Research Article Open Access CrossMark

Development of digital gait monitoring software for diagnosis of neuromuscular disorder

Abstract

Analysis of human gait is very useful in several fields of study with applications such as prosthetics, orthotics and diagnosis of neuromuscular disorder. Modern gait assessment relies on three basic methods; namely qualitative observation, quantitative measurement and biomechanical analysis. This research carried out the development of a user friendly application gait monitoring software incorporated in a developed complete gait monitoring system, for quantitative diagnosis of neuromuscular disorder. Microchip MPLAB Integrated Development Environment 2000 (MPLAB IDE 2000) was used for the firmware development of the software. The language for the program was written using Visual C++. The interface on the system was designed with a pull-down menu bar that consists of File, Edit, View, Tools, Settings, Help, and Gait Monitor Controller. The application software employs Direct Plotting and Fast Fourier Transform (FFT) for frequency (or Harmonic) analysis, Peak values and RMS Values. The Cadence Frequencies were recorded and stored for future purpose which is one of the major contributions of the project. The programmes were compiled and tested successfully, running well on Window operating systems, serving as interface for monitoring gait analysis of individual with a case of neuromuscular disorder.

Keywords: software, gait, neuromuscular, disorder

Volume 4 Issue 6 - 2018

Adegbenro Sunday Ajani,[1] Olusoji Amos Ogunbode,[2] Adebayo Segun Adewumi,[2] Akanni Olaosebikan Aremu,[3] Oluwatayo Sandra Ajani,[4] Michel Olusope Alade,[2] Taiwo Faidat Jimoh[1]

[1]Department of Physics and Material Science, Kwara State University, Nigeria
[2]Department of Pure and Applied Physics, Ladoke Akintola University of Technology, Nigeria
[3]Physics Department, The Polytechnic Ibadan, Nigeria
[4]Department of Surgery, Ladoke Akintola University of Technology Teaching Hospital, Nigeria

Correspondence: Adegbenro Sunday Ajani, Department of Physics and Material Science, Kwara State University, P.M.B 1530, Malete, Ilorin, Nigeria, Email adegbanro.ajani@kwasu.edu.ng

Received: August 13, 2018 | **Published:** November 07, 2018

Introduction

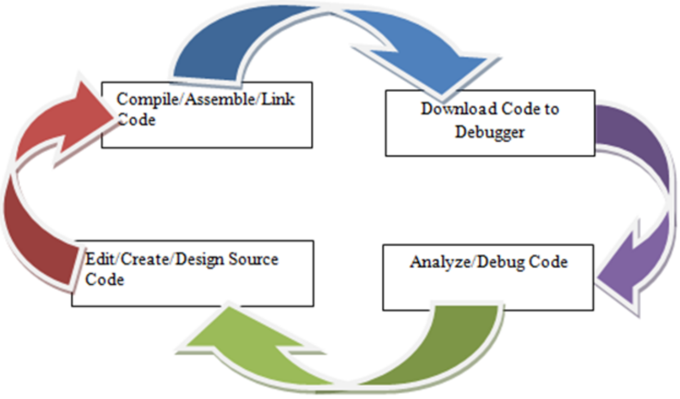
Human gait is observed as the pattern of walking an individual adopts during locomotion process controlled by the interaction of the nervous and muscular systems. It is categorized as either normal or pathological gait. Normal gait is attributed to the walking pattern of persons who have neither neuromuscular disorders nor articulate (joint) in their lower extremities, while the pathological gait refers to that of a person with a disorder or handicaps who respond to walking in an unconventional way. The analysis of gait is very applicable in several fields of investigation as well as application such as prosthetics and orthotics. In normal gait, the variation in most parameters will, in general be smaller than that in the pathological gait.[1] This variation requires development of flexible, robust and user friendly application software for the diagnosis of neuromuscular disorder and measurement for effective analysis of pathological gait.

The research development approach

Development of firmware is one of the significant tasks in the design of software for diagnosis neuromuscular disorders. The basic procedure employed in the design is described in Figure 1.

Software development cycle

The procedure for writing an application is often described as a development cycle, since it is rare that all the steps from design to demonstration can be done flawlessly the first time. Most times, codes written are tested and then modified in order to generate an application that performs correctly. An Integrated Development Environment (IDE) permits the embedded systems programmer to move through this cycle without the rigors of switching between arrays of tools. The Microchip MPLAB Integrated Development Environment (MPLAB IDE) was used for the firmware development of this work. MPLAB IDE is a Windows Operating System (OS) software program that runs on computers to develop applications for Microchip microcontrollers and digital signal controllers. It is called an IDE because it supplies a single integrated "environment" to develop code of embedded microcontrollers. By employing MPLAB IDE, functions were integrated which allowed the developer to concentrate on completing the application without the interruption of separate tools and different modes of operation.



Compile/Assemble/Link Code
Download Code to Debugger
Edit/Create/Design Source Code
Analyze/Debug Code

Figure 1 Firmware design cycle.

MPLAB IDE is a "wrapper" that organizes all the tools from a graphical user interface, automatically. Once codes are written, they can be converted to executable instructions and downloaded into a microcontroller to ascertain the performance. In this process, multiple tools were required: an editor to write the codes, a project manager to coordinate files and settings, a compiler or assembler that converts the source codes to machine codes and some sort of hardware or software that either connects to a featured microcontroller or simulates the operation of a microcontroller.

Implementation of embedded system design with MPLAB IDE

Developed system for embedded controllers is a system of programs that run on a desktop PC to write, edit, debug and program code, which fuses the intelligence of embedded systems applications into a microcontroller. MPLAB IDE runs on computers and contains all the components required to design and deploy embedded application system. The actual tasks for developing an embedded controller application are namely: Create high level design: Compile or assemble and link; and Test Codes and burn them into the device. These tasks are described as follows.

