HAL Team - Source code documentation - BNO055

1.04

Generated by Doxygen 1.9.7

Chapter 1

File Index

1.1 File List

Here is a list of all documented files with brief descriptions:

Chapter 2

File Documentation

2.1 HAL_BNO055/BNO055/bno055.c File Reference

```
#include ¨bno055.h¨
#include <stdint.h>
#include <stddef.h>
```

Macros

- #define **NULL** ((void ∗) 0)

Functions

- BNO055_RETURN_FUNCTION_TYPE bno055_init (struct bno055_t ∗bno055)

  This API is used for initialize bus read, bus write function pointers,device address,accel revision id, gyro revision id mag revision id, software revision id, boot loader revision id and page id.
- BNO055_RETURN_FUNCTION_TYPE bno055_write_register (u8 addr_u8, u8 ∗data_u8, u8 len_u8)

  This API gives data to the given register and the data is written in the corresponding register address.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_register (u8 addr_u8, u8 ∗data_u8, u8 len_u8)

  This API reads the data from the given register address.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_chip_id (u8 ∗chip_id_u8)

  This API reads chip id from register 0x00 it is a byte of data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_sw_rev_id (u16 ∗sw_id_u8)

  This API reads software revision id from register 0x04 and 0x05 it is a two byte of data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_page_id (u8 ∗page_id_u8)

  This API reads page id from register 0x07 it is a byte of data.
- BNO055_RETURN_FUNCTION_TYPE bno055_write_page_id (u8 page_id_u8)

  This API used to write the page id register 0x07.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_accel_rev_id (u8 ∗accel_rev_id_u8)

  This API reads accel revision id from register 0x01 it is a byte of value.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_mag_rev_id (u8 ∗mag_rev_id_u8)

  This API reads mag revision id from register 0x02 it is a byte of value.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_gyro_rev_id (u8 ∗gyro_rev_id_u8)

  This API reads gyro revision id from register 0x03 it is a byte of value.

- BNO055_RETURN_FUNCTION_TYPE bno055_read_bl_rev_id (u8 *bl_rev_id_u8)

  This API used to read boot loader revision id from register 0x06 it is a byte of value.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_accel_x (s16 *accel_x_s16)

  This API reads acceleration data X values from register 0x08 and 0x09 it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_accel_y (s16 *accel_y_s16)

  This API reads acceleration data Y values from register 0x0A and 0x0B it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_accel_z (s16 *accel_z_s16)

  This API reads acceleration data z values from register 0x0C and 0x0D it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_accel_xyz (struct bno055_accel_t *accel)

  This API reads acceleration data xyz values from register 0x08 to 0x0D it is a six byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_mag_x (s16 *mag_x_s16)

  This API reads mag data x values from register 0x0E and 0x0F it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_mag_y (s16 *mag_y_s16)

  This API reads mag data y values from register 0x10 and 0x11 it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_mag_z (s16 *mag_z_s16)

  This API reads mag data z values from register 0x12 and 0x13 it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_mag_xyz (struct bno055_mag_t *mag)

  This API reads mag data xyz values from register 0x0E to 0x13 it is a six byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_gyro_x (s16 *gyro_x_s16)

  This API reads gyro data x values from register 0x14 and 0x15 it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_gyro_y (s16 *gyro_y_s16)

  This API reads gyro data y values from register 0x16 and 0x17 it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_gyro_z (s16 *gyro_z_s16)

  This API reads gyro data z values from register 0x18 and 0x19 it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_gyro_xyz (struct bno055_gyro_t *gyro)

  This API reads gyro data xyz values from register 0x14 to 0x19 it is a six byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_euler_h (s16 *euler_h_s16)

  This API reads gyro data z values from register 0x1A and 0x1B it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_euler_r (s16 *euler_r_s16)

  This API reads Euler data r values from register 0x1C and 0x1D it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_euler_p (s16 *euler_p_s16)

  This API reads Euler data p values from register 0x1E and 0x1F it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_euler_hrp (struct bno055_euler_t *euler)

  This API reads Euler data hrp values from register 0x1A to 0x1F it is a six byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_quaternion_w (s16 *quaternion_w_s16)

  This API reads quaternion data w values from register 0x20 and 0x21 it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_quaternion_x (s16 *quaternion_x_s16)

  This API reads quaternion data x values from register 0x22 and 0x23 it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_quaternion_y (s16 *quaternion_y_s16)

  This API reads quaternion data y values from register 0x24 and 0x25 it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_quaternion_z (s16 *quaternion_z_s16)

  This API reads quaternion data z values from register 0x26 and 0x27 it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_quaternion_wxyz (struct bno055_quaternion_t *quaternion)

  This API reads Quaternion data wxyz values from register 0x20 to 0x27 it is a six byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_linear_accel_x (s16 *linear_accel_x_s16)

  This API reads Linear accel data x values from register 0x29 and 0x2A it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_linear_accel_y (s16 *linear_accel_y_s16)

  This API reads Linear accel data x values from register 0x2B and 0x2C it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_linear_accel_z (s16 *linear_accel_z_s16)

  This API reads Linear accel data x values from register 0x2C and 0x2D it is a two byte data.

- BNO055_RETURN_FUNCTION_TYPE bno055_read_linear_accel_xyz (struct bno055_linear_accel_↩ t *linear_accel)

  This API reads Linear accel data xyz values from register 0x28 to 0x2D it is a six byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_gravity_x (s16 *gravity_x_s16)

  This API reads gravity data x values from register 0x2E and 0x2F it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_gravity_y (s16 *gravity_y_s16)

  This API reads gravity data y values from register 0x30 and 0x31 it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_gravity_z (s16 *gravity_z_s16)

  This API reads gravity data z values from register 0x32 and 0x33 it is a two byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_gravity_xyz (struct bno055_gravity_t *gravity)

  This API reads gravity data xyz values from register 0x2E to 0x33 it is a six byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_temp_data (s8 *temp_s8)

  This API reads temperature values from register 0x33 it is a byte data.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_accel_x_msq (float *accel_x_f)

  This API is used to convert the accel x raw data to meterpersecseq output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_accel_x_mg (float *accel_x_f)

  This API is used to convert the accel x raw data to millig output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_accel_y_msq (float *accel_y_f)

  This API is used to convert the accel x raw data to meterpersecseq output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_accel_y_mg (float *accel_y_f)

  This API is used to convert the accel y raw data to millig output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_accel_z_msq (float *accel_z_f)

  This API is used to convert the accel z raw data to meterpersecseq output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_accel_z_mg (float *accel_z_f)

  This API is used to convert the accel z raw data to millig output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_accel_xyz_msq (struct bno055_accel_float↩ _t *accel_xyz)

  This API is used to convert the accel xyz raw data to meterpersecseq output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_accel_xyz_mg (struct bno055_accel_float_↩ t *accel_xyz)

  This API is used to convert the accel xyz raw data to millig output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_mag_x_uT (float *mag_x_f)

  This API is used to convert the mag x raw data to microTesla output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_mag_y_uT (float *mag_y_f)

  This API is used to convert the mag y raw data to microTesla output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_mag_z_uT (float *mag_z_f)

  This API is used to convert the mag z raw data to microTesla output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_mag_xyz_uT (struct bno055_mag_float_t *mag_xyz_data)

  This API is used to convert the mag yz raw data to microTesla output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_gyro_x_dps (float *gyro_x_f)

  This API is used to convert the gyro x raw data to dps output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_gyro_x_rps (float *gyro_x_f)

  This API is used to convert the gyro x raw data to rps output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_gyro_y_dps (float *gyro_y_f)

  This API is used to convert the gyro y raw data to dps output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_gyro_y_rps (float *gyro_y_f)

  This API is used to convert the gyro y raw data to rps output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_gyro_z_dps (float *gyro_z_f)

  This API is used to convert the gyro z raw data to dps output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_gyro_z_rps (float *gyro_z_f)

This API is used to convert the gyro z raw data to rps output as float.

- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_gyro_xyz_dps (struct bno055_gyro_float_t *gyro_xyz_data)

  This API is used to convert the gyro xyz raw data to dps output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_gyro_xyz_rps (struct bno055_gyro_float_t *gyro_xyz_data)

  This API is used to convert the gyro xyz raw data to rps output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_euler_h_deg (float *euler_h_f)

  This API is used to convert the Euler h raw data to degree output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_euler_h_rad (float *euler_h_f)

  This API is used to convert the Euler h raw data to radians output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_euler_r_deg (float *euler_r_f)

  This API is used to convert the Euler r raw data to degree output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_euler_r_rad (float *euler_r_f)

  This API is used to convert the Euler r raw data to radians output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_euler_p_deg (float *euler_p_f)

  This API is used to convert the Euler p raw data to degree output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_euler_p_rad (float *euler_p_f)

  This API is used to convert the Euler p raw data to radians output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_euler_hpr_deg (struct bno055_euler_float_t *euler_hpr)

  This API is used to convert the Euler hrp raw data to degree output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_euler_hpr_rad (struct bno055_euler_float_t *euler_hpr)

  This API is used to convert the Euler xyz raw data to radians output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_linear_accel_x_msq (float *linear_accel_x_f)

  This API is used to convert the linear accel x raw data to meterpersecseq output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_linear_accel_y_msq (float *linear_accel_y_f)

  This API is used to convert the linear accel y raw data to meterpersecseq output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_linear_accel_z_msq (float *linear_accel_z_f)

  This API is used to convert the linear accel z raw data to meterpersecseq output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_linear_accel_xyz_msq (struct bno055_linear_accel_float_t *linear_accel_xyz)

  This API is used to convert the linear accel xyz raw data to meterpersecseq output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_gravity_float_x_msq (float *gravity_x_f)

  This API is used to convert the gravity x raw data to meterpersecseq output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_gravity_float_y_msq (float *gravity_y_f)

  This API is used to convert the gravity y raw data to meterpersecseq output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_gravity_float_z_msq (float *gravity_z_f)

  This API is used to convert the gravity z raw data to meterpersecseq output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_gravity_xyz_msq (struct bno055_gravity_float_t *gravity_xyz)

  This API is used to convert the gravity xyz raw data to meterpersecseq output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_temp_fahrenheit (float *temp_f)

  This API is used to convert the temperature data to Fahrenheit output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_temp_celsius (float *temp_f)

  This API is used to convert the temperature data to Celsius output as float.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_accel_x_msq (double *accel_x_d)

  This API is used to convert the accel x raw data to meterpersecseq output as double.

- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_accel_x_mg (double *accel_x_d)

  This API is used to convert the accel x raw data to millig output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_accel_y_msq (double *accel_y_d)

  This API is used to convert the accel y raw data to meterpersecseq output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_accel_y_mg (double *accel_y_d)

  This API is used to convert the accel y raw data to millig output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_accel_z_msq (double *accel_z_d)

  This API is used to convert the accel z raw data to meterpersecseq output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_accel_z_mg (double *accel_z_d)

  This API is used to convert the accel z raw data to millig output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_accel_xyz_msq (struct bno055_accel_↩ double_t *accel_xyz)

  This API is used to convert the accel xyz raw data to meterpersecseq output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_accel_xyz_mg (struct bno055_accel_↩ double_t *accel_xyz)

  This API is used to convert the accel xyz raw data to millig output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_mag_x_uT (double *mag_x_d)

  This API is used to convert the mag x raw data to microTesla output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_mag_y_uT (double *mag_y_d)

  This API is used to convert the mag y raw data to microTesla output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_mag_z_uT (double *mag_z_d)

  This API is used to convert the mag z raw data to microTesla output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_mag_xyz_uT (struct bno055_mag_↩ double_t *mag_xyz)

  This API is used to convert the mag yz raw data to microTesla output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_gyro_x_dps (double *gyro_x_d)

  This API is used to convert the gyro x raw data to dps output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_gyro_x_rps (double *gyro_x_d)

  This API is used to convert the gyro x raw data to rps output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_gyro_y_dps (double *gyro_y_d)

  This API is used to convert the gyro y raw data to dps output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_gyro_y_rps (double *gyro_y_d)

  This API is used to convert the gyro y raw data to rps output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_gyro_z_dps (double *gyro_z_d)

  This API is used to convert the gyro z raw data to dps output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_gyro_z_rps (double *gyro_z_d)

  This API is used to convert the gyro z raw data to rps output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_gyro_xyz_dps (struct bno055_gyro_↩ double_t *gyro_xyz)

  This API is used to convert the gyro xyz raw data to dps output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_gyro_xyz_rps (struct bno055_gyro_↩ double_t *gyro_xyz)

  This API is used to convert the gyro xyz raw data to rps output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_euler_h_deg (double *euler_h_d)

  This API is used to convert the Euler h raw data to degree output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_euler_h_rad (double *euler_h_d)

  This API is used to convert the Euler h raw data to radians output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_euler_r_deg (double *euler_r_d)

  This API is used to convert the Euler r raw data to degree output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_euler_r_rad (double *euler_r_d)

  This API is used to convert the Euler r raw data to radians output as double.

- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_euler_p_deg (double *euler_p_d)

  This API is used to convert the Euler p raw data to degree output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_euler_p_rad (double *euler_p_d)

  This API is used to convert the Euler p raw data to radians output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_euler_hpr_deg (struct bno055_euler_↩ double_t *euler_hpr)

  This API is used to convert the Euler hpr raw data to degree output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_euler_hpr_rad (struct bno055_euler_↩ double_t *euler_hpr)

  This API is used to convert the Euler hpr raw data to radians output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_linear_accel_x_msq (double *linear_↩ accel_x_d)

  This API is used to convert the linear accel x raw data to meterpersecseq output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_linear_accel_y_msq (double *linear_↩ accel_y_d)

  This API is used to convert the linear accel y raw data to meterpersecseq output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_linear_accel_z_msq (double *linear_↩ accel_z_d)

  This API is used to convert the linear accel z raw data to meterpersecseq output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_linear_accel_xyz_msq (struct bno055_↩ linear_accel_double_t *linear_accel_xyz)

  This API is used to convert the linear accel xyz raw data to meterpersecseq output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_gravity_double_x_msq (double *gravity_x_d)

  This API is used to convert the gravity x raw data to meterpersecseq output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_gravity_double_y_msq (double *gravity_y_d)

  This API is used to convert the gravity y raw data to meterpersecseq output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_gravity_double_z_msq (double *gravity_z_d)

  This API is used to convert the gravity z raw data to meterpersecseq output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_gravity_xyz_msq (struct bno055_gravity↩ _double_t *gravity_xyz)

  This API is used to convert the gravity xyz raw data to meterpersecseq output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_temp_fahrenheit (double *temp_d)

  This API is used to convert the temperature data to Fahrenheit output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_temp_celsius (double *temp_d)

  This API is used to convert the temperature data to Celsius output as double.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_mag_calib_stat (u8 *mag_calib_u8)

  This API used to read mag calibration status from register from 0x35 bit 0 and 1.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_calib_stat (u8 *accel_calib_u8)

  This API used to read accel calibration status from register from 0x35 bit 2 and 3.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_calib_stat (u8 *gyro_calib_u8)

  This API used to read gyro calibration status from register from 0x35 bit 4 and 5.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_sys_calib_stat (u8 *sys_calib_u8)

  This API used to read system calibration status from register from 0x35 bit 6 and 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_selftest_accel (u8 *selftest_accel_u8)

  This API used to read self test of accel from register from 0x36 bit 0.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_selftest_mag (u8 *selftest_mag_u8)

  This API used to read self test of mag from register from 0x36 bit 1.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_selftest_gyro (u8 *selftest_gyro_u8)

  This API used to read self test of gyro from register from 0x36 bit 2.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_selftest_mcu (u8 *selftest_mcu_u8)

  This API used to read self test of micro controller from register from 0x36 bit 3.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_stat_gyro_any_motion (u8 *gyro_any_motion_u8)

This API used to read the stat_s8 of gyro anymotion interrupt from register from 0x37 bit 2.

- BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_stat_gyro_highrate (u8 *gyro_highrate_u8)

  This API used to read the stat_s8 of gyro highrate interrupt from register from 0x37 bit 3.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_stat_accel_high_g (u8 *accel_high_g_u8)

  This API used to read the stat_s8 of accel highg interrupt from register from 0x37 bit 5.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_stat_accel_any_motion (u8 *accel_any_motion_↩ u8)

  This API used to read the stat_s8 of accel anymotion interrupt from register from 0x37 bit 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_stat_accel_no_motion (u8 *accel_no_motion_u8)

  This API used to read the stat_s8 of accel nomotion/slowmotion interrupt from register from 0x37 bit 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_stat_main_clk (u8 *stat_main_clk_u8)

  This API is used to read status of main clock from the register 0x38 bit 0.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_sys_stat_code (u8 *sys_stat_u8)

  This API is used to read system status code from the register 0x39 it is a byte of data.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_sys_error_code (u8 *sys_error_u8)

  This API is used to read system BNO055_ERROR code from the register 0x3A it is a byte of data.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_unit (u8 *accel_unit_u8)

  This API used to read the accel unit from register from 0x3B bit 0.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_unit (u8 accel_unit_u8)

  This API used to write the accel unit from register from 0x3B bit 0.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_unit (u8 *gyro_unit_u8)

  This API used to read the gyro unit from register from 0x3B bit 1.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_unit (u8 gyro_unit_u8)

  This API used to write the gyro unit from register from 0x3B bit 1.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_euler_unit (u8 *euler_unit_u8)

  This API used to read the Euler unit from register from 0x3B bit 2.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_euler_unit (u8 euler_unit_u8)

  This API used to write the Euler unit from register from 0x3B bit 2.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_tilt_unit (u8 *tilt_unit_u8)

  This API used to write the tilt unit from register from 0x3B bit 3.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_tilt_unit (u8 tilt_unit_u8)

  This API used to write the tilt unit from register from 0x3B bit 3.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_temp_unit (u8 *temp_unit_u8)

  This API used to read the temperature unit from register from 0x3B bit 4.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_temp_unit (u8 temp_unit_u8)

  This API used to write the temperature unit from register from 0x3B bit 4.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_data_output_format (u8 *data_output_format_u8)

  This API used to read the current selected orientation mode from register from 0x3B bit 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_data_output_format (u8 data_output_format_u8)

  This API used to write the current selected orientation mode from register from 0x3B bit 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_operation_mode (u8 *operation_mode_u8)

  This API used to read the operation mode from register from 0x3D bit 0 to 3.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_operation_mode (u8 operation_mode_u8)

  This API used to write the operation mode from register from 0x3D bit 0 to 3.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_power_mode (u8 *power_mode_u8)

  This API used to read the power mode from register from 0x3E bit 0 to 1.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_power_mode (u8 power_mode_u8)

  This API used to write the power mode from register from 0x3E bit 0 to 1.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_rst (u8 *intr_rst_u8)

  This API used to read the reset interrupt from register from 0x3F bit 6 It resets all the interrupt bit and interrupt output.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_rst (u8 intr_rst_u8)

This API used to write the reset interrupt from register from 0x3F bit 6 It resets all the interrupt bit and interrupt output.

