VIETNAM NATIONAL UNIVERSITY HANOI
UNIVERSITY OF ENGINEERING AND TECHNOLOGY

Phạm Văn Thành

DEVELOPMENT OF A REAL-TIME SUPPORTED SYSTEM FOR FIREFIGHTERS ON-DUTY

MASTER'S THESIS IN ELECTRONICS AND COMMUNICATIONS ENGINEERING

Hanoi - 2016

VIETNAM NATIONAL UNIVERSITY HANOI

UNIVERSITY OF ENGINEERING AND TECHNOLOGY

Phạm Văn Thành

DEVELOPMENT OF A REAL-TIME SUPPORTED SYSTEM FOR FIREFIGHTERS ON-DUTY

Field: Electronics and Communications Engineering
Major: Electronic Engineering
Code: 60520203

MASTER'S THESIS IN ELECTRONICS AND COMMUNICATIONS ENGINEERING

SUPERVISOR: Assoc. Prof. Dr. Trần Đức Tân

Hanoi - 2016

AUTHORSHIP

I hereby declare that the work contained in this thesis is of my own and has not been previously submitted for a degree or diploma at this or any other higher education institution. To the best of my knowledge and belief, the thesis contains no materials previously published or written by another person except where due reference or acknowledgement is made.

Author

Student

Phạm Văn Thành

ACKNOWLEDGEMENT

I would like to express my sincere thanks to my advisor Assoc. Prof. Tran Duc-Tan, the professional of Faculty of Electronics and Telecommunication, University of Engineering and Technology – Vietnam National University, Hanoi for the guidance and support given to me throughout the thesis.

Special thanks to the lecturers of Faculty of Electronic and Communication for their help and guidance me in all thesis process.

Thanks for all members of the MEMS Lab for their help and discussed conversations.

At the end, I would like to thank my parents, my relatives and my friends because their comfort and supporting are the power for me going to success.

Sincerely

Pham Van Thanh

TABLE OF CONTENTS

Abstract

The firefighters can be injured by unintentional falls during the implementation tasks because of the broken in floors, structure elements; gas bombs; liquid boil ejection and toxic gases… in a fire. Therefore, this thesis aims to develop a portable and efficient device to monitor the falls by integrating a micro controller, a 3-DOF (Degrees of Freedom) accelerometer sensor, a MQ7 sensor (Semiconductor Sensor for Carbon Monoxide), a GSM/GPRS (Group Special Mobile/General packet radio service) modem, and the corresponding embedded fall detection algorithms. By developing algorithms and the corresponding simulations to monitor the fall event which can distinguish between being fall and the other daily activities (ADLs) such as standing, walking, running, sitting, lying. The signals from accelerometer are sent to the micro controller to monitor and alert the fall events. The cascade posture recognition is proposed to enhance the fall detection accuracy by determining if the posture is a result of a fall. Furthermore, MQ7 sensor is integrated into the proposed system to confirm the fall directly in emergency situations when air supporting device is working in failure. Based on the detection results, if a person falls with faint, an alert message will be sent to their leader via the GSM/GPRS modem. We had carefully investigated the threshold values (to determine the fall events) and the window size(to determine the time frame for analyzing) by MATLAB. After that, we selected the most suitable values for these parameters to achieve the optimal performance when it is working in emergency places.

Keywords: Firefighters, Acceleration, Fall detection, Posture recognitions, CO detection, Threshold investigations.

List of Figures

List of Tables

List of Abbreviations

ADLs	Daily activities
CCS	Cascading Style
CO	Carbonmonioxide
DOF	Degree Of Freedom
GPRS	General Packet Radio Service
I^2C	Inter – Integrated Circuit
LCD	Liquid Crystal Display
MCU	Microcontroller Unit
PASS	Personal Alert Safety System
SMS	Short Message Service
SPI	Serial Peripheral Interface
UART	Universal Asynchronous Receiver/Transmitter
UFFP	University of Firefighting and Prevention
ZCR	Zero Cross Rate

Chapter 1

INTRODUCTION

1.1. Overview about Firefighters

According to [1] there are 63,350 US firefighter injuries in 2014 with 27,015 occurred in fireground operations and a total of 64 firefighters died on-duty at the same year [4].

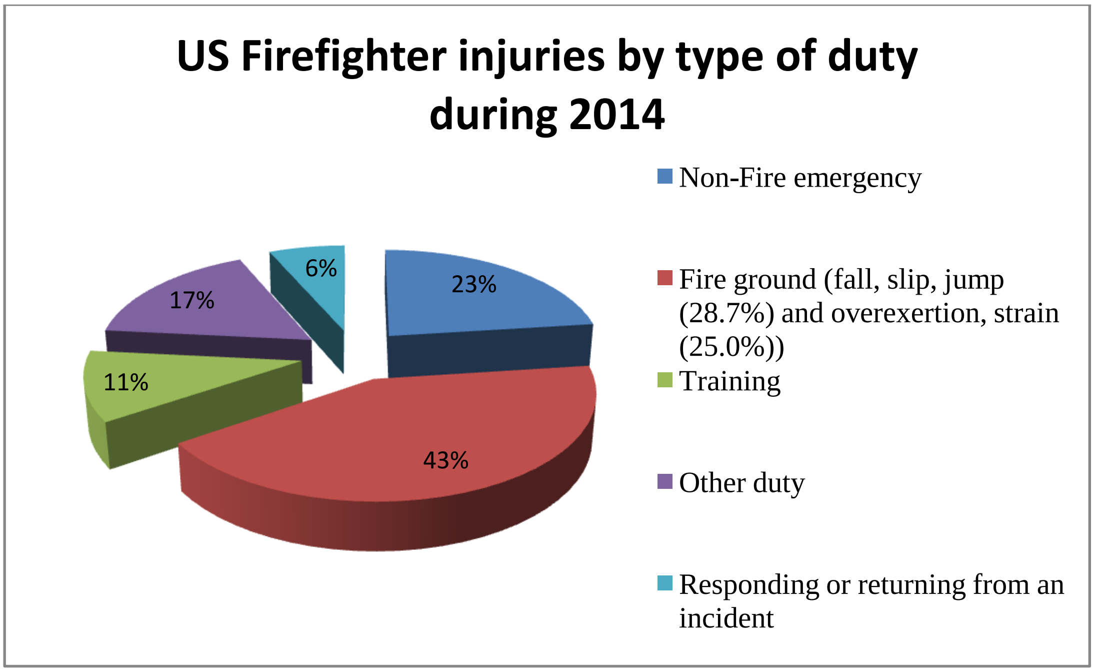


Figure 1-1– US Firefighter injuries by type of duty during 2014 [1]

In Vietnam, there are thousand of fires burning every year such as: 2357 and 2792 in 2014 and 2015 respectively [2] [3]. This is an alarm signal to alert about unsafely for firefighters in both US, Vietnam and in worldwide because they always are working and facing with a lot of dangers while they still have not enough and suitable supporting systems to protect their lives such as the fall detection systems in order to help them to escape from the dangerous situations.

Figure 1-2– Firefighter injury on-duty [5]

1.2. The research objectives

Based on the actual problem, this thesis mainly focus on improving the fall detection algorithms to distinguish between being fall and other activities of firefighters on-duty combined with CO level measurement to prevent the death because of the broken in floors, structure elements; gas bombs; liquid boil ejection and toxic gases and broken in air supporting devices...

1.3. The role of fall detection system

Fall detection system plays very essential role to support Firefighters on-duty to avoid the death because of the heat, smoke as others dangerous problems which can be appreared in a fire as discussed above or any other situations. When facing with the death if they donot have enough and siutable supporting systems, it will effect directly to their lives. Hence, the thesis mainly focus on proposed a system that can detect the fall events and CO threshold level as well as, and send out the message content to their leader and relative member for the help.

The proposed system can distinguish between being fall and others daily as on-duty activities of firefighters as running, walking, sitting, jumping,... in actual recorded data. Furthermore, most of firefighters and pedestrians were died by toxic smokes, CO is one of the most dangerous gas with the name “silent killer” and the process to find out the danger CO value in a fire is extremly important to protect the health, lives of firefighters.

1.4. The available supporting systems for Firefighters

There are several published methods used to detect the fall events in recent year such as: image processing [7], location sensors [8], smart phones and accelerometers [9][10]…but the reported publications were only used for the elderly and patients in clean air environments with long time to confirm self-stand up ability. Therefore, it is not suitable for firefighter’s activities in the fire environment conditions.

The department of Homeland Security also was developed a Personal alert safety system (PASS) [6] devices to equip for firefighters to detect high heat and smoke of a fire. PASS devices are designed to signal for aid via an audible alarm signal if a fire fighter becomes incapacitated on the fire ground. Furthermore, it can sense movement or lack of movement and activate a 95 decibel alarm if lack of motion exceeds a specific time period. Nevertheless, in a real fire situation, there are variety of noise like people voice; the operation of fire protection systems, fire truck, fire pumpes...Therefore, audible alarm signal is not useful in a big fire burning.

Figure 1-3– Personal alert safety system (PASS) devices from various manufacturers [6]

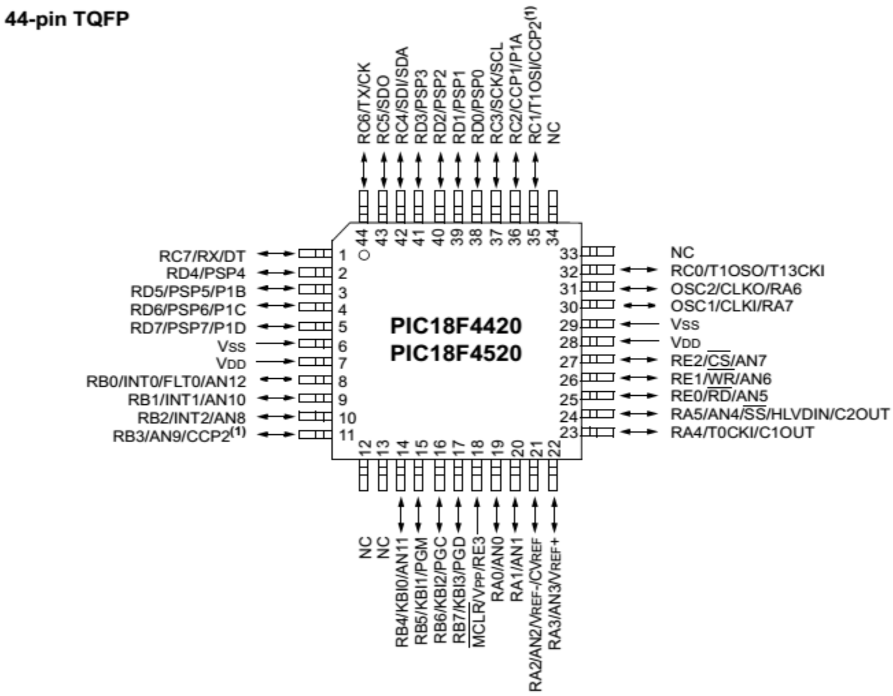
Based on the above limitations, this paper proposed to develop a real-time, low cost and high accuracy system which uses a 3-DOF accelerometer, MQ7 CO sensor combined with development the algorithms and the corresponding simulation process to monitor the fall events, which can be distinguished between fall and ADLs. It’s good for the fire environment and firefighters activities. Furthermore, we have used MATLAB to simulate and chosen the best size of the window and values of the threshold to improve the accuracy and performance of the system. The system can work well both in clean and fire environments with the first scenario that combined fall detection and posture recognitions and re-checked after 3 seconds to confirm they are faint or not. The second scenario is the output of both fall detection and CO detection modules to confirm they were fell or not, which caused by having air supporting devices broken.

Chapter 2

BACKGROUND AND HARDWARE DESIGN

2.1. Hardware

2.1.1. MCU PIC18f 4520



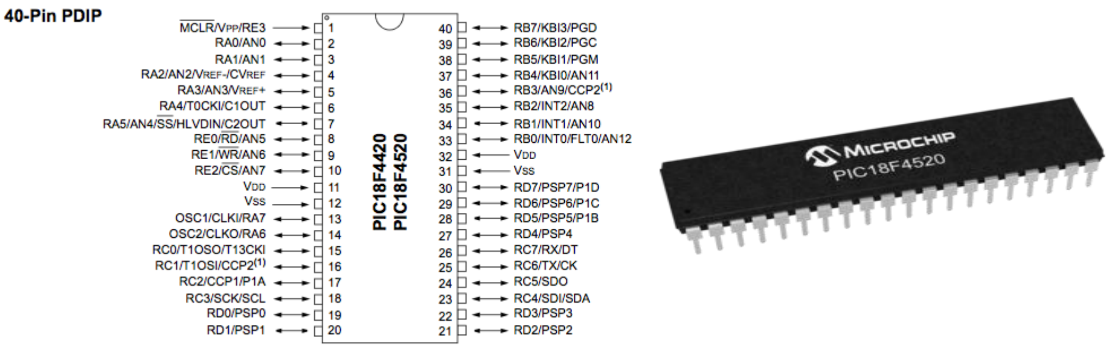


Figure 2-1– PIC18f 4520 pins [30]

Pic 18f4520 is a 10-Bit A/D and nanoWatt Technology microcontroller was developed by Microchip with some features as bellow:

Table 1: The Pic18f4520 features [30]

Features	PIC18F4520
Operating Frequency	DC – 40 MHz
Program Memory (Bytes)	32768
Program Memory (Instructions)	16384
Data Memory (Bytes)	1536
Data EEPROM Memory (Bytes)	256
Interrupt Sources	20
I/O Ports	Ports A, B, C, D, E
Timers	4
Capture/Compare/PWM Modules	1
Enhanced Capture/Compare/PWM Modules	1
Serial Communications	MSSP, Enhanced USART
Parallel Communications (PSP)	Yes
10-Bit Analog-to-Digital Module	13 Input Channels
Resets (and Delays)	POR, BOR, RESETInstruction, Stack Full, Stack Underflow (PWRT, OST), MCLR(optional), WDT
Programmable High/Low-Voltage Detect	Yes
Programmable Brown-out Reset	Yes
Instruction Set	75 Instructions; 83 with Extended Instruction Set Enabled
Packages	40-Pin PDIP, 44-Pin QFN,44-Pin TQFP

