# Downlink Payload Preparation Script Documentation

*5/1/2024*

*Author: Jatan Pandya*

*QuireTech LLC*

## Introduction

The Downlink Payload Preparation Script is a Python script to configure settings for remote sidewalk devices and send configuration payloads using AWS IoT Wireless.

## Usage

### Command-line Arguments

- `--config <path_to_config_file>`: Specifies the path to the configuration file (default: "./config.json").
- `--routine <routine_name>`: Specifies the routine to use from the configuration file (default: "default").

### Script Execution

To execute the script, run the following command: `python3 downlinker.py [--config <path_to_config_file>] [--routine <routine_name>]`

- If no arguments are provided, the script uses default settings.
- To specify a routine, use the `--routine` argument followed by the routine name defined in the configuration file.

### Example Usages

- Run with default settings: `python3 downlinker.py`
- Run with a specific routine: `python3 downlinker.py --routine routine1`

## Configuration

The script utilizes a JSON configuration file (`config.json`) to define different routines and their corresponding settings. Each routine can have customized values for various parameters. Below is the structure of the configuration file:

```
{
  "default": {
      "Buzzer_Set": "DEFAULT",
      "NFC_Set": "DEFAULT",
      ...
  },
```

```
    "routine1": {
        "Buzzer_Set": "LOW",
        "NFC_Set": "ENABLE",
        ...
    },
    "routine2": {
        "Buzzer_Set": "MEDIUM",
        "NFC_Set": "DISABLE",
        ...
    }
}
```

- Each routine is identified by a unique name (default, routine1, routine2, etc.).
- Settings such as Buzzer_Set, NFC_Set, Bin_Level, etc., can be customized for each routine.
- The N and FREQ parameters define the number of payloads to send and the frequency of transmission, respectively.
- The SEP parameter specifies the separator character between configuration values.
- The DEVICE_ID parameter identifies the AWS device ID to send the payload.

## Payload

| Event | Description |
|---|---|
| Buzzer_Set (01) | Sets the buzzer volume to Low, Medium, or High (0, 1, 2) |
| NFC_Set (02) | Enables or disables the NFC reader. |
| Bin_Level (03) | Sets the "Bin Full" sensor alert distance to 0, 1, or 2. |
| UHF_Power (04) | Adjusts the UHF reader power to 0, 1, or 2. |
| Display_Set (05) | Customizes the LED display for Venue 1, Venue 2, or Venue 3. |
| BinID_Set (06) | Assigns a unique ID to each Topper board. |
| NFC_Merch_Set (07) | Configures NFC parameters for updating the merchant ID. |
| BOOT_MODE (08) | Configure topper to enter in boot mode. |

## Script Structure

The script consists of the following components:

- Class Downlink: Handles configuration, payload preparation, encoding, and AWS downlink transmission.
- load_config Function: Loads configuration settings from the JSON file.
- main Function: Parses command-line arguments, loads configurations, and initiates payload transmission.

## Dependencies

- AWS account configured and authorized.
- Boto3 SDK (Python3 AWS SDK) installed. Run pip3 install -r requirements.txt for dependencies.