A. Create the high level design: From the features and performance desired, it was decided which PIC Microcontroller Unit or Digital Signal Processor (dsPIC) device is best suited to the application, and then design of the associated hardware circuitry was done. After determining which peripherals and pins to be used then the control of the hardware and the firmware were written. A language too, an assembler, which is translatable into machine code directly, or a compiler which permits more language like C-language for creating programs, should be selected to write and edit code. Assembles and compilers help make the code understandable, allowing function labels to identify code routines with variables that have names associated with their use, and with constructions that help organize the code in a maintainable structure.

B. Compile assemble and link the software: This section was introduced to convert the code into "ones and zeroes" machine codes for the PIC MCUs. Machine codes will become the firmware programmed into the microcontroller.

C. Testing and burning codes into device: A complex program usually does not work exactly the way as envisaged, hence "bugs" need to be removed from the program to get adequate results. The debugger helps to see the machine codes executed, related to the source code written, with the symbols and function identities from the program. Debugging gives easy experimentation with the codes to see the value of variables at different points in the program, and to perform "what if" checks, to change variable values and steps through routines. The codes "Burn" into the microcontroller and verifications that it executes correctly in the finished application was as well carried out. Step I is driven by the designer; MPLAB IDE helps with steps 2 to 4. The MPLAB IDE Programmer's Editor writes correct code with the language tools desired. The developer is aware of the assembler and compiler programming constructions and automatically "color-keys" the sources codes to ensure it is correct syntactically. Project Manager enables one to array the various files used in the application: source files, processor description header files and library files. Once the codes are built, one can control how rigorously code will be optimized for speed and size by the compiler and where individual variables and program data will be programmed into the system. If the language tools run into errors when building the application, the offending line is shown and can be "double clicked" to go to the corresponding source file for immediate editing. One has to press the "build" button to try again after editing. This write-compile-fix loop is executed many times for complex code such as the one developed in this research, even as the sub-sections were written and tested. MPLAB IDE goes through the loop with maximum speed, allowing it to get on to the next stage.

Once the codes are built, they need to be tested. MPLAB IDE has debuggers and free software simulators for all PIC MCU and dsPIC DSC devices to test the codes. If the hardware is not completed, it can begin by testing the codes with the simulator. The simulator can accept a simulated input into the microcontroller in order to model the firmware response to external signals. The simulator measures codes execution time, single step through code to observe variables and the peripherals, and trace the codes to generate a detailed record of the program performance. When the hardware is in a prototype stage, a hardware debugger, MPLAB ICE 2000 in circuit emulator, MPLAB REAL ICE in-circuit emulator, or MPLAB ICD 2 in-circuit debugger can be employed. These debuggers run the codes in real time on the application. MPLAB ICE 2000 emulator was used in this stage to replace the microcontroller in the target using a high-speed probe to give full control over the hardware in this design. MPLAB REAL ICE emulator and MPLAB ICD 2 debugger use special circuitry in-built Microchip MCUs with Flash program memory. They can "see into" the target microcontroller program and data memory. These debuggers can initiate and terminate program execution, helping to test the codes with the microcontroller in place on the application. When the application is run correctly, a microcontroller can be programmed with one of Microchip's device programmers such as MPLAB PM3 or PICSTART Plus. These programmers are used to verify that the finished codes run as designed. The flow chart for data fetching and data capture are described in (Figure 2) (Figure 3) respectively.

Results and discussion

The experimental results of the gait monitoring software for diagnosis are discussed in this section of the work.

Gait system software

Modern software design applies several techniques in order to achieve a program that is logically correct, efficient, easy to understand and maintain. Gait Monitoring System (GMS) was developed to acquire, analyze and show the gait results of any subject. This software was developed using Visual C++ and is running well on Window operating systems. It has a pull-down menu bar that consists of file, edit, view, tools, settings, help, and gait monitor controller.

Interfaces for the operation of the gait system software

The Gate Monitoring Software is developed in such a way that several interfaces incorporated can function with much simplicity without much ado from the user. These interfaces are dialog box for communication port and other component interfaces.

Dialog box for communication port

This section indicates the actual communication port that is opened for gait monitoring task. If the USB connection of the receiver is not properly done this interface displays ''unable to connect''. Figure 4 shows the dialog box for communication port.

Components interface for the gait monitoring system

This interface is the domain where actual data capturing is being carried out. This is explained through Figure 5

The Gait Monitoring System has the following features:

i. Get A Patient -This command receives the subject's information and creates a file such that all the information that relates to that subject is stored, analyzed and can be monitored by a physician.

ii. Clear Window -This is used to remove the content of the actively fetched data that shows on the monitor from the mobile unit.

iii. Terminate -This command is used to exit the capturing section.

iv. End Active Session - This command ends an active session in preparation for further analysis or capturing of another session.

v. Resume Active Session -This command restores the operation back to normal after being paused for one reason or the other.

vi. Get Offset Sensors - This provides dynamic offset nulling for Op-Amp circuit.

vii. Communication Port Settings -This is an interface for configuring the serial communication port for different settings (Baud rate, Flow Control, etc).

viii. Capture raw data -This command initiates the fetching of the transmitted raw data from the sensors. The fetched raw data will be shown on the interface window without hardware fault or failing radio link. This is very helpful in troubleshooting the system in case of hardware fault or failing radio link.

Menu bar for gait monitoring system

Menu bar for different commands of the Gait System is shown in Figure 6.

