SHIMMER 3

Basic fall detection with heartbeat

Written by Steffan Lildholdt (steffan@lildholdt.dk)

TABLE OF CONTENT

1. INTRODUCTION

Shimmer is a small sensor platform well suited for wearable applications. The integrated kinematic sensors, large storage and low-power standards based communication capabilities enable emerging applications in motion capture, long-term data acquisition and real-time monitoring. Shimmer offers the possibility for custom firmware integration so that the device can be tailored to fit many different applications. Shimmer3_FallDetection is an example of customized firmware.

This firmware is an extension of the “Basic fall detection” firmware. A key element in fall detection is detecting when the device is not able to transmit either due lack of power or a lost connection. In this extended firmware heartbeat functionality has been added, so that it transmits a character each 10 seconds which the PC listens for. If this character is not detected by the PC an alert is given.

Section 2 and 3 describes the architecture and design of the firmware so an overview can be obtained quickly. Section 4 is about how to test the firmware along with expected outputs and section 5 explains the key areas of the code and how to replace the current fall detection algorithm.

2. FIRMWARE ARCHITECTURE

The software architecture of the fall detection firmware is a combination of layered and module based and is illustrated in Figure 2.1. Each element is explained in the following.

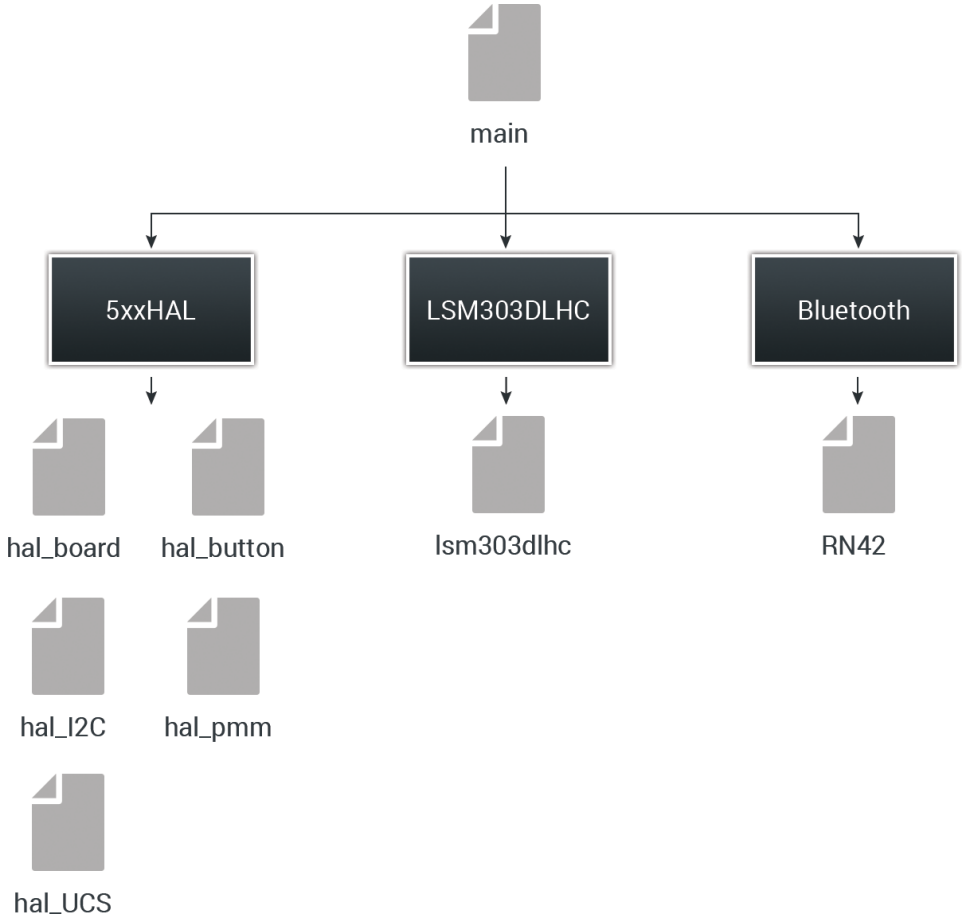

Figure 2.1 – Software architecture of fall detection firmware

Main
The main file is responsible for initializing all external components and controlling timers. Due to limited data processing all of this has also been placed in the main file.

