GNSS

Global positioning module with NMEA 0183 protocol support. Contains data structures, initialization code, validation functions, and UART configuration for the GY-NEO6MV2 sensor.

Hardware Specifications

Parameter	Value
Model	GY-NEO6MV2 (NEO-6M)
Interface	UART (TTL serial)
Baud Rate	9600 bps
Protocol	NMEA 0183
Update Rate	1 Hz (configurable)

GPS Fix Quality Types

```
typedef enum {
    GPS_NO_FIX = 0,        // No GPS fix available
    GPS_GPS_FIX = 1,       // Standard GPS fix
    GPS_DGPS_FIX = 2,      // Differential GPS fix
    GPS_PPS_FIX = 3,       // PPS (Pulse Per Second) fix
    GPS_RTK_FIX = 4,       // Real-Time Kinematic fix
    GPS_RTK_FLOAT = 5      // RTK float solution
} gps_fix_quality_t;
```

Data Structure

```
typedef struct {
    // Position Data
    double latitude;           // ±90 degrees (North/South)
    double longitude;          // ±180 degrees (East/West)
    float altitude;            // Meters above sea level

    // Velocity & Direction
```

```
    float speed;               // Meters per second
    float heading;             // 0-360 degrees (course over ground)

    // Accuracy Indicators
    float hdop;                // Horizontal dilution of precision
    float vdop;                // Vertical dilution of precision
    int32_t satellites;        // Number of satellites in view
    gps_fix_quality_t fix_quality;

    // Timestamps
    int64_t utc_timestamp;     // Unix epoch (milliseconds)
    uint32_t timestamp;        // System timestamp (last reading)
    uint32_t last_timestamp;   // System timestamp (previous reading)

    // Status
    bool is_valid;             // Data validity flag

    // UART Parsing State
    char rx_buffer[256];       // UART receive buffer
    uint16_t rx_index;         // Current buffer position
    bool sentence_ready;       // Complete sentence available
} gps_data_t;
```

NMEA Sentences Supported

Sentence	Purpose
GGA	Fix data (time, position, fix quality, satellite count)
RMC	Recommended Min. Navigation Info (position, speed, heading, date)
GSA	DOP and active satellites (fix type, DOP values)

Initialization & Usage

Initialize GPS

```
gps_data_t gps_data;
gps_sensor_init(&gps_data);
```

Poll GPS Data

```
result_t gps_poll_result = poll_gps_sensor(&gps_data);

if (gps_poll_result == RESULT_OK) {
    double latitude = gps_data.latitude;
    double longitude = gps_data.longitude;
    float altitude = gps_data.altitude;
    int32_t satellites = gps_data.satellites;
    float hdop = gps_data.hdop;
}
```

Validation Functions

```
// Validate latitude (-90 to +90)
result_t validate_gps_latitude(double latitude);

// Validate longitude (-180 to +180)
result_t validate_gps_longitude(double longitude);

// Validate HDOP (horizontal dilution of precision)
// Typical range: 0-50
result_t validate_gps_hdop(float hdop);

// Validate satellite count
// Typical: 0-30 satellites
result_t validate_gps_satellite_count(int32_t satellites);
```

Protobuf Message Format

```
message SensorBoardGPSInfo {
    double latitude;
    double longitude;
    float altitude;
    float speed;
    float heading;
```

```
    float hdop;
    float vdop;
    int32 satellites;
    SensorState state;
    GPSErrorCode error_code;
}
```

Error Handling

```
if (gps_poll_result == RESULT_ERR_UNIMPLEMENTED) {
    // Hardware not connected
    diagnostics.gps_sensor_1.state = SensorState_SENSOR_IDLE;
    diagnostics.gps_sensor_1.error_code = GPSErrorCode_GPS_COMMUNICATION_FAILURE;
} else if (gps_poll_result == RESULT_ERR_COMMS) {
    // Communication error (timeout/CRC)
    diagnostics.gps_sensor_1.state = SensorState_SENSOR_ERROR;
} else if (gps_poll_result == RESULT_OK) {
    // Validate data before accepting
    if (validate_gps_latitude(gps_data.latitude) == RESULT_OK) {
        diagnostics.gps_sensor_1.state = SensorState_SENSOR_OPERATING;
    }
}
```

Integration Notes

- Configured for dual GPS redundancy capability
- UART buffer size: 256 bytes
- NMEA sentence max length: 83 characters
- Data published to network at main loop interval (5 seconds default)
- All GPS data transmitted via UDP with Protobuf encoding
- Temperature range: -40°C to +85°C (standard)

Revision #3
Created 2026-04-14 15:28:17 UTC by Shishir Nambiar
Updated 2026-04-14 17:16:32 UTC by Shishir Nambiar