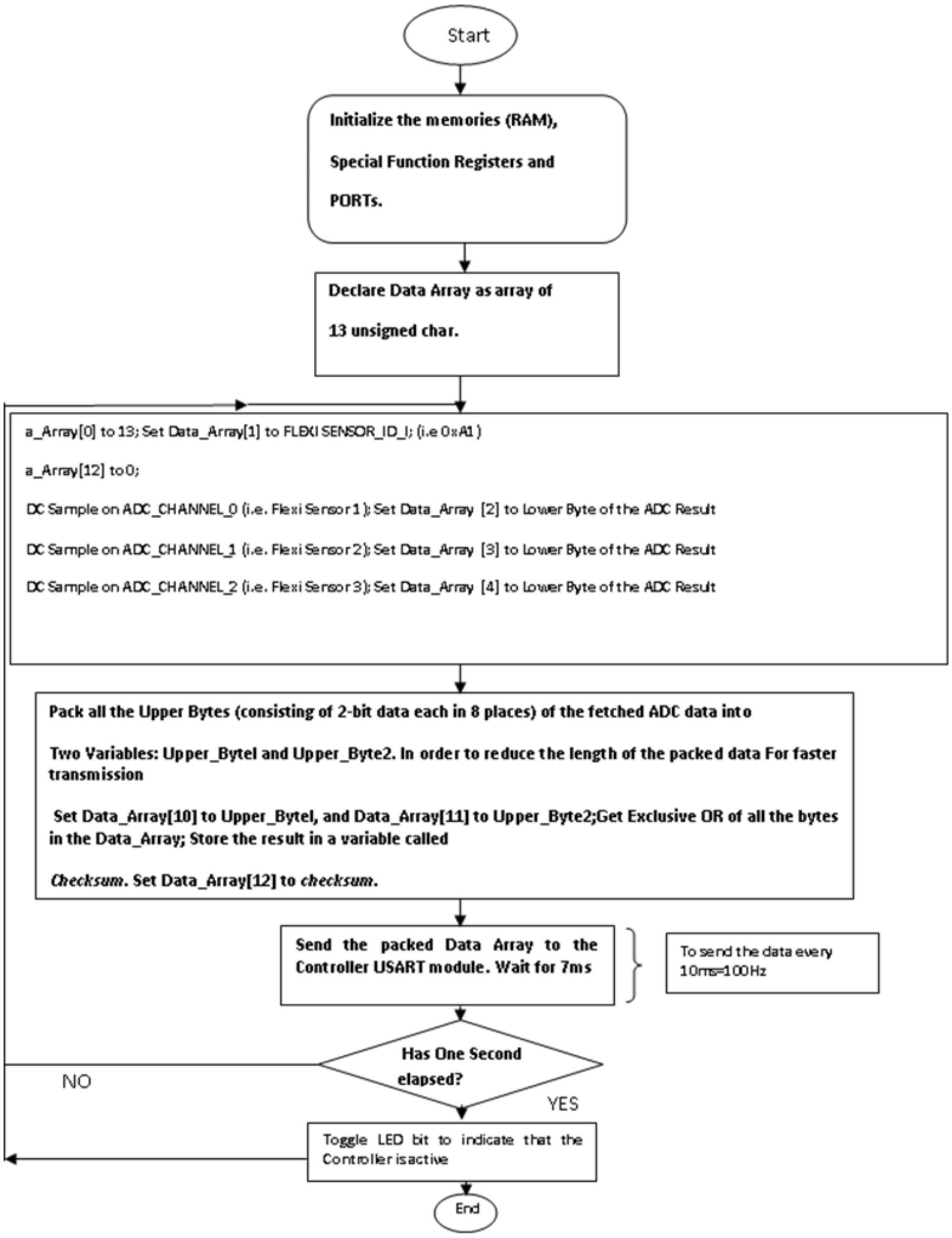

Start
Initialize the memories (RAM),
Special Function Registers and
PORTs.
Declare Data Array as array of
13 unsigned char.
a_Array[0] to 13; Set Data_Array[1] to FLEXI SENSOR_ID_I; (i.e 0xA1)
a_Array[12] to 0;
DC Sample on ADC_CHANNEL_0 (i.e. Flexi Sensor 1); Set Data_Array [2] to Lower Byte of the ADC Result
DC Sample on ADC_CHANNEL_1 (i.e. Flexi Sensor 2); Set Data_Array [3] to Lower Byte of the ADC Result
DC Sample on ADC_CHANNEL_2 (i.e. Flexi Sensor 3); Set Data_Array [4] to Lower Byte of the ADC Result
Pack all the Upper Bytes (consisting of 2-bit data each in 8 places) of the fetched ADC data into
Two Variables: Upper_Bytel and Upper_Byte2. In order to reduce the length of the packed data For faster transmission
Set Data_Array[10] to Upper_Bytel, and Data_Array[11] to Upper_Byte2;Get Exclusive OR of all the bytes in the Data_Array; Store the result in a variable called
Checksum. Set Data_Array[12] to checksum.
Send the packed Data Array to the Controller USART module. Wait for 7ms
To send the data every 10ms=100Hz
Has One Second elapsed?
NO
YES
Toggle LED bit to indicate that the Controller is active
End

Figure 2 Flowchart for data fetching and transmission.