- BNO055_RETURN_FUNCTION_TYPE bno055_get_clk_src (u8 *clk_src_u8)

  This API used to read the clk source from register from 0x3F bit 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_clk_src (u8 clk_src_u8)

  This API used to write the clk source from register from 0x3F bit 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_sys_rst (u8 *sys_rst_u8)

  This API used to read the reset system from register from 0x3F bit 5.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_sys_rst (u8 sys_rst_u8)

  This API used to write the reset system from register from 0x3F bit 5.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_selftest (u8 *selftest_u8)

  This API used to read the self test from register from 0x3F bit 0.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_selftest (u8 selftest_u8)

  This API used to write the self test from register from 0x3F bit 0.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_temp_source (u8 *temp_source_u8)

  This API used to read the temperature source from register from 0x40 bit 0 and 1.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_temp_source (u8 temp_source_u8)

  This API used to write the temperature source from register from 0x40 bit 0 and 1.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_axis_remap_value (u8 *remap_axis_u8)

  This API used to read the axis remap value from register from 0x41 bit 0 and 5.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_axis_remap_value (u8 remap_axis_u8)

  This API used to write the axis remap value from register from 0x41 bit 0 and 5.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_remap_x_sign (u8 *remap_x_sign_u8)

  This API used to read the x-axis remap sign from register from 0x42 bit 2.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_remap_x_sign (u8 remap_x_sign_u8)

  This API used to write the x-axis remap sign from register from 0x42 bit 2.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_remap_y_sign (u8 *remap_y_sign_u8)

  This API used to read the y-axis remap sign from register from 0x42 bit 1.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_remap_y_sign (u8 remap_y_sign_u8)

  This API used to write the y-axis remap sign from register from 0x42 bit 1.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_remap_z_sign (u8 *remap_z_sign_u8)

  This API used to read the z-axis remap sign from register from 0x42 bit 0.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_remap_z_sign (u8 remap_z_sign_u8)

  This API used to write the z-axis remap sign from register from 0x42 bit 0.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_sic_matrix (struct bno055_sic_matrix_t *sic_matrix)

  This API is used to read soft iron calibration matrix from the register 0x43 to 0x53 it is a 18 bytes of data.
- BNO055_RETURN_FUNCTION_TYPE bno055_write_sic_matrix (struct bno055_sic_matrix_t *sic_matrix)

  This API is used to write soft iron calibration matrix from the register 0x43 to 0x53 it is a 18 bytes of data.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_accel_offset (struct bno055_accel_offset_t *accel_↩ offset)

  This API is used to read accel offset and accel radius offset form register 0x55 to 0x5A and radius form 0x67 and 0x68.
- BNO055_RETURN_FUNCTION_TYPE bno055_write_accel_offset (struct bno055_accel_offset_t *accel_↩ offset)

  This API is used to write accel offset and accel radius offset form register 0x55 to 0x5A and radius form 0x67 and 0x68.
- BNO055_RETURN_FUNCTION_TYPE bno055_read_mag_offset (struct bno055_mag_offset_t *mag_↩ offset)

  This API is used to read mag offset offset form register 0x69 to 0x6A.
- BNO055_RETURN_FUNCTION_TYPE bno055_write_mag_offset (struct bno055_mag_offset_t *mag_↩ offset)

  This API is used to read mag offset offset form register 0x69 to 0x6A.

- BNO055_RETURN_FUNCTION_TYPE bno055_read_gyro_offset (struct bno055_gyro_offset_t *gyro_↩ offset)

  This API is used to read gyro offset offset form register 0x61 to 0x66.
- BNO055_RETURN_FUNCTION_TYPE bno055_write_gyro_offset (struct bno055_gyro_offset_t *gyro_↩ offset)

  This API is used to read gyro offset offset form register 0x61 to 0x66.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_range (u8 *accel_range_u8)

  This API used to read the accel range from page one register from 0x08 bit 0 and 1.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_range (u8 accel_range_u8)

  This API used to write the accel range from page one register from 0x08 bit 0 and 1.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_bw (u8 *accel_bw_u8)

  This API used to read the accel bandwidth from page one register from 0x08 bit 2 to 4.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_bw (u8 accel_bw_u8)

  This API used to write the accel bandwidth from page one register from 0x08 bit 2 to 4.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_power_mode (u8 *accel_power_mode_u8)

  This API used to read the accel power mode from page one register from 0x08 bit 5 to 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_power_mode (u8 accel_power_mode_u8)

  This API used to write the accel power mode from page one register from 0x08 bit 5 to 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_mag_data_output_rate (u8 *mag_data_output_rate_↩ u8)

  This API used to read the mag output data rate from page one register from 0x09 bit 0 to 2.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_mag_data_output_rate (u8 mag_data_output_rate_u8)

  This API used to write the mag output data rate from page one register from 0x09 bit 0 to 2.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_mag_operation_mode (u8 *mag_operation_mode_u8)

  This API used to read the mag operation mode from page one register from 0x09 bit 3 to 4.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_mag_operation_mode (u8 mag_operation_mode_u8)

  This API used to write the mag operation mode from page one register from 0x09 bit 3 to 4.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_mag_power_mode (u8 *mag_power_mode_u8)

  This API used to read the mag power mode from page one register from 0x09 bit 4 to 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_mag_power_mode (u8 mag_power_mode_u8)

  This API used to write the mag power mode from page one register from 0x09 bit 4 to 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_range (u8 *gyro_range_u8)

  This API used to read the gyro range from page one register from 0x0A bit 0 to 3.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_range (u8 gyro_range_u8)

  This API used to write the gyro range from page one register from 0x0A bit 0 to 3.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_bw (u8 *gyro_bw_u8)

  This API used to read the gyro bandwidth from page one register from 0x0A bit 3 to 5.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_bw (u8 gyro_bw_u8)

  This API used to write the gyro bandwidth from page one register from 0x0A bit 3 to 5.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_power_mode (u8 *gyro_power_mode_u8)

  This API used to read the gyro power mode from page one register from 0x0B bit 0 to 2.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_power_mode (u8 gyro_power_mode_u8)

  This API used to write the gyro power mode from page one register from 0x0B bit 0 to 2.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_sleep_tmr_mode (u8 *sleep_tmr_u8)

  This API used to read the accel sleep mode from page one register from 0x0C bit 0.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_sleep_tmr_mode (u8 sleep_tmr_u8)

  This API used to write the accel sleep mode from page one register from 0x0C bit 0.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_sleep_durn (u8 *sleep_durn_u8)

  This API used to read the accel sleep duration from page one register from 0x0C bit 1 to 4.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_sleep_durn (u8 sleep_durn_u8)

  This API used to write the accel sleep duration from page one register from 0x0C bit 1 to 4.

- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_sleep_durn (u8 *sleep_durn_u8)

  This API used to write the gyro sleep duration from page one register from 0x0D bit 0 to 2.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_sleep_durn (u8 sleep_durn_u8)

  This API used to write the gyro sleep duration from page one register from 0x0D bit 0 to 2.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_auto_sleep_durn (u8 *auto_sleep_durn_u8)

  This API used to read the gyro auto sleep duration from page one register from 0x0D bit 3 to 5.
- BNO055_RETURN_FUNCTION_TYPE bno055_gyro_set_auto_sleep_durn (u8 auto_sleep_durn_u8, u8 bw)

  This API used to write the gyro auto sleep duration from page one register from 0x0D bit 3 to 5.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_mag_sleep_mode (u8 *sleep_mode_u8)

  This API used to read the mag sleep mode from page one register from 0x0E bit 0.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_mag_sleep_mode (u8 sleep_mode_u8)

  This API used to write the mag sleep mode from page one register from 0x0E bit 0.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_mag_sleep_durn (u8 *sleep_durn_u8)

  This API used to read the mag sleep duration from page one register from 0x0E bit 1 to 4.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_mag_sleep_durn (u8 sleep_durn_u8)

  This API used to write the mag sleep duration from page one register from 0x0E bit 1 to 4.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_mask_gyro_any_motion (u8 *gyro_any_motion_↩ u8)

  This API used to read the gyro anymotion interrupt mask from page one register from 0x0F bit 2.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_mask_gyro_any_motion (u8 gyro_any_motion_u8)

  This API used to write the gyro anymotion interrupt mask from page one register from 0x0F bit 2.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_mask_gyro_highrate (u8 *gyro_highrate_u8)

  This API used to read the gyro highrate interrupt mask from page one register from 0x0F bit 3.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_mask_gyro_highrate (u8 gyro_highrate_u8)

  This API used to write the gyro highrate interrupt mask from page one register from 0x0F bit 3.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_mask_accel_high_g (u8 *accel_high_g_u8)

  This API used to read the accel highg interrupt mask from page one register from 0x0F bit 5.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_mask_accel_high_g (u8 accel_high_g_u8)

  This API used to write the accel highg interrupt mask from page one register from 0x0F bit 5.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_mask_accel_any_motion (u8 *accel_any_↩ motion_u8)

  This API used to read the accel anymotion interrupt mask from page one register from 0x0F bit 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_mask_accel_any_motion (u8 accel_any_motion↩ _u8)

  This API used to write the accel anymotion interrupt mask from page one register from 0x0F bit 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_mask_accel_no_motion (u8 *accel_nomotion_u8)

  This API used to read the accel nomotion interrupt mask from page one register from 0x0F bit 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_mask_accel_no_motion (u8 accel_nomotion_u8)

  This API used to write the accel nomotion interrupt mask from page one register from 0x0F bit 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_gyro_any_motion (u8 *gyro_any_motion_u8)

  This API used to read the gyro anymotion interrupt from page one register from 0x10 bit 2.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_gyro_any_motion (u8 gyro_any_motion_u8)

  This API used to write the gyro anymotion interrupt from page one register from 0x10 bit 2.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_gyro_highrate (u8 *gyro_highrate_u8)

  This API used to read the gyro highrate interrupt from page one register from 0x10 bit 3.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_gyro_highrate (u8 gyro_highrate_u8)

  This API used to write the gyro highrate interrupt from page one register from 0x10 bit 3.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_accel_high_g (u8 *accel_high_g_u8)

  This API used to read the accel highg interrupt from page one register from 0x10 bit 5.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_accel_high_g (u8 accel_high_g_u8)

This API used to write the accel highg interrupt from page one register from 0x10 bit 5.

- BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_accel_any_motion (u8 *accel_any_motion_u8)

  This API used to read the accel anymotion interrupt from page one register from 0x10 bit 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_accel_any_motion (u8 accel_any_motion_u8)

  This API used to write the accel anymotion interrupt from page one register from 0x10 bit 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_accel_no_motion (u8 *accel_nomotion_u8)

  This API used to read the accel nomotion interrupt from page one register from 0x10 bit 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_accel_no_motion (u8 accel_nomotion_u8)

  This API used to write the accel nomotion interrupt from page one register from 0x10 bit 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_any_motion_thres (u8 *accel_any_motion_thres_u8)

  This API used to read the accel any motion threshold from page one register from 0x11 bit 0 to 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_any_motion_thres (u8 accel_any_motion_thres_u8)

  This API used to write the accel any motion threshold from page one register from 0x11 bit 0 to 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_any_motion_durn (u8 *accel_any_motion_durn_u8)

  This API used to read the accel anymotion duration from page one register from 0x12 bit 0 to 1.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_any_motion_durn (u8 accel_any_motion_durn_u8)

  This API used to write the accel anymotion duration from page one register from 0x12 bit 0 to 1.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_any_motion_no_motion_axis_enable (u8 channel_u8, u8 *data_u8)

  This API used to read the accel anymotion enable from page one register from 0x12 bit 2 to 4.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_any_motion_no_motion_axis_enable (u8 channel_u8, u8 data_u8)

  This API used to write the accel anymotion enable from page one register from 0x12 bit 2 to 4.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_high_g_axis_enable (u8 channel_u8, u8 *data_u8)

  This API used to read the accel highg enable from page one register from 0x12 bit 5 to 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_high_g_axis_enable (u8 channel_u8, u8 data_u8)

  This API used to write the accel highg enable from page one register from 0x12 bit 5 to 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_high_g_durn (u8 *accel_high_g_durn_u8)

  This API used to read the accel highg duration from page one register from 0x13 bit 0 to 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_high_g_durn (u8 accel_high_g_durn_u8)

  This API used to write the accel highg duration from page one register from 0x13 bit 0 to 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_high_g_thres (u8 *accel_high_g_thres_u8)

  This API used to read the accel highg threshold from page one register from 0x14 bit 0 to 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_high_g_thres (u8 accel_high_g_thres_u8)

  This API used to write the accel highg threshold from page one register from 0x14 bit 0 to 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_slow_no_motion_thres (u8 *accel_slow_no_motion_thres_u8)

  This API used to read the accel slownomotion threshold from page one register from 0x15 bit 0 to 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_slow_no_motion_thres (u8 accel_slow_no_motion_thres_u8)

  This API used to write the accel slownomotion threshold from page one register from 0x15 bit 0 to 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_slow_no_motion_enable (u8 *accel_slow_no_motion_en_u8)

  This API used to read accel slownomotion enable from page one register from 0x16 bit 0.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_slow_no_motion_enable (u8 accel_slow_no_motion_en_u8)

This API used to write accel slownomotion enable from page one register from 0x16 bit 0.

- BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_slow_no_motion_durn (u8 *accel_slow_no_motion_durn_u8)

  This API used to read accel slownomotion duration from page one register from 0x16 bit 1 to 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_slow_no_motion_durn (u8 accel_slow_no_motion_durn_u8)

  This API used to write accel slownomotion duration from page one register from 0x16 bit 1 to 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_any_motion_axis_enable (u8 channel_u8, u8 *data_u8)

  This API used to read the gyro anymotion enable from page one register from 0x17 bit 0 to 2.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_any_motion_axis_enable (u8 channel_u8, u8 data_u8)

  This API used to write the gyro anymotion enable from page one register from 0x17 bit 0 to 2.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_axis_enable (u8 channel_u8, u8 *data_u8)

  This API used to read the gyro highrate enable from page one register from 0x17 bit 3 to 5.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_axis_enable (u8 channel_u8, u8 data_u8)

  This API used to write the gyro highrate enable from page one register from 0x17 bit 3 to 5.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_any_motion_filter (u8 *gyro_any_motion_filter_u8)

  This API used to read gyro anymotion filter from page one register from 0x17 bit 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_any_motion_filter (u8 gyro_any_motion_filter_u8)

  This API used to write gyro anymotion filter from page one register from 0x17 bit 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_filter (u8 *gyro_highrate_filter_u8)

  This API used to read gyro highrate filter from page one register from 0x17 bit 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_filter (u8 gyro_highrate_filter_u8)

  This API used to write gyro highrate filter from page one register from 0x17 bit 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_x_thres (u8 *gyro_highrate_x_thres_u8)

  This API used to read gyro highrate x threshold from page one register from 0x18 bit 0 to 4.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_x_thres (u8 gyro_highrate_x_thres_u8)

  This API used to write gyro highrate x threshold from page one register from 0x18 bit 0 to 4.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_x_hyst (u8 *gyro_highrate_x_hyst_u8)

  This API used to read gyro highrate x hysteresis from page one register from 0x18 bit 5 to 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_x_hyst (u8 gyro_highrate_x_hyst_u8)

  This API used to write gyro highrate x hysteresis from page one register from 0x18 bit 5 to 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_x_durn (u8 *gyro_highrate_x_durn_u8)

  This API used to read gyro highrate x duration from page one register from 0x19 bit 0 to 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_x_durn (u8 gyro_highrate_x_durn_u8)

  This API used to write gyro highrate x duration from page one register from 0x19 bit 0 to 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_y_thres (u8 *gyro_highrate_y_thres_u8)

  This API used to read gyro highrate y threshold from page one register from 0x1A bit 0 to 4.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_y_thres (u8 gyro_highrate_y_thres_u8)

  This API used to write gyro highrate y threshold from page one register from 0x1A bit 0 to 4.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_y_hyst (u8 *gyro_highrate_y_hyst_u8)

  This API used to read gyro highrate y hysteresis from page one register from 0x1A bit 5 to 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_y_hyst (u8 gyro_highrate_y_hyst_u8)

  This API used to write gyro highrate y hysteresis from page one register from 0x1A bit 5 to 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_y_durn (u8 *gyro_highrate_y_durn_u8)

  This API used to read gyro highrate y duration from page one register from 0x1B bit 0 to 7.

- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_y_durn (u8 gyro_highrate_y_durn_u8)

  This API used to write gyro highrate y duration from page one register from 0x1B bit 0 to 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_z_thres (u8 ∗gyro_highrate_z_thres_↩ u8)

  This API used to read gyro highrate z threshold from page one register from 0x1C bit 0 to 4.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_z_thres (u8 gyro_highrate_z_thres_u8)

  This API used to write gyro highrate z threshold from page one register from 0x1C bit 0 to 4.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_z_hyst (u8 ∗gyro_highrate_z_hyst_u8)

  This API used to read gyro highrate z hysteresis from page one register from 0x1C bit 5 to 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_z_hyst (u8 gyro_highrate_z_hyst_u8)

  This API used to write gyro highrate z hysteresis from page one register from 0x1C bit 5 to 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_z_durn (u8 ∗gyro_highrate_z_durn_u8)

  This API used to read gyro highrate z duration from page one register from 0x1D bit 0 to 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_z_durn (u8 gyro_highrate_z_durn_u8)

  This API used to write gyro highrate z duration from page one register from 0x1D bit 0 to 7.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_any_motion_thres (u8 ∗gyro_any_motion_thres↩ _u8)

  This API used to read gyro anymotion threshold from page one register from 0x1E bit 0 to 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_any_motion_thres (u8 gyro_any_motion_thres_↩ u8)

  This API used to write gyro anymotion threshold from page one register from 0x1E bit 0 to 6.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_any_motion_slope_samples (u8 ∗gyro_any_↩ motion_slope_samples_u8)

  This API used to read gyro anymotion slope samples from page one register from 0x1F bit 0 to 1.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_any_motion_slope_samples (u8 gyro_any_↩ motion_slope_samples_u8)

  This API used to write gyro anymotion slope samples from page one register from 0x1F bit 0 to 1.
- BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_any_motion_awake_durn (u8 ∗gyro_awake_↩ durn_u8)

  This API used to read gyro anymotion awake duration from page one register from 0x1F bit 2 to 3.
- BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_any_motion_awake_durn (u8 gyro_awake_↩ durn_u8)

  This API used to write gyro anymotion awake duration from page one register from 0x1F bit 2 to 3.

2.1.1 Detailed Description

Date

10/01/2020

Version

2.0.6

2.1.2 Function Documentation

2.1.2.1 bno055_convert_double_accel_x_mg()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_accel_x_mg (
            double * accel_x_d )
```

This API is used to convert the accel x raw data to millig output as double.

Parameters

accel_x↩ _d	: The accel x millig data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.2 bno055_convert_double_accel_x_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_accel_x_msq (
            double * accel_x_d )
```

This API is used to convert the accel x raw data to meterpersecseq output as double.

Parameters

accel_x↩ _d	: The accel x meterpersecseq data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.3 bno055_convert_double_accel_xyz_mg()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_accel_xyz_mg (
            struct bno055_accel_double_t * accel_xyz )
```

This API is used to convert the accel xyz raw data to millig output as double.

Parameters

accel_xyz	: The millig data of accel xyz

Parameter	result
x	millig data of accel
y	millig data of accel
z	millig data of accel

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.4 bno055_convert_double_accel_xyz_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_accel_xyz_msq (
            struct bno055_accel_double_t * accel_xyz )
```

This API is used to convert the accel xyz raw data to meterpersecseq output as double.