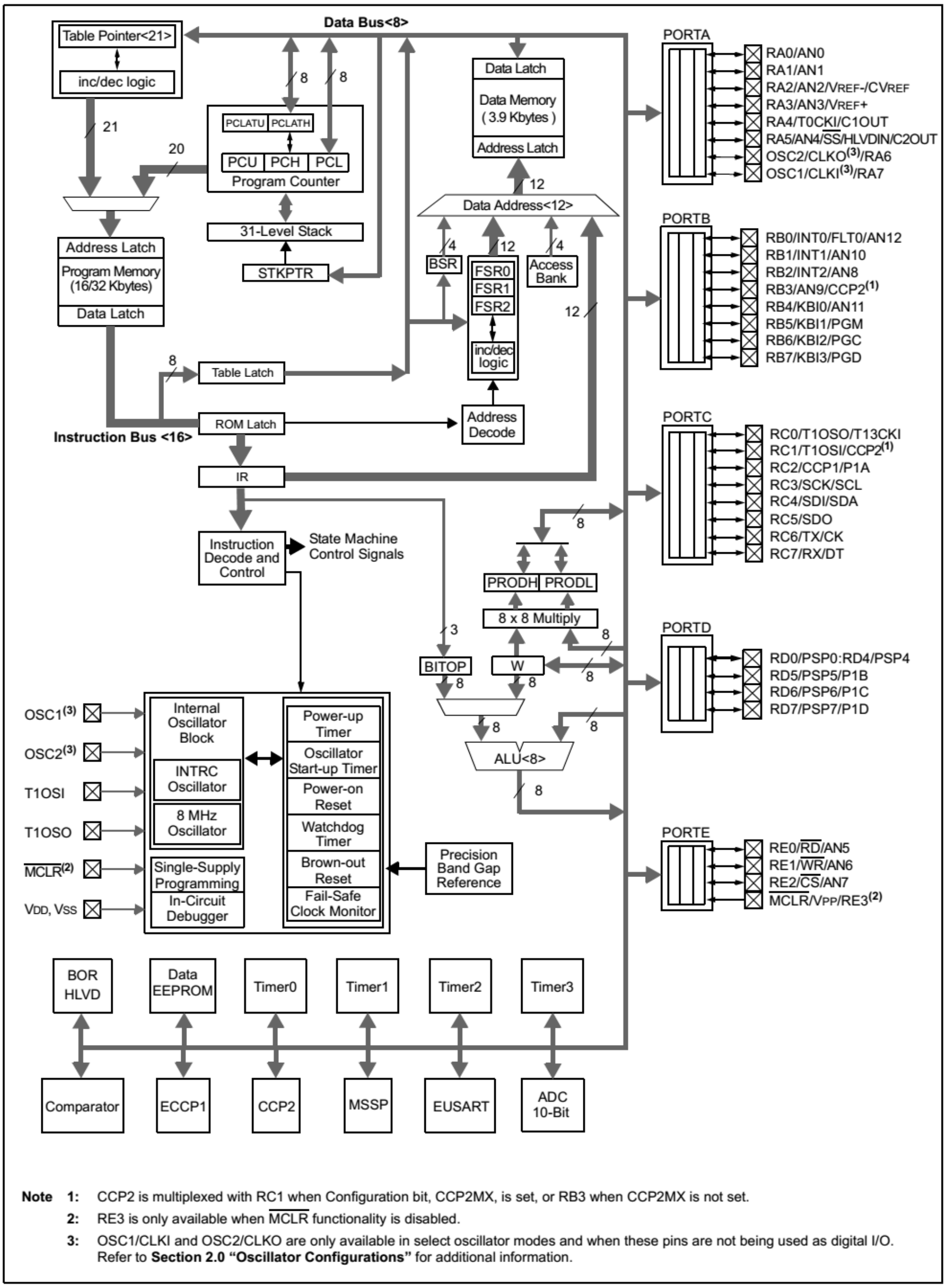


Figure 2-2– The structure of PIC18f 4520 [30]

2.1.2. ADXL345 accelerometers sensor

The ADXL345 is a small, thin, low power, 3-axis accelerometer with highresolution (13-bit) measurement at up to ±16 g [31]. Digital output data is formatted as 16-bit twos complement and is accessible through either a SPI (3- or 4-wire) or I^2C digital interface

Highlight features [31]:

- Ultralow power: as low as 40 μA in measurement mode and 0.1 μA in standby mode at V_S= 2.5 V (typical)
- Power consumption scales automatically with bandwidth
- User-selectable resolution. Fixed 10-bit resolution. Full resolution, where resolution increases with grange, up to 13-bit resolution at ±16 g (maintaining 4 mg/LSB scale factor in all granges)
- Tap/double tap detection
- Activity/inactivity monitoring
- Free-fall detection
- Supply voltage range: 2.0 V to 3.6 V
- SPI (3- and 4-wire) and I^2C digital interfaces
- Measurement ranges selectable via serial command
- Wide temperature range (−40°C to +85°C)

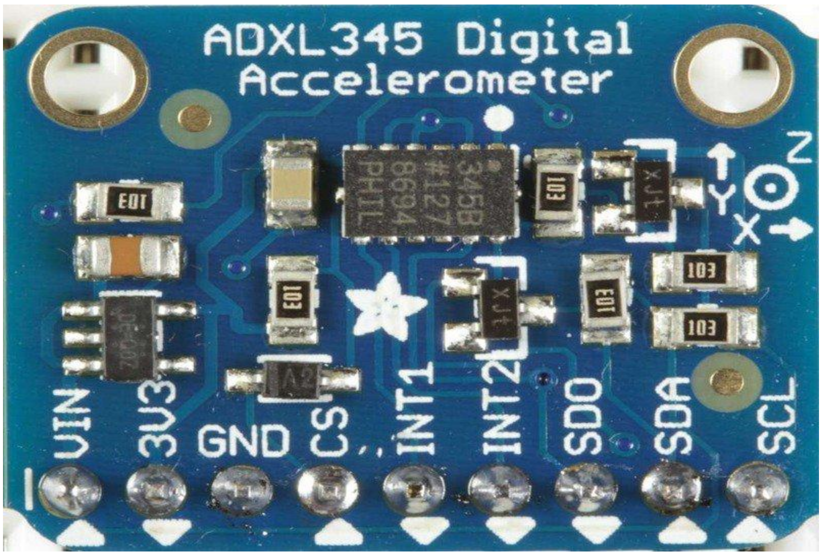


Figure 2-3– ADXL345 Digital Accelerometer

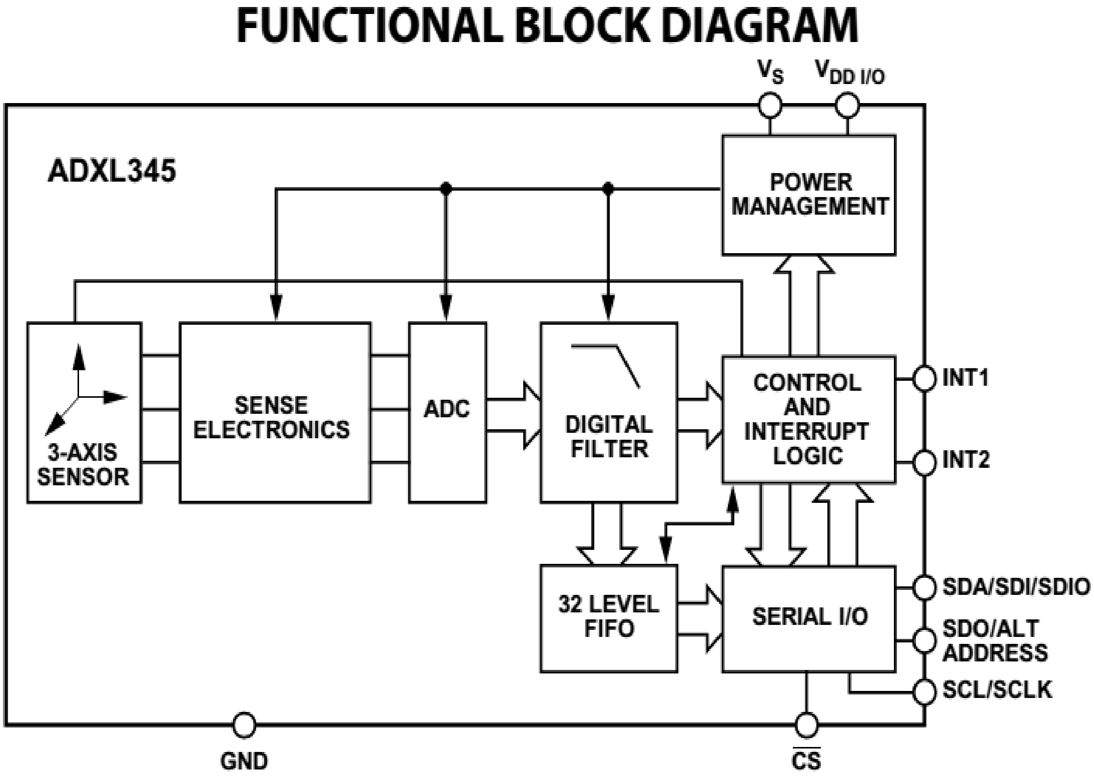


Figure 2-4– The functional block diagram of ADXL345 [31]

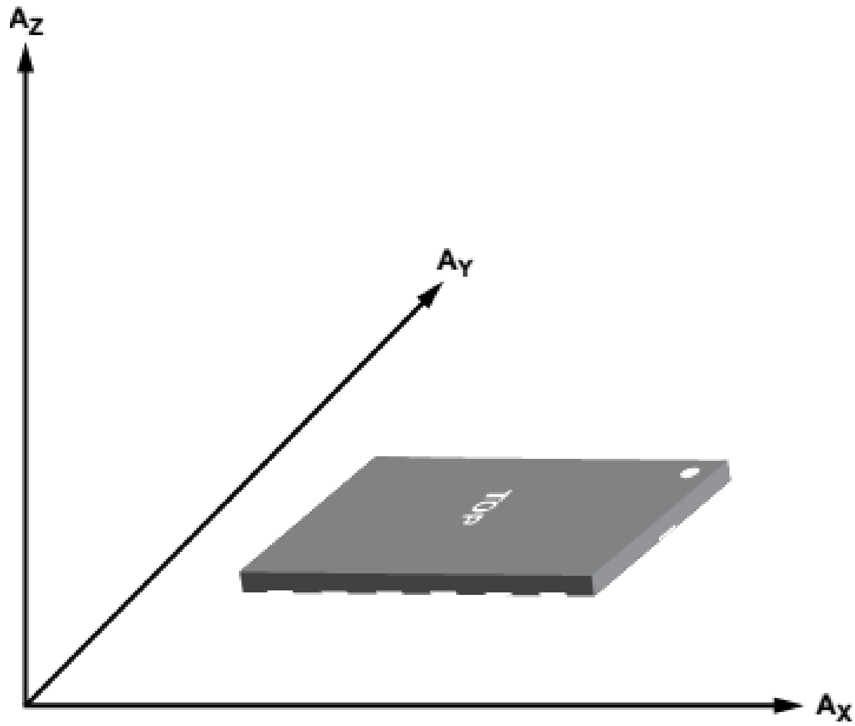


Figure 2-5– The axis of ADXL345 Accelerometer [31]

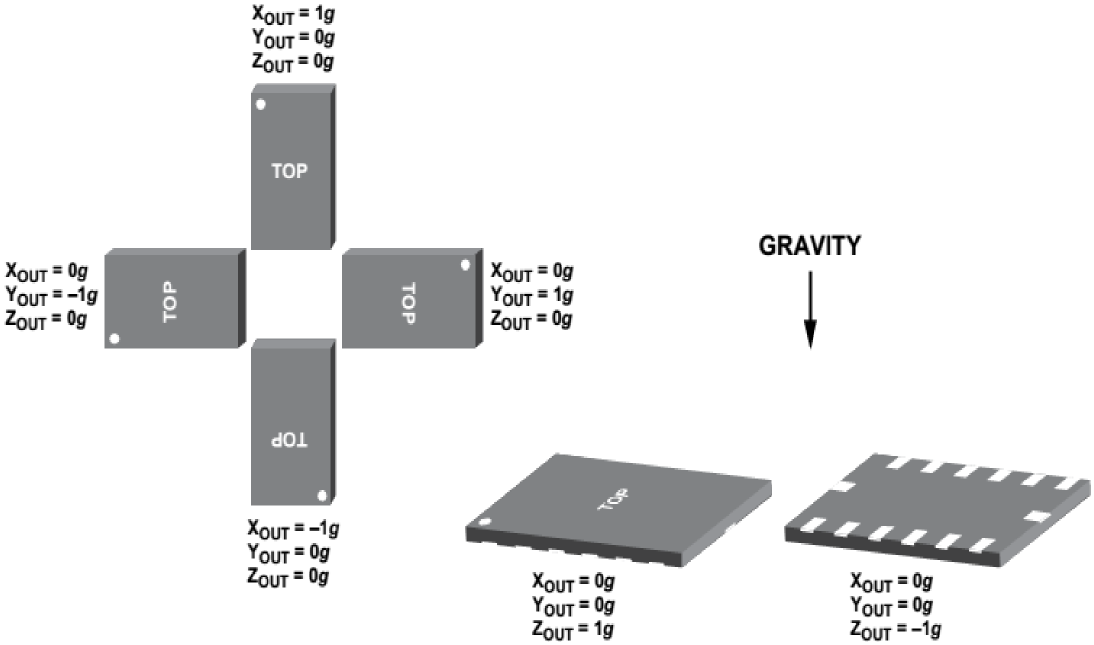


Figure 2-6– The positions and output responses [31]

2.1.3. SIM900

Featuring an industry-standard interface, the SIM900 delivers GSM/GPRS 850/900/1800/1900MHz performance for voice, SMS, Data, and Fax in a small form factor and with low power consumption. With a tiny configuration of 24mm x 24mm x 3mm, SIM900 can fit almost all the space requirements in your M2M application, especially for slimand compact demand of design [34].

Figure 2-7– The SIM900 Module [34]

The main features of Sim900 [34]:

- Quad-Band 850/ 900/ 1800/ 1900 MHz
- GPRS multi-slot class 10/8
- GPRS mobile station class B
- Compliant to GSM phase 2/2+
    + Class 4 (2 W @850/ 900 MHz)
    + Class 1 (1 W @ 1800/1900MHz)
- Dimensions: 24mm* 24mm * 3mm
- Weight: 3.4g
- Control via AT commands (GSM 07.07 ,07.05 and SIMCOM enhanced AT Commands)
- SIM application toolkit
- Supply voltage range 3.4 ... 4.5 V
- Low power consumption
- Operation temperature: -30 °C to +80 °C

2.1.4. MQ7 CO sensor

Sensitive material of MQ-7 gas sensor is SnO2, which with lower conductivity in clean air. It make detection by method of cycle high and low temperature, and detect CO when low temperature (heated by 1.5V). The sensor's conductivity is more higher along with the gas concentration rising. When high temperature (heated by 5.0V), it cleans the other gases adsorbed under low temperature [35].

MQ-7 gas sensor has high sensitity to Carbon Monoxide. The sensor could be used to detect different gases contains CO, it is with low cost and suitable for different application [35].

