

Expanding Visual Programming for Educational Robots/IoT

Authors: Van Nguyen & Monica Welfert

Director: Yinong Chen

Second Reader: Mutsumi Nakamura

New Additions	4
Documentation	4
1_globalVar.ino	4
BUZZER_PIN	4
SCREEN_WIDTH	4
SCREEN_HEIGHT	4
OLED_RESET	4
Adafruit_SSD1306 display	4
beeper.ino	4
setupBeeper()	4
beep(int freq, int ms)	4
happyBirthday()	5
display.ino	5
setupDisplay()	5
drawLeftArrow()	5
drawRightArrow()	5
drawUpArrow()	5
drawDownArrow()	5
drawMessage(String message)	5
Pinout	5
Ultrasonic Sensor	5
TRIG	5
ECHO	5
Infrared Sensors	5
TOUCH1	5
TOUCH2	5
Motors	6
LEFT_IN1	6
LEFT_IN2	6
RIGHT_IN1	6
RIGHT_IN2	6
Beeper	6
BUZZER_PIN	6
Display	6
SDA	6
SCL	6
Libraries	6

New Additions

In this update to the ESPBot, we have introduced new libraries for a small OLED display and a beeper. This functionality can be easily expanded to multiple beepers and displays, but requires more GPIO pins, or for the user to not use some of the infrared sensors or the ultrasonic sensor. We have also relocated some of the pins

The display can be updated to display 1 of 4 predefined shapes, or to display user-defined text. New shapes can be added by defining new methods within `display.ino` and calling the appropriate functions while parsing the JSON data in `v1p1e.ino`.

The beeper can be controlled by user-defined input to play any frequency for any amount of time. There is also a function added to play the happy birthday song. More songs can be added by defining new methods within `beeper.ino` and calling the appropriate functions while parsing the JSON data in `v1p1e.ino`. More functionality can be added to allow the user to input a list of frequencies along with a list of time so the user can define their own songs or sequences on the fly.

Documentation

1_globalVar.ino

`BUZZER_PIN`

Defines the GPIO pin for the buzzer.

`SCREEN_WIDTH`

Defines the screen width, in pixels, for the OLED display.

`SCREEN_HEIGHT`

Defines the screen height, in pixels for the OLED display.

`OLED_RESET`

Defines the reset pin for the OLED display. -1 if sharing reset pin with Arduino.

`Adafruit_SSD1306 display`

Object of `Adafruit_SSD1306` that allows control of display via I2C.

beeper.ino

`setupBeeper()`

Method used to set up beeper pins and ensure that beeper is silent on boot.

`beep(int freq, int ms)`

Method used to make the beeper emit a tone at a desired frequency, `freq`, for a desired time in milliseconds, `ms`.

happyBirthday()

Method used to make the beeper play the happy birthday song. Can be copied/adapted to play different songs/sequences.

display.ino

setupDisplay()

Method used to set up the display over I2C using the D1 and D2 pins. Also displays a default image on boot for ensuring OLED is setup properly.

drawLeftArrow()

Method used to draw a left arrow on the display.

drawRightArrow()

Method used to draw a right arrow on the display.

drawUpArrow()

Method used to draw an up arrow on the display.

drawDownArrow()

Method used to draw a down arrow on the display.

drawMessage(String message)

Method used to print a message, message, on the display.

Pinout

Ultrasonic Sensor

TRIG

Default as GPIO9. Corresponds to SD2. Used to send ultrasonic signal.

ECHO

Default as GPIO10. Corresponds to SD3. Used to read ultrasonic signal.

Infrared Sensors

TOUCH1

Default as GPIO2. Corresponds to D4. Used for left infrared sensor.

TOUCH2

Default as GPIO0. Corresponds to D3. Used for right infrared sensor.

Motors

LEFT_IN1

Default as GPIO14. Corresponds to D5. Used for left motor.

LEFT_IN2

Default as GPIO12. Corresponds to D6. Used for left motor.

RIGHT_IN1

Default as GPIO13. Corresponds to D7. Used for right motor.

RIGHT_IN2

Default as GPIO15. Corresponds to D8. Used for right motor.

Beeper

BUZZER_PIN

Default as GPIO16. Corresponds to D0. Used for beeper signal.

Display

NOTE: These pins are not present in 1_globalVar.ino because they are always bound to these GPIO pins, as the display is using I2C.

SDA

Connect to GPIO5. Corresponds to D2.

SCL

Connect to GPIO4. Corresponds to D1.

Libraries

- ESP8266WiFi.h
- ESP8266WebServer.h
- ESP8266mDNS.h
- ESP8266HTTPUpdateServer.h
- EEPROM.h
- WebSocketsServer.h
- SPI.h
- Wire.h
- Adafruit_GFX.h
- Adafruit_SSD1306.h