Parameters

accel_xyz	: The meterpersecseq data of accel xyz

Parameter	result
x	meterpersecseq data of accel
y	meterpersecseq data of accel
z	meterpersecseq data of accel

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.5 bno055_convert_double_accel_y_mg()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_accel_y_mg (
            double * accel_y_d )
```

This API is used to convert the accel y raw data to millig output as double.

Parameters

accel_y_d	: The accel y millig data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.6 bno055_convert_double_accel_y_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_accel_y_msq (
            double * accel_y_d )
```

This API is used to convert the accel y raw data to meterpersecseq output as double.

Parameters

accel_y_d	: The accel y meterpersecseq data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.7 bno055_convert_double_accel_z_mg()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_accel_z_mg (
            double * accel_z_d )
```

This API is used to convert the accel z raw data to millig output as double.

Parameters

accel_z_d	: The accel z millig data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.8 bno055_convert_double_accel_z_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_accel_z_msq (
            double * accel_z_d )
```

This API is used to convert the accel z raw data to meterpersecseq output as double.

Parameters

accel_z_d	: The accel z meterpersecseq data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.9 bno055_convert_double_euler_h_deg()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_euler_h_deg (
            double * euler_h_d )
```

This API is used to convert the Euler h raw data to degree output as double.

Parameters

euler_h↩ _d	: The double value of Euler h degree

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.10 bno055_convert_double_euler_h_rad()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_euler_h_rad (
            double * euler_h_d )
```

This API is used to convert the Euler h raw data to radians output as double.

Parameters

euler_h↩ _d	: The double value of Euler h radians

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.11 bno055_convert_double_euler_hpr_deg()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_euler_hpr_deg (
            struct bno055_euler_double_t * euler_hpr )
```

This API is used to convert the Euler hpr raw data to degree output as double.

Parameters

euler_hpr	: The degree data of Euler hpr

Parameter	result
h	degree data of Euler
r	degree data of Euler
p	degree data of Euler

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.12 bno055_convert_double_euler_hpr_rad()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_euler_hpr_rad (
            struct bno055_euler_double_t * euler_hpr )
```

This API is used to convert the Euler hpr raw data to radians output as double.

Parameters

euler_hpr	: The radians data of Euler hpr

Parameter	result
h	radians data of Euler
r	radians data of Euler
p	radians data of Euler

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.13 bno055_convert_double_euler_p_deg()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_euler_p_deg (
            double * euler_p_d )
```

This API is used to convert the Euler p raw data to degree output as double.

Parameters

euler_p↩ _d	: The double value of Euler p degree

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.14 bno055_convert_double_euler_p_rad()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_euler_p_rad (
            double * euler_p_d )
```

This API is used to convert the Euler p raw data to radians output as double.

Parameters

euler_p↩ _d	: The double value of Euler p radians

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.15 bno055_convert_double_euler_r_deg()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_euler_r_deg (
            double * euler_r_d )
```

This API is used to convert the Euler r raw data to degree output as double.

Parameters

euler_r_d	: The double value of Euler r degree

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.16 bno055_convert_double_euler_r_rad()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_euler_r_rad (
            double * euler_r_d )
```

This API is used to convert the Euler r raw data to radians output as double.

Parameters

euler_r_d	: The double value of Euler r radians

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.17 bno055_convert_double_gravity_xyz_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_gravity_xyz_msq (
            struct bno055_gravity_double_t * gravity_xyz )
```

This API is used to convert the gravity xyz raw data to meterpersecseq output as double.

Parameters

gravity_xyz	: The meterpersecseq data of gravity xyz

Parameter	result
x	meterpersecseq data of gravity
y	meterpersecseq data of gravity
z	meterpersecseq data of gravity

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.18 bno055_convert_double_gyro_x_dps()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_gyro_x_dps (
            double * gyro_x_d )
```

This API is used to convert the gyro x raw data to dps output as double.

Parameters

gyro_x↩ _d	: The gyro x dps double data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.19 bno055_convert_double_gyro_x_rps()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_gyro_x_rps (
            double * gyro_x_d )
```

This API is used to convert the gyro x raw data to rps output as double.

Parameters

gyro_x_d	: The gyro x dps double data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.20 bno055_convert_double_gyro_xyz_dps()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_gyro_xyz_dps (
            struct bno055_gyro_double_t * gyro_xyz )
```

This API is used to convert the gyro xyz raw data to dps output as double.

Parameters

gyro_xyz	: The dps data of gyro xyz

Parameter	result
x	dps data of gyro
y	dps data of gyro
z	dps data of gyro

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.21 bno055_convert_double_gyro_xyz_rps()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_gyro_xyz_rps (
            struct bno055_gyro_double_t * gyro_xyz )
```

This API is used to convert the gyro xyz raw data to rps output as double.

Parameters

gyro_xyz	: The rps data of gyro xyz

Parameter	result
x	rps data of gyro
y	rps data of gyro
z	rps data of gyro

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.22 bno055_convert_double_gyro_y_dps()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_gyro_y_dps (
            double * gyro_y_d )
```

This API is used to convert the gyro y raw data to dps output as double.

Parameters

gyro_y_d	: The gyro y dps double data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.23 bno055_convert_double_gyro_y_rps()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_gyro_y_rps (
            double * gyro_y_d )
```

This API is used to convert the gyro y raw data to rps output as double.

Parameters

gyro_y_d	: The gyro y dps double data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.24 bno055_convert_double_gyro_z_dps()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_gyro_z_dps (
            double * gyro_z_d )
```

This API is used to convert the gyro z raw data to dps output as double.

Parameters

gyro_z_d	: The gyro z dps double data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.25 bno055_convert_double_gyro_z_rps()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_gyro_z_rps (
            double * gyro_z_d )
```

This API is used to convert the gyro z raw data to rps output as double.

Parameters

gyro_z↩ _d	: The gyro z rps double data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.26 bno055_convert_double_linear_accel_x_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_linear_accel_x_msq (
            double * linear_accel_x_d )
```

This API is used to convert the linear accel x raw data to meterpersecseq output as double.

Parameters

linear_accel_x↩ _d	: The double value of linear accel x meterpersecseq

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.27 bno055_convert_double_linear_accel_xyz_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_linear_accel_xyz_msq (
            struct bno055_linear_accel_double_t * linear_accel_xyz )
```

This API is used to convert the linear accel xyz raw data to meterpersecseq output as double.

Parameters

linear_accel_xyz	: The meterpersecseq data of linear accel xyz

Parameter	result
x	meterpersecseq data of linear accel
y	meterpersecseq data of linear accel
z	meterpersecseq data of linear accel

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.28 bno055_convert_double_linear_accel_y_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_linear_accel_y_msq (
            double * linear_accel_y_d )
```

This API is used to convert the linear accel y raw data to meterpersecseq output as double.

Parameters

linear_accel_y_d	: The double value of linear accel y meterpersecseq

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.29 bno055_convert_double_linear_accel_z_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_linear_accel_z_msq (
            double * linear_accel_z_d )
```

This API is used to convert the linear accel z raw data to meterpersecseq output as double.

Parameters

linear_accel_z_d	: The double value of linear accel z meterpersecseq

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.30 bno055_convert_double_mag_x_uT()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_mag_x_uT (
            double * mag_x_d )
```

This API is used to convert the mag x raw data to microTesla output as double.

Parameters

mag_x↩ _d	: The mag x microTesla data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.31 bno055_convert_double_mag_xyz_uT()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_mag_xyz_uT (
            struct bno055_mag_double_t * mag_xyz )
```

This API is used to convert the mag yz raw data to microTesla output as double.

Parameters

mag_xyz	: The microTesla data of mag xyz

Parameter	result
x	microTesla data of mag
y	microTesla data of mag
z	microTesla data of mag

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.32 bno055_convert_double_mag_y_uT()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_mag_y_uT (
            double * mag_y_d )
```

This API is used to convert the mag y raw data to microTesla output as double.

Parameters

mag_y↩ _d	: The mag y microTesla data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.33 bno055_convert_double_mag_z_uT()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_mag_z_uT (
            double * mag_z_d )
```

This API is used to convert the mag z raw data to microTesla output as double.

Parameters

mag_z↩ _d	: The mag z microTesla data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.34 bno055_convert_double_temp_celsius()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_temp_celsius (
            double * temp_d )
```

This API is used to convert the temperature data to Celsius output as double.

Parameters

temp↩ _d	: The double value of temperature Celsius

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.35 bno055_convert_double_temp_fahrenheit()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_double_temp_fahrenheit (
            double * temp_d )
```

This API is used to convert the temperature data to Fahrenheit output as double.

Parameters

temp↩ _d	: The double value of temperature Fahrenheit

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.36 bno055_convert_float_accel_x_mg()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_accel_x_mg (
            float * accel_x_f )
```

This API is used to convert the accel x raw data to millig output as float.

Parameters

accel_x_f	: The accel x millig data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.37 bno055_convert_float_accel_x_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_accel_x_msq (
            float * accel_x_f )
```

This API is used to convert the accel x raw data to meterpersecseq output as float.

Parameters

accel_x_f	: The accel x meterpersecseq data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.38 bno055_convert_float_accel_xyz_mg()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_accel_xyz_mg (
            struct bno055_accel_float_t * accel_xyz )
```

This API is used to convert the accel xyz raw data to millig output as float.

Parameters

accel_xyz	: The millig data of accel xyz

Parameter	result
x	millig data of accel
y	millig data of accel
z	millig data of accel

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.39 bno055_convert_float_accel_xyz_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_accel_xyz_msq (
            struct bno055_accel_float_t * accel_xyz )
```

This API is used to convert the accel xyz raw data to meterpersecseq output as float.

Parameters

accel_xyz	: The meterpersecseq data of accel xyz

Parameter	result
x	meterpersecseq data of accel
y	meterpersecseq data of accel
z	meterpersecseq data of accel

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.40 bno055_convert_float_accel_y_mg()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_accel_y_mg (
            float * accel_y_f )
```

This API is used to convert the accel y raw data to millig output as float.

Parameters

accel_↩ y_f	: The accel y millig data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.41 bno055_convert_float_accel_y_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_accel_y_msq (
            float * accel_y_f )
```

This API is used to convert the accel x raw data to meterpersecseq output as float.

Parameters

accel_↩ y_f	: The accel y meterpersecseq data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.42 bno055_convert_float_accel_z_mg()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_accel_z_mg (
            float * accel_z_f )
```

This API is used to convert the accel z raw data to millig output as float.

Parameters

accel_↩ z_f	: The accel z millig data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.43 bno055_convert_float_accel_z_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_accel_z_msq (
            float * accel_z_f )
```

This API is used to convert the accel z raw data to meterpersecseq output as float.

Parameters

accel_↩ z_f	: The accel z meterpersecseq data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.44 bno055_convert_float_euler_h_deg()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_euler_h_deg (
            float * euler_h_f )
```

This API is used to convert the Euler h raw data to degree output as float.

Parameters

euler_↩ h_f	: The float value of Euler h degree

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.45 bno055_convert_float_euler_h_rad()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_euler_h_rad (
            float * euler_h_f )
```

This API is used to convert the Euler h raw data to radians output as float.

Parameters

euler_h_f	: The float value of Euler h radians

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.46 bno055_convert_float_euler_hpr_deg()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_euler_hpr_deg (
            struct bno055_euler_float_t * euler_hpr )
```

This API is used to convert the Euler hrp raw data to degree output as float.

Parameters

euler_hpr	: The degree data of Euler hrp

Parameter	result
h	degree data of Euler
r	degree data of Euler
p	degree data of Euler

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.47 bno055_convert_float_euler_hpr_rad()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_euler_hpr_rad (
            struct bno055_euler_float_t * euler_hpr )
```

This API is used to convert the Euler xyz raw data to radians output as float.

Parameters

euler_hpr	: The radians data of Euler hrp

Parameter	result
h	radians data of Euler
r	radians data of Euler
p	radians data of Euler

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.48 bno055_convert_float_euler_p_deg()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_euler_p_deg (
            float * euler_p_f )
```

This API is used to convert the Euler p raw data to degree output as float.

Parameters

euler_↩ p_f	: The float value of Euler p degree

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.49 bno055_convert_float_euler_p_rad()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_euler_p_rad (
            float * euler_p_f )
```

This API is used to convert the Euler p raw data to radians output as float.

Parameters

euler_↩ p_f	: The float value of Euler p radians

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.50 bno055_convert_float_euler_r_deg()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_euler_r_deg (
            float * euler_r_f )
```

This API is used to convert the Euler r raw data to degree output as float.

Parameters

euler_↩ r_f	: The float value of Euler r degree

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.51 bno055_convert_float_euler_r_rad()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_euler_r_rad (
            float * euler_r_f )
```

This API is used to convert the Euler r raw data to radians output as float.

Parameters

euler_r_f	: The float value of Euler r radians

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.52 bno055_convert_float_gravity_xyz_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_gravity_xyz_msq (
            struct bno055_gravity_float_t * gravity_xyz )
```

This API is used to convert the gravity xyz raw data to meterpersecseq output as float.

Parameters

gravity_xyz	: The meterpersecseq data of gravity xyz

Parameter	result
x	meterpersecseq data of gravity
y	meterpersecseq data of gravity
z	meterpersecseq data of gravity

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.53 bno055_convert_float_gyro_x_dps()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_gyro_x_dps (
            float * gyro_x_f )
```

This API is used to convert the gyro x raw data to dps output as float.

Parameters

gyro_↩ x_f	: The gyro x dps float data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.54 bno055_convert_float_gyro_x_rps()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_gyro_x_rps (
            float * gyro_x_f )
```

This API is used to convert the gyro x raw data to rps output as float.

Parameters

gyro_↩ x_f	: The gyro x dps float data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.55 bno055_convert_float_gyro_xyz_dps()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_gyro_xyz_dps (
            struct bno055_gyro_float_t * gyro_xyz_data )
```

This API is used to convert the gyro xyz raw data to dps output as float.

Parameters

gyro_xyz_data	: The dps data of gyro xyz

Parameter	result
x	dps data of gyro
y	dps data of gyro
z	dps data of gyro

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.56 bno055_convert_float_gyro_xyz_rps()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_gyro_xyz_rps (
            struct bno055_gyro_float_t * gyro_xyz_data )
```

This API is used to convert the gyro xyz raw data to rps output as float.

Parameters

gyro_xyz_data	: The rps data of gyro xyz

Parameter	result
x	rps data of gyro
y	rps data of gyro
z	rps data of gyro

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.57 bno055_convert_float_gyro_y_dps()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_gyro_y_dps (
            float * gyro_y_f )
```

This API is used to convert the gyro y raw data to dps output as float.

Parameters

gyro_↩ y_f	: The gyro y dps float data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.58 bno055_convert_float_gyro_y_rps()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_gyro_y_rps (
            float * gyro_y_f )
```

This API is used to convert the gyro y raw data to rps output as float.

Parameters

gyro_↩ y_f	: The gyro y dps float data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.59 bno055_convert_float_gyro_z_dps()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_gyro_z_dps (
            float * gyro_z_f )
```

This API is used to convert the gyro z raw data to dps output as float.

Parameters

gyro_z_f	: The gyro z dps float data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.60 bno055_convert_float_gyro_z_rps()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_gyro_z_rps (
            float * gyro_z_f )
```

This API is used to convert the gyro z raw data to rps output as float.

Parameters

gyro_z_f	: The gyro z rps float data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.61 bno055_convert_float_linear_accel_x_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_linear_accel_x_msq (
            float * linear_accel_x_f )
```

This API is used to convert the linear accel x raw data to meterpersecseq output as float.

Parameters

linear_accel_x_f	: The float value of linear accel x meterpersecseq

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.62 bno055_convert_float_linear_accel_xyz_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_linear_accel_xyz_msq (
            struct bno055_linear_accel_float_t * linear_accel_xyz )
```

This API is used to convert the linear accel xyz raw data to meterpersecseq output as float.

Parameters

linear_accel_xyz	: The meterpersecseq data of linear accel xyz

Parameter	result
x	meterpersecseq data of linear accel
y	meterpersecseq data of linear accel
z	meterpersecseq data of linear accel

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.63 bno055_convert_float_linear_accel_y_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_linear_accel_y_msq (
            float * linear_accel_y_f )
```

This API is used to convert the linear accel y raw data to meterpersecseq output as float.

Parameters

linear_accel_↩ y_f	: The float value of linear accel y meterpersecseq

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.64 bno055_convert_float_linear_accel_z_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_linear_accel_z_msq (
            float * linear_accel_z_f )
```

This API is used to convert the linear accel z raw data to meterpersecseq output as float.

Parameters

linear_accel_↩ z_f	: The float value of linear accel z meterpersecseq

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.65 bno055_convert_float_mag_x_uT()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_mag_x_uT (
            float * mag_x_f )
```

This API is used to convert the mag x raw data to microTesla output as float.

Parameters

mag_↩ x_f	: The mag x microTesla data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.66 bno055_convert_float_mag_xyz_uT()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_mag_xyz_uT (
            struct bno055_mag_float_t * mag_xyz_data )
```

This API is used to convert the mag yz raw data to microTesla output as float.

Parameters

mag_xyz_data	: The microTesla data of mag xyz

Parameter	result
x	microTesla data of mag
y	microTesla data of mag
z	microTesla data of mag

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.67 bno055_convert_float_mag_y_uT()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_mag_y_uT (
            float * mag_y_f )
```

This API is used to convert the mag y raw data to microTesla output as float.

Parameters

mag_↩ y_f	: The mag y microTesla data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.68 bno055_convert_float_mag_z_uT()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_mag_z_uT (
            float * mag_z_f )
```

This API is used to convert the mag z raw data to microTesla output as float.

Parameters

mag_z_f	: The mag z microTesla data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.69 bno055_convert_float_temp_celsius()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_temp_celsius (
            float * temp_f )
```

This API is used to convert the temperature data to Celsius output as float.

Parameters

temp_f	: The float value of temperature Celsius

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.70 bno055_convert_float_temp_fahrenheit()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_float_temp_fahrenheit (
            float * temp_f )
```

This API is used to convert the temperature data to Fahrenheit output as float.

Parameters

temp_f	: The float value of temperature Fahrenheit

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.71 bno055_convert_gravity_double_x_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_gravity_double_x_msq (
            double * gravity_x_d )
```

This API is used to convert the gravity x raw data to meterpersecseq output as double.

Parameters

gravity_x_d	: The double value of gravity x meterpersecseq

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.72 bno055_convert_gravity_double_y_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_gravity_double_y_msq (
            double * gravity_y_d )
```

This API is used to convert the gravity y raw data to meterpersecseq output as double.

Parameters

gravity_y_d	: The double value of gravity y meterpersecseq

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.73 bno055_convert_gravity_double_z_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_gravity_double_z_msq (
            double * gravity_z_d )
```

This API is used to convert the gravity z raw data to meterpersecseq output as double.

Parameters

gravity_z_d	: The double value of gravity z meterpersecseq

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.74 bno055_convert_gravity_float_x_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_gravity_float_x_msq (
            float * gravity_x_f )
```

This API is used to convert the gravity x raw data to meterpersecseq output as float.

Parameters

gravity_x_f	: The float value of gravity x meterpersecseq

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.75 bno055_convert_gravity_float_y_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_gravity_float_y_msq (
            float * gravity_y_f )
```

This API is used to convert the gravity y raw data to meterpersecseq output as float.

Parameters

gravity_y_f	: The float value of gravity y meterpersecseq

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.76 bno055_convert_gravity_float_z_msq()

```
BNO055_RETURN_FUNCTION_TYPE bno055_convert_gravity_float_z_msq (
            float * gravity_z_f )
```

This API is used to convert the gravity z raw data to meterpersecseq output as float.

Parameters

gravity_z_f	: The float value of gravity z meterpersecseq

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.77 bno055_get_accel_any_motion_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_any_motion_durn (
            u8 * accel_any_motion_durn_u8 )
```

This API used to read the accel anymotion duration from page one register from 0x12 bit 0 to 1.

