# Receiving Sensor Data over BLE with a mobile application

In the Project titled ‘nodusC’, you will find a file named TopDesign.cysch. Opening this file with PSOC Creator 3.3, you will see a design window and several tabs. On the ‘Communication’ tab, there is a block labeled BLE. Double clicking this block will allow you to see the BLE configuration being used for this particular project. The ‘Profiles’ tab up top is where you will find the Custom profile, as well as the services and characteristics it contains.

The service labeled ‘NodusC\_Service’ contains a characteristic named ‘NodusC\_Characteristic’, which is where sensor data is written to. The characteristic has a notify property, meaning that notifications for it can be enabled or disabled. The code written in the main file of this project is configured to behave differently based on whether these notifications are enabled or not. The Client Characteristic Configuration contains a property where the notifications are actually set.

There is also a characteristic within the same service labeled ‘Config\_sample\_frequency’, which can be edited directly with a write request or read in the mobile application with a read request. This determines the frequency at which the ADC samples. It is automatically set to 10Hz as a default, but can be set to any valid uint8 value once a connection between the BLE devices is established.

Inside the main.c file, there is a function called ‘CustomEventHandler’ which determines the response of the Nodus C device to certain BLE events. The code within the CYBLE\_EVT\_GATTS\_WRITE\_REQ case runs when the mobile device requests to write to the Nodus device. If the ‘Config\_sample\_frequency’ characteristic is the one being written to, the global variable containing the sampling frequency will be changed to whatever value is given. If the ‘NodusC\_Characteristic’ characteristic is the one selected, this means that notifications have either been disabled or enabled, and the global variable containing the Boolean value to represent whether notifications are allowed to be sent or not is what is updated accordingly.

The ‘SendADCData’ function is where the Notification handle/value pair is updated with whatever data is passed in as an argument. This method is called within the readADC\_routine selected by the scheduler, but only if a BLE connection exists and notifications have been set to true. It is called with the filtered ADC data passed into it. How this data is interpreted is decided by the mobile application being used to communicate with the Nodus device.