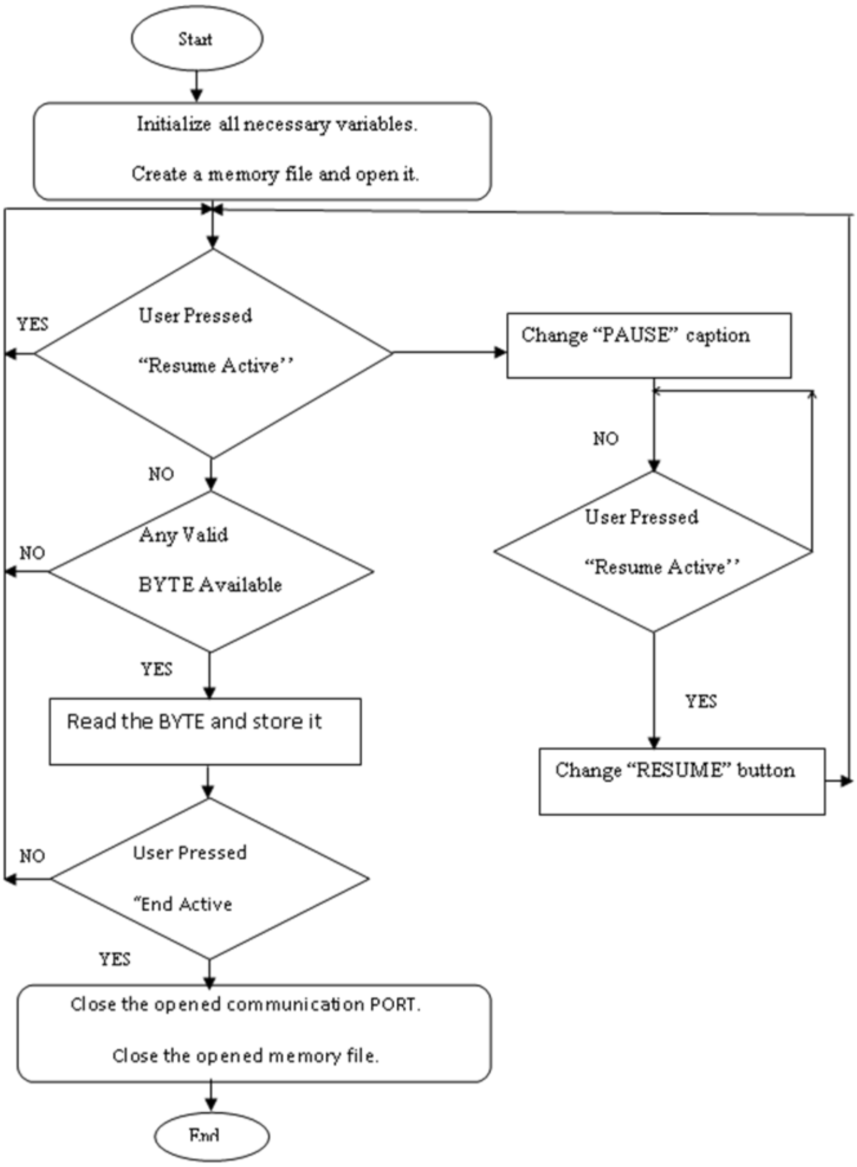
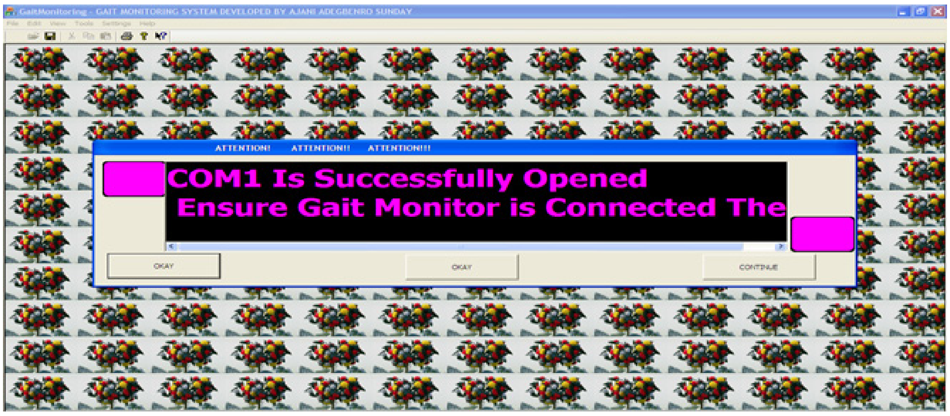


Figure 3 Flowchart for data capturing.


GaitMonitoring - GAIT MONITORING SYSTEM DEVELOPED BY AJANI ADEGBENRO SUNDAY
File Edit View Tools Settings Help
ATTENTION!! ATTENTION!! ATTENTION!!!
OKAY
OKAY
CONTINUE

COM1 Is Successfully Opened
Ensure Gait Monitor is Connected The

Figure 4 Dialog box for communication port.

Citation: Ajani AS, Ogunbode OA, Adewumi AS, et al. Development of digital gait monitoring software for diagnosis of neuromuscular disorder. *Int J Biosen Bioelectron.* 2018;4(6):249‒258. DOI: 10.15406/ijbsbe.2018.04.00135

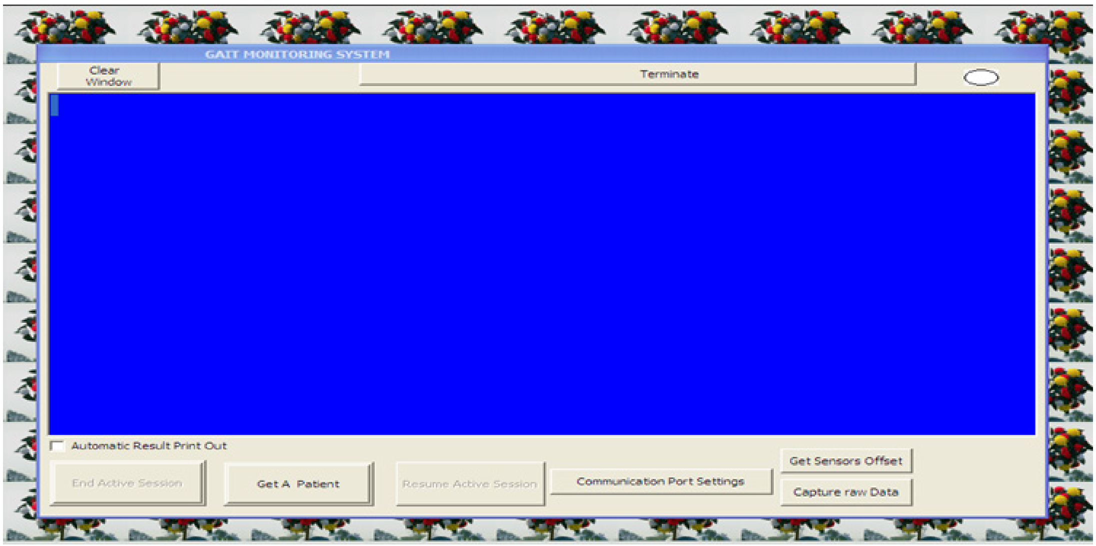


Figure 5 Components interface for the gate monitoring system.

