

### Change the header file:

```
// I2C0 registers
#define I2C_MST_CTRL          0x01
#define I2C_MST_DELAY_CTRL 0x02
#define I2C_SLV0_ADDR        0x03
#define I2C_SLV0_REG         0x04
#define I2C_SLV0_CTRL        0x05
#define I2C_SLV0_DO          0x06
#define I2C_MST_ODR_CONFIG 0x00
#define LP_CONFIG            0x05

// Magnetometer registers
#define MAG_DEVICE_ID        0x01
#define MAG_STATUS1         0x10
#define MAG_DATA_ONSET      0x11
#define MAG_STATUS2         0x18
#define MAG_CTRL2           0x31
#define MAG_CTRL3           0x32

// Magnetometer address
#define AK09916_ADDRESS 0x0c
```

### New static functions:

```
static void ak09916_mag_init();
static void ak09916_read_reg(uint8_t onset_reg, uint8_t len);
static void ak09916_write_reg(uint8_t reg, uint8_t data);
```

```
static void ak09916_mag_init()
{
    uint8_t temp_data;
    // I2C master reset, page 36
    icm_20948_read_reg(_b0, USER_CTRL, &temp_data);
    temp_data |= 0x02;
    icm_20948_write_reg(_b0, USER_CTRL, temp_data);
    HAL_Delay(100);
    // I2C Master enable, page 36
    icm_20948_read_reg(_b0, USER_CTRL, &temp_data);
    temp_data |= 0x20;
    icm_20948_write_reg(_b0, USER_CTRL, temp_data);
    HAL_Delay(10);

    // I2C Master clock: 7 (400 kHz), page 68
    temp_data = 0x07;
    icm_20948_write_reg(_b3, I2C_MST_CTRL, temp_data);
    HAL_Delay(10);
}
```

```

// LP_CONFIG:ODR is determined by I2C_MST_ODR_CONFIG register,page 37
temp_data = 0x40;
icm_20948_write_reg(_b0, LP_CONFIG, temp_data);
HAL_Delay(10);

// I2C_MST_ODR_CONFIG: 1.1 kHz/(2^3) = 136 Hz, page 68
temp_data = 0x03;
icm_20948_write_reg(_b3, I2C_MST_ODR_CONFIG, temp_data);
HAL_Delay(10);

// I2C_MST_DELAY_CTRL: delays shadowing of external sensors, page 69
temp_data = 0x80;
icm_20948_write_reg(_b3, I2C_MST_DELAY_CTRL, temp_data);
HAL_Delay(10);

// Magnetometer reset, page, page 80
ak09916_write_reg(MAG_CTRL3, 0x01);
HAL_Delay(100);

// continuous mode 4: 100 Hz, page 79
ak09916_write_reg(MAG_CTRL2, 0x08);
}

static void ak09916_write_reg(uint8_t reg, uint8_t data)
{
    icm_20948_write_reg(_b3, I2C_SLV0_ADDR, AK09916_ADDRESS);
    icm_20948_write_reg(_b3, I2C_SLV0_REG , reg);
    icm_20948_write_reg(_b3, I2C_SLV0_DO , data);
    // Enable and single data write
    HAL_Delay(50);
    icm_20948_write_reg(_b3, I2C_SLV0_CTRL, 0x80|0x01);
    HAL_Delay(50);
}

static void ak09916_read_reg(uint8_t onset_reg, uint8_t len)
{
    icm_20948_write_reg(_b3, I2C_SLV0_ADDR, 0x80|AK09916_ADDRESS);
    icm_20948_write_reg(_b3, I2C_SLV0_REG , onset_reg);
    HAL_Delay(50);
    icm_20948_write_reg(_b3, I2C_SLV0_CTRL, 0x80|len);
    HAL_Delay(50);
}

```

### **Inside of the initialization function:**

```

ak09916_mag_init();
ak09916_read_reg(MAG_DATA_ONSET, 8);

```

### **Inside of icm\_20948\_read\_data function:**

1. Change the length of rx\_data to 22: static uint8\_t data\_rx[22];
2. Read 22 bytes: HAL\_SPI\_Receive(&IMU\_SPI, data\_rx, **22**, 1000);
3. Store the compass data:  
data ->x\_magnet = ((int16\_t)data\_rx[15]<<8) | (int16\_t) data\_rx[14];  
data ->y\_magnet = ((int16\_t)data\_rx[17]<<8) | (int16\_t)data\_rx[16];  
data ->z\_magnet = ((int16\_t)data\_rx[19]<<8) | (int16\_t)data\_rx[18];