MQ7 sensor used in gas detecting equipment for cacbon monoxide (CO) in family and industry or car.

Table 2: The technical data of MQ7 [35]

Model No.			MQ-7
Sensor Type			Semiconductor
Standard Encapsulation			Plastic
Detection Gas			Carbon Monoxide
Concentration			10-10000ppm CO
Circuit	Loop Voltage	Vc	≤10V DC
	Heater Voltage	VH	5.0V±0.2V ACorDC（High） 1.5V±0.1V ACorDC（Low）
	Heater Time	TL	60±1S（High）90±1S（Low）
	Load Resistance	RL	Adjustable
Character	Heater Resistance	RH	31Ω±3Ω（Room Tem.）
	Heater consumption	PH	≤350mW
	Sensing Resistance	Rs	2KΩ-20KΩ(in 100ppm CO)
	Sensitivity	S	Rs(in air)/Rs(100ppm CO)≥5
	Slope	α	≤0.6(R300ppm/R100ppm CO)
Condition	Tem. Humidity		20℃±2℃；65%±5%RH
	Standard test circuit		Vc:5.0V±0.1V； VH（High）: 5.0V±0.1V； VH（Low）: 1.5V±0.1V
	Preheat time		Over 48 hours

Figure 2-8– The CO sensor [36]

2.2. Solfware

2.2.1. I^{2}C Interface

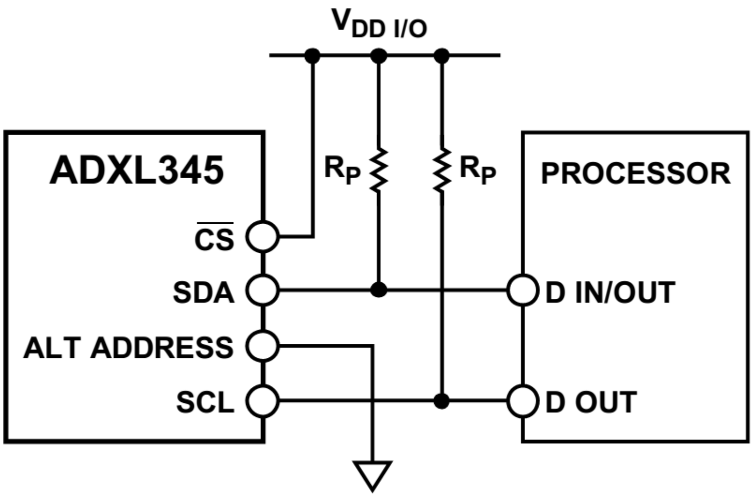


Figure 2-9– I^{2}C connection diagram [37]

The physical I^{2}C bus is just two wires, called SCL and SDA. SCL is the clock line. It is used to synchronize all data transfers over the I^{2}C bus. SDA is the data line. The SCL & SDA lines are connected to all devices on the I^{2}C bus. There needs to be a third wire, which is just the ground or 0 volts. There may also be a 5volt wire is power is being distributed to the devices. Both SCL and SDA lines are "open drain" drivers. What this means is that the chip can drive its output low, but it cannot drive it high. For the line to be able to go high, you must provide pull-up resistors to the 5v supply. There should be a resistor from the SCL line to the 5v line and another from the SDA line to the 5V line. You only need one set of pull-up resistors for the whole I^{2}C bus, not for each device, as illustrated below [32].

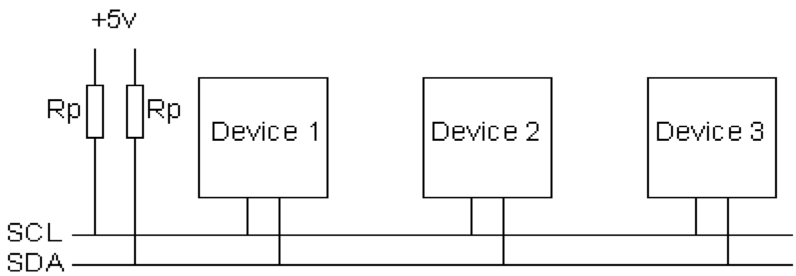


Figure 2-10– The physical I^{2}C bus [32]

The value of the resistors is not critical. I have seen anything from 1k8 (1800 ohms) to 47k (47000 ohms) used. 1k8, 4k7 and 10k are common values, but anything in this range should work OK. I recommend 1k8 as this gives you the best performance. If the resistors are missing, the SCL and SDA lines will always be low - nearly 0 volts - and the I^2C bus will not work [32].

2.2.1.1. Masters and Slaves

The devices on the I^2C bus are either masters or slaves. The master is always the device that drives the SCL clock line. The slaves are the devices that respond to the master. A slave cannot initiate a transfer over the I^2C bus, only a master can do that. There can be, and usually are, multiple slaves on the I^2C bus, however there is normally only one master. It is possible to have multiple masters, but it is unusual and not covered here. On our application, the master will be pic 18f4520 micro controller and the slaves will be three-axis accelerometer ADXL345 sensor. Slaves will never initiate a transfer. Both master and slave can transfer data over the I^2C bus, but that transfer is always controlled by the master [32].

2.2.1.2. The I^2C Physical Protocol

When the master pic 18f4520 micro controller wishes to talk to a slave (our ADXL345 sensor for example), it begins by issuing a start sequence on the I^2C bus [1]. A start sequence is one of two special sequences defined for the I^2C bus, the other being the stop sequence. The start sequence and stop sequence are special in that these are the only places where the SDA (data line) is allowed to change while the SCL (clock line) is high. When data is being transferred, SDA must remain stable and not change whilst SCL is high. The start and stop sequences mark the beginning and end of a transaction with the slave device [32].

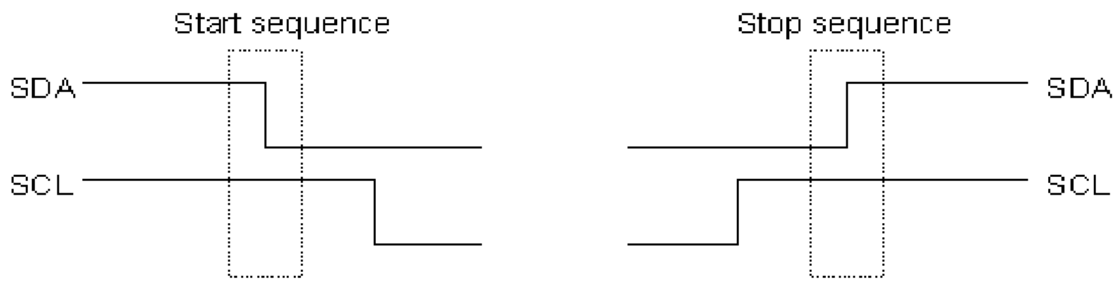
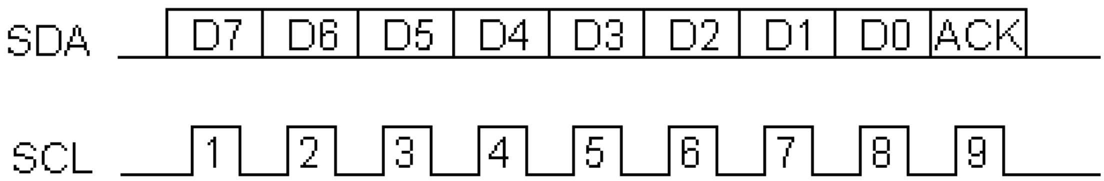


Figure 2-11– Start and stop sequences [32]

Data is transferred in sequences of 8 bits. The bits are placed on the SDA line starting with the MSB (Most Significant Bit). The SCL line is then pulsed high, then low. Remember that the chip cannot really drive the line high, it simply "let’s go" of it and the resistor actually pulls it high. For every 8 bits transferred, the device receiving the data sends back an acknowledge bit, so there are actually 9 SCL clock pulses to transfer each 8 bit byte of data. If the receiving device sends back a low ACK bit, then it has received the data and is ready to accept another byte. If it sends back a high then it is indicating it cannot accept any further data and the master should terminate the transfer by sending a stop sequence [32].

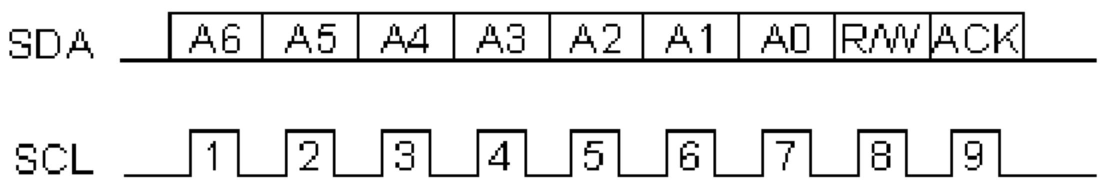


2.2.1.3. Clock

The standard clock (SCL) speed for I^2C up to 100KHz. Philips do define faster speeds: Fast mode, which is up to 400KHz and High Speed mode which is up to 3.4MHz [32].

2.2.1.4. I^2C Device Addressing

All I^2C addresses are either 7 bits or 10 bits. The use of 10 bit addresses is rare and is not covered here. All of our modules and the common chips you will use will have 7 bit addresses. This means that you can have up to 128 devices on the I^2C bus, since a 7 bit number can be from 0 to 127. When sending out the 7 bit address, we still always send 8 bits. The extra bit is used to inform the slave if the master is writing to it or reading from it. If the bit is zero the master is writing to the slave. If the bit is 1 the master is reading from the slave. The 7 bit address is placed in the upper 7 bits of the byte and the Read/Write (R/W) bit is in the LSB (Least Significant Bit). The address of slave ADXL345 is 0x53 [32].



2.2.1.5. The I^2C Software Protocol

The first thing that will happen is that the master will send out a start sequence. This will alert all the slave devices on the bus that a transaction is starting and they should listen in incase it is for them. Next the master will send out the device address. The slave that matches this address will continue with the transaction, any others will ignore the rest of this transaction and wait for the next. Having addressed the slave device the master must now send out the internal location or register number inside the slave that it wishes to write to or read from. This number is obviously dependant on what the slave actually is and how many internal registers it has. Some very simple devices do not have any, but most do, including all of our modules. Our CMPS03 has 16 locations numbered 0-15. The SRF08 has 36. Having sent the I2C address and the internal register address the master can now send the data byte (or bytes, it doesn't have to be just one). The master can continue to send data bytes to the slave and these will normally be placed in the following registers because the slave will automatically increment the internal register address after each byte. When the master has finished writing all data to the slave, it sends a stop sequence which completes the transaction. So to write to a slave device [32]:

- Send a start sequence
- Send the I2C address of the slave with the R/W bit low (even address)
- Send the internal register number you want to write to
- Send the data byte
- [Optionally, send any further data bytes]
- Send the stop sequence.

2.2.1.6. Reading from the Slave

Before reading data from the slave device, you must tell it which of its internal addresses you want to read. So a read of the slave actually starts off by writing to it. This is the same as when you want to write to it: You send the start sequence, the I^2C address of the slave with the R/W bit low (even address) and the internal register number you want to write to. Now you send another start sequence (sometimes called a restart) and the I^2C address again - this time with the read bit set. You then read as many data bytes as you wish and terminate the transaction with a stop sequence [32]:

- Send a start sequence
- Send 0x53 (I^2C address of the ADXL345)
- Send address (Internal address of the bearing register)

- Send a start sequence again (repeated start)
- Send 0xC1 (I^2C address of the ADXL345 with the R/W bit high (odd address)
- Read data byte from ADXL345
- Send the stop sequence.

2.2.2. UART communication

The Universal Asynchronous Receiver/Transmitter (UART) controller is the key component of the serial communications subsystem of a computer [33]. UART is also a common integrated feature in most microcontrollers. 3 pins we must care are Tx (transmitter), Rx (Receiver) and Ground.

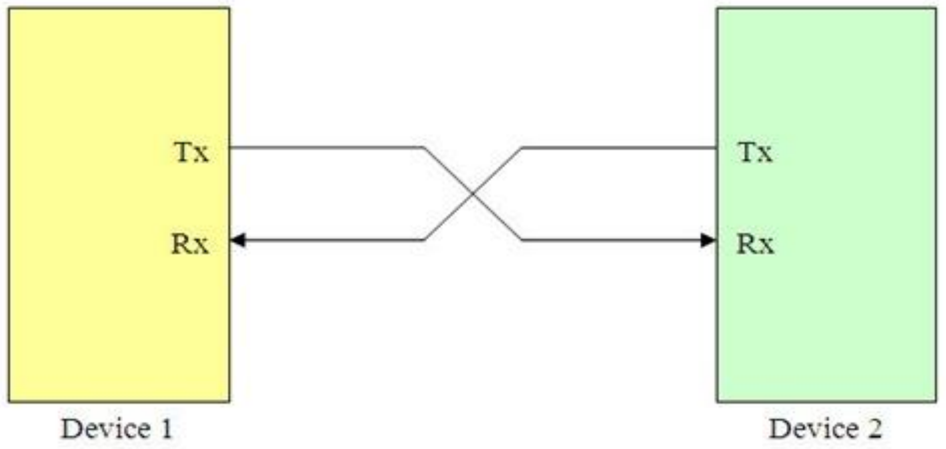


Figure 2-12– Communication between two devices [33]

2.2.2.1. The Asynchronous Receiving and Transmitting Protocol

The asynchronous communication it mean that both transmitter and receiving works in different clocks but must not exceed 10%. Start and stop bits are also sent with each data byte to identify the data. In this case, the sender and receiver must agree on timing parameters (Baud Rate) prior transmission and special bits are added to each word to synchronize the sending and receiving units [33].

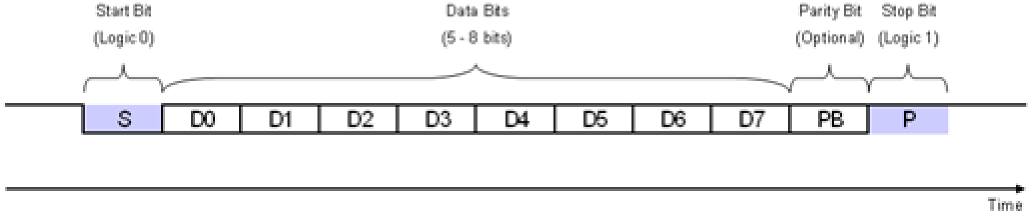


Figure 2-13– Basic UART packet form: 1 start bit, 8 data bits, 1 parity and 1 stop bit [33]

Every operation of the UART hardware is controlled by a clock signal, which runs at much faster rate than the baud rate. Transmitting and receiving UARTs must be set at the same baud rate, character length, parity, and stop bits for proper operation. The typical format for serial ports used with PC connected to modems is 1 Start bit, 8 data bits, no Parity and 1 Stop bit. UART is the simplest form of communication between microcontroller and PC. However, due to the mushrooming growth of technology, serial port is slowly being replaced by other means of communication port such as USB to RS-232 [33].