Parameters

accel_any_motion_durn_u8	: The value of accel anymotion duration

accel_any_motion_durn_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.78 bno055_get_accel_any_motion_no_motion_axis_enable()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_any_motion_no_motion_axis_enable (
            u8 channel_u8,
            u8 * data_u8 )
```

This API used to read the accel anymotion enable from page one register from 0x12 bit 2 to 4.

Parameters

data_u8	: The value of accel anymotion enable

data_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Parameters

channel_u8	: The value of accel anymotion axis selection

channel_u8	value
BNO055_ACCEL_ANY_MOTION_NO_↩ MOTION_X_AXIS	0
BNO055_ACCEL_ANY_MOTION_NO_↩ MOTION_Y_AXIS	1
BNO055_ACCEL_ANY_MOTION_NO_↩ MOTION_Y_AXIS	2

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.79 bno055_get_accel_any_motion_thres()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_any_motion_thres (
            u8 * accel_any_motion_thres_u8 )
```

This API used to read the accel any motion threshold from page one register from 0x11 bit 0 to 7.

Parameters

accel_any_motion_thres_u8	: The value of any motion threshold

accel_any_motion_thres_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Accel anymotion threshold dependent on the range values

accel_range_u8	threshold	LSB
2g	3.19mg	1LSB
4g	7.81mg	1LSB
8g	15.63mg	1LSB
16g	31.25mg	1LSB

2.1.2.80 bno055_get_accel_bw()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_bw (
            u8 * accel_bw_u8 )
```

This API used to read the accel bandwidth from page one register from 0x08 bit 2 to 4.

Parameters

accel_bw_u8	: The value of accel bandwidth accel_bw_u8 | result ---------- | ---------- 0x00 | BNO055_ACCEL_BW_7_81HZ 0x01 | BNO055_ACCEL_BW_15_63HZ 0x02 | BNO055_ACCEL_BW_31_25HZ 0x03 | BNO055_ACCEL_BW_62_5HZ 0x04 | BNO055_ACCEL_BW_125HZ 0x05 | BNO055_ACCEL_BW_250HZ 0x06 | BNO055_ACCEL_BW_500HZ 0x07 | BNO055_ACCEL_BW_1000HZ

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.81 bno055_get_accel_calib_stat()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_calib_stat (
            u8 * accel_calib_u8 )
```

This API used to read accel calibration status from register from 0x35 bit 2 and 3.

Parameters

accel_calib_u8	: The value of accel calib status

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.82 bno055_get_accel_high_g_axis_enable()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_high_g_axis_enable (
            u8 channel_u8,
            u8 * data_u8 )
```

This API used to read the accel highg enable from page one register from 0x12 bit 5 to 7.

Parameters

data_u8	: The value of accel highg enable **data_u8** / **result** 0x01 / BNO055_BIT_ENABLE 0x00 / BNO055_BIT_DISABLE
channel_u8	: The value of accel highg axis selection **channel_u8** / **value** BNO055_ACCEL_HIGH_G_X_AXIS / 0 BNO055_ACCEL_HIGH_G_Y_AXIS / 1 BNO055_ACCEL_HIGH_G_Z_AXIS / 2

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.83 bno055_get_accel_high_g_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_high_g_durn (
            u8 * accel_high_g_durn_u8 )
```

This API used to read the accel highg duration from page one register from 0x13 bit 0 to 7.

Parameters

accel_high_g_durn_u8	: The value of accel highg duration

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

The high-g interrupt trigger delay according to [highg duration + 1] ∗ 2 ms

in a range from 2 ms to 512 ms

2.1.2.84 bno055_get_accel_high_g_thres()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_high_g_thres (
            u8 * accel_high_g_thres_u8 )
```

This API used to read the accel highg threshold from page one register from 0x14 bit 0 to 7.

Parameters

accel_high_g_thres_u8	: The value of accel highg threshold

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Accel highg interrupt threshold dependent for accel g range

accel_range_u8	threshold	LSB
2g	7.81mg	1LSB
4g	15.63mg	1LSB
8g	31.25mg	1LSB
16g	62.5mg	1LSB

2.1.2.85 bno055_get_accel_power_mode()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_power_mode (
            u8 * accel_power_mode_u8 )
```

This API used to read the accel power mode from page one register from 0x08 bit 5 to 7.

Parameters

accel_power_mode_u8 : The value of accel power mode

accel_power_mode_u8	result
0x00	BNO055_ACCEL_NORMAL
0x01	BNO055_ACCEL_SUSPEND
0x02	BNO055_ACCEL_LOWPOWER_1
0x03	BNO055_ACCEL_STANDBY
0x04	BNO055_ACCEL_LOWPOWER_2
0x05	BNO055_ACCEL_DEEPSUSPEND

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.86 bno055_get_accel_range()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_range (
            u8 * accel_range_u8 )
```

This API used to read the accel range from page one register from 0x08 bit 0 and 1.

Parameters

accel_range_u8 : The value of accel range

accel_range_u8	result
0x00	BNO055_ACCEL_RANGE_2G
0x01	BNO055_ACCEL_RANGE_4G
0x02	BNO055_ACCEL_RANGE_8G
0x03	BNO055_ACCEL_RANGE_16G

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.87 bno055_get_accel_sleep_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_sleep_durn (
            u8 * sleep_durn_u8 )
```

This API used to read the accel sleep duration from page one register from 0x0C bit 1 to 4.

Parameters

sleep_durn_u8	: The value of accel sleep duration

sleep_durn_u8	result
0x05	BNO055_ACCEL_SLEEP_DURN_0_5MS
0x06	BNO055_ACCEL_SLEEP_DURN_1MS
0x07	BNO055_ACCEL_SLEEP_DURN_2MS
0x08	BNO055_ACCEL_SLEEP_DURN_4MS
0x09	BNO055_ACCEL_SLEEP_DURN_6MS
0x0A	BNO055_ACCEL_SLEEP_DURN_10MS
0x0B	BNO055_ACCEL_SLEEP_DURN_25MS
0x0C	BNO055_ACCEL_SLEEP_DURN_50MS
0x0D	BNO055_ACCEL_SLEEP_DURN_100MS
0x0E	BNO055_ACCEL_SLEEP_DURN_500MS
0x0F	BNO055_ACCEL_SLEEP_DURN_1S

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.88 bno055_get_accel_sleep_tmr_mode()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_sleep_tmr_mode (
            u8 * sleep_tmr_u8 )
```

This API used to read the accel sleep mode from page one register from 0x0C bit 0.

Parameters

sleep_tmr_u8	: The value of accel sleep mode

sleep_tmr_u8	result
0x00	enable EventDrivenSampling(EDT)
0x01	enable Equidistant sampling mode(EST)

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.89 bno055_get_accel_slow_no_motion_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_slow_no_motion_durn (
            u8 * accel_slow_no_motion_durn_u8 )
```

This API used to read accel slownomotion duration from page one register from 0x16 bit 1 to 6.

Parameters

accel_slow_no_motion_durn_u8	: The value of accel slownomotion duration

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.90 bno055_get_accel_slow_no_motion_enable()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_slow_no_motion_enable (
            u8 * accel_slow_no_motion_en_u8 )
```

This API used to read accel slownomotion enable from page one register from 0x16 bit 0.

Parameters

accel_slow_no_motion_en_u8	: The value of accel slownomotion enable

accel_slow_no_motion_en_u8	result
0x01	Slow motion
0x00	No motion

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.91 bno055_get_accel_slow_no_motion_thres()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_slow_no_motion_thres (
            u8 * accel_slow_no_motion_thres_u8 )
```

This API used to read the accel slownomotion threshold from page one register from 0x15 bit 0 to 7.

Parameters

accel_slow_no_motion_thres_u8	: The value of accel slownomotion threshold

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Accel slow no motion interrupt threshold dependent for accel g range

accel_range_u8	threshold	LSB
2g	3.19mg	1LSB
4g	7.81mg	1LSB
8g	15.63mg	1LSB
16g	31.25mg	1LSB

2.1.2.92 bno055_get_accel_unit()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_accel_unit (
            u8 * accel_unit_u8 )
```

This API used to read the accel unit from register from 0x3B bit 0.

Parameters

accel_unit_u8	: The value of accel unit

accel_unit_u8	result
0x00	BNO055_ACCEL_UNIT_MSQ
0x01	BNO055_ACCEL_UNIT_MG

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.93 bno055_get_axis_remap_value()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_axis_remap_value (
            u8 * remap_axis_u8 )
```

This API used to read the axis remap value from register from 0x41 bit 0 and 5.

Parameters

remap_axis_u8	: The value of axis remapping

remap_axis_u8	result	comments
0X21	BNO055_REMAP_X_Y	Z=Z;X=Y;Y=X
0X18	BNO055_REMAP_Y_Z	X=X;Y=Z;Z=Y
0X06	BNO055_REMAP_Z_X	Y=Y;X=Z;Z=X
0X12	BNO055_REMAP_X_Y_Z_TYPE0	X=Z;Y=X;Z=Y
0X09	BNO055_REMAP_X_Y_Z_TYPE1	X=Y;Y=Z;Z=X
0X24	BNO055_DEFAULT_AXIS	X=X;Y=Y;Z=Z

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

: For axis sign remap refer the following APIs x-axis :

bno055_set_x_remap_sign()

y-axis :

bno055_set_y_remap_sign()

z-axis :

bno055_set_z_remap_sign()

2.1.2.94 bno055_get_clk_src()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_clk_src (
            u8 * clk_src_u8 )
```

This API used to read the clk source from register from 0x3F bit 7.

Parameters

clk_src_u8	: The value of clk source

clk_src_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.95 bno055_get_data_output_format()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_data_output_format (
            u8 * data_output_format_u8 )
```

This API used to read the current selected orientation mode from register from 0x3B bit 7.

Parameters

data_output_format_u8	: The value of data output format

data_output_format_u8	result
0x00	Windows
0x01	Android

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.96 bno055_get_euler_unit()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_euler_unit (
            u8 * euler_unit_u8 )
```

This API used to read the Euler unit from register from 0x3B bit 2.

Parameters

euler_unit_u8	: The value of accel unit

euler_unit_u8	result
0x00	BNO055_EULER_UNIT_DEG
0x01	BNO055_EULER_UNIT_RAD

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.97 bno055_get_gyro_any_motion_awake_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_any_motion_awake_durn (
            u8 * gyro_awake_durn_u8 )
```

This API used to read gyro anymotion awake duration from page one register from 0x1F bit 2 to 3.

Parameters

gyro_awake_durn_u8	: The value of gyro anymotion awake duration

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.98 bno055_get_gyro_any_motion_axis_enable()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_any_motion_axis_enable (
            u8 channel_u8,
            u8 * data_u8 )
```

This API used to read the gyro anymotion enable from page one register from 0x17 bit 0 to 2.

Parameters

data_u8	: The value of gyro anymotion enable **data_u8** / **result** 0x01 / BNO055_BIT_ENABLE 0x00 / BNO055_BIT_DISABLE
channel_u8	: The value of gyro anymotion axis selection **channel_u8** / **value** BNO055_GYRO_ANY_MOTIONX_AXIS / 0 BNO055_GYRO_ANY_MOTIONY_AXIS / 1 BNO055_GYRO_ANY_MOTIONZ_AXIS / 2

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.99 bno055_get_gyro_any_motion_filter()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_any_motion_filter (
            u8 * gyro_any_motion_filter_u8 )
```

This API used to read gyro anymotion filter from page one register from 0x17 bit 6.

Parameters

gyro_any_motion_filter_u8	: The value of gyro anymotion filter

gyro_any_motion_filter_u8	result
0x00	BNO055_GYRO_FILTERED_↩ CONFIG
0x01	BNO055_GYRO_UNFILTERED_↩ CONFIG

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.100 bno055_get_gyro_any_motion_slope_samples()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_any_motion_slope_samples (
            u8 * gyro_any_motion_slope_samples_u8 )
```

This API used to read gyro anymotion slope samples from page one register from 0x1F bit 0 to 1.

Parameters

gyro_any_motion_slope_samples_u8	: The value of gyro anymotion slope samples

gyro_any_motion_slope_↩ samples_u8	result
0	8 samples
1	16 samples
2	32 samples
3	64 samples

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.101 bno055_get_gyro_any_motion_thres()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_any_motion_thres (
            u8 * gyro_any_motion_thres_u8 )
```

This API used to read gyro anymotion threshold from page one register from 0x1E bit 0 to 6.

Parameters

gyro_any_motion_thres_u8	: The value of gyro anymotion threshold

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro anymotion interrupt threshold dependent on the selection of gyro range

gyro_range_u8	threshold	LSB
2000	1dps	1LSB
1000	0.5dps	1LSB
500	0.25dps	1LSB

2.1.2.102 bno055_get_gyro_auto_sleep_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_auto_sleep_durn (
            u8 * auto_sleep_durn_u8 )
```

This API used to read the gyro auto sleep duration from page one register from 0x0D bit 3 to 5.

Parameters

auto_sleep_durn_u8	: The value of gyro auto sleep duration

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.103 bno055_get_gyro_bw()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_bw (
            u8 * gyro_bw_u8 )
```

This API used to read the gyro bandwidth from page one register from 0x0A bit 3 to 5.

Parameters

gyro_bw_u8	: The value of gyro bandwidth

gyro_bw_u8	result
0x00	BNO055_GYRO_BW_523HZ
0x01	BNO055_GYRO_BW_230HZ
0x02	BNO055_GYRO_BW_116HZ
0x03	BNO055_GYRO_BW_47HZ
0x04	BNO055_GYRO_BW_23HZ
0x05	BNO055_GYRO_BW_12HZ
0x06	BNO055_GYRO_BW_64HZ
0x07	BNO055_GYRO_BW_32HZ

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.104 bno055_get_gyro_calib_stat()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_calib_stat (
            u8 * gyro_calib_u8 )
```

This API used to read gyro calibration status from register from 0x35 bit 4 and 5.

Parameters

gyro_calib_u8	: The value of gyro calib status

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.105 bno055_get_gyro_highrate_axis_enable()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_axis_enable (
            u8 channel_u8,
            u8 * data_u8 )
```

This API used to read the gyro highrate enable from page one register from 0x17 bit 3 to 5.

Parameters

data_u8	: The value of gyro highrate enable

data_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

channel_u8	: The value of gyro highrate axis selection

channel_u8	value
BNO055_GYRO_HIGHRATE_X_AXIS	0
BNO055_GYRO_HIGHRATE_Y_AXIS	1
BNO055_GYRO_HIGHRATE_Z_AXIS	2

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.106 bno055_get_gyro_highrate_filter()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_filter (
            u8 * gyro_highrate_filter_u8 )
```

This API used to read gyro highrate filter from page one register from 0x17 bit 7.

Parameters

gyro_highrate_filter_u8	: The value of gyro highrate filter

gyro_highrate_filter_u8	result
0x00	BNO055_GYRO_FILTERED_CONFIG
0x01	BNO055_GYRO_UNFILTERED_↩ CONFIG

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.107 bno055_get_gyro_highrate_x_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_x_durn (
            u8 * gyro_highrate_x_durn_u8 )
```

This API used to read gyro highrate x duration from page one register from 0x19 bit 0 to 7.

Parameters

gyro_highrate_x_durn_u8	: The value of gyro highrate x duration

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro highrate duration calculate by using the formula

(1 + gyro_highrate_x_durn_u8)∗2.5ms

2.1.2.108 bno055_get_gyro_highrate_x_hyst()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_x_hyst (
            u8 * gyro_highrate_x_hyst_u8 )
```

This API used to read gyro highrate x hysteresis from page one register from 0x18 bit 5 to 6.

Parameters

gyro_highrate_x_hyst_u8	: The value of gyro highrate x hysteresis

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro high rate hysteresis calculated by

using this (255 + 256 ∗ gyro_highrate_x_hyst_u8) ∗4 LSB

The high rate value scales with the range setting

gyro_range_u8	hysteresis	LSB
2000	62.26dps	1LSB
1000	31.13dps	1LSB
500	15.56dps	1LSB

2.1.2.109 bno055_get_gyro_highrate_x_thres()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_x_thres (
            u8 * gyro_highrate_x_thres_u8 )
```

This API used to read gyro highrate x threshold from page one register from 0x18 bit 0 to 4.

Parameters

gyro_highrate_x_thres_u8	: The value of gyro x highrate threshold

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro highrate threshold dependent on the selection of gyro range

gyro_range_u8	threshold	LSB
2000	62.5dps	1LSB
1000	31.25dps	1LSB
500	15.625dps	1LSB
125	7.8125dps	1LSB

2.1.2.110 bno055_get_gyro_highrate_y_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_y_durn (
            u8 * gyro_highrate_y_durn_u8 )
```

This API used to read gyro highrate y duration from page one register from 0x1B bit 0 to 7.

Parameters

gyro_highrate_y_durn_u8	: The value of gyro highrate y duration

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro highrate duration calculate by using the formula

(1 + gyro_highrate_y_durn_u8)∗2.5ms

2.1.2.111 bno055_get_gyro_highrate_y_hyst()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_y_hyst (
            u8 * gyro_highrate_y_hyst_u8 )
```

This API used to read gyro highrate y hysteresis from page one register from 0x1A bit 5 to 6.

Parameters

gyro_highrate_y_hyst_u8	: The value of gyro highrate y hysteresis

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro high rate hysteresis calculated by

using this (255 + 256 ∗ gyro_highrate_y_hyst_u8) ∗4 LSB

The high rate value scales with the range setting

gyro_range_u8	hysteresis	LSB
2000	62.26dps	1LSB
1000	31.13dps	1LSB
500	15.56dps	1LSB

2.1.2.112 bno055_get_gyro_highrate_y_thres()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_y_thres (
            u8 * gyro_highrate_y_thres_u8 )
```

This API used to read gyro highrate y threshold from page one register from 0x1A bit 0 to 4.

Parameters

gyro_highrate_y_thres_u8	: The value of gyro highrate y threshold

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro highrate threshold dependent on the selection of gyro range

gyro_range_u8	threshold	LSB
2000	62.5dps	1LSB
1000	31.25dps	1LSB
500	15.625dps	1LSB
125	7.8125dps	1LSB

2.1.2.113 bno055_get_gyro_highrate_z_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_z_durn (
            u8 * gyro_highrate_z_durn_u8 )
```

This API used to read gyro highrate z duration from page one register from 0x1D bit 0 to 7.

Parameters

gyro_highrate_z_durn_u8	: The value of gyro highrate z duration

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro highrate duration calculate by using the formula

(1 + gyro_highrate_z_durn_u8)∗2.5ms

2.1.2.114 bno055_get_gyro_highrate_z_hyst()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_z_hyst (
             u8 * gyro_highrate_z_hyst_u8 )
```

This API used to read gyro highrate z hysteresis from page one register from 0x1C bit 5 to 6.

Parameters

gyro_highrate_z_hyst_u8	: The value of gyro highrate z hysteresis

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro high rate hysteresis calculated by

using this (255 + 256 ∗ gyro_highrate_z_hyst_u8) ∗4 LSB

The high rate value scales with the range setting

gyro_range_u8	hysteresis	LSB
2000	62.26dps	1LSB
1000	31.13dps	1LSB
500	15.56dps	1LSB

2.1.2.115 bno055_get_gyro_highrate_z_thres()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_highrate_z_thres (
            u8 * gyro_highrate_z_thres_u8 )
```

This API used to read gyro highrate z threshold from page one register from 0x1C bit 0 to 4.

