```c
/**
 * @file bno055_support.h
 *
 */

/*-----------------------------------------------------------------------*
 *  Includes
 *-----------------------------------------------------------------------*/
#include "bno055.h"

#define  BNO055_API

#define  FLAG_MEAS_ON    1
#define  FLAG_MEAS_OFF   0
/*-----------------------------------------------------------------------*
 *  The following APIs are used for reading and writing of
 *   sensor data using I2C communication
 *-----------------------------------------------------------------------*/
#ifdef  BNO055_API
#define BNO055_I2C_BUS_WRITE_ARRAY_INDEX ((u8)1)

/*  \Brief: The API is used as I2C bus read
 *  \Return : Status of the I2C read
 *  \param dev_addr : The device address of the sensor
 *  \param reg_addr : Address of the first register,
 *   will data is going to be read
 *  \param reg_data : This data read from the sensor,
 *   which is hold in an array
 *  \param cnt : The no of byte of data to be read
 */
s8 BNO055_I2C_bus_read(u8 dev_addr, u8 reg_addr, u8 *reg_data, u8 cnt);

/*  \Brief: The API is used as SPI bus write
 *  \Return : Status of the SPI write
 *  \param dev_addr : The device address of the sensor
 *  \param reg_addr : Address of the first register,
 *   will data is going to be written
 *  \param reg_data : It is a value hold in the array,
 *  will be used for write the value into the register
 *  \param cnt : The no of byte of data to be write
 */
s8 BNO055_I2C_bus_write(u8 dev_addr, u8 reg_addr, u8 *reg_data, u8 cnt);

/*
 * \Brief: I2C init routine
 */
s8 I2C_routine(void);

/*  Brief : The delay routine
 *  \param : delay in ms
 */
void BNO055_delay_msek(u32 msek);

#endif

/********************End of I2C APIs declarations********************/

/* This API is an example for reading sensor data
 *  \param: None
 *  \return: communication result
 */
s32 bno055_init_readout(void);

s32 bno055_read_routine(s_bno055_data *data);

/*-----------------------------------------------------------------------*
 *  struct bno055_t parameters can be accessed by using BNO055
 *  BNO055_t having the following parameters
 *  Bus write function pointer: BNO055_WR_FUNC_PTR
 *  Bus read function pointer: BNO055_RD_FUNC_PTR
 *  Burst read function pointer: BNO055_BRD_FUNC_PTR
 *  Delay function pointer: delay_msec
 *  I2C address: dev_addr
```

```c
 *  Chip id of the sensor: chip_id
 *---------------------------------------------------------------------------*/
struct bno055_t bno055;
```