2.2.3. Timer

Timer as the name suggests pertain to time-related operations. They are mostly used for exact delay generation. Timers are also used in various other operations like PWM signal generation, auto-triggering of several other peripherals. In our project, we used timer0 for calculating data sample rate and timer1 for calculating exactly time to detect falls. Each of the four timers of Pic f84520 has certain special features some of which are explained below. The detailed list of these features can be obtained from PIC18f4520 datasheet [38].

2.2.3.1. Timer0 features [30]:

- Software selectable operation as a timer or counter in both 8-bit or 16-bit modes
- Readable and writable registers
- Dedicated 8-bit, software programmable prescaler
- Selectable clock source (internal or external)
- Edge select for external clock
- Interrupt-on-overflow

2.2.3.2. Timer1 features [30]:

- Software selectable operation as a 16-bit timer or counter
- Readable and writable 8-bit registers (TMR1H and TMR1L)
- Selectable clock source (internal or external) with device clock or Timer1 oscillator internal options
- Interrupt-on-overflow

- Reset on CCP Special Event Trigger
- Device clock status flag (T1RUN)

2.2.3.3. Timer2 features [30]:

- 8-Bit Timer and Period registers (TMR2 and PR2, respectively)
- Readable and writable (both registers)
- Software programmable prescaler (1:1, 1:4 and 1:16)
- Software programmable postscaler (1:1 through 1:16)
- Interrupt on TMR2 to PR2 match
- Optional use as the shift clock for the MSSP module

2.2.3.4. Timer3 features [30]:

- Software selectable operation as a 16-bit timer or counter
- Readable and writable 8-bit registers (TMR3H and TMR3L)
- Selectable clock source (internal or external) with device clock or Timer1 oscillator internal options
- Interrupt-on-overflow
- Module Reset on CCP Special Event Trigger.

2.3. The integrated system

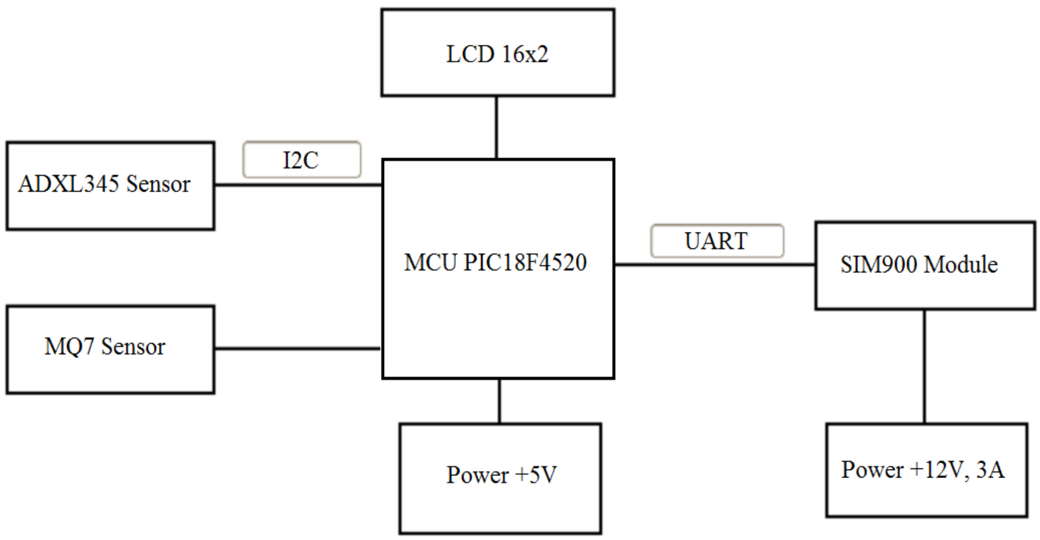


Figure 2-14– The connected modules in the proposed system

2.3.1. Power module

In automatic fall detect system. There are two level power sources, one for MCU and sensor accelerometer ADXL345 and other for module SIM900. With module SIM900, it only works when the current is larger than 2A. Consequently, we have been using an adapter 12V, 3A with LM2576 to get +5V, 2A. Power for MCU and sensor is +5V voltage, so one branch from the adapter with LM7805 voltage regulator we will receive it [38].

2.3.2. MCU module

The Microcontroller 18f4520 has been being used and clock source frequency (Crystal) is 20 MHz, which fast enough to execute fall, detect program [38].

2.3.3. SIM900 module

The using pins [38]:

Power on or down: PWRKEY should be pulled down at least 1 second and then released power on/down the module

Status: STATUS Power on the status

NETLIGHT Network status

SIM interface: SIM_VDD Voltage supply for SIM card +3V

SIM_DATA SIM_DAT input/output

SIM_CLK SIM clock

SIM_RST SIM reset

Serial port: RXD and TXD for UART communication between MCU and SIM900 module

2.3.4. Sensor ADXL345

I^2C interface used to connect between MCU and sensor. There are 4 pins has been used:

- Vcc 5VDC
- Ground
- SCL: the clock line
- SDA is the data line.

2.3.5. Sensor MQ7

There are 4 pins has used: Vcc, ground, A0 and D0; Vcc for +5VDC, A0 is the analog signal, D0 is the digital signal.

Chapter 3

METHODS

3.1. The 3-DOF accelerometer

The accelerometer is the heart of our proposed system to detect the fall events of firefighter's on-duty. The sensor used in the system is ADXL345 that can sense the acceleration in three dimensions x, y, z axes subtracted by the gravity vector G (G=9.81 m/s^2). Output data are accessible through the I^2C (Inter – Integrated Circuit) digital interface. The accelerometer is positioned in the waist so that y-axis must be paralleled with the earth's gravity to have expected reading results of accelerometers approximately in [0, 9.81, 0] m/s^2 as in Fig. 3-1 with the rate of 10 samples per second. Then, we applied a preprocessing step before taking data into the attribute extraction module to formulate the mean, orientation and standard deviation. The final step is data mining between fall detection and posture recognition and CO detection modules as well in the real-time.

Figure 3-1– Position of the 3-DOF accelerometer in waist body

3.2. Model of fall data processing

Fig. 3-2 shows the configuration for algorithm verification. The purpose is to investigate the best working conditions for the fall detection device before it would run independently. Firstly, ADXL345 used to get acceleration data in x, y, z – axes and transfer to MCU Pic18F4520 through a standard I^2C interface. Then, the MCU will send acquired data to computer through UART (Universal Asynchronous Receiver/Transmitter) communication cable for analyzing algorithms by Matlab. The acceleration data are stored in the buffer of 20 to 40 samples with the sampling rate of 10 Hz.

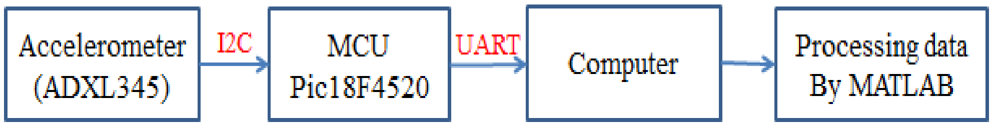


Figure 3-2– Fall data processing for fall detection system

3.3. The fall detection algorithms

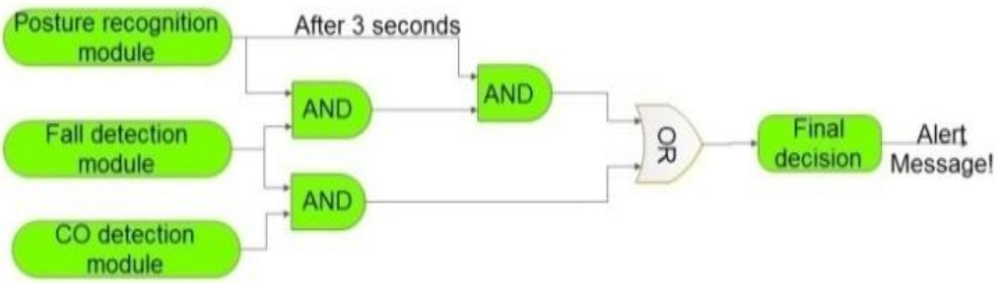


Figure 3-3– The summary of fall detection system

The final decision of fall based on the results of either fall detection module and posture recognition or CO detection module (see Fig. 3-3). Because firefighters are strong, the falling reasons usually come from the external causes such as the broken of floors and constructional elements; gas bombs, toxic gases, liquid boil ejection, etc. Fig. 3-4 shows the algorithmic architecture embedded in the MCU. The accelerometer will sense acceleration in three dimensions x, y and z, then posture recognition used to confirm the state of firefighters through the combination of three components: posture data base, suitable adjustment mode and acceleration values. Moreover, the proposed system also cares about dangerous events by the broken in air supporting devices in the high CO environment through the combination of fall detection and CO detection module which mounted inside of the mask.

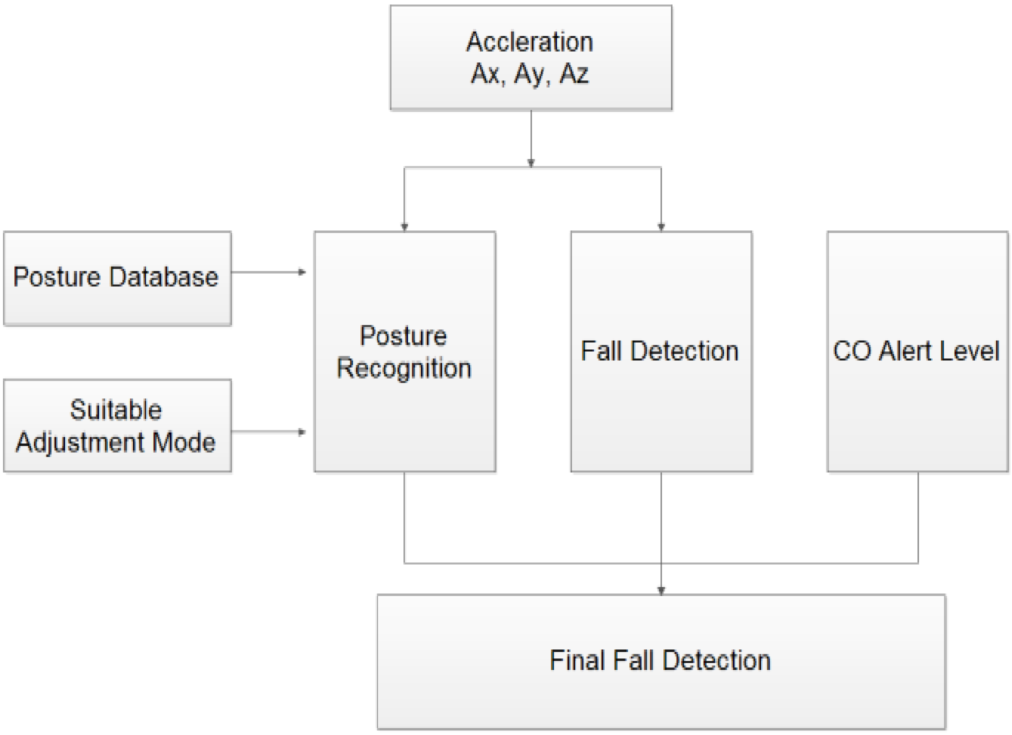


Figure 3-4– The proposed algorithms of fall detection

3.4. Posture Recognition Module

Fig. 3-5 shows the posture recognition module which will declare whenever people is standing, lying, walking, running, sitting or Null statues (Null state is the state of undefined confirmation). The target of this module that detects the fall events basing on the third threshold. Hence, we do not care about any kinds of postures. In this diagram, A_n is the average acceleration of three accelerations in x, y and z directions as below formula:

$$A_n = \sqrt{A_x^2 + A_y^2 + A_z^2}, \tag{1}$$

where n denotes the discrete time and the n[th] sliding window is formulated as:

$$W_n = [A_n \quad A_{n-1} \ \dots \ A_{n-19}]. \tag{2}$$

After that, the zero cross rate (ZCR) is computed by:

$$ZRC_n = \sum_{i=2}^{20} (A_{n+i} - DC < 0)(A_{n+i-1} - DC > 0), \tag{3}$$

where DC is the DC component of the A_n with ten acceleration samples are averaged and stored in a buffer of the MCU. As can be seen from Fig. 3-5 that when ZCR equals to zero, it means that the firefighters are in steady states.

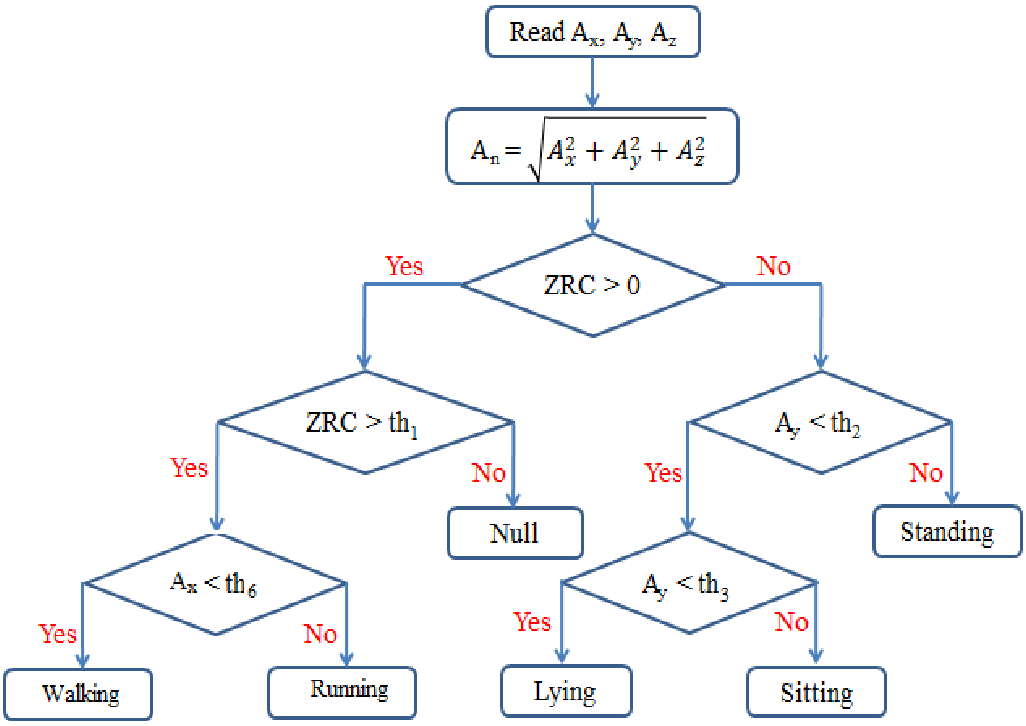


Figure 3-5– Flow chart of posture recognition

Fig. 3-6, illustrates the roles of two thresholds, th_1 and th_2 with a real experiment data. In the case of the person is moving (i.e. walking or Null), the threshold th_1 is used to confirm that the person is walking [16]. Otherwise, th_2 is used to confirm standing or lying postures. It is obvious that if the person is standing, the A_y (vertical acceleration) would show large enough amplitude.

