scaled to a 0–249 range (PR2), matching our 500 Hz PWM on CCP2 (RC1).
- **Watch the IR sensor**
  RB1 is the IR input (active LOW).
    - If RB1 is LOW, we turn on the LED on RC7 and load CCPR2L with the calculated duty — the pump runs at that speed.
    - If RB1 is HIGH, we turn the LED off and set CCPR2L to 0 so the pump stops.

All of this is set up in init_io_adc_pwm():

- PORTB pull-ups on, RB1 input, RC1/RC4/RC5/RC7 outputs, RA0 analog input
- ADCON1/ADCON0 enable only AN0, Fosc/16
- Timer2 at 1:16 prescale, PR2 = 249 for 500 Hz PWM
- CCP2 in PWM mode
- RC4/RC5 fixed for forward direction on our H-bridge

Then main() repeats: read ADC, convert to PWM duty, check RB1, toggle LED, and update CCPR2L. Because the PWM hardware handles the pulses, our loop stays very small and easy to follow.

Problems & Recommendations

Early on, we built our circuit on a green board by soldering our wires together. However, this made our project much less dynamic, meaning that changing anything later in the project was made much more difficult. When we realized certain Port D connections were no longer functioning properly, we urgently switched to a breadboard in order to use Port B instead.

Another problem we faced was the power requirements of the servo motor; above 5 V, the value of Vcc, and the motor burned. This occurred on one occasion. Furthermore, in the early testing phases certain EasyPICs seemed to exhibit unexpected behavior, due to preconfigured settings on each EasyPIC.

Our recommendations are as follows:

- Students should have more training in soldering, both with regards to technique and best practices.
- The available EasyPICs should not be preconfigured in particular ways; they should be regularly reset to their default configurations in order to avoid unexpected behavior.
- Certain C libraries would have been immensely useful; a small number of C libraries may be deserving of due consideration.

Conclusion

Our device meets its goal of cutting down on germs by removing the need to touch trash lids or water taps. The fast lid action and steady pump flow prove that a simple microcontroller and a few sensors can make a reliable, low-cost system. This design could help keep hospitals, schools, and food areas cleaner. It shows that basic embedded electronics can improve public health and make everyday tasks easier.