Parameters

gyro_highrate_z_thres_u8	: The value of gyro highrate z threshold

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro highrate threshold dependent on the selection of gyro range

gyro_range_u8	threshold	LSB
2000	62.5dps	1LSB
1000	31.25dps	1LSB
500	15.625dps	1LSB
125	7.8125dps	1LSB

2.1.2.116 bno055_get_gyro_power_mode()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_power_mode (
            u8 * gyro_power_mode_u8 )
```

This API used to read the gyro power mode from page one register from 0x0B bit 0 to 2.

Parameters

gyro_power_mode_u8	: The value of gyro power mode

gyro_power_mode_u8	result
0x00	GYRO_OPR_MODE_NORMAL
0x01	GYRO_OPR_MODE_FASTPOWERUP
0x02	GYRO_OPR_MODE_DEEPSUSPEND
0x03	GYRO_OPR_MODE_SUSPEND
0x04	GYRO_OPR_MODE_ADVANCE_POWERSAVE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.117 bno055_get_gyro_range()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_range (
            u8 * gyro_range_u8 )
```

This API used to read the gyro range from page one register from 0x0A bit 0 to 3.

Parameters

gyro_range_u8	: The value of gyro range

gyro_range_u8	result
0x00	BNO055_GYRO_RANGE_2000DPS
0x01	BNO055_GYRO_RANGE_1000DPS
0x02	BNO055_GYRO_RANGE_500DPS
0x03	BNO055_GYRO_RANGE_250DPS
0x04	BNO055_GYRO_RANGE_125DPS

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.118 bno055_get_gyro_sleep_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_sleep_durn (
```

```
            u8 * sleep_durn_u8 )
```

This API used to write the gyro sleep duration from page one register from 0x0D bit 0 to 2.

Parameters

sleep_durn_u8	: The value of gyro sleep duration

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.119 bno055_get_gyro_unit()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_gyro_unit (
            u8 * gyro_unit_u8 )
```

This API used to read the gyro unit from register from 0x3B bit 1.

Parameters

gyro_unit_u8	: The value of accel unit

gyro_unit_u8	result
0x00	BNO055_GYRO_UNIT_DPS
0x01	BNO055_GYRO_UNIT_RPS

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.120 bno055_get_intr_accel_any_motion()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_accel_any_motion (
            u8 * accel_any_motion_u8 )
```

This API used to read the accel anymotion interrupt from page one register from 0x10 bit 6.

Parameters

accel_any_motion_u8	: The value of accel anymotion interrupt

accel_any_motion_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While enabling the accel anymotion interrupt configure the following settings

Axis:

bno055_set_accel_any_motion_no_motion_axis_enable()

Duration:

bno055_set_accel_any_motion_durn()

Threshold:

bno055_set_accel_any_motion_thres()

2.1.2.121 bno055_get_intr_accel_high_g()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_accel_high_g (
            u8 * accel_high_g_u8 )
```

This API used to read the accel highg interrupt from page one register from 0x10 bit 5.

Parameters

accel_high_g_u8	: The value of accel highg interrupt

accel_high_g_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While enabling the accel highg interrupt configure the below settings by using the following APIs

Axis :

bno055_set_accel_high_g_axis_enable()

Threshold :

bno055_set_accel_high_g_thres()

Duration :

bno055_set_accel_high_g_durn()

2.1.2.122 bno055_get_intr_accel_no_motion()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_accel_no_motion (
            u8 * accel_nomotion_u8 )
```

This API used to read the accel nomotion interrupt from page one register from 0x10 bit 6.

Parameters

accel_nomotion_u8	: The value of accel nomotion interrupt

accel_nomotion_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While enabling the accel nomotion interrupt configure the following settings

Axis:

bno055_set_accel_any_motion_no_motion_axis_enable()

Threshold :

bno055_set_accel_slow_no_motion_thres()

Duration :

bno055_set_accel_slow_no_motion_durn()

Slow/no motion enable:

bno055_set_accel_slow_no_motion_enable()

2.1.2.123 bno055_get_intr_gyro_any_motion()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_gyro_any_motion (
            u8 * gyro_any_motion_u8 )
```

This API used to read the gyro anymotion interrupt from page one register from 0x10 bit 2.

Parameters

gyro_any_motion_u8	: The value of gyro anymotion interrupt

gyro_any_motion_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While enabling the gyro anymotion interrupt configure the following settings

Axis: bno055_set_gyro_any_motion_axis_enable()

Filter setting: bno055_set_gyro_any_motion_filter()

Threshold :

bno055_set_gyro_any_motion_thres()

Slope samples :

bno055_set_gyro_any_motion_slope_samples()

Awake duration :

bno055_set_gyro_any_motion_awake_durn()

2.1.2.124 bno055_get_intr_gyro_highrate()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_gyro_highrate (
            u8 * gyro_highrate_u8 )
```

This API used to read the gyro highrate interrupt from page one register from 0x10 bit 3.

Parameters

gyro_highrate_u8	: The value of gyro highrate interrupt

gyro_highrate_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While enabling the gyro highrate interrupt configure the below settings by using the following APIs

Axis :

bno055_set_gyro_highrate_axis_enable()

Filter :

bno055_set_gyro_highrate_filter()

Threshold :

bno055_get_gyro_highrate_x_thres()

bno055_get_gyro_highrate_y_thres()

bno055_get_gyro_highrate_z_thres()

Hysteresis :

bno055_set_gyro_highrate_x_hyst()

bno055_set_gyro_highrate_y_hyst()

bno055_set_gyro_highrate_z_hyst()

Duration :

bno055_set_gyro_highrate_x_durn()

bno055_set_gyro_highrate_y_durn()

bno055_set_gyro_highrate_z_durn()

2.1.2.125 bno055_get_intr_mask_accel_any_motion()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_mask_accel_any_motion (
            u8 * accel_any_motion_u8 )
```

This API used to read the accel anymotion interrupt mask from page one register from 0x0F bit 6.

Parameters

accel_any_motion_u8	: The value of accel anymotion interrupt mask

accel_any_motion_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While enabling the accel highg interrupt configure the below settings by using the following APIs

Axis :

bno055_set_accel_high_g_axis_enable()

Threshold :

bno055_set_accel_high_g_thres()

Duration :

bno055_set_accel_high_g_durn()

2.1.2.126 bno055_get_intr_mask_accel_high_g()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_mask_accel_high_g (
            u8 * accel_high_g_u8 )
```

This API used to read the accel highg interrupt mask from page one register from 0x0F bit 5.

Parameters

accel_high_g_u8	: The value of accel highg interrupt mask

accel_high_g_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While enabling the accel highg interrupt configure the below settings by using the following APIs

Axis :

bno055_set_accel_high_g_axis_enable()

Threshold :

bno055_set_accel_high_g_thres()

Duration :

bno055_set_accel_high_g_durn()

2.1.2.127 bno055_get_intr_mask_accel_no_motion()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_mask_accel_no_motion (
            u8 * accel_nomotion_u8 )
```

This API used to read the accel nomotion interrupt mask from page one register from 0x0F bit 7.

Parameters

accel_nomotion_u8	: The value of accel nomotion interrupt mask

accel_nomotion_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS

Note

While enabling the accel anymotion interrupt configure the following settings

Axis:

bno055_set_accel_any_motion_no_motion_axis_enable()

Duration:

bno055_set_accel_any_motion_durn()

Threshold:

bno055_set_accel_any_motion_thres())

2.1.2.128 bno055_get_intr_mask_gyro_any_motion()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_mask_gyro_any_motion (
            u8 * gyro_any_motion_u8 )
```

This API used to read the gyro anymotion interrupt mask from page one register from 0x0F bit 2.

Parameters

gyro_any_motion_u8	: The value of gyro anymotion interrupt mask

gyro_any_motion_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While enabling the gyro anymotion interrupt configure the following settings

Axis: bno055_set_gyro_any_motion_axis_enable()

Filter setting: bno055_set_gyro_any_motion_filter()

Threshold :

bno055_set_gyro_any_motion_thres()

Slope samples :

bno055_set_gyro_any_motion_slope_samples()

Awake duration :

bno055_set_gyro_any_motion_awake_durn()

2.1.2.129 bno055_get_intr_mask_gyro_highrate()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_mask_gyro_highrate (
            u8 * gyro_highrate_u8 )
```

This API used to read the gyro highrate interrupt mask from page one register from 0x0F bit 3.

Parameters

gyro_highrate_u8	: The value of gyro highrate interrupt mask

gyro_highrate_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While enabling the gyro highrate interrupt configure the below settings by using the following API

Axis :

bno055_set_gyro_highrate_axis_enable()

Filter :

bno055_set_gyro_highrate_filter()

Threshold :

bno055_get_gyro_highrate_x_thres()

bno055_get_gyro_highrate_y_thres()

bno055_get_gyro_highrate_z_thres()

Hysteresis :

bno055_set_gyro_highrate_x_hyst()

bno055_set_gyro_highrate_y_hyst()

bno055_set_gyro_highrate_z_hyst()

Duration :

bno055_set_gyro_highrate_x_durn()

bno055_set_gyro_highrate_y_durn()

bno055_set_gyro_highrate_z_durn()

2.1.2.130 bno055_get_intr_rst()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_rst (
            u8 * intr_rst_u8 )
```

This API used to read the reset interrupt from register from 0x3F bit 6 It resets all the interrupt bit and interrupt output.

Parameters

intr_rst_u8	: The value of reset interrupt

intr_rst_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.131 bno055_get_intr_stat_accel_any_motion()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_stat_accel_any_motion (
            u8 * accel_any_motion_u8 )
```

This API used to read the stat_s8 of accel anymotion interrupt from register from 0x37 bit 6.

Parameters

accel_any_motion_u8	: The value of accel anymotion interrupt

accel_any_motion_u8	result
0x00	indicates no interrupt triggered
0x01	indicates interrupt triggered

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Accel anymotion interrupt can be configured by the following APIs

bno055_set_intr_mask_accel_any_motion()

bno055_set_intr_accel_any_motion()

2.1.2.132 bno055_get_intr_stat_accel_high_g()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_stat_accel_high_g (
            u8 * accel_high_g_u8 )
```

This API used to read the stat_s8 of accel highg interrupt from register from 0x37 bit 5.

Parameters

accel_high_g_u8	: The value of accel highg interrupt

accel_high_g_u8	result
0x00	indicates no interrupt triggered
0x01	indicates interrupt triggered

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Accel highg interrupt can be configured by the following APIs

bno055_set_intr_mask_accel_high_g()

bno055_set_intr_accel_high_g()

2.1.2.133 bno055_get_intr_stat_accel_no_motion()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_stat_accel_no_motion (
            u8 * accel_no_motion_u8 )
```

This API used to read the stat_s8 of accel nomotion/slowmotion interrupt from register from 0x37 bit 6.

Parameters

accel_no_motion_u8	: The value of accel nomotion/slowmotion interrupt

accel_no_motion_u8	result
0x00	indicates no interrupt triggered
0x01	indicates interrupt triggered

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Accel nomotion/slowmotion interrupt can be configured by the following APIs

bno055_set_intr_mask_accel_nomotion()

bno055_set_intr_accel_nomotion()

2.1.2.134 bno055_get_intr_stat_gyro_any_motion()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_stat_gyro_any_motion (
            u8 * gyro_any_motion_u8 )
```

This API used to read the stat_s8 of gyro anymotion interrupt from register from 0x37 bit 2.

Parameters

gyro_any_motion_u8	: The value of gyro anymotion interrupt

gyro_any_motion_u8	result
0x00	indicates no interrupt triggered
0x01	indicates interrupt triggered

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro anymotion interrupt can be BNO055_BIT_ENABLE by the following APIs

bno055_set_intr_mask_gyro_any_motion()

bno055_set_intr_gyro_any_motion()

2.1.2.135 bno055_get_intr_stat_gyro_highrate()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_intr_stat_gyro_highrate (
            u8 * gyro_highrate_u8 )
```

This API used to read the stat_s8 of gyro highrate interrupt from register from 0x37 bit 3.

Parameters

gyro_highrate_u8	: The value of gyro highrate interrupt

gyro_highrate_u8	result
0x00	indicates no interrupt triggered
0x01	indicates interrupt triggered

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro highrate interrupt can be configured by the following APIs

bno055_set_intr_mask_gyro_highrate()

bno055_set_intr_gyro_highrate()

2.1.2.136 bno055_get_mag_calib_stat()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_mag_calib_stat (
            u8 * mag_calib_u8 )
```

This API used to read mag calibration status from register from 0x35 bit 0 and 1.

Parameters

mag_calib_u8	: The value of mag calib status

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.137 bno055_get_mag_data_output_rate()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_mag_data_output_rate (
            u8 * mag_data_output_rate_u8 )
```

This API used to read the mag output data rate from page one register from 0x09 bit 0 to 2.

Parameters

mag_data_output_rate_u8	: The value of mag output data rate

mag_data_output_rate_u8	result
0x00	MAG_DATA_OUTPUT_RATE_2HZ
0x01	MAG_DATA_OUTPUT_RATE_6HZ
0x02	MAG_DATA_OUTPUT_RATE_8HZ
0x03	MAG_DATA_OUTPUT_RATE_10HZ
0x04	MAG_DATA_OUTPUT_RATE_15HZ
0x05	MAG_DATA_OUTPUT_RATE_20HZ
0x06	MAG_DATA_OUTPUT_RATE_25HZ
0x07	MAG_DATA_OUTPUT_RATE_30HZ

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.138 bno055_get_mag_operation_mode()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_mag_operation_mode (
            u8 * mag_operation_mode_u8 )
```

This API used to read the mag operation mode from page one register from 0x09 bit 3 to 4.

Parameters

mag_operation_mode_u8	: The value of mag operation mode

mag_operation_mode_u8	result
0x00	MAG_OPR_MODE_LOWPOWER
0x01	MAG_OPR_MODE_REGULAR
0x02	MAG_OPR_MODE_ENHANCED_REGULAR
0x03	MAG_OPR_MODE_HIGH_ACCURACY

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.139 bno055_get_mag_power_mode()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_mag_power_mode (
            u8 * mag_power_mode_u8 )
```

This API used to read the mag power mode from page one register from 0x09 bit 4 to 6.

Parameters

mag_power_mode_u8	: The value of mag power mode

mag_power_mode_u8	result
0x00	BNO055_MAG_POWER_MODE_NORMAL
0x01	BNO055_MAG_POWER_MODE_SLEEP
0x02	BNO055_MAG_POWER_MODE_SUSPEND
0x03	BNO055_MAG_POWER_MODE_FORCE_MODE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.140 bno055_get_mag_sleep_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_mag_sleep_durn (
            u8 * sleep_durn_u8 )
```

This API used to read the mag sleep duration from page one register from 0x0E bit 1 to 4.

Parameters

sleep_durn_u8	: The value of mag sleep duration

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.141 bno055_get_mag_sleep_mode()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_mag_sleep_mode (
            u8 * sleep_mode_u8 )
```

This API used to read the mag sleep mode from page one register from 0x0E bit 0.

Parameters

sleep_mode_u8	: The value of mag sleep mode

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.142 bno055_get_operation_mode()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_operation_mode (
            u8 * operation_mode_u8 )
```

This API used to read the operation mode from register from 0x3D bit 0 to 3.

Parameters

operation_mode_u8	: The value of operation mode

operation_mode_u8	result	comments
0x00	BNO055_OPERATION_MODE_CONFIG	Configuration mode

operation_mode_u8	result	comments
0x01	BNO055_OPERATION_MODE_ACCONLY	Reads accel data alone
0x02	BNO055_OPERATION_MODE_MAGONLY	Reads mag data alone
0x03	BNO055_OPERATION_MODE_GYRONLY	Reads gyro data alone
0x04	BNO055_OPERATION_MODE_ACCMAG	Reads accel and mag data
0x05	BNO055_OPERATION_MODE_ACCGYRO	Reads accel and gyro data
0x06	BNO055_OPERATION_MODE_MAGGYRO	Reads accel and mag data
0x07	OPERATION_MODE_ANY_MOTION	Reads accel mag and gyro data
0x08	BNO055_OPERATION_MODE_IMUPLUS	Inertial measurement unit
-	-	Reads accel,gyro and fusion data
0x09	BNO055_OPERATION_MODE_COMPASS	Reads accel, mag data
-	-	and fusion data
0x0A	BNO055_OPERATION_MODE_M4G	Reads accel, mag data
-	-	and fusion data
0x0B	BNO055_OPERATION_MODE_NDOF_FMC_OFF	Nine degrees of freedom with
-	-	fast magnetic calibration
-	-	Reads accel,mag, gyro
-	-	and fusion data
0x0C	BNO055_OPERATION_MODE_NDOF	Nine degrees of freedom
-	-	Reads accel,mag, gyro
-	-	and fusion data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

In the config mode, all sensor and fusion data becomes zero and it is mainly derived to configure the various settings of the BNO

2.1.2.143 bno055_get_power_mode()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_power_mode (
            u8 * power_mode_u8 )
```

This API used to read the power mode from register from 0x3E bit 0 to 1.

Parameters

power_mode_u8	: The value of power mode

power_mode_u8	result	comments
0x00	BNO055_POWER_MODE_NORMAL	In the NORMAL mode the register
-	-	map and the internal peripherals
-	-	of the MCU are always
-	-	operative in this mode
0x01	BNO055_POWER_MODE_LOWPOWER	This is first level of power

| - | saving mode 0x02 |BNO055_POWER_MODE_SUSPEND | In suspend mode the system is

- | - | paused and all the sensors and
- | - | the micro controller are
- | - | put into sleep mode.

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

For detailed about LOWPOWER mode refer data sheet 3.4.2

2.1.2.144 bno055_get_remap_x_sign()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_remap_x_sign (
              u8 * remap_x_sign_u8 )
```

This API used to read the x-axis remap sign from register from 0x42 bit 2.

Parameters

remap_x_sign_u8	: The value of x-axis remap sign

remap_x_sign_u8	result
0X00	BNO055_REMAP_AXIS_POSITIVE
0X01	BNO055_REMAP_AXIS_NEGATIVE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.145 bno055_get_remap_y_sign()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_remap_y_sign (
            u8 * remap_y_sign_u8 )
```

This API used to read the y-axis remap sign from register from 0x42 bit 1.

Parameters

remap_y_sign_u8	: The value of y-axis remap sign

remap_y_sign_u8	result
0X00	BNO055_REMAP_AXIS_POSITIVE
0X01	BNO055_REMAP_AXIS_NEGATIVE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.146 bno055_get_remap_z_sign()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_remap_z_sign (
            u8 * remap_z_sign_u8 )
```

This API used to read the z-axis remap sign from register from 0x42 bit 0.

Parameters

remap_z_sign_u8	: The value of z-axis remap sign

remap_z_sign_u8	result
0X00	BNO055_REMAP_AXIS_POSITIVE
0X01	BNO055_REMAP_AXIS_NEGATIVE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.147 bno055_get_selftest()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_selftest (
            u8 * selftest_u8 )
```

This API used to read the self test from register from 0x3F bit 0.

