

Ezlopi v2.x: Custom device/sensor integration guide

Prerequisite: Idea of **cJSON** is required before starting

Step-1: Create a respective folder in **devices** folder (eg. digital_io)

Step-2: Create a src file which contains the sensor/device script (eg. digital_io.c)

Step-3: Create a header file to declare the public function (eg. digital_io.h)

Step-4: Create **CMakeLists.txt** to link the required modules (eg. please see devices/digital_io/CMakeLists.txt for reference)

Step-5: Define the primary function for the device/sensor

eg. please follow the digital_io

```
int digital_io(e_ezlopi_actions_t action,
s_ezlopi_device_properties_t *properties, void *arg)
```

@details This function will be called whenever any event occurs (event such as `EZLOPI_ACTION_PREPARE` or `EZLOPI_ACTION_INITIALIZE` or `EZLOPI_ACTION_NOTIFY_1000_MS` or `EZLOPI_ACTION_GET_EZLOPI_VALUE`, etc.)

@arg action respective action will be fed when called this function
@arg properties the cloud and interface properties of sensor/device
@arg arg pointer of any specific data required to fed (eg. cJSON packed of request on `hub.items.value.set`)

@return int 0 if function execution were not success, 1 when function execution was a success, or return device properties type casted to int on `'EZLOPI_ACTION_PREPARE'`.

Use switch-case to implement the different event/action as in digital_io

Essential Actions:

EZLOPI_ACTION_PREPARE: This action stores the required properties for the particular device/sensor in a structure

EZLOPI_ACTION_INITIALIZE: Initialize the device/sensor's interface or GPIO pins

EZLOPI_ACTION_GET_EZLOPI_VALUE: add the value parts of cJSON as in digital_io example. This event will be executed whenever the value needs to send to the cloud.

`EZLOPI_ACTION_NOTIFY_1000_MS`: timer event. This event should be implemented if the device/sensor needs to sample the data each seconds. Similarly `EZLOPI_ACTION_NOTIFY_500_MS` for 500 ms, `EZLOPI_ACTION_NOTIFY_100_MS` for 100 ms and so on. For other than listed interval, user needs to implement themselves.

Similarly other events can be implemented.

Step-6: Put the device/sensor primary function on the `device_array`

Eg. Please look at `ezlopi-core/ezlopi_devices_list.c`

```
#ifndef EZLOPI_DEVICE_0001_LED
{
    .id = EZLOPI_DEVICE_0001_LED,
    .func = digital_io,
},
#endif
```

Define the macro for a particular device from the reference of the ezlogic app.

Step-7: Make sure the particular device is implemented on the ezlogic app.