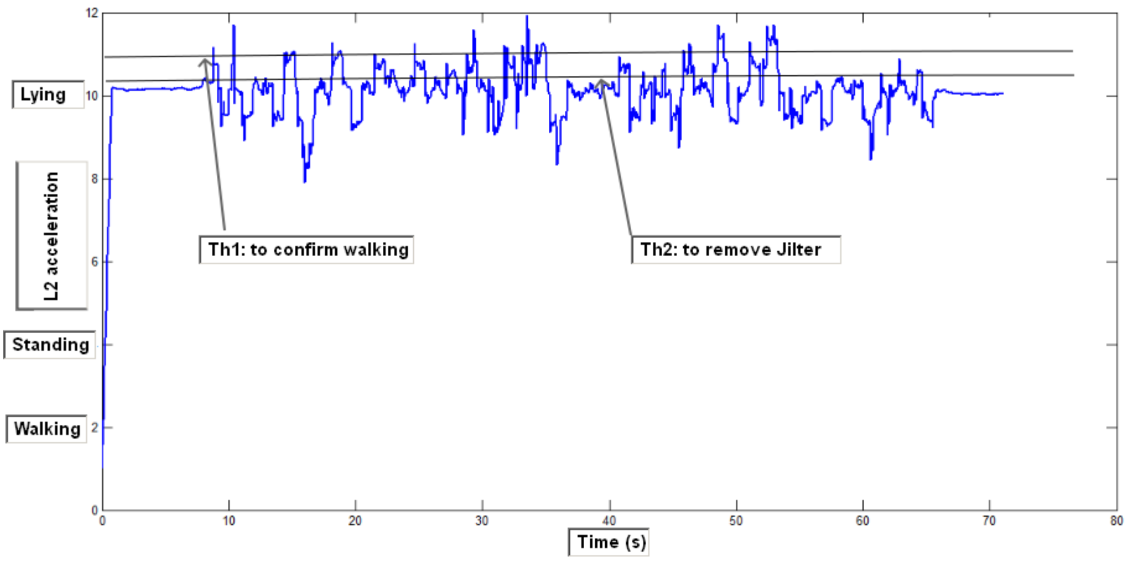


Figure 3-6– Illustration of two threshold th_1 and th_2 [39]

By using ZRC, L_1 norm of A_y acceleration, th_1 and th_2, four postures can be identified and assigned by corresponding values as shown in Table 2. The Boolean values in the third column would be used in the final decision (see Fig. 3-4. Note that, these values in the second column have illustrating meaning only (see Fig. 3-7). This figure shows the result of our posture recognition of a human in a period of 95 seconds with several phases of postures such as standing - walking - standing - lying -standing. All estimated postures are matched to experimented postures.

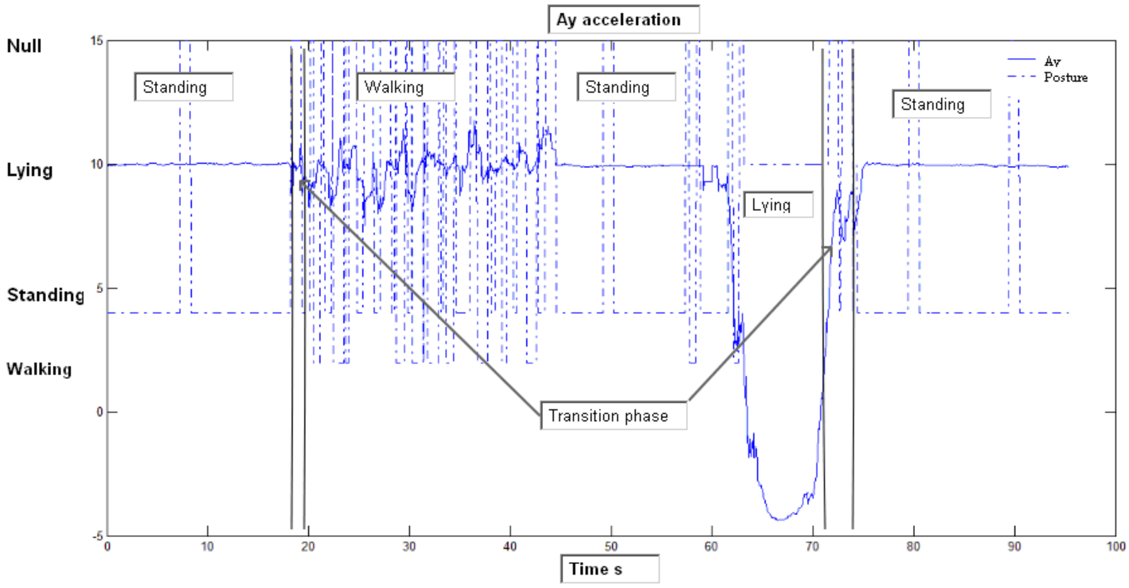


Figure 3-7– A_y acceleration vs. posture cognitions [39]

Table 3: Assigned Values for Different Postures [38]

State	**Values for illustration**	**Boolean values**
Walking	2	0
Standing	4	0
Lying	10	1
Null	15	1

3.5. Cascade Posture Recognition

Cascade posture recognition is very essential role in fall detection system because it will check the posture of elderly after 3 seconds to confirm the fall. If a firefighter was falled without stand-up ability, it means that the posture will keep at the steady states (lying or Null states) after falling and they need the help from leaders and relative members. Furethermore, some firefighters can self-stand up after fall and they do not the help from others, this is the fact problem which recorded during the process of getting real-life datasets. Hence, cascade posture recognition will check and auto remove sending out message in fall-like events, self-stand up ability and posture recognition failed.

3.6. Fall Detection Module

The fall detection module is the difference between two consecutive L_2 acceleration as below:

$$D_n = A_n - A_{n-1}. \tag{4}$$

The searching algorithm utilizing D_n is applied to find two positions corresponding to minimum and maximum of A_n, the difference between A_n and A_{n-1} would be compared with threshold th_4 to determine whether the fall event happens. If th_4 is chosen large, the fall events may be ignored, for small value there are many activities that will be detected as falls.

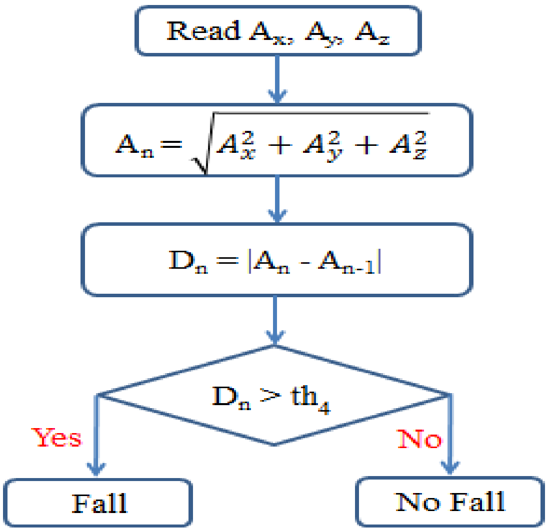


Figure 3-8– Fall detection module

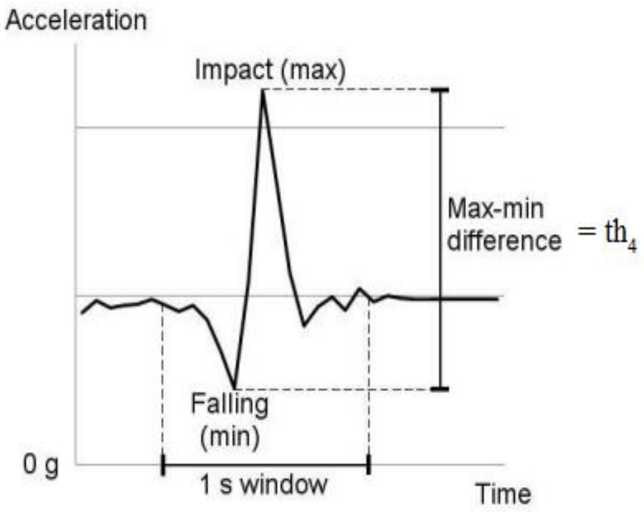


Figure 3-9– L_2 acceleration pattern of a fall sample [9]

3.7. CO Detection Module

Fig. 3-10 describes about the process of using MQ7 sensor to detect the fall events by using the threshold value th_5 to distinguish between clean and smoke environments. The reasons in choosing MQ7 sensor that there are a lot toxic gases from the fire burning process [13] such as: CO, CO_2, N_xO, NO_x…it depends on the type of burning materials. Nevertheless, CO named as the "silent killer" is the most dangerous to people's lives.

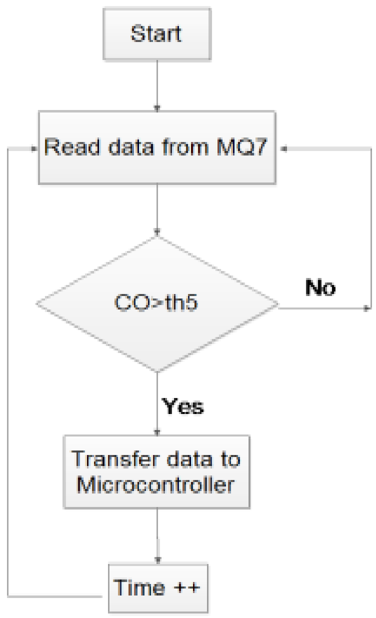


Figure 3-10– CO detection algorithm

Table 4: Carbon Monoxide Concentrations, COHb Levels, and Associated Symptoms [11]

Carbon Monoxide Concentration	COHb Level	Signs and Symptoms
35 ppm	<10%	Headache and dizziness within 6 to 8 h of constant exposure
100 ppm	>10%	Slight headache in 2 to 3 h
200 ppm	20%	Slight headache within 2 to 3 h; loss of judgment
400 ppm	25%	Frontal headache within 1 to 2 h
800 ppm	30%	Dizziness, nausea, and convulsions within 45 min; insensible within 2 h
1600 ppm	40%	Headache, tachycardia, dizziness, and nausea within 20 min; death in less than 2 h
3200 ppm	50%	Headache, dizziness, and nausea in 5 to 10 min; death within 30 min
6400 ppm	60%	Headache and dizziness in 1 to 2 min; convulsions, respiratory arrest, and death in less than 20 min
12,800 ppm	>70%	Death in less than 3 min

MQ7 sensor was mounted inside of the mask as Fig. 3-11 to detect CO gas when air supporting devices are working in failure, the value th_5 was chosen based on the process of testing carefully with 10 times between clean and dry wood burning rooms at UFFP. It can be seen clearly in Fig. 4-3, the values of th_5 between clean and smoke environment in ppm (parts per millions).

Figure 3-11– CO sensor location

3.8. Final Decision

The final decision is based on either output of fall detection combined with posture recognition or CO detection module as shown in Fig. 3-4. The more details are shown in Fig. 3-12. When a fall is detected, the first posture recognition is executed to determine if the posture state is Lying or Null. After 3 seconds, the second posture recognition is used to confirm a true fall alarm. The reason for the need of cascade posture recognitions is that after a short time, some firefighters can stand up by themselves together with improving the algorithms's accuracy. Furthermore, the detection high CO level, it means that air supporting devices are working in failure and they need the help of their leader soon to avoid the coming death.

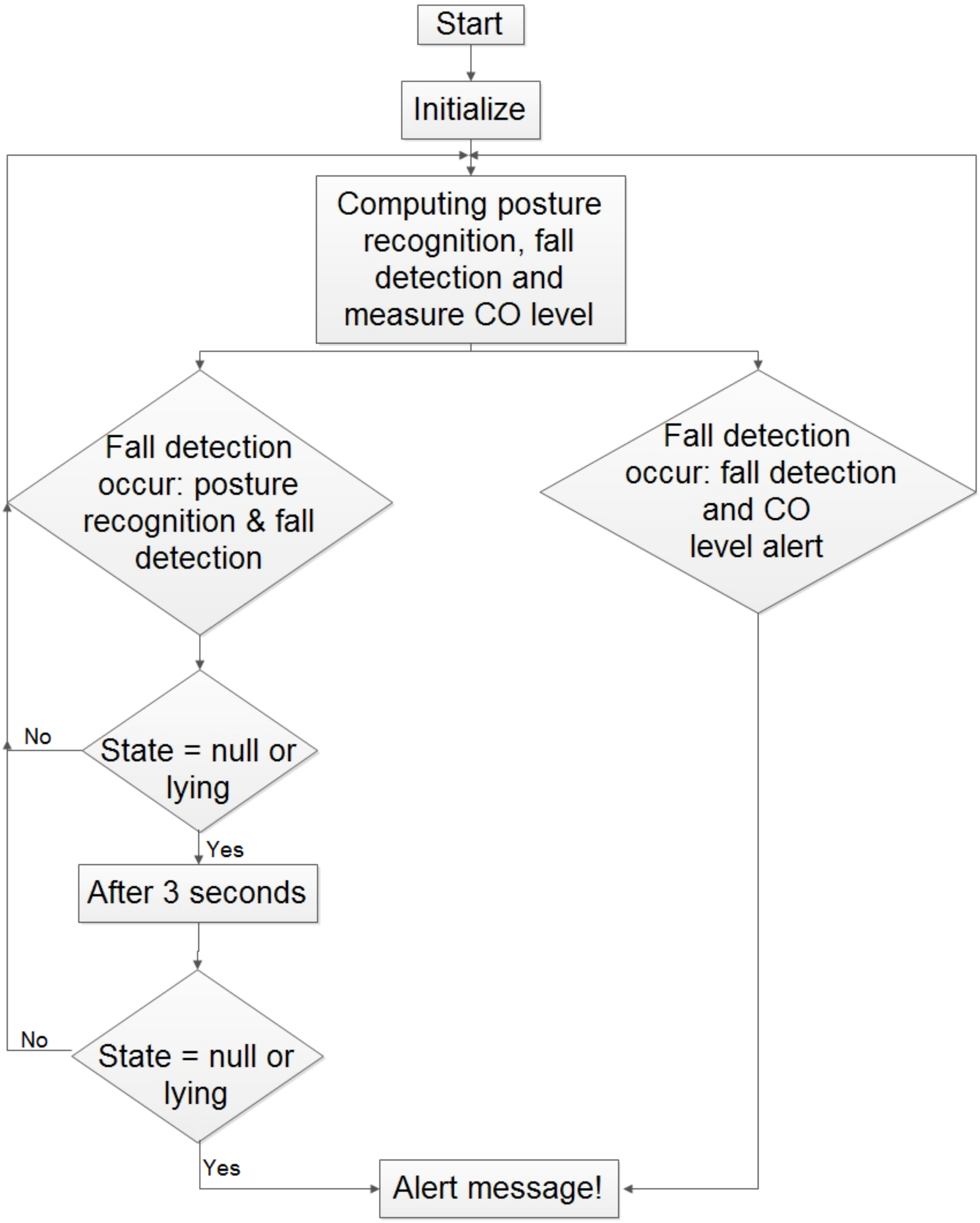


Figure 3-12– Fall decision using fall detection combined cascade posture recognitions and CO alert level

Table 5: Final Decision of Fall using Cascade Posture Cognition.