Parameters

selftest_u8	: The value of self test

selftest_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

It triggers the self test

2.1.2.148 bno055_get_selftest_accel()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_selftest_accel (
            u8 * selftest_accel_u8 )
```

This API used to read self test of accel from register from 0x36 bit 0.

Parameters

selftest_accel_u8	: The value of self test of accel

selftest_accel_u8	result
0x00	indicates test failed
0x01	indicated test passed

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.149 bno055_get_selftest_gyro()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_selftest_gyro (
            u8 * selftest_gyro_u8 )
```

This API used to read self test of gyro from register from 0x36 bit 2.

Parameters

selftest_gyro_u8	: The value of self test of gyro

selftest_gyro_u8	result
0x00	indicates test failed
0x01	indicated test passed

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.150 bno055_get_selftest_mag()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_selftest_mag (
            u8 * selftest_mag_u8 )
```

This API used to read self test of mag from register from 0x36 bit 1.

Parameters

selftest_mag_u8	: The value of self test of mag

selftest_mag_u8	result
0x00	indicates test failed
0x01	indicated test passed

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.151 bno055_get_selftest_mcu()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_selftest_mcu (
              u8 * selftest_mcu_u8 )
```

This API used to read self test of micro controller from register from 0x36 bit 3.

Parameters

selftest_mcu_u8	: The value of self test of micro controller

selftest_mcu_u8	result
0x00	indicates test failed
0x01	indicated test passed

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.152 bno055_get_stat_main_clk()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_stat_main_clk (
              u8 * stat_main_clk_u8 )
```

This API is used to read status of main clock from the register 0x38 bit 0.

Parameters

stat_main_clk_u8	: the status of main clock

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.153 bno055_get_sys_calib_stat()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_sys_calib_stat (
            u8 * sys_calib_u8 )
```

This API used to read system calibration status from register from 0x35 bit 6 and 7.

Parameters

sys_calib_u8	: The value of system calib status

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.154 bno055_get_sys_error_code()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_sys_error_code (
            u8 * sys_error_u8 )
```

This API is used to read system BNO055_ERROR code from the register 0x3A it is a byte of data.

Parameters

sys_error_u8	: The value of system BNO055_ERROR code

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.155 bno055_get_sys_rst()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_sys_rst (
            u8 * sys_rst_u8 )
```

This API used to read the reset system from register from 0x3F bit 5.

Parameters

sys_rst_u8	: The value of reset system

sys_rst_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

It resets the whole system

2.1.2.156 bno055_get_sys_stat_code()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_sys_stat_code (
            u8 * sys_stat_u8 )
```

This API is used to read system status code from the register 0x39 it is a byte of data.

Parameters

sys_stat_u8	: the status of system

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.157 bno055_get_temp_source()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_temp_source (
            u8 * temp_source_u8 )
```

This API used to read the temperature source from register from 0x40 bit 0 and 1.

Parameters

temp_source_u8	: The value of selected temperature source

temp_source_u8	result
0x00	BNO055_ACCEL_TEMP_EN
0X01	BNO055_GYRO_TEMP_EN
0X03	BNO055_MCU_TEMP_EN

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.158 bno055_get_temp_unit()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_temp_unit (
            u8 * temp_unit_u8 )
```

This API used to read the temperature unit from register from 0x3B bit 4.

Parameters

temp_unit_u8	: The value of temperature unit

temp_unit_u8	result
0x00	BNO055_TEMP_UNIT_CELSIUS

temp_unit_u8	result
0x01	BNO055_TEMP_UNIT_FAHRENHEIT

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.159 bno055_get_tilt_unit()

```
BNO055_RETURN_FUNCTION_TYPE bno055_get_tilt_unit (
            u8 * tilt_unit_u8 )
```

This API used to write the tilt unit from register from 0x3B bit 3.

Parameters

tilt_unit_u8	: The value of tilt unit

tilt_unit_u8	result
0x00	degrees
0x01	radians

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.160 bno055_gyro_set_auto_sleep_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_gyro_set_auto_sleep_durn (
            u8 auto_sleep_durn_u8,
            u8 bw )
```

This API used to write the gyro auto sleep duration from page one register from 0x0D bit 3 to 5.

Parameters

auto_sleep_durn_u8	: The value of gyro auto sleep duration
bw	: The value of gyro bandwidth

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.161 bno055_init()

```
BNO055_RETURN_FUNCTION_TYPE bno055_init (
            struct bno055_t * bno055 )
```

This API is used for initialize bus read, bus write function pointers,device address,accel revision id, gyro revision id mag revision id, software revision id, boot loader revision id and page id.

Parameters

bno055	- structure pointer

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While changing the parameter of the bno055_t consider the following point: Changing the reference value of the parameter will changes the local copy or local reference make sure your changes will not affect the reference value of the parameter (Better case don't change the reference value of the parameter)

2.1.2.162 bno055_read_accel_offset()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_accel_offset (
            struct bno055_accel_offset_t * accel_offset )
```

This API is used to read accel offset and accel radius offset form register 0x55 to 0x5A and radius form 0x67 and 0x68.

Parameters

accel_offset	: The value of accel offset and radius bno055_accel_offset_t | result ---------- | ---------- x | accel offset x y | accel offset y z | accel offset z r | accel offset r

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

The range of the accel offset varies based on the G-range of accel sensor.

accel G range	offset range
BNO055_ACCEL_RANGE_2G	+/-2000
BNO055_ACCEL_RANGE_4G	+/-4000
BNO055_ACCEL_RANGE_8G	+/-8000
BNO055_ACCEL_RANGE_16G	+/-16000

accel G range can be configured by using the bno055_set_accel_range() API

2.1.2.163 bno055_read_accel_rev_id()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_accel_rev_id (
            u8 * accel_rev_id_u8 )
```

This API reads accel revision id from register 0x01 it is a byte of value.

Parameters

accel_rev_id_u8	: The accel revision id 0xFB

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.164 bno055_read_accel_x()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_accel_x (
            s16 * accel_x_s16 )
```

This API reads acceleration data X values from register 0x08 and 0x09 it is a two byte data.

Parameters

accel_x_s16	: The X raw data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.165 bno055_read_accel_xyz()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_accel_xyz (
            struct bno055_accel_t * accel )
```

This API reads acceleration data xyz values from register 0x08 to 0x0D it is a six byte data.

Parameters

accel	: The value of accel xyz data

Parameter	result
x	The accel x data
y	The accel y data
z	The accel z data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.166 bno055_read_accel_y()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_accel_y (
            s16 * accel_y_s16 )
```

This API reads acceleration data Y values from register 0x0A and 0x0B it is a two byte data.

Parameters

accel_y_s16	: The Y raw data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.167 bno055_read_accel_z()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_accel_z (
            s16 * accel_z_s16 )
```

This API reads acceleration data z values from register 0x0C and 0x0D it is a two byte data.

Parameters

accel_z_s16	: The z raw data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.168 bno055_read_bl_rev_id()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_bl_rev_id (
            u8 * bl_rev_id_u8 )
```

This API used to read boot loader revision id from register 0x06 it is a byte of value.

Parameters

bl_rev_id_u8	: The boot loader revision id

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.169 bno055_read_chip_id()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_chip_id (
            u8 * chip_id_u8 )
```

This API reads chip id from register 0x00 it is a byte of data.

Parameters

chip_id_u8	: The chip id value 0xA0

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.170 bno055_read_euler_h()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_euler_h (
            s16 * euler_h_s16 )
```

This API reads gyro data z values from register 0x1A and 0x1B it is a two byte data.

Parameters

euler_h_s16	: The raw h data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.171 bno055_read_euler_hrp()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_euler_hrp (
            struct bno055_euler_t * euler )
```

This API reads Euler data hrp values from register 0x1A to 0x1F it is a six byte data.

Parameters

euler	: The Euler hrp data's

Parameter	result
h	The Euler h data
r	The Euler r data
p	The Euler p data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.172 bno055_read_euler_p()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_euler_p (
            s16 * euler_p_s16 )
```

This API reads Euler data p values from register 0x1E and 0x1F it is a two byte data.

Parameters

euler_p_s16	: The raw p data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.173 bno055_read_euler_r()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_euler_r (
            s16 * euler_r_s16 )
```

This API reads Euler data r values from register 0x1C and 0x1D it is a two byte data.

Parameters

euler_r_s16	: The raw r data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.174 bno055_read_gravity_x()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_gravity_x (
            s16 * gravity_x_s16 )
```

This API reads gravity data x values from register 0x2E and 0x2F it is a two byte data.

Parameters

gravity_x_s16	: The raw x data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.175 bno055_read_gravity_xyz()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_gravity_xyz (
            struct bno055_gravity_t * gravity )
```

This API reads gravity data xyz values from register 0x2E to 0x33 it is a six byte data.

Parameters

gravity	: The value of gravity xyz data's

Parameter	result
x	The gravity x data
y	The gravity y data
z	The gravity z data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.176 bno055_read_gravity_y()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_gravity_y (
            s16 * gravity_y_s16 )
```

This API reads gravity data y values from register 0x30 and 0x31 it is a two byte data.

Parameters

gravity_y_s16	: The raw y data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.177 bno055_read_gravity_z()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_gravity_z (
            s16 * gravity_z_s16 )
```

This API reads gravity data z values from register 0x32 and 0x33 it is a two byte data.

Parameters

gravity_z_s16	: The raw z data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.178 bno055_read_gyro_offset()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_gyro_offset (
            struct bno055_gyro_offset_t * gyro_offset )
```

This API is used to read gyro offset offset form register 0x61 to 0x66.

Parameters

gyro_offset	: The value of gyro offset bno055_gyro_offset_t | result ---------- | ---------- x | gyro offset x y | gyro offset y z | gyro offset z

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

The range of the gyro offset varies based on the range of gyro sensor

gyro G range	offset range
BNO055_GYRO_RANGE_2000DPS	+/-32000
BNO055_GYRO_RANGE_1000DPS	+/-16000
BNO055_GYRO_RANGE_500DPS	+/-8000
BNO055_GYRO_RANGE_250DPS	+/-4000
BNO055_GYRO_RANGE_125DPS	+/-2000

Gyro range can be configured by using the bno055_set_gyro_range() API

2.1.2.179 bno055_read_gyro_rev_id()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_gyro_rev_id (
            u8 * gyro_rev_id_u8 )
```

This API reads gyro revision id from register 0x03 it is a byte of value.

Parameters

gyro_rev_id_u8	: The gyro revision id 0xF0

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.180 bno055_read_gyro_x()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_gyro_x (
            s16 * gyro_x_s16 )
```

This API reads gyro data x values from register 0x14 and 0x15 it is a two byte data.

Parameters

gyro_x_s16	: The x raw data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.181 bno055_read_gyro_xyz()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_gyro_xyz (
            struct bno055_gyro_t * gyro )
```

This API reads gyro data xyz values from register 0x14 to 0x19 it is a six byte data.

Parameters

gyro	: The value of gyro xyz data's

Parameter	result
x	The gyro x data
y	The gyro y data
z	The gyro z data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.182 bno055_read_gyro_y()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_gyro_y (
            s16 * gyro_y_s16 )
```

This API reads gyro data y values from register 0x16 and 0x17 it is a two byte data.

Parameters

gyro_y_s16	: The y raw data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.183 bno055_read_gyro_z()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_gyro_z (
            s16 * gyro_z_s16 )
```

This API reads gyro data z values from register 0x18 and 0x19 it is a two byte data.

Parameters

gyro_z_s16	: The z raw data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.184 bno055_read_linear_accel_x()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_linear_accel_x (
            s16 * linear_accel_x_s16 )
```

This API reads Linear accel data x values from register 0x29 and 0x2A it is a two byte data.

Parameters

linear_accel_x_s16	: The raw x data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.185 bno055_read_linear_accel_xyz()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_linear_accel_xyz (
            struct bno055_linear_accel_t * linear_accel )
```

This API reads Linear accel data xyz values from register 0x28 to 0x2D it is a six byte data.

Parameters

linear_accel	: The value of linear accel xyz data's

Parameter	result
x	The linear accel x data
y	The linear accel y data
z	The linear accel z data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.186 bno055_read_linear_accel_y()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_linear_accel_y (
            s16 * linear_accel_y_s16 )
```

This API reads Linear accel data x values from register 0x2B and 0x2C it is a two byte data.

Parameters

linear_accel_y_s16	: The raw y data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.187 bno055_read_linear_accel_z()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_linear_accel_z (
            s16 * linear_accel_z_s16 )
```

This API reads Linear accel data x values from register 0x2C and 0x2D it is a two byte data.

Parameters

linear_accel_z_s16	: The raw z data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.188 bno055_read_mag_offset()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_mag_offset (
            struct bno055_mag_offset_t * mag_offset )
```

This API is used to read mag offset offset form register 0x69 to 0x6A.

Parameters

mag_offset	: The value of mag offset and radius bno055_mag_offset_t | result ---------- | ---------- x | mag offset x y | mag offset y z | mag offset z r | mag radius r

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

The range of the magnetometer offset is +/-6400 in LSB

2.1.2.189 bno055_read_mag_rev_id()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_mag_rev_id (
            u8 * mag_rev_id_u8 )
```

This API reads mag revision id from register 0x02 it is a byte of value.

Parameters

mag_rev_id_u8	: The mag revision id 0x32

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.190 bno055_read_mag_x()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_mag_x (
            s16 * mag_x_s16 )
```

This API reads mag data x values from register 0x0E and 0x0F it is a two byte data.

Parameters

mag_x_s16	: The x raw data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.191 bno055_read_mag_xyz()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_mag_xyz (
            struct bno055_mag_t * mag )
```

This API reads mag data xyz values from register 0x0E to 0x13 it is a six byte data.

Parameters

mag	: The mag xyz values

Parameter	result
x	The mag x data
y	The mag y data
z	The mag z data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.192 bno055_read_mag_y()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_mag_y (
            s16 * mag_y_s16 )
```

This API reads mag data y values from register 0x10 and 0x11 it is a two byte data.

Parameters

mag_y_s16	: The y raw data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.193 bno055_read_mag_z()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_mag_z (
            s16 * mag_z_s16 )
```

This API reads mag data z values from register 0x12 and 0x13 it is a two byte data.

Parameters

mag_z_s16	: The z raw data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.194 bno055_read_page_id()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_page_id (
            u8 * page_id_u8 )
```

This API reads page id from register 0x07 it is a byte of data.

Parameters

page_id_u8	: The value of page id

BNO055_PAGE_ZERO -> 0x00 BNO055_PAGE_ONE -> 0x01

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.195 bno055_read_quaternion_w()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_quaternion_w (
            s16 * quaternion_w_s16 )
```

This API reads quaternion data w values from register 0x20 and 0x21 it is a two byte data.

Parameters

quaternion_w_s16	: The raw w data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.196 bno055_read_quaternion_wxyz()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_quaternion_wxyz (
            struct bno055_quaternion_t * quaternion )
```

This API reads Quaternion data wxyz values from register 0x20 to 0x27 it is a six byte data.

Parameters

quaternion	: The value of quaternion wxyz data's

Parameter	result
w	The quaternion w data
x	The quaternion x data
y	The quaternion y data
z	The quaternion z data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.197 bno055_read_quaternion_x()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_quaternion_x (
            s16 * quaternion_x_s16 )
```

This API reads quaternion data x values from register 0x22 and 0x23 it is a two byte data.

Parameters

quaternion_x_s16	: The raw x data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.198 bno055_read_quaternion_y()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_quaternion_y (
            s16 * quaternion_y_s16 )
```

This API reads quaternion data y values from register 0x24 and 0x25 it is a two byte data.

Parameters

quaternion_y_s16	: The raw y data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.199 bno055_read_quaternion_z()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_quaternion_z (
            s16 * quaternion_z_s16 )
```

This API reads quaternion data z values from register 0x26 and 0x27 it is a two byte data.

Parameters

quaternion_z_s16	: The raw z data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.200 bno055_read_register()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_register (
            u8 addr_u8,
            u8 * data_u8,
            u8 len_u8 )
```

This API reads the data from the given register address.

Parameters

addr_u8	: Address of the register
data_u8	: address of the variable, read value will be kept
len_u8	: Length of the data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.201 bno055_read_sic_matrix()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_sic_matrix (
            struct bno055_sic_matrix_t * sic_matrix )
```

This API is used to read soft iron calibration matrix from the register 0x43 to 0x53 it is a 18 bytes of data.

Parameters

sic_matrix	: The value of soft iron calibration matrix

sic_matrix	result
sic_0	soft iron calibration matrix zero
sic_1	soft iron calibration matrix one
sic_2	soft iron calibration matrix two
sic_3	soft iron calibration matrix three
sic_4	soft iron calibration matrix four
sic_5	soft iron calibration matrix five
sic_6	soft iron calibration matrix six
sic_7	soft iron calibration matrix seven
sic_8	soft iron calibration matrix eight

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

: Each soft iron calibration matrix range from -32768 to +32767

2.1.2.202 bno055_read_sw_rev_id()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_sw_rev_id (
            u16 * sw_id_u8 )
```

This API reads software revision id from register 0x04 and 0x05 it is a two byte of data.

Parameters

sw_id_u8	: The SW revision id

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.203 bno055_read_temp_data()

```
BNO055_RETURN_FUNCTION_TYPE bno055_read_temp_data (
```

```
            s8 * temp_s8 )
```

This API reads temperature values from register 0x33 it is a byte data.

Parameters

temp_s8	: The raw temperature data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.204 bno055_set_accel_any_motion_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_any_motion_durn (
            u8 accel_any_motion_durn_u8 )
```

This API used to write the accel anymotion duration from page one register from 0x12 bit 0 to 1.

Parameters

accel_any_motion_durn_u8	: The value of accel anymotion duration

accel_any_motion_durn_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.205 bno055_set_accel_any_motion_no_motion_axis_enable()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_any_motion_no_motion_axis_enable (
            u8 channel_u8,
            u8 data_u8 )
```

This API used to write the accel anymotion enable from page one register from 0x12 bit 2 to 4.

Parameters

data_u8	: The value of accel anymotion enable **data_u8** / **result** 0x01 / BNO055_BIT_ENABLE 0x00 / BNO055_BIT_DISABLE
channel_u8	: The value of accel anymotion axis selection **channel_u8** / **value** BNO055_ACCEL_ANY_MOTION_NO_↩MOTION_X_AXIS / 0 BNO055_ACCEL_ANY_MOTION_NO_↩MOTION_Y_AXIS / 1 BNO055_ACCEL_ANY_MOTION_NO_↩MOTION_Y_AXIS / 2

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.206 bno055_set_accel_any_motion_thres()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_any_motion_thres (
            u8 accel_any_motion_thres_u8 )
```

This API used to write the accel any motion threshold from page one register from 0x11 bit 0 to 7.

