Sensor Interfacing with Wi-SUN Explorer Kit (EFR32ZG28)

Step 1: Creating Application Monitoring Project

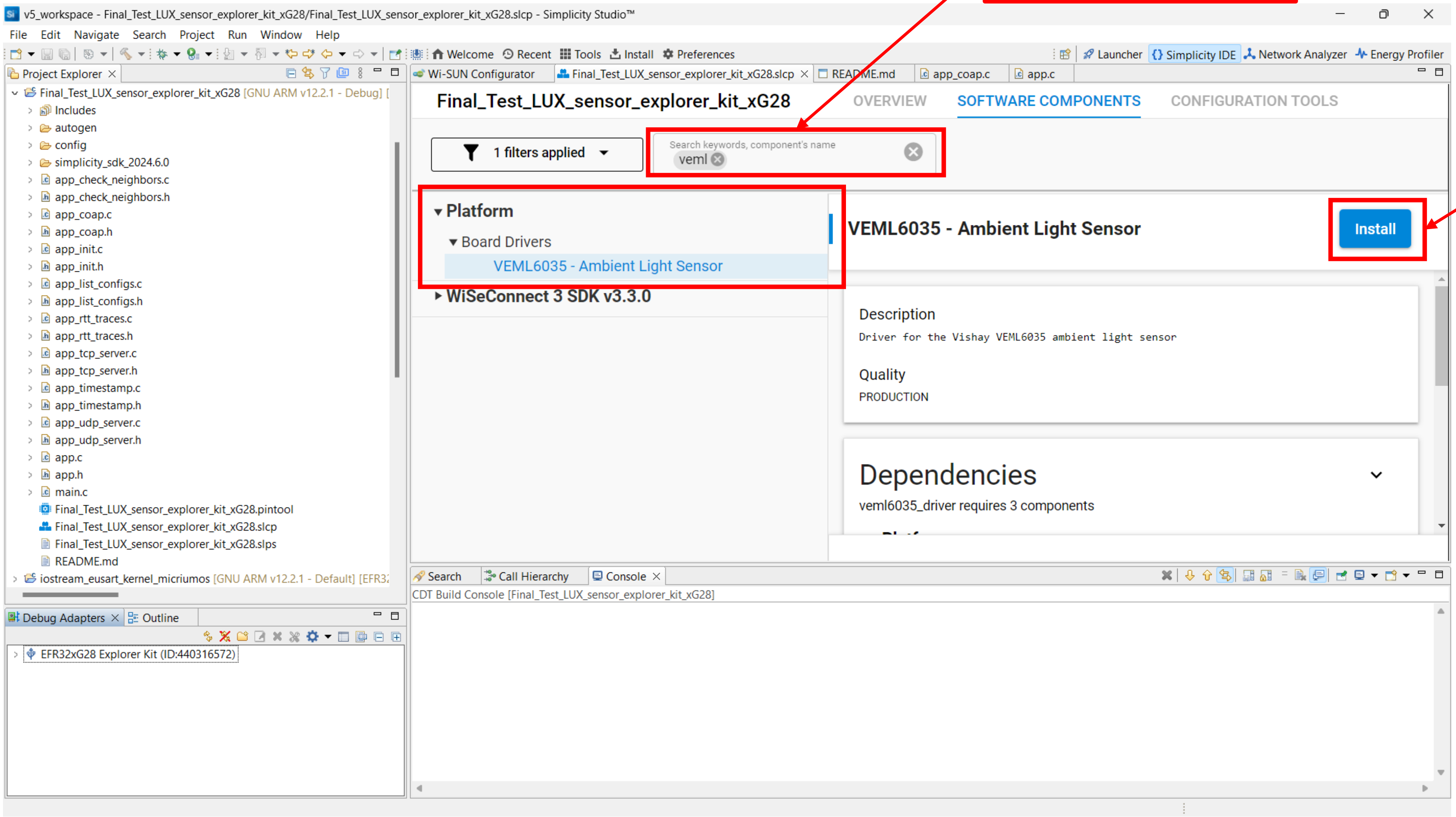
1. Create Wi-SUN Node Monitoring Application
2. Goto .slcp file
3. Click on Software Component
4. Remove “Wi-SUN OTA DFU” Component
5. Add “CoAP” Component