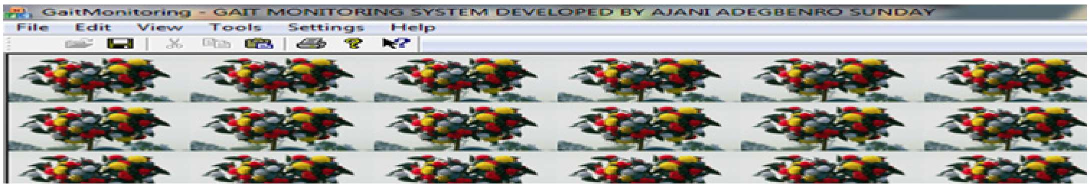

GaitMonitoring - GAIT MONITORING SYSTEM DEVELOPED BY AJANI ADEGBENRO SUNDAY
File
Edit
View
Tools
Settings
Help

Figure 6 Showing the menu bar of the gait system.

File menu

The file menu comprises of open gait file using fast Fourier transform (FFT) plot, open gait file using direct plot, save active result into gait file as, print, print setup, send and exit. Figure 7 shows the interface of the file menu of the gait system.

a) Open gait file: This command is meant to open the existing saved gait files of a particular subject either using Amplitude-Frequency (FFT) or Amplitude-Time Plot. Figure 8 shows the interface for Open Gait file of the existing patients' data in Gait folder on the computer.

b) Save active result into gait file As: This command is used to save the current file using a particular filename into a folder for a particular subject.

c) Print section: This is used to print the hardcopy of the gait monitored result of a particular subject.

d) Print setup: This command can be used to adjust the settings of the printer.

e) Exit: This command is used to close the gait monitoring software windows.

Tools menu bar

Tools menu bar, as described in Figure 9 contains the data capturing mode, data analysis mode and Gait monitor performance analyzer. The data analysis mode is subdivided into direct sample analysis plots and custom data analysis.

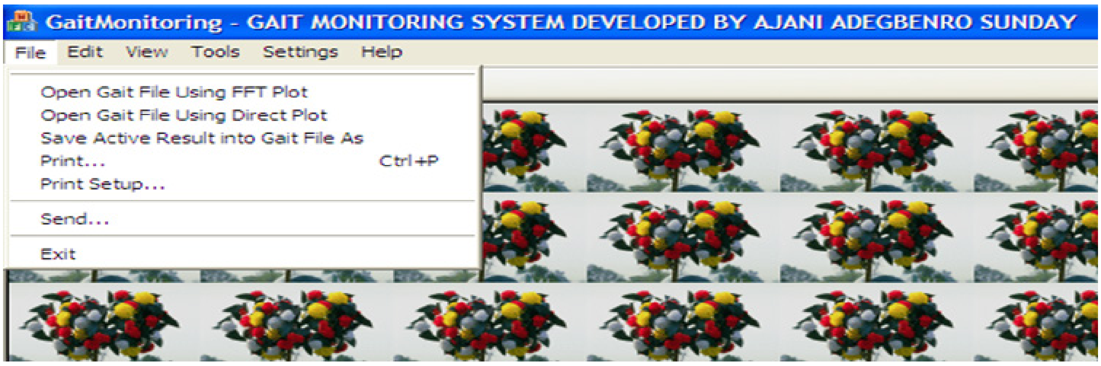
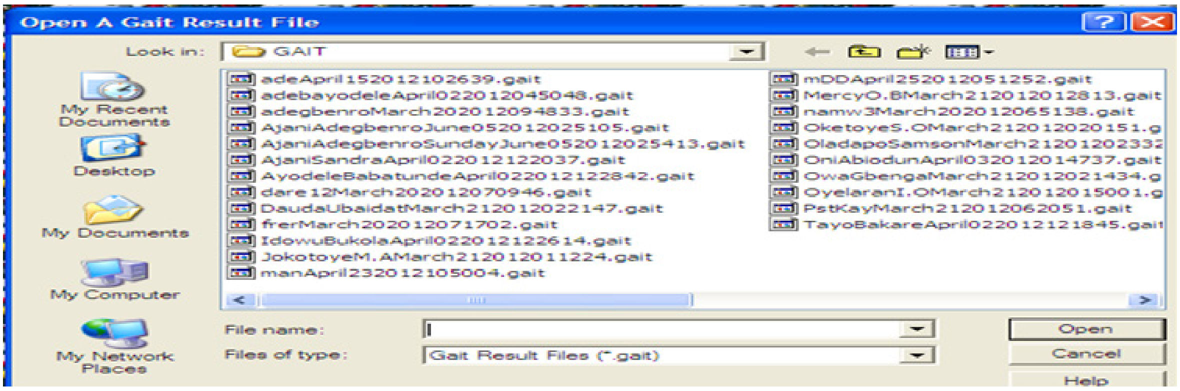

GaitMonitoring - GAIT MONITORING SYSTEM DEVELOPED BY AJANI ADEGBENRO SUNDAY
File
Edit
View
Tools
Settings
Help
Open Gait File Using FFT Plot
Open Gait File Using Direct Plot
Save Active Result into Gait File As
Print... Ctrl+P
Print Setup...
Send...
Exit

Figure 7 Interface showing the file menu of the gait system.