Parameters

accel_any_motion_thres_u8	: The value of any motion threshold **accel_any_motion_thres_u8** / **result** 0x01 / BNO055_BIT_ENABLE 0x00 / BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Accel anymotion threshold dependent on the range values

accel_range_u8	threshold	LSB
2g	3.19mg	1LSB
4g	7.81mg	1LSB
8g	15.63mg	1LSB
16g	31.25mg	1LSB

2.1.2.207 bno055_set_accel_bw()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_bw (
            u8 accel_bw_u8 )
```

This API used to write the accel bandwidth from page one register from 0x08 bit 2 to 4.

Parameters

accel_bw_u8	: The value of accel bandwidth accel_bw_u8 | result ---------- | ---------- 0x00 | BNO055_ACCEL_BW_7_81HZ 0x01 | BNO055_ACCEL_BW_15_63HZ 0x02 | BNO055_ACCEL_BW_31_25HZ 0x03 | BNO055_ACCEL_BW_62_5HZ 0x04 | BNO055_ACCEL_BW_125HZ 0x05 | BNO055_ACCEL_BW_250HZ 0x06 | BNO055_ACCEL_BW_500HZ 0x07 | BNO055_ACCEL_BW_1000HZ

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.208 bno055_set_accel_high_g_axis_enable()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_high_g_axis_enable (
            u8 channel_u8,
            u8 data_u8 )
```

This API used to write the accel highg enable from page one register from 0x12 bit 5 to 7.

Parameters

data_u8	: The value of accel highg enable **data_u8** / **result** 0x01 / BNO055_BIT_ENABLE 0x00 / BNO055_BIT_DISABLE
channel_u8	: The value of accel highg axis selection **channel_u8** / **value** BNO055_ACCEL_HIGH_G_X_AXIS / 0 BNO055_ACCEL_HIGH_G_Y_AXIS / 1 BNO055_ACCEL_HIGH_G_Z_AXIS / 2

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.209 bno055_set_accel_high_g_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_high_g_durn (
            u8 accel_high_g_durn_u8 )
```

This API used to write the accel highg duration from page one register from 0x13 bit 0 to 7.

Parameters

accel_high_g_durn_u8	: The value of accel highg duration

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

The high-g interrupt trigger delay according to [highg duration + 1] $*$ 2 ms

in a range from 2 ms to 512 ms

2.1.2.210 bno055_set_accel_high_g_thres()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_high_g_thres (
            u8 accel_high_g_thres_u8 )
```

This API used to write the accel highg threshold from page one register from 0x14 bit 0 to 7.

Parameters

accel_high_g_thres_u8	: The value of accel highg threshold

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Accel highg interrupt threshold dependent for accel g range

accel_range_u8	threshold	LSB
2g	7.81mg	1LSB
4g	15.63mg	1LSB
8g	31.25mg	1LSB
16g	62.5mg	1LSB

2.1.2.211 bno055_set_accel_power_mode()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_power_mode (
            u8 accel_power_mode_u8 )
```

This API used to write the accel power mode from page one register from 0x08 bit 5 to 7.

Parameters

accel_power_mode_u8 : The value of accel power mode

accel_power_mode_u8	result
0x00	BNO055_ACCEL_NORMAL
0x01	BNO055_ACCEL_SUSPEND
0x02	BNO055_ACCEL_LOWPOWER_1
0x03	BNO055_ACCEL_STANDBY
0x04	BNO055_ACCEL_LOWPOWER_2
0x05	BNO055_ACCEL_DEEPSUSPEND

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.212 bno055_set_accel_range()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_range (
            u8 accel_range_u8 )
```

This API used to write the accel range from page one register from 0x08 bit 0 and 1.

Parameters

accel_range_u8	: The value of accel range accel_range_u8 | result ---------- | ---------- 0x00 | BNO055_ACCEL_RANGE_2G 0x01 | BNO055_ACCEL_RANGE_4G 0x02 | BNO055_ACCEL_RANGE_8G 0x03 | BNO055_ACCEL_RANGE_16G

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.213 bno055_set_accel_sleep_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_sleep_durn (
            u8 sleep_durn_u8 )
```

This API used to write the accel sleep duration from page one register from 0x0C bit 1 to 4.

Parameters

sleep_durn_u8	: The value of accel sleep duration

sleep_durn_u8	result
0x05	BNO055_ACCEL_SLEEP_DURN_0_5MS

sleep_durn_u8	result
0x06	BNO055_ACCEL_SLEEP_DURN_1MS
0x07	BNO055_ACCEL_SLEEP_DURN_2MS
0x08	BNO055_ACCEL_SLEEP_DURN_4MS
0x09	BNO055_ACCEL_SLEEP_DURN_6MS
0x0A	BNO055_ACCEL_SLEEP_DURN_10MS
0x0B	BNO055_ACCEL_SLEEP_DURN_25MS
0x0C	BNO055_ACCEL_SLEEP_DURN_50MS
0x0D	BNO055_ACCEL_SLEEP_DURN_100MS
0x0E	BNO055_ACCEL_SLEEP_DURN_500MS
0x0F	BNO055_ACCEL_SLEEP_DURN_1S

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.214 bno055_set_accel_sleep_tmr_mode()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_sleep_tmr_mode (
            u8 sleep_tmr_u8 )
```

This API used to write the accel sleep mode from page one register from 0x0C bit 0.

Parameters

sleep_tmr_u8	: The value of accel sleep mode

sleep_tmr_u8	result
0x00	enable EventDrivenSampling(EDT)
0x01	enable Equidistant sampling mode(EST)

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.215 bno055_set_accel_slow_no_motion_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_slow_no_motion_durn (
            u8 accel_slow_no_motion_durn_u8 )
```

This API used to write accel slownomotion duration from page one register from 0x16 bit 1 to 6.

Parameters

accel_slow_no_motion_durn_u8	: The value of accel slownomotion duration

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.216 bno055_set_accel_slow_no_motion_enable()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_slow_no_motion_enable (
            u8 accel_slow_no_motion_en_u8 )
```

This API used to write accel slownomotion enable from page one register from 0x16 bit 0.

Parameters

accel_slow_no_motion_en_u8	: The value of accel slownomotion enable

accel_slow_no_motion_en_u8	result
0x01	Slow motion
0x00	No motion

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.217 bno055_set_accel_slow_no_motion_thres()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_slow_no_motion_thres (
            u8 accel_slow_no_motion_thres_u8 )
```

This API used to write the accel slownomotion threshold from page one register from 0x15 bit 0 to 7.

Parameters

accel_slow_no_motion_thres_u8	: The value of accel slownomotion threshold

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Accel slow no motion interrupt threshold dependent for accel g range

accel_range_u8	threshold	LSB
2g	3.19mg	1LSB
4g	7.81mg	1LSB
8g	15.63mg	1LSB
16g	31.25mg	1LSB

2.1.2.218 bno055_set_accel_unit()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_accel_unit (
              u8 accel_unit_u8 )
```

This API used to write the accel unit from register from 0x3B bit 0.

Parameters

accel_unit_u8	: The value of accel unit

accel_unit_u8	result
0x00	BNO055_ACCEL_UNIT_MSQ
0x01	BNO055_ACCEL_UNIT_MG

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.219 bno055_set_axis_remap_value()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_axis_remap_value (
            u8 remap_axis_u8 )
```

This API used to write the axis remap value from register from 0x41 bit 0 and 5.

Parameters

remap_axis_u8	: The value of axis remapping

remap_axis_u8	result	comments
0X21	BNO055_REMAP_X_Y	Z=Z;X=Y;Y=X
0X18	BNO055_REMAP_Y_Z	X=X;Y=Z;Z=Y
0X06	BNO055_REMAP_Z_X	Y=Y;X=Z;Z=X
0X12	BNO055_REMAP_X_Y_Z_TYPE0	X=Z;Y=X;Z=Y
0X09	BNO055_REMAP_X_Y_Z_TYPE1	X=Y;Y=Z;Z=X
0X24	BNO055_DEFAULT_AXIS	X=X;Y=Y;Z=Z

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

: For axis sign remap refer the following APIs x-axis :

bno055_set_x_remap_sign()

y-axis :

bno055_set_y_remap_sign()

z-axis :

bno055_set_z_remap_sign()

2.1.2.220 bno055_set_clk_src()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_clk_src (
            u8 clk_src_u8 )
```

This API used to write the clk source from register from 0x3F bit 7.

Parameters

clk_src_u8	: The value of clk source

clk_src_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.221 bno055_set_data_output_format()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_data_output_format (
            u8 data_output_format_u8 )
```

This API used to write the current selected orientation mode from register from 0x3B bit 7.

Parameters

data_output_format_u8	: The value of data output format

data_output_format_u8	result
0x00	Windows
0x01	Android

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.222 bno055_set_euler_unit()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_euler_unit (
            u8 euler_unit_u8 )
```

This API used to write the Euler unit from register from 0x3B bit 2.

Parameters

euler_unit_u8	: The value of Euler unit

euler_unit_u8	result
0x00	BNO055_EULER_UNIT_DEG
0x01	BNO055_EULER_UNIT_RAD

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.223 bno055_set_gyro_any_motion_awake_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_any_motion_awake_durn (
            u8 gyro_awake_durn_u8 )
```

This API used to write gyro anymotion awake duration from page one register from 0x1F bit 2 to 3.

Parameters

gyro_awake_durn_u8	: The value of gyro anymotion awake duration

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.224 bno055_set_gyro_any_motion_axis_enable()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_any_motion_axis_enable (
            u8 channel_u8,
            u8 data_u8 )
```

This API used to write the gyro anymotion enable from page one register from 0x17 bit 0 to 2.

Parameters

data_u8	: The value of gyro anymotion enable **data_u8** / **result** 0x01 / BNO055_BIT_ENABLE 0x00 / BNO055_BIT_DISABLE
channel_u8	: The value of gyro anymotion axis selection **channel_u8** / **value** BNO055_GYRO_ANY_MOTIONX_AXIS / 0 BNO055_GYRO_ANY_MOTIONY_AXIS / 1 BNO055_GYRO_ANY_MOTIONZ_AXIS / 2

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.225 bno055_set_gyro_any_motion_filter()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_any_motion_filter (
            u8 gyro_any_motion_filter_u8 )
```

This API used to write gyro anymotion filter from page one register from 0x17 bit 6.

Parameters

gyro_any_motion_filter_u8	: The value of gyro anymotion filter **gyro_any_motion_filter_u8** / **result** 0x00 / BNO055_GYRO_FILTERED_↩ CONFIG 0x01 / BNO055_GYRO_UNFILTERED_↩ CONFIG

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.226 bno055_set_gyro_any_motion_slope_samples()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_any_motion_slope_samples (
            u8 gyro_any_motion_slope_samples_u8 )
```

This API used to write gyro anymotion slope samples from page one register from 0x1F bit 0 to 1.

Parameters

gyro_any_motion_slope_samples_u8	: The value of gyro anymotion slope samples

gyro_any_motion_slope_samples_u8	result
0	8 samples
1	16 samples
2	32 samples
3	64 samples

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.227 bno055_set_gyro_any_motion_thres()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_any_motion_thres (
            u8 gyro_any_motion_thres_u8 )
```

This API used to write gyro anymotion threshold from page one register from 0x1E bit 0 to 6.

Parameters

gyro_any_motion_thres_u8	: The value of gyro anymotion threshold

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro anymotion interrupt threshold dependent on the selection of gyro range

gyro_range_u8	threshold	LSB
2000	1dps	1LSB
1000	0.5dps	1LSB
500	0.25dps	1LSB

2.1.2.228 bno055_set_gyro_bw()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_bw (
            u8 gyro_bw_u8 )
```

This API used to write the gyro bandwidth from page one register from 0x0A bit 3 to 5.

Parameters

gyro_bw_u8	: The value of gyro bandwidth

gyro_bw_u8	result
0x00	BNO055_GYRO_BW_523HZ
0x01	BNO055_GYRO_BW_230HZ
0x02	BNO055_GYRO_BW_116HZ
0x03	BNO055_GYRO_BW_47HZ
0x04	BNO055_GYRO_BW_23HZ
0x05	BNO055_GYRO_BW_12HZ
0x06	BNO055_GYRO_BW_64HZ
0x07	BNO055_GYRO_BW_32HZ

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.229 bno055_set_gyro_highrate_axis_enable()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_axis_enable (
            u8 channel_u8,
            u8 data_u8 )
```

This API used to write the gyro highrate enable from page one register from 0x17 bit 3 to 5.

Parameters

data_u8	: The value of gyro highrate enable **data_u8** / **result** 0x01 / BNO055_BIT_ENABLE 0x00 / BNO055_BIT_DISABLE
channel_u8	: The value of gyro highrate axis selection **channel_u8** / **value** BNO055_GYRO_HIGHRATE_X_AXIS / 0 BNO055_GYRO_HIGHRATE_Y_AXIS / 1 BNO055_GYRO_HIGHRATE_Z_AXIS / 2

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.230 bno055_set_gyro_highrate_filter()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_filter (
            u8 gyro_highrate_filter_u8 )
```

This API used to write gyro highrate filter from page one register from 0x17 bit 7.

Parameters

gyro_highrate_filter_u8	: The value of gyro highrate filter **gyro_highrate_filter_u8** / **result** 0x00 / BNO055_GYRO_FILTERED_CONFIG 0x01 / BNO055_GYRO_UNFILTERED_↩ CONFIG

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.231 bno055_set_gyro_highrate_x_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_x_durn (
            u8 gyro_highrate_x_durn_u8 )
```

This API used to write gyro highrate x duration from page one register from 0x19 bit 0 to 7.

Parameters

gyro_highrate_x_durn_u8	: The value of gyro highrate x duration

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro highrate duration calculate by using the formula

(1 + gyro_highrate_x_durn_u8)∗2.5ms

2.1.2.232 bno055_set_gyro_highrate_x_hyst()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_x_hyst (
            u8 gyro_highrate_x_hyst_u8 )
```

This API used to write gyro highrate x hysteresis from page one register from 0x18 bit 5 to 6.

Parameters

gyro_highrate_x_hyst_u8	: The value of gyro highrate x hysteresis

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro high rate hysteresis calculated by

using this (255 + 256 ∗ gyro_highrate_x_hyst_u8) ∗4 LSB

The high rate value scales with the range setting

gyro_range_u8	hysteresis	LSB
2000	62.26dps	1LSB
1000	31.13dps	1LSB
500	15.56dps	1LSB

2.1.2.233 bno055_set_gyro_highrate_x_thres()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_x_thres (
            u8 gyro_highrate_x_thres_u8 )
```

This API used to write gyro highrate x threshold from page one register from 0x18 bit 0 to 4.

Parameters

gyro_highrate_x_thres_u8	: The value of gyro x highrate threshold

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro highrate threshold dependent on the selection of gyro range

gyro_range_u8	threshold	LSB
2000	62.5dps	1LSB
1000	31.25dps	1LSB
500	15.625dps	1LSB
125	7.8125dps	1LSB

2.1.2.234 bno055_set_gyro_highrate_y_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_y_durn (
            u8 gyro_highrate_y_durn_u8 )
```

This API used to write gyro highrate y duration from page one register from 0x1B bit 0 to 7.

Parameters

gyro_highrate_y_durn_u8	: The value of gyro highrate y duration

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro highrate duration calculate by using the formula

(1 + gyro_highrate_y_durn_u8)∗2.5ms

2.1.2.235 bno055_set_gyro_highrate_y_hyst()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_y_hyst (
              u8 gyro_highrate_y_hyst_u8 )
```

This API used to write gyro highrate y hysteresis from page one register from 0x1A bit 5 to 6.

Parameters

gyro_highrate_y_hyst_u8	: The value of gyro highrate y hysteresis

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro high rate hysteresis calculated by

using this (255 + 256 ∗ gyro_highrate_y_hyst_u8) ∗4 LSB

The high rate value scales with the range setting

gyro_range_u8	hysteresis	LSB
2000	62.26dps	1LSB
1000	31.13dps	1LSB
500	15.56dps	1LSB

2.1.2.236 bno055_set_gyro_highrate_y_thres()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_y_thres (
            u8 gyro_highrate_y_thres_u8 )
```

This API used to write gyro highrate y threshold from page one register from 0x1A bit 0 to 4.

Parameters

gyro_highrate_y_thres_u8	: The value of gyro highrate y threshold

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro highrate threshold dependent on the selection of gyro range

gyro_range_u8	threshold	LSB
2000	62.5dps	1LSB
1000	31.25dps	1LSB
500	15.625dps	1LSB
125	7.8125dps	1LSB

2.1.2.237 bno055_set_gyro_highrate_z_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_z_durn (
            u8 gyro_highrate_z_durn_u8 )
```

This API used to write gyro highrate z duration from page one register from 0x1D bit 0 to 7.

Parameters

gyro_highrate_z_durn_u8	: The value of gyro highrate z duration

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro highrate duration calculate by using the formula

(1 + gyro_highrate_z_durn_u8)∗2.5ms

2.1.2.238 bno055_set_gyro_highrate_z_hyst()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_z_hyst (
            u8 gyro_highrate_z_hyst_u8 )
```

This API used to write gyro highrate z hysteresis from page one register from 0x1C bit 5 to 6.

Parameters

gyro_highrate_z_hyst_u8	: The value of gyro highrate z hysteresis

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro high rate hysteresis calculated by

using this (255 + 256 ∗ gyro_highrate_z_hyst_u8) ∗4 LSB

The high rate value scales with the range setting

gyro_range_u8	hysteresis	LSB
2000	62.26dps	1LSB
1000	31.13dps	1LSB
500	15.56dps	1LSB

2.1.2.239 bno055_set_gyro_highrate_z_thres()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_highrate_z_thres (
            u8 gyro_highrate_z_thres_u8 )
```

This API used to write gyro highrate z threshold from page one register from 0x1C bit 0 to 4.

Parameters

gyro_highrate_z_thres_u8	: The value of gyro highrate z threshold

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

Gyro highrate threshold dependent on the selection of gyro range

gyro_range_u8	threshold	LSB
2000	62.5dps	1LSB
1000	31.25dps	1LSB
500	15.625dps	1LSB
125	7.8125dps	1LSB

2.1.2.240 bno055_set_gyro_power_mode()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_power_mode (
            u8 gyro_power_mode_u8 )
```

This API used to write the gyro power mode from page one register from 0x0B bit 0 to 2.

Parameters

gyro_power_mode_u8	: The value of gyro power mode

gyro_power_mode_u8	result
0x00	GYRO_OPR_MODE_NORMAL
0x01	GYRO_OPR_MODE_FASTPOWERUP
0x02	GYRO_OPR_MODE_DEEPSUSPEND
0x03	GYRO_OPR_MODE_SUSPEND
0x04	GYRO_OPR_MODE_ADVANCE_POWERSAVE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.241 bno055_set_gyro_range()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_range (
            u8 gyro_range_u8 )
```

This API used to write the gyro range from page one register from 0x0A bit 0 to 3.

Parameters

gyro_range_u8	: The value of gyro range

gyro_range_u8	result
0x00	BNO055_GYRO_RANGE_2000DPS
0x01	BNO055_GYRO_RANGE_1000DPS
0x02	BNO055_GYRO_RANGE_500DPS
0x03	BNO055_GYRO_RANGE_250DPS
0x04	BNO055_GYRO_RANGE_125DPS

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.242 bno055_set_gyro_sleep_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_sleep_durn (
            u8 sleep_durn_u8 )
```

This API used to write the gyro sleep duration from page one register from 0x0D bit 0 to 2.

Parameters

sleep_durn_u8	: The value of gyro sleep duration

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.243 bno055_set_gyro_unit()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_gyro_unit (
            u8 gyro_unit_u8 )
```

This API used to write the gyro unit from register from 0x3B bit 1.

Parameters

gyro_unit_u8	: The value of accel unit

gyro_unit_u8	result
0x00	BNO055_GYRO_UNIT_DPS
0x01	BNO055_GYRO_UNIT_RPS

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.244 bno055_set_intr_accel_any_motion()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_accel_any_motion (
            u8 accel_any_motion_u8 )
```

This API used to write the accel anymotion interrupt from page one register from 0x10 bit 6.