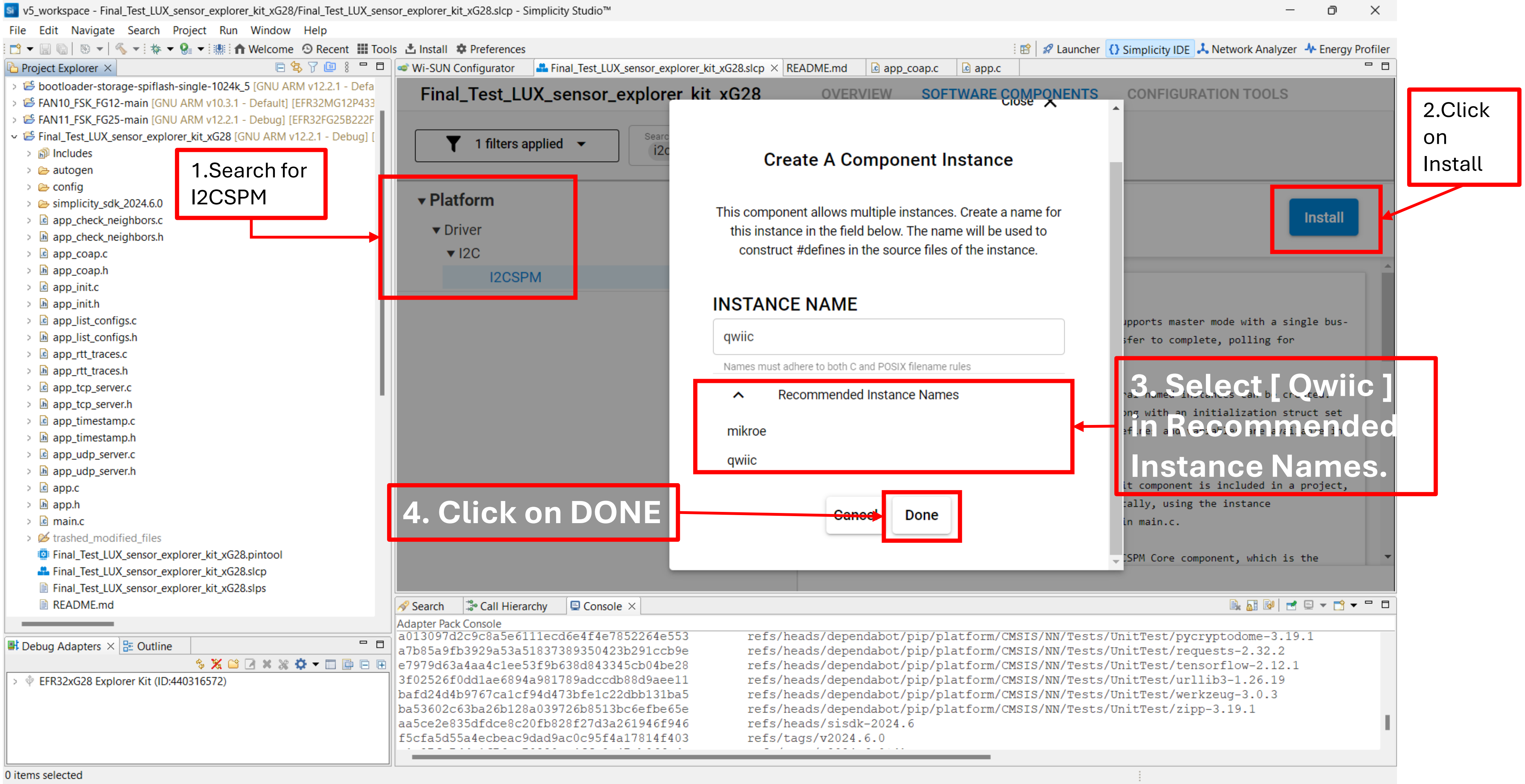
Note:- This steps are same as that of the Wi-SUN node monitoring application.

Step 2: Add Software components for Sensor Interfacing.