Bluetooth
This is a driver for the RN42 Bluetooth module.

LSM303DLHC
This is a driver for the digital LSM303DLHC wide range accelerometer.

5xxHAL
A collection of various helper files. Hal_board is used to setup the board and control LEDs. Hal_button is a driver for the button. Hal_I2C is a driver for the I2C communication. Hal_pmm controls the power consumption (Power Management Module). Hal_UCS is driver for the timers (Unified Clock System).

3. FIRMWARE DESIGN

An overview of the fall detection algorithm is illustrated in the activity diagram in Figure 3.1.

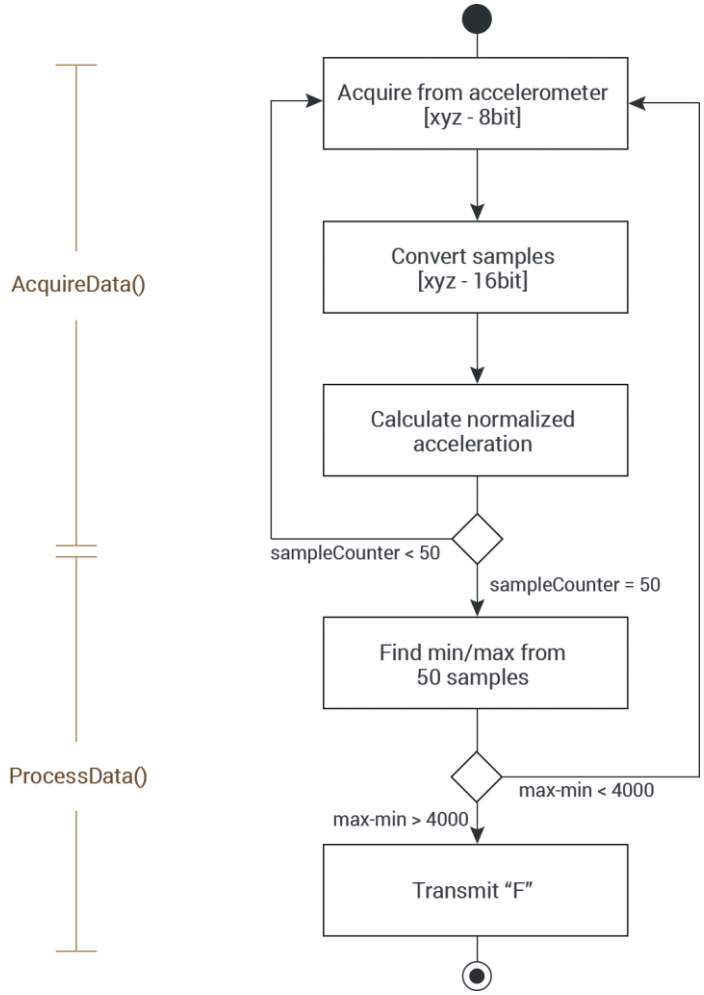

Figure 3.1 – Activity diagram of the fall detection algorithm

The digital LSM303DLHC accelerometer provides six 8bit values as the acceleration of the x, y and z axis. The structure is as follows.

$$[X\_LSB\ X\_MSB\ Y\_LSB\ Y\_MSB\ Z\_LSB\ Z\_MSB]$$

This needs to be turned into three 16bit values so that data processing can be applied later on. When this is done the normalized acceleration is calculated using the following formula

$$norm\_acc = \sqrt{X^2 + Y^2 + Z^2}$$

Samples are acquired 50 times per second. When the 50 samples have been acquired the minimum and maximum sample value is found. The difference between these is compared to a fixed threshold value of 4000. If the difference is higher an "F" is transmitted via Bluetooth otherwise the samples are cleared and the acquisition starts over.

4. TESTING THE FIRMWARE

To test the firmware, follow these steps.