Citation: Ajani AS, Ogunbode OA, Adewumi AS, et al. Development of digital gait monitoring software for diagnosis of neuromuscular disorder. *Int J Biosen Bioelectron.* 2018;4(6):249–258. DOI: 10.15406/ijbsbe.2018.04.00135



adeApril1520121026З9.gait
adebayodeleApril022012045048.gait
adegbenroMarch202012094833.gait
AjaniAdegbenroJune052012025105.gait
AjaniAdegbenroSundayJune052012025413.gait
AjaniSandraApril022012122037.gait
AyodeleBabatundeApril022012122842.gait
dare12March202012070946.gait
DaudaUbaidatMarch212012022147.gait
frerMarch202012071702.gait
IdowuBukolaApril022012122614.gait
JokotoyeM.AMarch212012011224.gait
manApril232012105004.gait
mDDApril252012051252.gait
MercyO.BMarch212012012813.gait
namw3March202012065138.gait
OketoyeS.OMarch212012020151.g
OladapoSamsonMarch2120120233
OniAbiodunApril032012014737.gait
OwaGbengaMarch212012021434.g
OyelaranI.OMarch212012015001.g
PstKayMarch212012062051.gait
TayoBakareApril022012121845.gait

Figure 8 Software interface showing the open gait file.

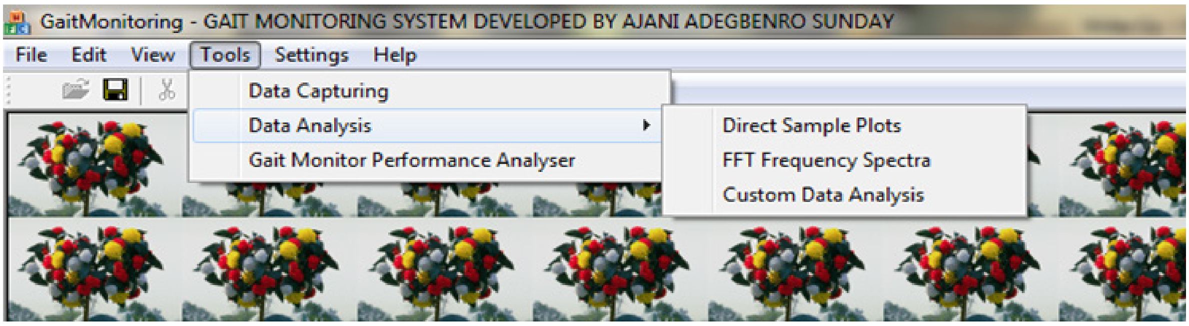


Figure 9 Interface showing the Tools Menu.

Data analysis

The first step here was to design a data recorder in tabular form, in which the measurements of individual can be recorded and saved through the data base program. The domain data form is captured by Table 1. Data collected from various subjects were run by the developed software and interpreted to make decisions. The format developed were both for normal and impaired subjects for conscious and unconscious moments of gait sessions. The details of the data collected are expressed (Table 2) (Table 3). The application software performs Direct Plotting and Fast Fourier Transform (FFT) for frequency analysis; and Peak values, RMS Values and the Cadence Frequencies were recorded and stored for further future investigation, which is one of the contributions of this research. This analysis however, requires a time consideration but it provides more precise information.

To interpret gait result, it is imperative to provide further investigational computations, which will estimate the parameters of interest and describe the physical process being measured. This provides analytical solutions that give the evaluation of the test procedures to help in making decisions. In the computation of the different parameters, scaling of the raw data is expedient since a large volume of data and problems of discontinuities are mostly encountered. The data processing deals with more than just plotting, a general and specialized subroutines and computer programs must be written to achieve the desired goals. Apart from the scaling, other operation that is usually performed on the data is Fast Fourier Transform for frequency or harmonic analysis. Frequency analysis is an essential operation in data processing because it gives information about the significant frequencies contained in the data. This analysis requires a considerable amount of computer time and memory but supplies accurate information easily.

Result obtained from the experiment

The obtained result from the experiments showed the peak value, the RMS value, and the mean frequency of each sensor for right and left legs. The software interface was designed in such a way that all the corresponding values are clearly shown on the screen. The corresponding waveforms of each sensor could be observed in either Amplitude-Time Plot or Amplitude-Frequency Plot using FFT. Figure 10 & Figure 11 represent samples of the waveforms of the ground-foot reaction force for pathological subjects in amplitude-frequency and amplitude-time modes, for the right and left legs. Each waveform reveals effects of the pressure on the sensor attached to the targeted points where the ground reaction force was measured. This target points could be the heel, metatarsals etc. According to Miyazaki[2] it has been shown that the total foot-force is the sum of the forces at the heel and the metatarsals part of the foot. It was also observed that the total foot-force defines two peak points (one on the heel and the other on metatarsal) and a trough, which corresponds to the period when the mass centre of the body is at its highest position. This occurs halfway through of the swing phase.