1.In Software Components Search for veml6035

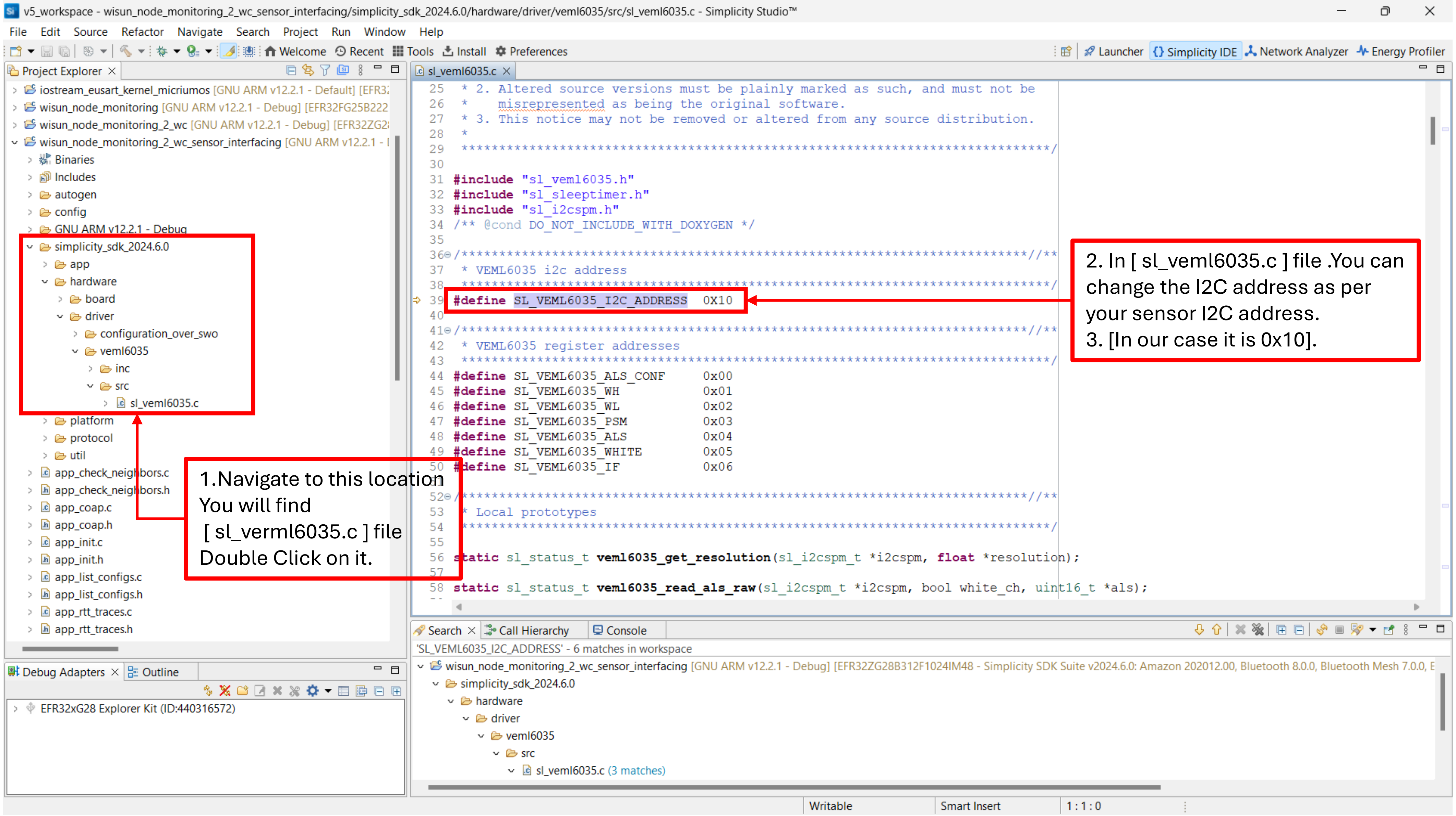
Click on Install.




