WIRELESS CHARGING STATION CODE EXPLATIONATION

Code Breakdown

1. **Blynk Configuration**:

```cpp
#define BLYNK_TEMPLATE_ID "TMPL3Je5RoGr0"
#define BLYNK_TEMPLATE_NAME "Wireless Charging"
#define BLYNK_AUTH_TOKEN "E-fKJZBx7p_fKAKURyLj6F48sEpb65qN"
#define BLYNK_PRINT Serial
```

   - These lines define constants for the Blynk template ID, template name, and authentication token. The `BLYNK_PRINT Serial` line is used to enable serial output for debugging.

2. **Library Inclusions**:

```cpp
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
#include <LiquidCrystal.h>
```

   - These lines include the necessary libraries for WiFi communication, Blynk integration, and controlling an LCD display.

3. **WiFi Credentials**:

```cpp
char auth[] = BLYNK_AUTH_TOKEN;
char ssid[] = "WIFI";
char pass[] = "12345678";
```

   - Here, the authentication token and WiFi credentials (SSID and password) are defined as character arrays.

4. **Sensor and PWM Pin Definitions**:

```cpp
int sense_2 = 15, Ir;
int sense_4 = 4, Ir2;
#define PWM_12 12
#define PWM_13 13
```

- `sense_2` and `sense_4` are GPIO pins connected to sensors that detect vehicles. `PWM_12` and `PWM_13` are defined for controlling the charging mechanisms.

5. **LCD Initialization**:

```cpp
LiquidCrystal lcd(D_05, D_18, D_19, D_21, D_22_SCL, D_23_SDA);
```

- This line initializes an instance of the `LiquidCrystal` class for controlling an LCD display. The parameters represent the pins connected to the LCD.

6. **Charging Flags and Widget Initialization**:

```cpp
bool charge_flag_1 = 0, charge_flag_2 = 0;
WidgetLED S_S1(V3);
WidgetLED S_S2(V4);
```

- `charge_flag_1` and `charge_flag_2` are boolean flags that indicate whether charging is requested for the two spots. `WidgetLED` instances are created for visual feedback in the Blynk app.

7. **Blynk Write Handlers**:

```cpp
BLYNK_WRITE(V1) { ... }
BLYNK_WRITE(V2) { ... }
```

   - These functions handle incoming commands from the Blynk app. When a button in the app is pressed, the corresponding flag (`charge_flag_1` or `charge_flag_2`) is set to indicate whether charging should be activated for that spot.

8. **Setup Function**:

```cpp
void setup() { ... }
```

   - This function initializes the pins, starts the Blynk connection, and scrolls a welcome message on the LCD. It sets the pin modes for the PWM and sensor pins and initializes the LCD.

9. **Loop Function**:

```cpp
void loop() { ... }
```

   - This function runs continuously and checks the state of the vehicle detection sensors. It updates the LCD display and the Blynk app based on whether a vehicle is detected and if charging is requested.