Navigation Sensor Interface Firmware

Purpose:
- Provides the sensor interface for the main navigation controller.
- Data is collected from navigation sensors.
- The raw sensor data is processed to provide logical information (e.g. object distance relative to the robot, acceleration).
- The processed information is written to local registers that are ready by the main controller over a high-speed bus.
- Emphasis is placed on timing of the sensor loop.

Firmware Flow Overview

1. Start
2. Initialize Sensors/Controller
3. Data Acquisition/Processing
4. Write to Externally Addressable Registers
5. Data Request Interrupt
   - Read Request?
     - Yes: Write Reg. Data Back to Master
     - No: Write New Data
   - Write Request?

Navigation Sensor Controller Initialization

1. System Initialization
   - EN = 0?
     - Yes: Init Oscillation
       - Oscillator Stable
         - Yes
     - No

2. Set/Enable Watchdog Timer
3. Reset I2C Data Registers to Default

4. Init. GPIO
5. Init. Communication SPI/I2C
6. ADC Setup
7. SPI Communication

---

**Sensor Initialization**

- Enable Power for IR Sensor (Switch FET ON)
- Enable Power for Ultrasonic Sensor (Sonar Auto Calibration Begins)
- Enable Accelerometer x, y, z axes

---

- Enable Power for IR Sensor
  - Conversion Ready?
    - Yes
      - Convert ADC value to Distance via Lookup table
      - Write result to temp register
    - No
      - Wait 100ms for IR Sensor Setup Time

- Set ADC Channel to IR Sensor R Channel (AN0)

---

- Send 50 us Pulse to Sonar Sensor Rx
- Wait 200 ms for Sonar Calibration to Finish
- Write result to temp register
- Convert ADC value to Distance via Lookup table

---

- Set ADC Channel to IR Sensor L Channel (AN1)
- Conversion Ready?
  - Yes
    - Sonar Sensor Counter = 0
    - Set Ready Bit in Status Register
    - Write buffered sensor values to I2C reg.
      - (atomic)
  - No
    - Sonar Sensor Counter >= 4?
      - Yes
        - Sonar Sensor Counter++
        - Write result to temp register
      - No

- Convert ADC value to Distance via Lookup table
Sensor Sampling Cycle

Scheme 1 : Sonar Sensor Group Firing
(Chosen Scheme)

Queue Sonar Sensors

Accelerometer x,y,z Reading
Left IR Ground Sensor
Sonar Sensor Group 1
Right IR Ground Sensor
Sonar Sensor Group 2

0 25 50 75 100 ms

Scheme 2 : Individual Sonar Firing

Queue Sonar Sensors

Accelerometer x,y,z Reading
Left IR Ground Sensor
Sonar Sensor Front Left
Right IR Ground Sensor
Sonar Sensor Front Center
Sonar Sensor Front Right
Sonar Sensor Back Center

0 25 50 75 100 125 150 175 200 ms
Sensor Sampling Cycle

B → Sensor Sampling Cycle → Send 50 us Pulse to Sonar Sensor Rx → Read Accel. Status Reg

New x,y,z data available?

No → Ground Sensor Counter = 0 → Write Accel. Data to Temp Buffer → Convert Readings to Units G-Force → Read x,y,z axes data

Yes → Conversion Ready?

No → Convert ADC value to Distance via Lookup table → Write Distance to Temp Buffer

Yes → Ground Sensor Counter >=2?

No → Ground Sensor Counter++

Yes → Convert ADC value to Distance via Lookup table

Sonar Sensor Counter = 0 → Set ADC Channel to Current Sonar Sensor (AN2-AN5) → Conversion Ready?

No → Set ADC Channel to Current Ground Sensor (AN0-AN1) → Conversion Ready?

Yes → Write Distance to Temp Buffer

No → Wait 50ms for Current Group Conversion → Sonar Group Counter = 0

Yes → Sonar Group Counter++

No → Sonar Group Counter >=2?

Yes → Write buffered sensor values to I2C regs. (atomic) → Set Ready Bit in Status Register → End

No → Sonar Group Counter++
Currently only RESET functionality over I2C Write, Any Other Needs?