1.Search for I2CSPM
2.Click on Install
3. Select [Qwiic] in Recommended Instance Names.
4. Click on DONE
Final_Test_LUX_sensor_explorer_kit_xG28
OVERVIEW
SOFTWARE COMPONENTS
CONFIGURATION TOOLS
1 filters applied
Platform
Driver
I2C
I2CSPM
Install
Close
Create A Component Instance
This component allows multiple instances. Create a name for this instance in the field below. The name will be used to construct #defines in the source files of the instance.
INSTANCE NAME
qwiic
Names must adhere to both C and POSIX filename rules
Recommended Instance Names
mikroe
qwiic
Cancel
Done

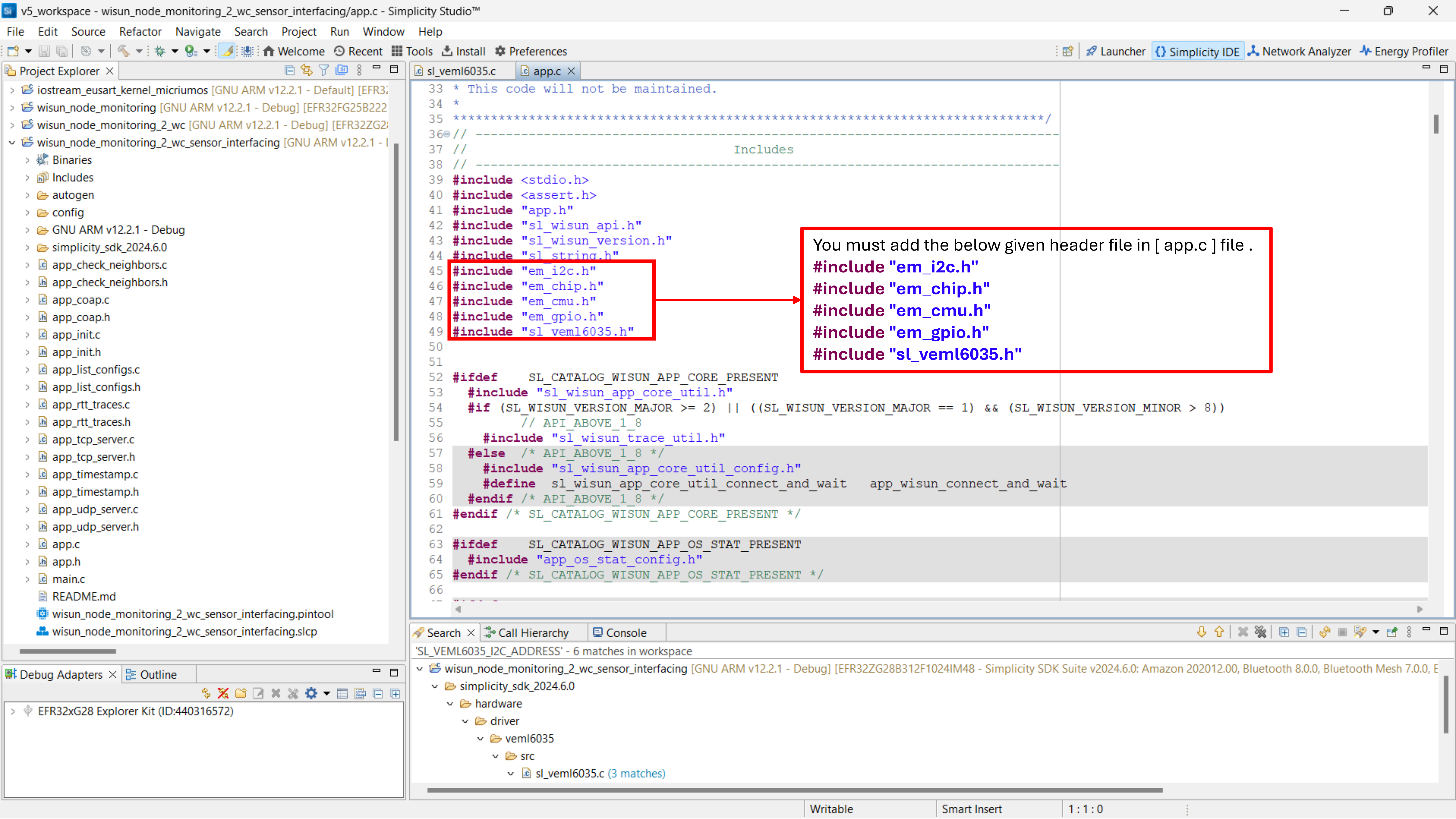
Step 3 : Add few lines of code

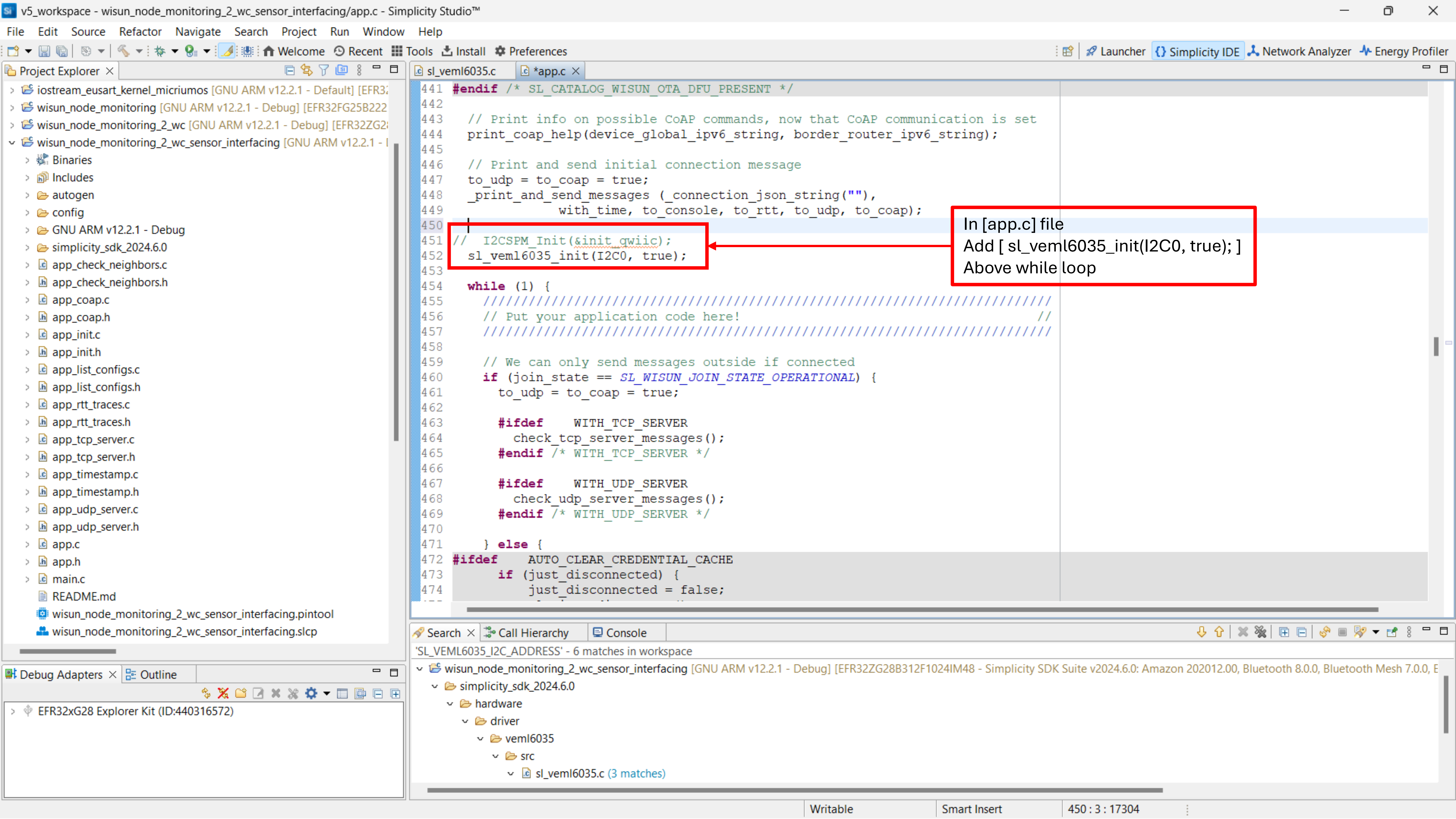
Note: while editing the [sl_veml6035.c] file it will ask you to save a copy.
So pls make sure to save a copy.