Fall Detection	**Posture Recognition**	**CO Detection Level**	**Posture after 3s of Fall**	**Final Decision**
Fall	Standing		Don’t care	False
Fall	Walking		Don’t care	False
Standing or Walking	Lying		Don’t care	False
Fall	**Lying**		Standing or Walking	False
			Lying or Null	**True**
Fall	**Null**		Standing or Walking	False
			Lying or Null	**True**
Fall	Don’t care	$\geq th_5$	Don’t care	**True**

Chapter 4

RESULTS AND DISCUSSIONS

4.1. Experimental setup and testing

For the experiment testing, we tested on 12 firefighters ages: 18-35, height: 1.68 – 1.75 m, weight: 62-75 kg who were randomly selected from various firefighters in UFFP to get actual data in their life, the volunteers placed the devices on the waist with the most comfortable and CO sensor is located inside the mask.

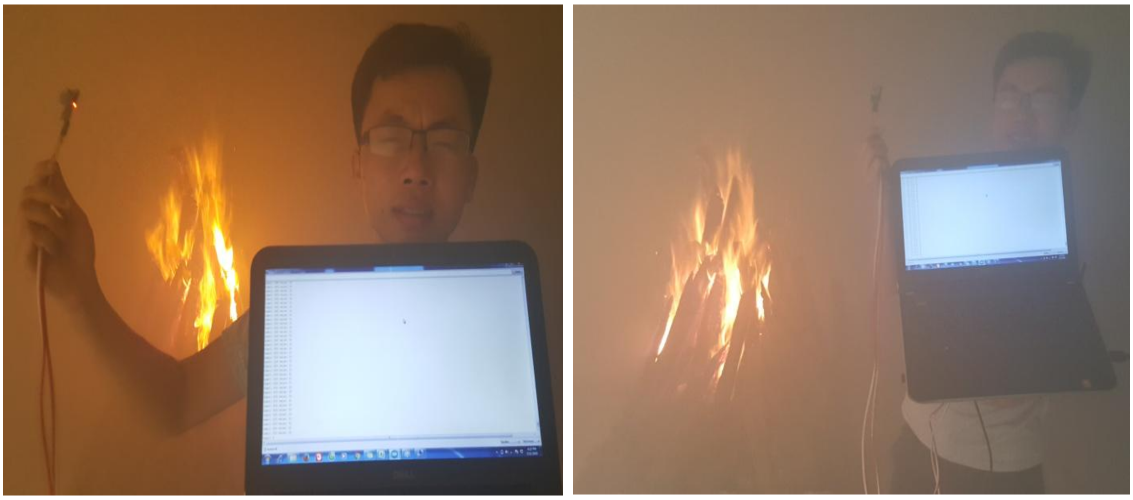


Figure 4-1– The author testing and measuring the CO level in the fire

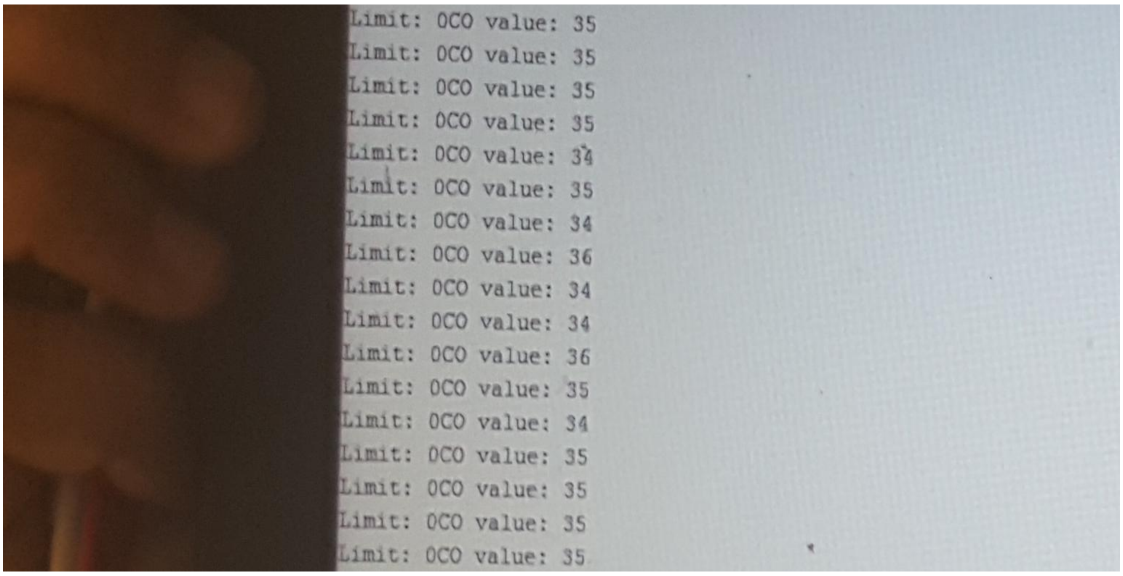


Figure 4-2– The CO level in the fire

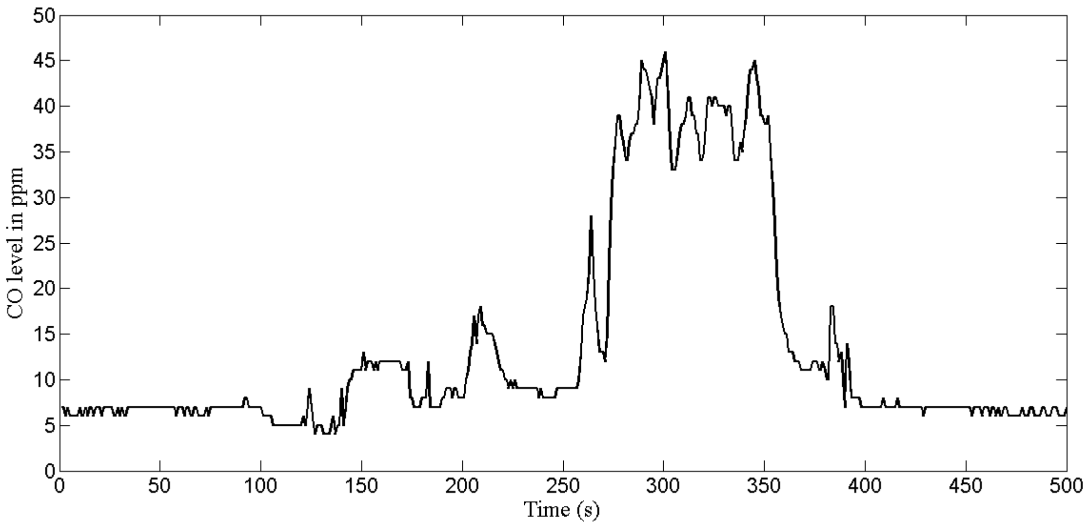


Figure 4-3– CO levels between clean and smoke environments

Fig. 4-1 shows the measuring CO process by volunteer in the fire condition and Fig. 4-2 is the result of CO level in firewood enviroment. Based on Fig. 4-3 can be seen that the clean environment, the CO value fluctuated around 7 ppm, and it had increased to 10 and 15 ppm when we moved MQ7 sensor nearby the main door and th_5 value varied from 33 to 45 ppm in the fire room. Based on the real testing process at UFFP that people feel breathless, headache…in the combination of all toxics and CO signs and symptom level, which published in [11], the authors chose th_5 = 33 ppm to protect firefighters on-duty.

Furthermore, we investigated this system with many single ADLs or combined ADLs. The data will be brought to computers to process and calculate suitable thresholds. After a careful test, analysis and calibration, the posture recognition works well with suitable thresholds. The results are shown in the following figures:

Fig. 4-4 shows A_x, A_y, A_z in standing statues: $A_x = A_z = 0$, $A_y = 10m/s^2$. The Fig. 4-5 is the posture recognition of the Fig. 4-4. The A_y axis value keeps the same around $10m/s^2$ and the value of posture recognition is $4m/s^2$. It is no fall detection during 70 seconds.