Parameters

accel_any_motion_u8	: The value of accel anymotion interrupt

accel_any_motion_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While enabling the accel anymotion interrupt configure the following settings

Axis:

bno055_set_accel_any_motion_no_motion_axis_enable()

Duration:

bno055_set_accel_any_motion_durn()

Threshold:

bno055_set_accel_any_motion_thres()

2.1.2.245 bno055_set_intr_accel_high_g()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_accel_high_g (
            u8 accel_high_g_u8 )
```

This API used to write the accel highg interrupt from page one register from 0x10 bit 5.

Parameters

accel_high_g_u8	: The value of accel highg interrupt

accel_high_g_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While enabling the accel highg interrupt configure the below settings by using the following APIs

Axis :

bno055_set_accel_high_g_axis_enable()

Threshold :

bno055_set_accel_high_g_thres()

Duration :

bno055_set_accel_high_g_durn()

2.1.2.246 bno055_set_intr_accel_no_motion()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_accel_no_motion (
            u8 accel_nomotion_u8 )
```

This API used to write the accel nomotion interrupt from page one register from 0x10 bit 6.

Parameters

accel_nomotion_u8	: The value of accel nomotion interrupt

accel_nomotion_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While enabling the accel nomotion interrupt configure the following settings

Axis:

bno055_set_accel_any_motion_no_motion_axis_enable()

Threshold :

bno055_set_accel_slow_no_motion_thres()

Duration :

bno055_set_accel_slow_no_motion_durn()

Slow/no motion enable:

bno055_set_accel_slow_no_motion_enable()

2.1.2.247 bno055_set_intr_gyro_any_motion()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_gyro_any_motion (
            u8 gyro_any_motion_u8 )
```

This API used to write the gyro anymotion interrupt from page one register from 0x10 bit 2.

Parameters

gyro_any_motion_u8	: The value of gyro anymotion interrupt

gyro_any_motion_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While enabling the gyro anymotion interrupt configure the following settings

Axis: bno055_set_gyro_any_motion_axis_enable()

Filter setting: bno055_set_gyro_any_motion_filter()

Threshold :

bno055_set_gyro_any_motion_thres()

Slope samples :

bno055_set_gyro_any_motion_slope_samples()

Awake duration :

bno055_set_gyro_any_motion_awake_durn()

2.1.2.248 bno055_set_intr_gyro_highrate()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_gyro_highrate (
            u8 gyro_highrate_u8 )
```

This API used to write the gyro highrate interrupt from page one register from 0x10 bit 3.

Parameters

gyro_highrate_u8	: The value of gyro highrate interrupt

gyro_highrate_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While enabling the gyro highrate interrupt configure the below settings by using the following APIs

Axis :

bno055_set_gyro_highrate_axis_enable()

Filter :

bno055_set_gyro_highrate_filter()

Threshold :

bno055_get_gyro_highrate_x_thres()

bno055_get_gyro_highrate_y_thres()

bno055_get_gyro_highrate_z_thres()

Hysteresis :

bno055_set_gyro_highrate_x_hyst()

bno055_set_gyro_highrate_y_hyst()

bno055_set_gyro_highrate_z_hyst()

Duration :

bno055_set_gyro_highrate_x_durn()

bno055_set_gyro_highrate_y_durn()

bno055_set_gyro_highrate_z_durn()

2.1.2.249 bno055_set_intr_mask_accel_any_motion()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_mask_accel_any_motion (
            u8 accel_any_motion_u8 )
```

This API used to write the accel anymotion interrupt mask from page one register from 0x0F bit 6.

Parameters

accel_any_motion_u8	: The value of accel anymotion interrupt mask

accel_any_motion_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While enabling the accel anymotion interrupt configure the following settings

Axis:

bno055_set_accel_any_motion_no_motion_axis_enable()

Duration:

bno055_set_accel_any_motion_durn()

Threshold:

bno055_set_accel_any_motion_thres()

2.1.2.250 bno055_set_intr_mask_accel_high_g()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_mask_accel_high_g (
            u8 accel_high_g_u8 )
```

This API used to write the accel highg interrupt mask from page one register from 0x0F bit 5.

Parameters

accel_high_g_u8	: The value of accel highg interrupt mask

accel_high_g_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While enabling the accel highg interrupt configure the below settings by using the following APIs

Axis :

bno055_set_accel_high_g_axis_enable()

Threshold :

bno055_set_accel_high_g_thres()

Duration :

bno055_set_accel_high_g_durn()

2.1.2.251 bno055_set_intr_mask_accel_no_motion()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_mask_accel_no_motion (
            u8 accel_nomotion_u8 )
```

This API used to write the accel nomotion interrupt mask from page one register from 0x0F bit 7.

Parameters

accel_nomotion_u8	: The value of accel nomotion interrupt mask

accel_nomotion_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While enabling the accel nomotion interrupt configure the following settings

Axis:

bno055_set_accel_any_motion_no_motion_axis_enable()

Threshold :

bno055_set_accel_slow_no_motion_thres()

Duration :

bno055_set_accel_slow_no_motion_durn()

Slow/no motion enable:

bno055_set_accel_slow_no_motion_enable()

2.1.2.252 bno055_set_intr_mask_gyro_any_motion()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_mask_gyro_any_motion (
            u8 gyro_any_motion_u8 )
```

This API used to write the gyro anymotion interrupt mask from page one register from 0x0F bit 2.

Parameters

gyro_any_motion_u8	: The value of gyro anymotion interrupt mask

gyro_any_motion_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While enabling the gyro anymotion interrupt configure the following settings

Axis: bno055_set_gyro_any_motion_axis_enable()

Filter setting: bno055_set_gyro_any_motion_filter()

Threshold :

bno055_set_gyro_any_motion_thres()

Slope samples :

bno055_set_gyro_any_motion_slope_samples()

Awake duration :

bno055_set_gyro_any_motion_awake_durn()

2.1.2.253 bno055_set_intr_mask_gyro_highrate()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_mask_gyro_highrate (
            u8 gyro_highrate_u8 )
```

This API used to write the gyro highrate interrupt mask from page one register from 0x0F bit 3.

Parameters

gyro_highrate_u8	: The value of gyro highrate interrupt mask

gyro_highrate_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

While enabling the gyro highrate interrupt configure the below settings by using the following APIs

Axis :

bno055_set_gyro_highrate_axis_enable()

Filter :

bno055_set_gyro_highrate_filter()

Threshold :

bno055_get_gyro_highrate_x_thres()

bno055_get_gyro_highrate_y_thres()

bno055_get_gyro_highrate_z_thres()

Hysteresis :

bno055_set_gyro_highrate_x_hyst()

bno055_set_gyro_highrate_y_hyst()

bno055_set_gyro_highrate_z_hyst()

Duration :

bno055_set_gyro_highrate_x_durn()

bno055_set_gyro_highrate_y_durn()

bno055_set_gyro_highrate_z_durn()

2.1.2.254 bno055_set_intr_rst()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_intr_rst (
            u8 intr_rst_u8 )
```

This API used to write the reset interrupt from register from 0x3F bit 6 It resets all the interrupt bit and interrupt output.

Parameters

intr_rst_u8	: The value of reset interrupt

intr_rst_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.255 bno055_set_mag_data_output_rate()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_mag_data_output_rate (
            u8 mag_data_output_rate_u8 )
```

This API used to write the mag output data rate from page one register from 0x09 bit 0 to 2.

Parameters

mag_data_output_rate_u8	: The value of mag output data rate

mag_data_output_rate_u8	result
0x00	MAG_DATA_OUTPUT_RATE_2HZ
0x01	MAG_DATA_OUTPUT_RATE_6HZ
0x02	MAG_DATA_OUTPUT_RATE_8HZ
0x03	MAG_DATA_OUTPUT_RATE_10HZ
0x04	MAG_DATA_OUTPUT_RATE_15HZ
0x05	MAG_DATA_OUTPUT_RATE_20HZ
0x06	MAG_DATA_OUTPUT_RATE_25HZ
0x07	MAG_DATA_OUTPUT_RATE_30HZ

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.256 bno055_set_mag_operation_mode()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_mag_operation_mode (
            u8 mag_operation_mode_u8 )
```

This API used to write the mag operation mode from page one register from 0x09 bit 3 to 4.

Parameters

mag_operation_mode_u8	: The value of mag operation mode

mag_operation_mode_u8	result
0x00	MAG_OPR_MODE_LOWPOWER
0x01	MAG_OPR_MODE_REGULAR
0x02	MAG_OPR_MODE_ENHANCED_REGULAR
0x03	MAG_OPR_MODE_HIGH_ACCURACY

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.257 bno055_set_mag_power_mode()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_mag_power_mode (
            u8 mag_power_mode_u8 )
```

This API used to write the mag power mode from page one register from 0x09 bit 4 to 6.

Parameters

mag_power_mode_u8	: The value of mag power mode

mag_power_mode_u8	result
0x00	BNO055_MAG_POWER_MODE_NORMAL
0x01	BNO055_MAG_POWER_MODE_SLEEP
0x02	BNO055_MAG_POWER_MODE_SUSPEND
0x03	BNO055_MAG_POWER_MODE_FORCE_MODE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.258 bno055_set_mag_sleep_durn()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_mag_sleep_durn (
            u8 sleep_durn_u8 )
```

This API used to write the mag sleep duration from page one register from 0x0E bit 1 to 4.

Parameters

sleep_durn_u8	: The value of mag sleep duration

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.259 bno055_set_mag_sleep_mode()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_mag_sleep_mode (
            u8 sleep_mode_u8 )
```

This API used to write the mag sleep mode from page one register from 0x0E bit 0.

Parameters

sleep_mode_u8	: The value of mag sleep mode

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.260 bno055_set_operation_mode()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_operation_mode (
            u8 operation_mode_u8 )
```

This API used to write the operation mode from register from 0x3D bit 0 to 3.

Parameters

operation_mode_u8	: The value of operation mode

operation_mode_u8	result	comments
0x00	BNO055_OPERATION_MODE_CONFIG	Configuration mode
0x01	BNO055_OPERATION_MODE_ACCONLY	Reads accel data alone
0x02	BNO055_OPERATION_MODE_MAGONLY	Reads mag data alone
0x03	BNO055_OPERATION_MODE_GYRONLY	Reads gyro data alone
0x04	BNO055_OPERATION_MODE_ACCMAG	Reads accel and mag data
0x05	BNO055_OPERATION_MODE_ACCGYRO	Reads accel and gyro data
0x06	BNO055_OPERATION_MODE_MAGGYRO	Reads accel and mag data
0x07	OPERATION_MODE_ANY_MOTION	Reads accel mag and

| - | gyro data 0x08 | BNO055_OPERATION_MODE_IMUPLUS | Inertial measurement unit

- | | Reads accel,gyro and | - | fusion data 0x09 | BNO055_OPERATION_MODE_COMPASS | Reads accel, mag data
- | - | and fusion data 0x0A | BNO055_OPERATION_MODE_M4G | Reads accel, mag data
  - | - | and fusion data 0x0B | BNO055_OPERATION_MODE_NDOF_FMC_OFF| Nine degrees of freedom with
- | - | fast magnetic calibration
- | - | Reads accel,mag, gyro
- | - | and fusion data 0x0C | BNO055_OPERATION_MODE_NDOF | Nine degrees of freedom
- | - | Reads accel,mag, gyro
- | - | and fusion data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

In the config mode, all sensor and fusion data becomes zero and it is mainly derived to configure the various settings of the BNO

2.1.2.261 bno055_set_power_mode()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_power_mode (
            u8 power_mode_u8 )
```

This API used to write the power mode from register from 0x3E bit 0 to 1.

Parameters

power_mode_u8	: The value of power mode

power_mode_u8	result	comments
0x00	BNO055_POWER_MODE_NORMAL	In the NORMAL mode the register
-	-	map and the internal peripherals
-	-	of the MCU are always
-	-	operative in this mode
0x01	BNO055_POWER_MODE_LOWPOWER	This is first level of power

| - | saving mode 0x02 | BNO055_POWER_MODE_SUSPEND | In suspend mode the system is

- | - | paused and all the sensors and
- | - | the micro controller are
- | - | put into sleep mode.

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

For detailed about LOWPOWER mode refer data sheet 3.4.2

2.1.2.262 bno055_set_remap_x_sign()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_remap_x_sign (
            u8 remap_x_sign_u8 )
```

This API used to write the x-axis remap sign from register from 0x42 bit 2.

Parameters

remap_x_sign_u8	: The value of x-axis remap sign

remap_x_sign_u8	result
0X00	BNO055_REMAP_AXIS_POSITIVE
0X01	BNO055_REMAP_AXIS_NEGATIVE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.263 bno055_set_remap_y_sign()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_remap_y_sign (
            u8 remap_y_sign_u8 )
```

This API used to write the y-axis remap sign from register from 0x42 bit 1.

Parameters

remap_y_sign_u8	: The value of y-axis remap sign

remap_y_sign_u8	result
0X00	BNO055_REMAP_AXIS_POSITIVE
0X01	BNO055_REMAP_AXIS_NEGATIVE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.264 bno055_set_remap_z_sign()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_remap_z_sign (
            u8 remap_z_sign_u8 )
```

This API used to write the z-axis remap sign from register from 0x42 bit 0.

Parameters

remap_z_sign_u8	: The value of z-axis remap sign

remap_z_sign_u8	result
0X00	BNO055_REMAP_AXIS_POSITIVE
0X01	BNO055_REMAP_AXIS_NEGATIVE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.265 bno055_set_selftest()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_selftest (
            u8 selftest_u8 )
```

This API used to write the self test from register from 0x3F bit 0.

Parameters

selftest_u8	: The value of self test

selftest_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

It triggers the self test

2.1.2.266 bno055_set_sys_rst()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_sys_rst (
            u8 sys_rst_u8 )
```

This API used to write the reset system from register from 0x3F bit 5.

Parameters

sys_rst_u8	: The value of reset system

sys_rst_u8	result
0x01	BNO055_BIT_ENABLE
0x00	BNO055_BIT_DISABLE

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

It resets the whole system

2.1.2.267 bno055_set_temp_source()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_temp_source (
            u8 temp_source_u8 )
```

This API used to write the temperature source from register from 0x40 bit 0 and 1.

Parameters

temp_source_u8	: The value of selected temperature source

temp_source_u8	result
0x00	BNO055_ACCEL_TEMP_EN
0X01	BNO055_GYRO_TEMP_EN
0X03	BNO055_MCU_TEMP_EN

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.268 bno055_set_temp_unit()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_temp_unit (
            u8 temp_unit_u8 )
```

This API used to write the temperature unit from register from 0x3B bit 4.

Parameters

temp_unit_u8	: The value of temperature unit

temp_unit_u8	result
0x00	BNO055_TEMP_UNIT_CELSIUS
0x01	BNO055_TEMP_UNIT_FAHRENHEIT

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.269 bno055_set_tilt_unit()

```
BNO055_RETURN_FUNCTION_TYPE bno055_set_tilt_unit (
            u8 tilt_unit_u8 )
```

This API used to write the tilt unit from register from 0x3B bit 3.

Parameters

tilt_unit_u8	: The value of tilt unit

tilt_unit_u8	result
0x00	degrees
0x01	radians

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Returns

Communication results

2.1.2.270 bno055_write_accel_offset()

```
BNO055_RETURN_FUNCTION_TYPE bno055_write_accel_offset (
             struct bno055_accel_offset_t * accel_offset )
```

This API is used to write accel offset and accel radius offset form register 0x55 to 0x5A and radius form 0x67 and 0x68.

Parameters

accel_offset	: The value of accel offset and radius bno055_accel_offset_t | result ---------- | ---------- x | accel offset x y | accel offset y z | accel offset z r | accel offset r

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

The range of the accel offset varies based on the G-range of accel sensor.

accel G range	offset range
BNO055_ACCEL_RANGE_2G	+/-2000
BNO055_ACCEL_RANGE_4G	+/-4000
BNO055_ACCEL_RANGE_8G	+/-8000
BNO055_ACCEL_RANGE_16G	+/-16000

accel G range can be configured by using the bno055_set_accel_range() API

2.1.2.271 bno055_write_gyro_offset()

```
BNO055_RETURN_FUNCTION_TYPE bno055_write_gyro_offset (
             struct bno055_gyro_offset_t * gyro_offset )
```

This API is used to read gyro offset offset form register 0x61 to 0x66.

Parameters

gyro_offset	: The value of gyro offset bno055_gyro_offset_t | result ---------- | ---------- x | gyro offset x y | gyro offset y z | gyro offset z

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

The range of the gyro offset varies based on the range of gyro sensor

gyro G range	offset range
BNO055_GYRO_RANGE_2000DPS	+/-32000
BNO055_GYRO_RANGE_1000DPS	+/-16000
BNO055_GYRO_RANGE_500DPS	+/-8000
BNO055_GYRO_RANGE_250DPS	+/-4000
BNO055_GYRO_RANGE_125DPS	+/-2000

Gyro range can be configured by using the bno055_set_gyro_range() API

2.1.2.272 bno055_write_mag_offset()

```
BNO055_RETURN_FUNCTION_TYPE bno055_write_mag_offset (
            struct bno055_mag_offset_t * mag_offset )
```

This API is used to read mag offset offset form register 0x69 to 0x6A.

Parameters

mag_offset	: The value of mag offset and radius bno055_mag_offset_t | result ---------- | ---------- x | mag offset x y | mag offset y z | mag offset z r | mag radius r

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

The range of the magnetometer offset is +/-6400 in LSB

2.1.2.273 bno055_write_page_id()

```
BNO055_RETURN_FUNCTION_TYPE bno055_write_page_id (
            u8 page_id_u8 )
```

This API used to write the page id register 0x07.

Parameters

page_id_u8	: The value of page id

BNO055_PAGE_ZERO -> 0x00 BNO055_PAGE_ONE -> 0x01

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.274 bno055_write_register()

```
BNO055_RETURN_FUNCTION_TYPE bno055_write_register (
            u8 addr_u8,
            u8 * data_u8,
            u8 len_u8 )
```

This API gives data to the given register and the data is written in the corresponding register address.

Parameters

addr_u8	: Address of the register
data_u8	: Data to be written to the register
len_u8	: Length of the Data

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

2.1.2.275 bno055_write_sic_matrix()

```
BNO055_RETURN_FUNCTION_TYPE bno055_write_sic_matrix (
            struct bno055_sic_matrix_t * sic_matrix )
```

This API is used to write soft iron calibration matrix from the register 0x43 to 0x53 it is a 18 bytes of data.

Parameters

sic_matrix	: The value of soft iron calibration matrix

sic_matrix	result
sic_0	soft iron calibration matrix zero
sic_1	soft iron calibration matrix one
sic_2	soft iron calibration matrix two
sic_3	soft iron calibration matrix three
sic_4	soft iron calibration matrix four
sic_5	soft iron calibration matrix five

sic_matrix	result
sic_6	soft iron calibration matrix six
sic_7	soft iron calibration matrix seven
sic_8	soft iron calibration matrix eight

Returns

results of bus communication function

Return values

0	-> BNO055_SUCCESS
1	-> BNO055_ERROR

Note

: Each soft iron calibration matrix range from -32768 to +32767

Index