1. If the Shimmer3 is not added as unit on your computer then follow the instructions in this link http://windows.microsoft.com/en-us/windows7/add-a-bluetooth-enabled-device-to-your-computer

2. Load the bootstrap "Shimmer3_FallDetectionHeartBeat.txt onto the Shimmer. The file is located in the folder "Resources/Firmware".
   (How to load custom firmware onto the Shimmer is described in tutorial 1)

3. Launch the PC application "Shimmer3_PCApp" located in the folder "Resources/PC Application/Shimmer3_PCApp"

4. Enter the COM port of the Shimmer Bluetooth connection (i.e. COM13)

5. The program should display the text "`Connected to COMxx`" and the blue LED on the Shimmer should be on. If the program keeps displaying the message "`Unable to connect to specified COM port`" try reload the firmware on the device.

6. Try to shake the Shimmer. The program should now display the text "`Alert: A fall has occurred`".

7. When the Shimmer is turned off the program will detect this after 15 seconds and display the text "`Warning: Lost connection to the Shimmer`", before asking for a COM port again in order to reconnect.

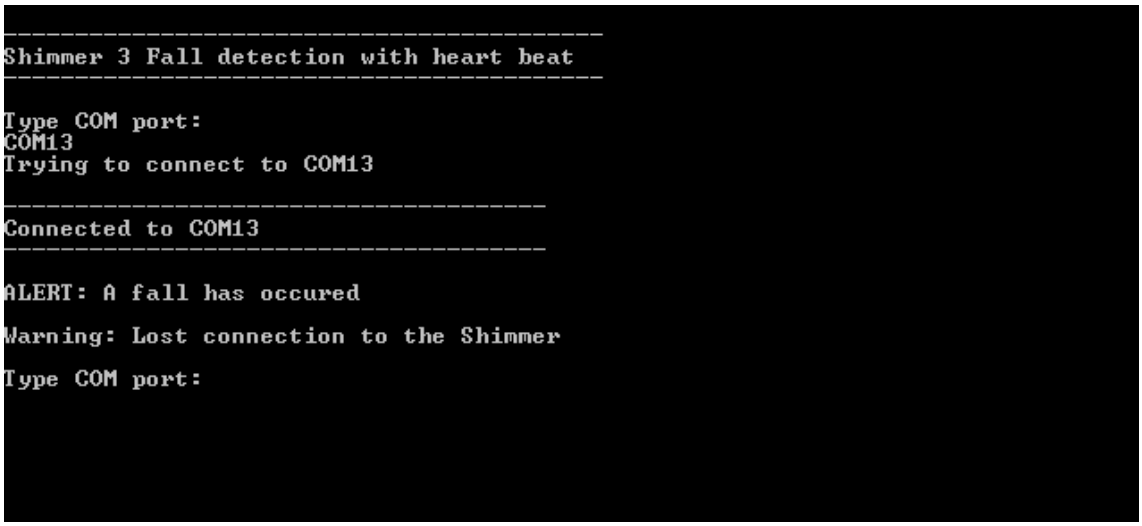

Figure 4.1 - The program output at successful connection, detection of fall and detection of lost connection to the Shimmer.

5. MODIFYING THE FIRMWARE

This section describes how to modify the heartbeat functionality. How to modify the actual fall detection algorithm is described in tutorial 2 – Basic fall detection.

5.1. Heart beat interval

Line 35 in the firmware:

```
#define HEARTBEAT_INTERVAL          10
```

This constant defines the interval of the heartbeat in seconds. If this is changed the PC application also needs to be modified in order for it to be aware of how often the Shimmer transmits a heartbeat.

Line 12 in the PC application:

```
const int HEARTBEAT_INTERVAL = 10;
```

These two intervals should always match. On the PC a buffer is added to the interval in order to cope with any delays on the Shimmer. As default the buffer is 5 seconds.

5.2. The heartbeat

The actual heart functionality is located at line 173-182 in the firmware.

```
TB0CCR3 += 32768;
if(heartbeatCounter == HEARTBEAT_INTERVAL)
{
     if(btIsConnected)
     {
            BT_write("H",1);
     }
     heartbeatCounter = 0;
}
heartbeatCounter++;
```

A timer is used to control how often the heartbeat is transmitted. Each second this timer increments the heartbeatCounter and when this reaches the HEARTBEAT_INTERVAL a single character (“H”) is transmitted.