1.Navigate to this location
You will find
[sl_verml6035.c] file
Double Click on it.

2. In [sl_veml6035.c] file .You can change the I2C address as per your sensor I2C address.
3. [In our case it is 0x10].


v5_workspace - wisun_node_monitoring_2_wc_sensor_interfacing/app.c - Simplicity Studio™
File Edit Source Refactor Navigate Search Project Run Window Help
Welcome Recent Tools Install Preferences
Launcher Simplicity IDE Network Analyzer Energy Profiler
Project Explorer
iostream_eusart_kernel_micriumos [GNU ARM v12.2.1 - Default] [EFR3
wisun_node_monitoring [GNU ARM v12.2.1 - Debug] [EFR32FG25B222
wisun_node_monitoring_2_wc [GNU ARM v12.2.1 - Debug] [EFR32ZG2
wisun_node_monitoring_2_wc_sensor_interfacing [GNU ARM v12.2.1 -
Binaries
Includes
autogen
config
GNU ARM v12.2.1 - Debug
simplicity_sdk_2024.6.0
app_check_neighbors.c
app_check_neighbors.h
app_coap.c
app_coap.h
app_init.c
app_init.h
app_list_configs.c
app_list_configs.h
app_rtt_traces.c
app_rtt_traces.h
app_tcp_server.c
app_tcp_server.h
app_timestamp.c
app_timestamp.h
app_udp_server.c
app_udp_server.h
app.c
app.h
main.c
README.md
wisun_node_monitoring_2_wc_sensor_interfacing.pintool
wisun_node_monitoring_2_wc_sensor_interfacing.slcp
sl_veml6035.c
app.c
* This code will not be maintained.
*
**/
// ---
// Includes
// ---
#include <stdio.h>
#include <assert.h>
#include "app.h"
#include "sl_wisun_api.h"
#include "sl_wisun_version.h"
#include "sl_string.h"
#include "em_i2c.h"
#include "em_chip.h"
#include "em_cmu.h"
#include "em_gpio.h"
#include "sl_veml6035.h"
#ifdef SL_CATALOG_WISUN_APP_CORE_PRESENT
#include "sl_wisun_app_core_util.h"
#if (SL_WISUN_VERSION_MAJOR >= 2) || ((SL_WISUN_VERSION_MAJOR == 1) && (SL_WISUN_VERSION_MINOR > 8))
// API_ABOVE_1_8
#include "sl_wisun_trace_util.h"
#else /* API_ABOVE_1_8 */
#include "sl_wisun_app_core_util_config.h"
#define sl_wisun_app_core_util_connect_and_wait app_wisun_connect_and_wait
#endif /* API_ABOVE_1_8 */
#endif /* SL_CATALOG_WISUN_APP_CORE_PRESENT */
#ifdef SL_CATALOG_WISUN_APP_OS_STAT_PRESENT
#include "app_os_stat_config.h"
#endif /* SL_CATALOG_WISUN_APP_OS_STAT_PRESENT */
You must add the below given header file in [app.c] file .
#include "em_i2c.h"
#include "em_chip.h"
#include "em_cmu.h"
#include "em_gpio.h"
#include "sl_veml6035.h"
Debug Adapters
Outline
EFR32xG28 Explorer Kit (ID:440316572)
Search
Call Hierarchy
Console
'SL_VEML6035_I2C_ADDRESS' - 6 matches in workspace
wisun_node_monitoring_2_wc_sensor_interfacing [GNU ARM v12.2.1 - Debug] [EFR32ZG28B312F1024IM48 - Simplicity SDK Suite v2024.6.0: Amazon 202012.00, Bluetooth 8.0.0, Bluetooth Mesh 7.0.0, E
simplicity_sdk_2024.6.0
hardware
driver
veml6035
src
sl_veml6035.c (3 matches)
Writable
Smart Insert
1 : 1 : 0


