

IMU

The Inertial Measurement Unit (IMU) provides three-axis acceleration, angular velocity, and magnetic field measurements for attitude determination and motion analysis.

Communication Interfaces

- **Accelerometer:** I2C or SPI
- **Gyroscope:** I2C or SPI
- **Magnetometer:** I2C or SPI
- **Update Rate:** Typically 1-100+ Hz (configurable)

Data Structure

```
typedef struct {
    // Acceleration (m/s2 or g-units)
    float accel[3];           // [X, Y, Z] acceleration

    // Angular Velocity (rad/s or °/s)
    float gyro[3];           // [X, Y, Z] rotation rate

    // Magnetic Field (Gauss or μT)
    float mag[3];           // [X, Y, Z] magnetic field

    // Timestamps
    uint32_t timestamp;      // Current reading timestamp
    uint32_t last_timestamp; // Previous reading timestamp
} imu_data_t;
```

Initialization & Usage

Initialize IMU

```
imu_data_t imu_data;
imu_sensor_init(&imu_data);
```

Poll IMU Data

```
result_t imu_result = poll_imu_sensor(&imu_data);

if (imu_result == RESULT_OK) {
    // Acceleration
    float accel_x = imu_data.accel[0];
    float accel_y = imu_data.accel[1];
    float accel_z = imu_data.accel[2];

    // Angular velocity
    float gyro_x = imu_data.gyro[0];
    float gyro_y = imu_data.gyro[1];
    float gyro_z = imu_data.gyro[2];

    // Magnetic field
    float mag_x = imu_data.mag[0];
    float mag_y = imu_data.mag[1];
    float mag_z = imu_data.mag[2];
}
```

Advanced Functions

Update with Raw Values

```
result_t imu_sensor_update(
    imu_data_t *imu,
    float ax, float ay, float az, // Accelerometer values
    float gx, float gy, float gz, // Gyroscope values
    float mx, float my, float mz, // Magnetometer values
    uint32_t timestamp
);
```

Calculate Acceleration Magnitude

```
float acceleration_magnitude = imu_get_acceleration_magnitude(&imu_data);  
// Useful for impact detection and free-fall detection
```

Thread-Safe Data Reading

```
imu_data_t imu_copy;  
imu_sensor_read(&imu_data, &imu_copy);  
// Use imu_copy in another thread without locking
```

Typical Sensor Ranges

Measurement	Typical Range	Units
Acceleration	± 16	g
Angular Velocity	± 2000	$^{\circ}/s$
Magnetic Field	± 4800	μT

Conversion Reference

From	To	Factor
g	m/s^2	$\times 9.81$
$^{\circ}/s$	rad/s	$\times \pi/180$
Gauss	μT	$\times 100$

Validation Functions

```
// Validate single accelerometer value  
// Typical range: -50 to +50 m/s2  
result_t validate_accelerometer_value(float accel_value);  
  
// Validate all three acceleration axes  
result_t validate_imu_data(  

```

```
float accel_x,  
float accel_y,  
float accel_z  
);
```

Protobuf Message Format

```
message SensorBoardIMUInfo {  
    float accel_x;  
    float accel_y;  
    float accel_z;  
    float gyro_x;  
    float gyro_y;  
    float gyro_z;  
    SensorState state;  
}
```

Common Applications

Impact Detection

```
float mag = imu_get_acceleration_magnitude(&imu_data);  
if (mag > IMPACT_THRESHOLD) {  
    // High acceleration detected  
}
```

Tilt Detection

```
// Calculate tilt angle from acceleration  
float tilt_angle = atan2(imu_data.accel[0], imu_data.accel[2]);
```

Motion Classification

```
// Static vs dynamic based on gyro magnitude
```

```
float gyro_mag = sqrt(gyro_x*gyro_x + gyro_y*gyro_y + gyro_z*gyro_z);
```

Integration Notes

- Single IMU instance with 3-axis sensors (dual planned)
- All three axes transmitted as independent fields
- Acceleration magnitude available for impact detection
- Timestamp tracking enables dead reckoning applications
- Typical IMU update rate: 10-100 Hz
- Filter algorithms can be applied to raw data for smoothing

Revision #4

Created 2026-04-14 15:34:08 UTC by Shishir Nambiar

Updated 2026-04-14 17:17:09 UTC by Shishir Nambiar