Citation: Ajani AS, Ogunbode OA, Adewumi AS, et al. Development of digital gait monitoring software for diagnosis of neuromuscular disorder. *Int J Biosen Bioelectron.* 2018;4(6):249–258. DOI: 10.15406/ijbsbe.2018.04.00135

Table 1 Format of the Record Designed to obtain Information from Pathological Subjects

S/N	Sample name	Date of data collection	Age	Weight (Kg)	Weight (m)	Cause if gait abnormality	Ground foot force readings											
							First reading (Initial)						Reading after a week					
							Left leg readings sensor 1			Right leg readings sensor 2			Left leg readings sensor 1			Right leg readings sensor 2		
							Peak value (mV)	RMS value (mV)	Mean Freq. (Hz)	Peak value (mV)	RMS value (mV)	Mean Freq. (Hz)	Peak value (mV)	RMS value (mV)	Mean Freq. (Hz)	Peak value (mV)	RMS value (mV)	Mean Freq. (Hz)
1																		
2																		
3																		
4																		
5																		
6																		
7																		
8																		
9																		
10																		

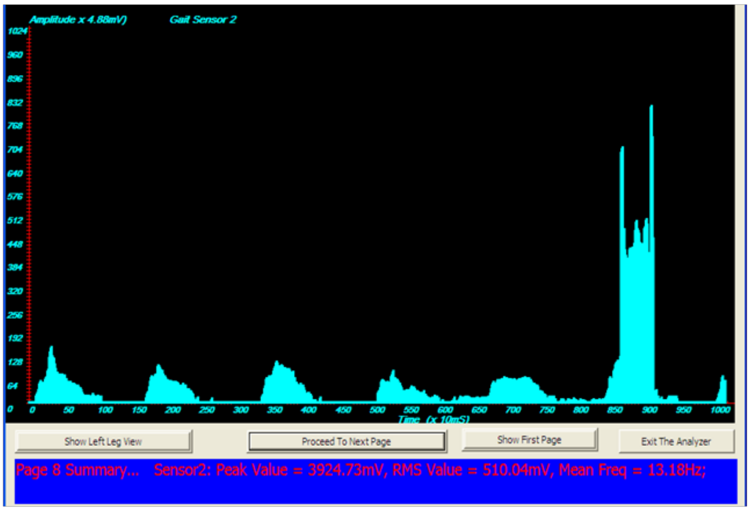
Table 2 Data obtained from the Gait Measurement of Pathological Subjects in March, 2017 at Private and Public Hospitals in Ogbomoso Metropolis, Ogbomoso, Oyo State, Nigeria, for a Period of 120 Seconds of Conscious Active Data Capturing

S/N	Sample name	Date of data collection	Age	Weight (Kg)	Weight (m)	Sex	Cause if gait abnormality	Ground foot force readings											
								First reading (Intial)						Reading after a week					
								Left leg readings sensor 1			Right leg readings sensor 2			Left leg readings sensor 1			Right leg readings sensor 2		
								Peak value (mV)	RMS value (mV)	Mean Freq. (Hz)	Peak value (mV)	RMS value (mV)	Mean Freq. (Hz)	Peak value (mV)	RMS value (mV)	Mean Freq. (Hz)	Peak value (mV)	RMS value (mV)	Mean Freq. (Hz)
1	Adeoye A.	16/3/17	55	66	1.72	M	Porosis	4408.3	3498.6	7.32	32010	3193.3	14.13	3708.3	2498.6	7.31	3701	2597.2	14.11
2	Ilugbile O.	16/3/17	45	72	1.68	M	Myelotis	2820.1	1270.2	9.24	11013	3445.4	8.54	3820.2	2270.2	9.45	3801	2465.7	8.52
3	Aba A.	16/3/17	22	65	1.65	M	Malacia	2868.2	1065.2	7.33	36764	2263.1	8.32	3762.1	2060.7	7.37	3775.6	2461.4	8.44
4	Sunday S.	16/3/17	61	75	1.74	M	Accident	5348.2	1970.6	9.84	31471	1465.5	8.76	2348.7	1971.1	9.89	2349.3	2070	8.91
5	Adesokan S.	16/3/17	21	57	1.72	M	Sarcoma	338.2	1892.6	6.66	29885	2403.4	14.75	3248.3	1892.9	7.47	3248.4	2003.7	14.81
6	David D.	16/3/17	19	82	1.67	M	Osteoma	4835.3	3520.6	11.46	36443	2444.4	13.15	3834.3	2520.6	10.49	3714.7	2550.2	13.21
7	Adewale A.	16/3/17	14	54	1.59	F	Accident	4521.5	3551.3	12.32	51012	2693.5	8.52	4321.2	2571.7	12.02	4501.2	2685.5	8.73
8	Ibiyemi A.	16/3/17	28	63	1.55	M	Accident	4014.3	3239.3	9.38	39333	1465.5	8.38	3814.5	2439.2	9.39	3734	2536.2	8.33
9	Sheu S.	16/3/17	44	72	1.74	M	Arthrits	2534.6	2233.5	6.24	28215	1702	6.89	334.8	1606.9	7.27	3221.8	1613.9	7.25
10	Ayodeji I.	16/3/17	32	77	1.66	F	Birth	5143.4	3584.3	9.26	41520	2734.9	8.23	4123.3	2486.8	9.96	4120.1	2556.8	8.22

Table 3 Data obtained from the Gait Measurement of Pathological Subjects in March, 2017 at Private and Public Hospitals in Ogbomoso Metropolis, Ogbomoso, Oyo State, Nigeria, for Period of 120 Seconds of Unconscious Active Data Capturing