v5_workspace - wisun_node_monitoring_2_wc_sensor_interfacing/app.c - Simplicity Studio™
File Edit Source Refactor Navigate Search Project Run Window Help
Welcome Recent Tools Install Preferences
Launcher Simplicity IDE Network Analyzer Energy Profiler
Project Explorer
iostream_eusart_kernel_micriumos [GNU ARM v12.2.1 - Default] [EFR3
wisun_node_monitoring [GNU ARM v12.2.1 - Debug] [EFR32FG25B222
wisun_node_monitoring_2_wc [GNU ARM v12.2.1 - Debug] [EFR32ZG2
wisun_node_monitoring_2_wc_sensor_interfacing [GNU ARM v12.2.1 -
Binaries
Includes
autogen
config
GNU ARM v12.2.1 - Debug
simplicity_sdk_2024.6.0
app_check_neighbors.c
app_check_neighbors.h
app_coap.c
app_coap.h
app_init.c
app_init.h
app_list_configs.c
app_list_configs.h
app_rtt_traces.c
app_rtt_traces.h
app_tcp_server.c
app_tcp_server.h
app_timestamp.c
app_timestamp.h
app_udp_server.c
app_udp_server.h
app.c
app.h
main.c
README.md
wisun_node_monitoring_2_wc_sensor_interfacing.pintool
wisun_node_monitoring_2_wc_sensor_interfacing.slcp
sl_veml6035.c
*app.c
#endif /* SL_CATALOG_WISUN_OTA_DFU_PRESENT */
// Print info on possible CoAP commands, now that CoAP communication is set
print_coap_help(device_global_ipv6_string, border_router_ipv6_string);
// Print and send initial connection message
to_udp = to_coap = true;
_print_and_send_messages (_connection_json_string(""),
with_time, to_console, to_rtt, to_udp, to_coap);
// I2CSPM_Init(&init_qwiic);
sl_veml6035_init(I2C0, true);
while (1) {
// Put your application code here! //
// We can only send messages outside if connected
if (join_state == SL_WISUN_JOIN_STATE_OPERATIONAL) {
to_udp = to_coap = true;
#ifdef WITH_TCP_SERVER
check_tcp_server_messages();
#endif /* WITH_TCP_SERVER */
#ifdef WITH_UDP_SERVER
check_udp_server_messages();
#endif /* WITH_UDP_SERVER */
} else {
#ifdef AUTO_CLEAR_CREDENTIAL_CACHE
if (just_disconnected) {
just_disconnected = false;
In [app.c] file
Add [sl_veml6035_init(I2C0, true);]
Above while loop
Debug Adapters
Outline
EFR32xG28 Explorer Kit (ID:440316572)
Search
Call Hierarchy
Console
'SL_VEML6035_I2C_ADDRESS' - 6 matches in workspace
wisun_node_monitoring_2_wc_sensor_interfacing [GNU ARM v12.2.1 - Debug] [EFR32ZG28B312F1024IM48 - Simplicity SDK Suite v2024.6.0: Amazon 202012.00, Bluetooth 8.0.0, Bluetooth Mesh 7.0.0, E
simplicity_sdk_2024.6.0
hardware
driver
veml6035
src
sl_veml6035.c (3 matches)
Writable
Smart Insert
450 : 3 : 17304

In [app_coap.c] file You must add the below given code.

```c
sl_wisun_coap_packet_t * lux_data (
    const  sl_wisun_coap_packet_t *const req_packet) {
  float data;
  sl_veml6035_get_als_lux(I2C0, &data);
  snprintf(coap_response, COAP_MAX_RESPONSE_LEN, "%.2f", data);
  return app_coap_reply(coap_response, req_packet); }
```

```c
coap_resource.data.uri_path = "/data";
coap_resource.data.resource_type = "json";
coap_resource.data.interface = "node";
coap_resource.auto_response = lux_data;
coap_resource.discoverable = true;
assert(sl_wisun_coap_rhnd_resource_add(&coap_resource) == SL_STATUS_OK);
count++;
```

Step 4: Upload [.s37] file and check the data on oneM2M

After That Just Build the Project and Flash [.s37] file to the Board