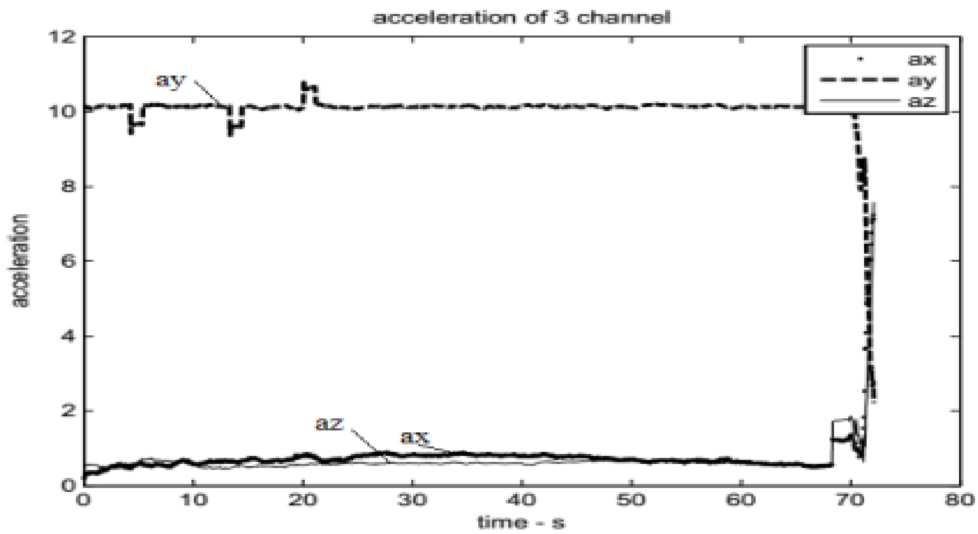


Figure 4-4– Standing

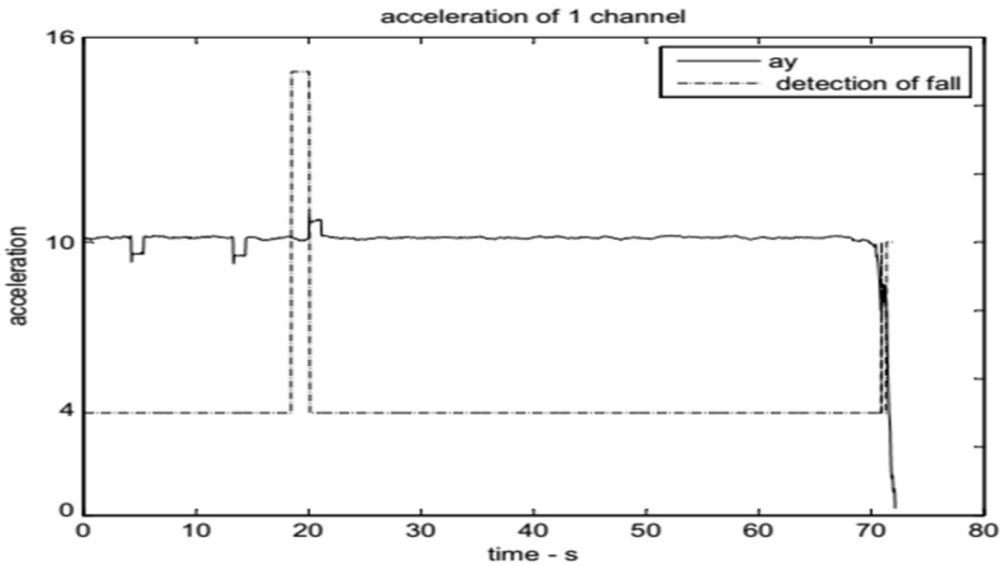


Figure 4-5– Standing posture

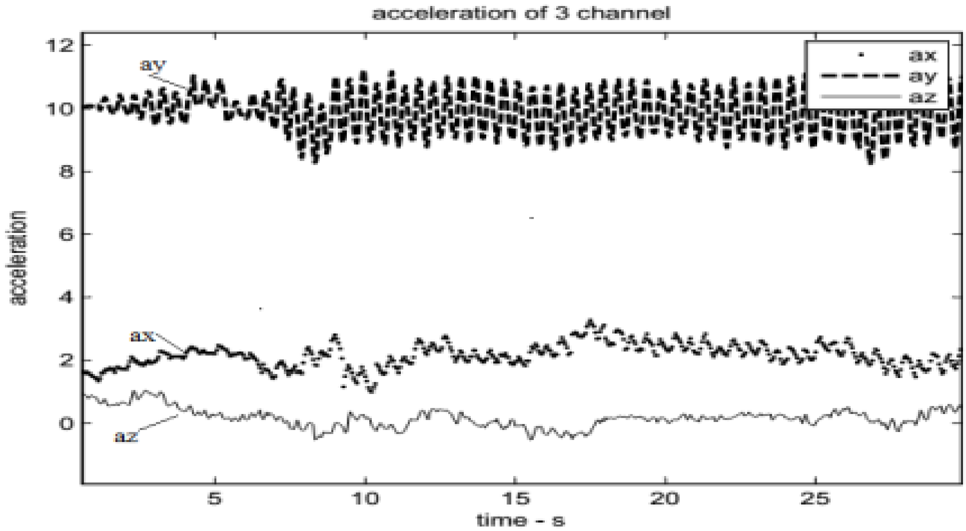


Figure 4-6– Walking

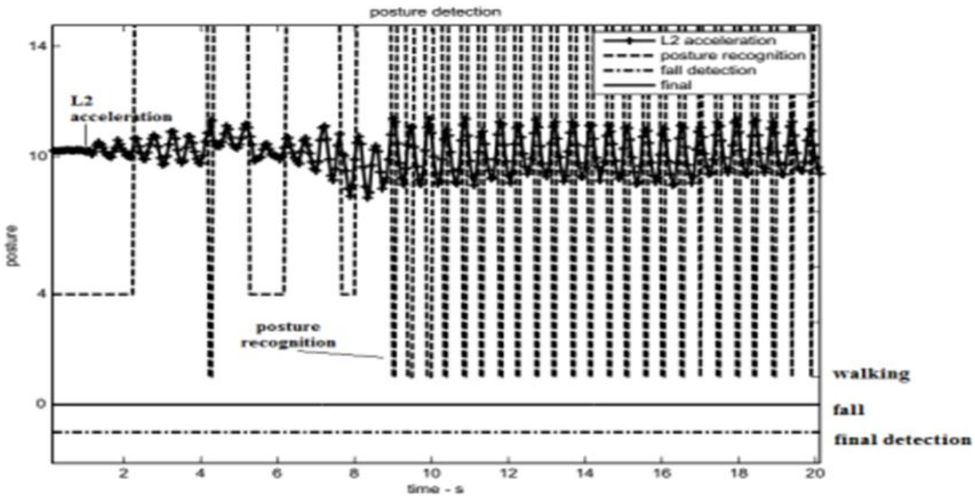


Figure 4-7– Walking posture

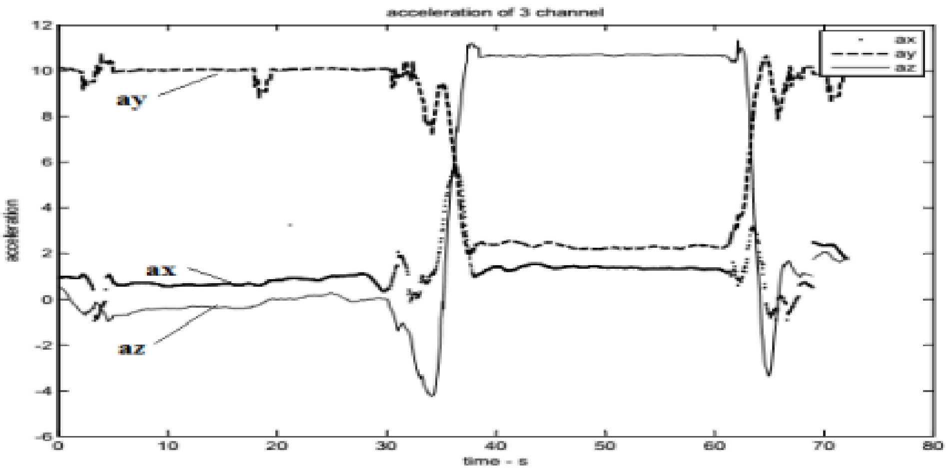


Figure 4-8– Standing and sitting

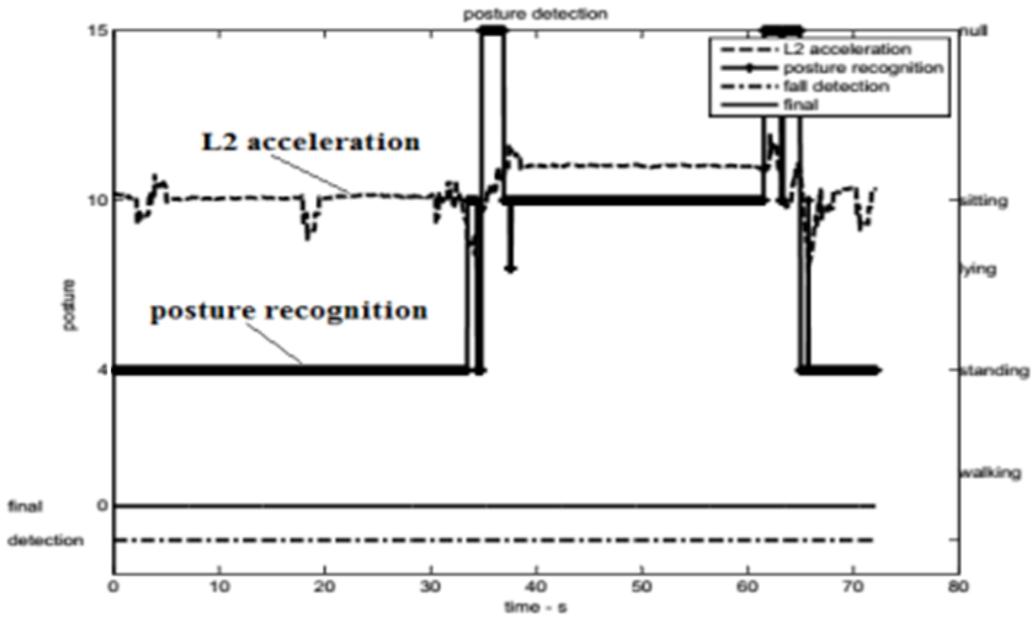


Figure 4-9– Recognition detection of standing and sitting

The Fig. 4-6 shows A_x, A_y, as in a walking status and A_y moves continuously around $10m/s^2$. The Fig. 4-7 is the posture of Fig. 4-6, the posture recognition result change continuously between the Null and the walking states. A_y is larger than threshold th_1, so the value of the posture detection equal $2m/s^2$ and no fall events were detected.

The Fig. 4-8 shows A_x, A_y, as in combined ADLs of standing and sitting. The Fig. 4-9 indicates the correct postures happening in this figure.

The accuracy of the system depends not only on the threshold, but also on the window size. Fig. 4-10 shows the result with window size of 10 samples and there are many fall events that were detected for this window size value. By increasing the window size to 20 samples, the result is relatively correct because the figure was clear at different time as shown in Fig. 4-11. While, Fig. 4-12 can not declare the actual fall event, then this window size can lose fall events information.

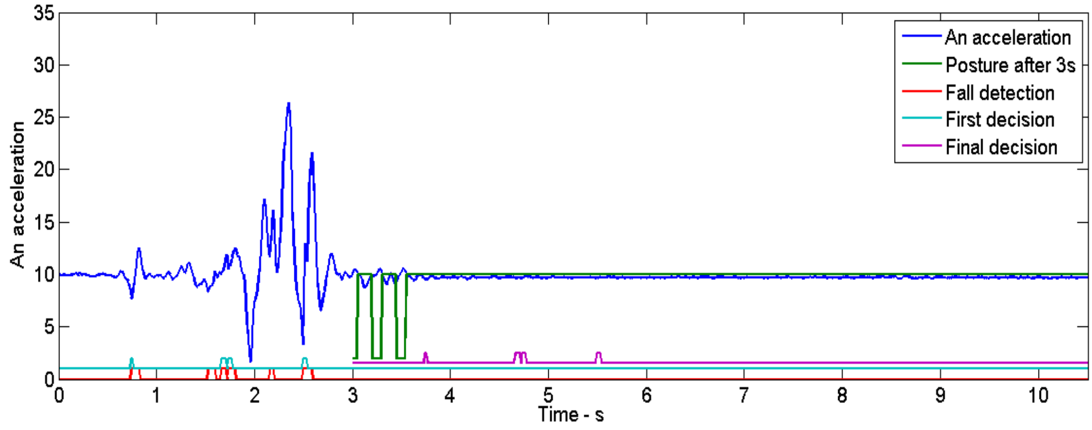


Figure 4-10– Fall detection with the window size of 10 samples and threshold th4 = 1.4 m/s2

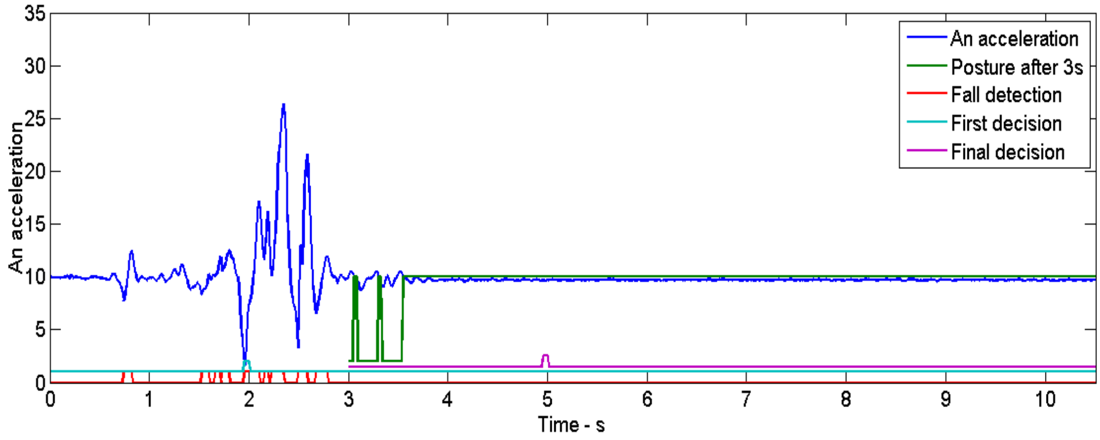


Figure 4-11– Fall detection with the window size of 20 samples and threshold th4 = 1.4 m/s2

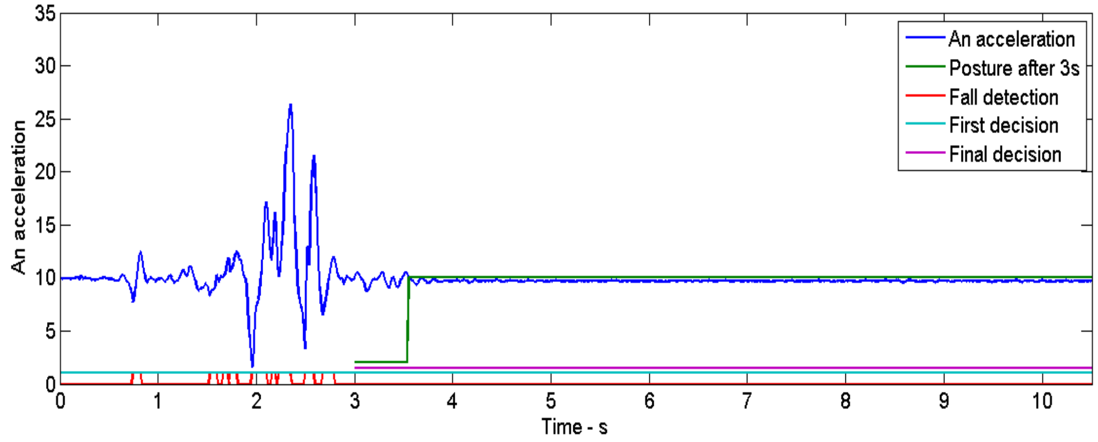


Figure 4-12– Fall detection with the window size of 30 samples and threshold th4 = 1.4 m/s2

We investigated and tested carefully to find out the suitable window size and threshold values because the accuracy of the system depends on both of them. The window size is 20 samples and threshold $th_4 = 1.4\ m/s^2$ are the best values that were chosen. The Fig. 4-13 shows the L_2 acceleration, posture recognition, fall detection and final decision without cascade posture recognitions, respectively. There are two fall events have been declared on the 13^{th} and 105^{th} second. The first decision is exactly fall event, but firefighters can self-stand up after falling and he was walking continuously. It is unnecessary to send out messages to their leader as relative members in this situation. Fig. 4-14 combined with cascade posture recognitions after 3 second was detected exactly fall event at 105^{th} second.

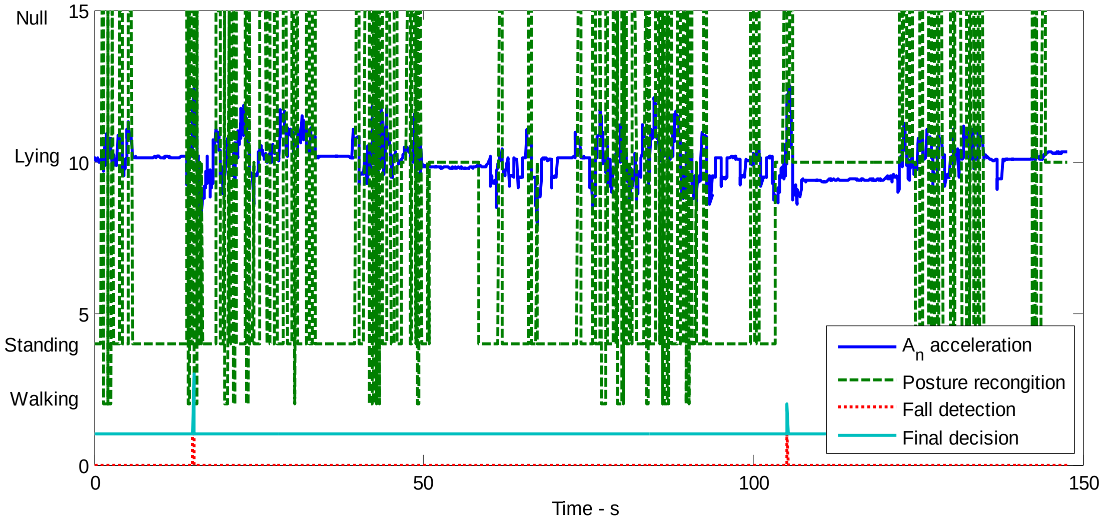


Figure 4-13– Fall decision without cascade posture recognitions [39]

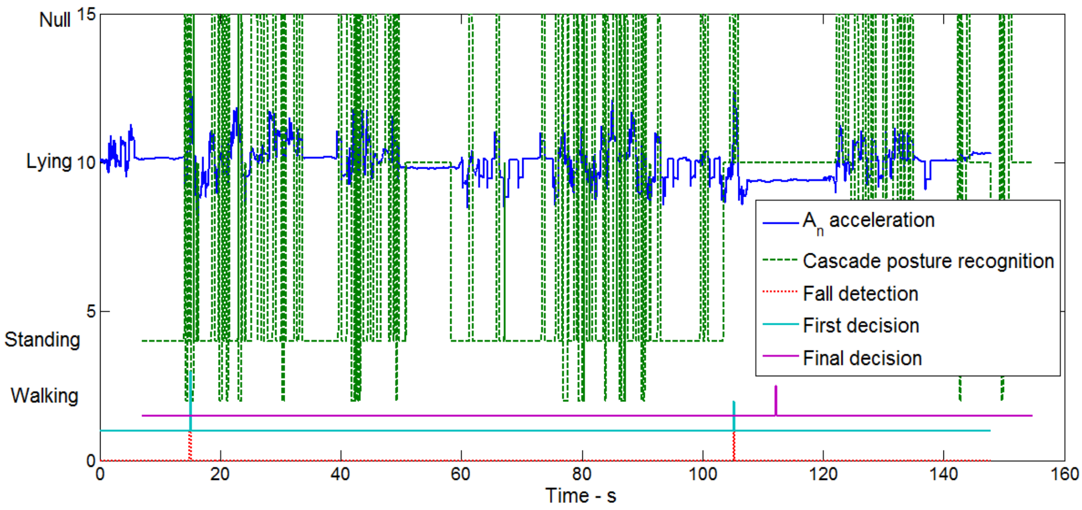


Figure 4-14– Fall decision with cascade posture recognitions [39]

4.2. The evaluation with other public datasets

In the next section, the authors will evaluate in a fair way our algorithms with our recorded and public datasets. The following table is the features of recorded and public data [22][23][24][25][26][27][28][29]

Public datasets are an important step towards allowing the self-evaluation of the current algorithms with various of fall events and ADLs activities. Based on the above datasets can be seen that the recorded and the public datasets have different samples that MobiFall and DLR datasets were recorded at 100 Hz, tFall dataset was recorded at 50 Hz while our dataset was recorded at 10 Hz. The reasons in choosing the frequency range at 10 Hz that [20] showed that the sampling at 100 Hz is not better than sampling at 50 Hz, it depends on the algorithms, the reasons for authors to analyzed and deciced to choose frequency at 10 Hz because the battery time life is very essential for firefighters on. Therefore, the author decided to choose 10Hz for the proposed system (energy consumption at 10 Hz and 100 Hz are 40 µA and 140 µA respectively).

Table 6: Features of the public and our recorded fall detection datasets

		Public data			**Our recorded data**
		DLR [24]	**MobiFall** [25]	**tFall** [26]	
Experiments	**No. Volunteers**	16	11	10	16
	Recorded from	Xsens MTx	Samsung Galaxy S3	Samsung Galaxy Mini	3-DOF (ADXL34)
	Position	Belt	Pocket	Pocket/Handbag	Waist
	Types of falls	Not specified	Forward-lying, front-knees-lying,	Forward straight, backward, lateral left and right,	Forward-lying, sideward and backward

			sideward-lying and back-sitting-lying	sitting on empty air, syncope and forward fall with obstacle	lying and back-sitting-lying
	Types of ADLs	Running, walking, jumping, standing, sitting, lying, getting up from lying or sitting, going down i.e.: from standing to sitting, walking upstairs, walking downstairs, transition between the activities	Standing, walking, jogging, jumping, stairs up , stairs down, sit chair, car-step in, car-step out	Not specified	Standing, walking, downstairs, upstairs, Sit chair or bed, getting up from lying or sitting, transition between the activities
Samples	**No. ADLs**	1077	831	7816	864
	No. Falls	53	288	503	168
	Sampling frequency	100 Hz	100 Hz	50 Hz	10 Hz
	Acc. Range	7g	2g	2g	2g

The minimum range (2g) is given by our recorded dataset, tFall and MobiFall. DLR dataset recorded originally at 7 g, these values were saturated to 2g because [20] also showed that no clear difference between 7 g and 2 g while they were compared these public datasets. Based on the exiting public datasets to evaluate the algorithm's performance is very essential because we can measure our algorithms' performance when we were fed with the different datasets and chose the best threshold values before it applied in the system to improve the system's performance and avoid any unexpected situations which we can not cover in real life.

To evaluate the proposed system, we use four following factors: True positive (TP) factor to determine if a fall occurred and the system can detect it. False positive (FP) factor to determine if a normal activity can be declared as a fall; True negative (TN) factor to determine if a fall-like event is declared correctly as a normal activity. False negative (FN) factor to determine if a fall occur, but the system cannot detect it [40].

After that, the sensitivity and the accuracy of the system can be evaluated by [40]:

$$Sen = \frac{TP}{TP + FN} \tag{5}$$

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

The proposed algorithms combined fall detection, posture recognition and cascade posture recognition, it can detect and distinguish most of the fall events with high sensitivity and accuracy around 92.8% and 96.8% respectively on our recorded dataset and the public datasets which recorded from other accelermeters such as: DLR, MobiFall and tFall datasets as well as. Based on the analysis and simulation results, we can choose the most suitable threshold values which can achieve quite well on sensitivity and accuracy performnace of DLR with 67.9% and 78%, MobiFall with 65.9% and 62.6%, the worst performance is tFall with 57.6% and 66.4% as Table 8. There are some fall positive detections on DLR, MobiFall and tFall datasets because the recorded positions were on two trousers pockets or in handbag, its were not fixed. Therefore, there were some false judgments in the public datasets. Based on the carefully analysis, simulation results on our recorded dataset and the public datasets for acceleration-based fall detection using MATLAB, we have chosen the most suitable value for these thresholds and the window size to work with our proposed system: the window size = 20 samples; th1= 0.7 m/s^2; th2 = 4.0 m/s^2; th3 = 1.1 m/s^2; th4 = 1.4 m/s^2 and th_5 = 33 ppm.

Table 7: The result of applying our algorithms to detect the fall events on other exiting datasets

Datasets	True positive (TP)	False positive (FP)	True negative (TN)	False negative (FN)	Algorithms results in validated	
					Sensitivity	Accuracy
Our recorded dataset	156/168	20/864	844/864	12/168	92.8%	96.8%
DLR dataset	36/53	231/1077	846/1077	17/53	67.9%	78%
MobiFall dataset	190/288	320/831	511/831	98/288	65.9%	62.6%
tFall dataset	290/503	2580/7816	5236/7816	213/503	57.6%	66.4%

Chapter 5

CONCLUSIONS

In this thesis, we have proposed a completed the algorithms, window size and threshold values for fall detection using a 3-DOF accelerometer, a MQ7 sensor, a micro controller, and the corresponding embedded algorithms. The posture recognition used to improve the fall detection; cascade posture recognitions have been introduced to significantly improve the accuracy of the fall detection system. Furthermore, the proposed system also found out the suitable threshold value of CO in the fire to protect firefighters' lives. The algorithms have firstly been simulated in MATLAB environment and re-programmed in C language for embedded in the micro controller. For the future study, we will integrate more sensors and improve the algorithms for online working to save the life of firefighters during the process of implementation tasks.

LIST OF AUTHOR'S PUBLICATIONS

1. Pham Van Thanh, Nguyen Thi Huyen Nga, Le Thi Thu Ha, Do Van Lam, Dinh-Chinh Nguyen, Duc-Tan Tran, (6/2016) "Development of a Real-time Supported System for Firefighters in Emergency Cases", the 6th International Conference on the Development of Biomedical Engineering, pp 341-344.

References

[1] Hylton J. G. Haynes, Joseph L. Molis (2015) U.S. Firefighter Injuries - 2014, NFPA Fire Analysis and Research.

[2] Fire Statistics Vietnam: 2014, at http://thoibaotaichinhvietnam.vn/pages/thoi-su-chinh-tri/2015-01-09/ca-nuoc-xay-ra-2357-vu-chay-trong-nam-2014-16989.aspx

[3] Fire Statistics Vietnam: 2015, at http://hanoimoi.com.vn/Tin-tuc/Xa-hoi/818206/nam-2015-xay-ra-gan-2800-vu-chay

[4] Rita F. Fahy, Paul R. LeBlanc, J. L. Molis (2015) Firefighter Fatalities in the United State – 2014, NFPA Fire Analysis and Research.

[5] http://blogs.wickedlocal.com/newton/2009/02/12/firefighter-richard-busa-survives-two-story-fall/#axzz4Cr3I48jk

[6] Nelson Bryner, Daniel Madrzykowski, David Stroup (2005) Performance of Thermal Exposure Sensors in Personal Alert Safety System (PASS) Devices, Building and Fire Research Laboratory, National Institute of Standards and Technology.

[7] Shih-Wei Yang, Shir-Kuan Lin (2014) Fall Detection for Multiple Pedestrians Using Depth Image Processing Technique, Computer Mehtods and Programs in Biomedicine, vol. 114, pp. 172-182, DOI: 10.1016/j.cmpb.2014.02.001

[8] Luštrek, M., Gjoreski, H., Kozina, S., Cvetković, B., Mirchevska, V., Gams, M. (2011) Detecting falls with location sensors and accelerometers, Twenty-Third IAAI Conference, pp. 1662-1667, DOI: 10.1109/MPRV.2015.84

[9] Hristijan Gjoreski, Mitja Luštrek, Matjaž Gams (2011) Accelerometer Placement for Posture Recognition and Fall Detection, Intelligent Environments (IE), 7th International Conference on. Nottingham: IEEE, pp. 47-54, DOI: 10.1109/IE.2011.11

[10] Wang, J., Zhang, Z., Bin, L., Lee, S. and Sherratt, S (2014) An enhanced fall detection system for elderly person monitoring using consumer home networks, IEEE Transactions on Consumer Electronics, 60 (1). pp. 20-29, DOI: 10.1109/TCE.2014.6780921

[11] Mark Goldstein, RN, BSN, EMT-P I/C, Royal Oak, Mich (2008) Carbon Monoxide Poisoning, Journal of Emergency Nursing, DOI: 10.1016/j.jen.2007.11.014

[12] Brian Y. Lattimer, Uri Vandsburger, Richard J. Roby (1998) Carbon Monoxide Levels in Structure Fires: Effects of Wood in the Upper Layer of a Post-Flashover Compartment Fire, Fire Technology, Volume 34, Issue 4, pp 325-355, DOI: 10.1023/A:1015366527753

[13] Barbara C. Levin, Erica D. Kuligowski, Toxicology of Fire and Smoke, National Institutes of Standards and Technology.

[14] Wang, J., Zhang, Z., Bin, L., Lee, S. and Sherratt, S , (2014) An enhanced fall detection system for elderly person monitoring using consumer home networks, IEEE Transactions on Consumer Electronics, 60 (1). pp. 20-29.

[15] Jose Carlos Castillo, Davide Carneiro, Juan Serrano-Cuerda, Paulo Novais, Antonio Fernández-Caballero and José Neves, (2014) A Multi-Modal Approach for Activity Classification and Fall Detection, International Journal of Systems Science, vol. 45, pp. 810–824.

[16] Babu, BR Prasad, SmitaPatil, and T. Gayathri. Baandhav, (2014) Smart Mobile Application for the Safety of Women and Elderly Population, International Journal of Innovative Research and Development, Vol.3(5), pp. 575-580.

[17] Petar Mostarac, Roman Malarić, Marko Jurčević, Hrvoje Hegeduš, AiméLay-Ekuakille, Patrizia Vergall, (2011) System for monitoring and fall detection of patients using mobile 3-axis accelerometers sensors, IEEE International Workshop on Medical Measurements and Applications Proceedings (MeMeA), pp. 456-459.

[18] P. Mazurek, Roman Z. Morawski, (2015) Application of Naĭve Bayes Classifier in a Fall Detection System Based on Infrared Depth Sensors, The 8th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, vol. 2, pp. 717 – 722.

[19] V. Bevilacqua, et al, (2014) Fall detection in indoor environment with kinect sensor, Proc. IEEE International Symposium on Innovations in Intelligent Systems and Applications, pp. 319–324.

[20] Raul Igual,Carlos Medrano,Inmaculada Plaza, (2015) A comparison of public datasets for acceleration-based fall detection, Medical Engineering and Physics.

[21] Medrano C, Igual R, Plaza I, Castro M. Detecting falls as novelties in acceleration patterns acquired with smartphones. PLoS ONE 2014;9(4):e94811.

[22] Korbinian F, Vera MJ, Robertson P, Pfeifer T. Bayesian recognition of motion related activities with inertial sensors. In: Proceedings of the 12th ACM International Conference on Ubiquitous Computing, Copenhagen, Denmark; 2010 26–29 Sep.

[23] Vavoulas G, Pediaditis M, Spanakis EG, Tsiknakis M. (2013) The MobiFall dataset: an initial evaluation of fall detection algorithms using smartphones. In: Proceedings of the 13th IEEE International Conference on Bioinformatics and Bioengineering (BIBE, Chania), p. 1–4.

[24] DLR dataset. Available online [accessed 05.02.16]: www.dlr.de/kn/en/Portaldata/27/Resources/dokumente/04_abteilungen_fs/kooperative_systeme/high_precision_reference_data/Activity_DataSet.zip

[25] MobiFall dataset. Available online [accessed 05.02.16]: http://www.bmi.teicrete.gr/index.php/research/mobiact

[26] tFall: EduQTech dataset. Available online [accessed 05.02.16]: http://eduqtech.unizar.es/fall-adl-data/

[27] Korbinian F, Vera MJ, Robertson P, Pfeifer T. (2010) Bayesian recognition of motion related activities with inertial sensors. In: Proceedings of the 12th ACM International Conference on Ubiquitous Computing, Copenhagen, Denmark.

[28] Vavoulas G, Pediaditis M, Spanakis EG, Tsiknakis M. (2013) The MobiFall dataset: an initial evaluation of fall detection algorithms using smartphones. In: Proceedings of the 13th IEEE International Conference on Bioinformatics and Bioengineering (BIBE, Chania).

[29] Fudickar S, Karth C, Mahr P, Schnor B. (2012) Fall-detection simulator for accelerometers with in-hardware preprocessing. In: Proceedings of the 5th International Conference on Pervasive Technologies Related to Assistive Environments, PETRA ’12 New York, USA.

[30] Microchip Technology Inc, " PIC18F2420/2520/4420/4520 Data Sheet", 2008, http://ww1.microchip.com/downloads/en/DeviceDoc/39631E.pdf.

[31] Sparkfun, "3-Axis, ±2g/ ±4g/ ±8g/ ±16g, Digital Accelerometer",https://www.sparkfun.com/datasheets/Sensors/Accelerometer/ADXL345.pdf.

[32] Robot Electronics, Using the I^2C Bus, http://www.robot-electronics.co.uk/.

[33] WWKong, UART – Universal Asynchronous Receiver and Transmitter, December 2010, http://tutorial.cytron.com

[34] Propox, SIM900 - The GSM/GPRS Module for M2M applications, http://www.propox.com/download/docs/SIM900.pdf

[35] Sparkfun, Techincal data mq-7 gas sensor, SparkFun Electronics, https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-7.pdf

[36] Aliexpress, MQ7 CO Carbon Monoxide Coal Gas Sensor Module, http://www.aliexpress.com/item-img/1Pc-MQ-7-MQ7-CO-Carbon-Monoxide-Coal-Gas-Sensor-Module-Newest/32418772592.html#

[37] Analog Devices, http://www.analog.com/en/design-center/reference-designs/hardware-reference-design/circuits-from-the-lab/cn0133.html#rd-commonvariations

[38] Nguyen Van Tinh, (2014) DETECTING HUMAN FALLS WITH A 3-DOF ACCELEROMETER, Bachelor's Thesis.

[39] Tran Duc Tan and Nguyen Van Tinh, (2014) Reliable Fall Detection System Using an 3-DOF Accelerometer and Cascade Posture Recognitions, Signal and Information Processing Association Annual Summit and Conference (APSIPA).

[40] N. Noury, P. Rumeau, A. Bourke, G. Laighin, J. Lundy, (2008) A proposal for the classification and evaluation of fall detectors, IRBM 29, pp. 340–349.