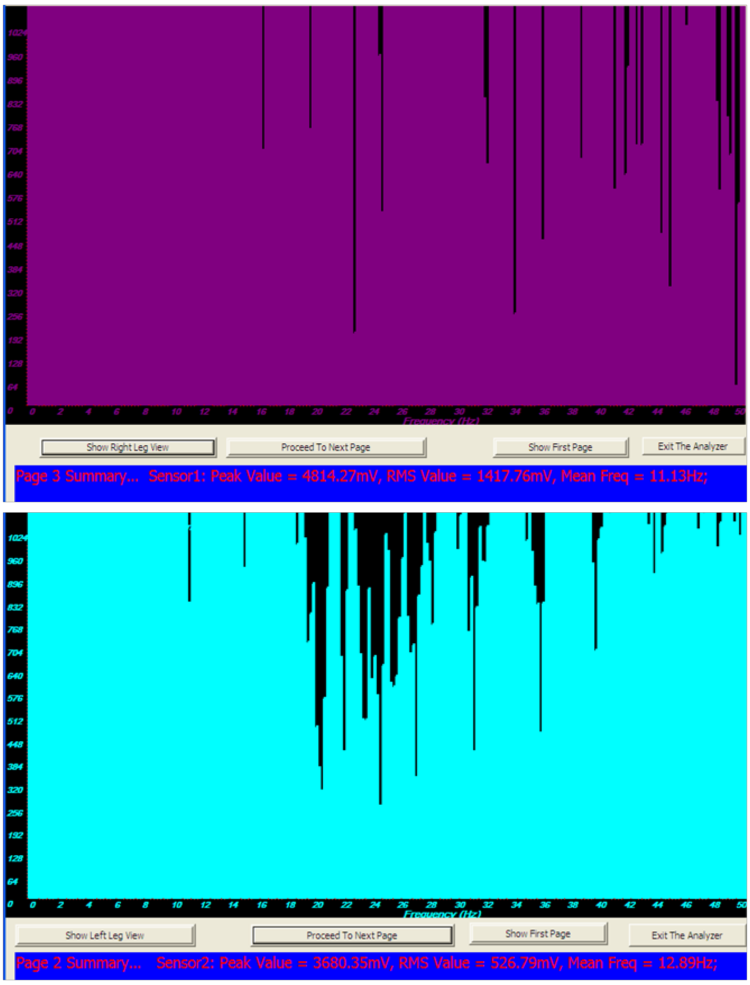
S/N	Sample name	Date of data collection	Age	Weight (Kg)	Weight (m)	Cause if gait abnormality	Ground foot force readings											
							First reading (Initial)						Reading after a week					
							Left leg readings sensor 1			Right leg readings sensor 2			Left leg readings sensor 1			Right leg readings sensor 2		
							Peak value (mV)	RMS value (mV)	Mean Freq. (Hz)	Peak value (mV)	RMS value (mV)	Mean Freq. (Hz)	Peak value (mV)	RMS value (mV)	Mean Freq. (Hz)	Peak value (mV)	RMS value (mV)	Mean Freq. (Hz)
1	Adeoye A.	16/3/17	55	66	1.72	Porosis	4408.1	3499.7	7.36	32017	3193.6	14.21	3708.8	2499.9	7.38	3701.5	2597.2	14.32
2	Ilugbile O.	16/3/17	45	72	1.68	Myelotis	2820.5	1270.8	9.28	11014	3445.8	8.61	3820.9	2271.8	9.49	3801.6	2465.7	8.73
3	Aba A.	16/3/17	22	65	1.65	Malacia	2868.7	1065.7	7.338	36767	2263.5	8.39	3762.7	2062.8	7.42	3775.9	2461.4	8.51
4	Sunday S.	16/3/17	61	75	1.74	Accident	5348.7	1972.3	9.89	31478	1465.9	8.84	2349.3	1973.5	9.99	2349.8	2070	8.96
5	Adesokan S.	16/3/17	21	57	1.72	Sarcoma	3348.8	1893.3	6.72	29882	2403.9	14.84	3249.2	1895.3	7.54	3248.7	2003.7	14.93
6	David D.	16/3/17	19	82	1.67	Osteoma	4835.7	3521.5	11.55	36447	2444.7	13.22	3834.9	2526.4	10.57	3714.9	2550.2	13.32
7	Adewale A.	16/3/17	14	54	1.59	Accident	4521.9	3551.7	12.37	51017	2693.65	8.59	4321.9	2573.6	12.83	4501.6	2685.5	8.77
8	Ibiyemi A.	16/3/17	28	63	1.55	Accident	4014.8	3239.8	9.41	39336	1465.7	8.45	3815.7	2454.6	9.48	3735.2	2536.2	8.38
9	Sheu S.	16/3/17	44	72	1.74	Arthrits	2535.5	2233.8	6.43	28216	1703.2	6.93	3335.5	1607.4	7.34	3223.5	1613.9	7.28
10	Ayodeji I.	16/3/17	32	77	1.66	Birth	5144.7	3584.8	9.34	41525	2735.6	8.24	4123.9	2487.3	9.98	4120.7	2556.8	8.31

Citation: Ajani AS, Ogunbode OA, Adewumi AS, et al. Development of digital gait monitoring software for diagnosis of neuromuscular disorder. *Int J Biosen Bioelectron.* 2018;4(6):249‒258. DOI: 10.15406/ijbsbe.2018.04.00135



Page 8 Summary... Sensor2: Peak Value = 3924.73mV, RMS Value = 510.04mV, Mean Freq = 13.18Hz;

Figure 10 Graph Sample obtained using Amplitude-Time Plot on an Impaired Subject.
10A) Sample of Graph Obtained Using Amplitude-Time Plot on an Impaired Subject's leg (Left).
10B) Graph Sample obtained using Amplitude-Time Plot on an Impaired Subject's leg (Right).



Page 3 Summary... Sensor1: Peak Value = 4814.27mV, RMS Value = 1417.76mV, Mean Freq = 11.13Hz;

Page 2 Summary... Sensor2: Peak Value = 3680.35mV, RMS Value = 526.79mV, Mean Freq = 12.89Hz;

Figure 11 Graph obtained using Amplitude Frequency Plot on Impaired Subject.
11A) Graph obtained using Amplitude-Frequency Plot on Impaired Subject's leg (Left).
11B) Graph sample obtained using Amplitude Frequency Plot on Impaired Subject's leg (Right).

Conclusion

The development of flexible and user friendly application software for the diagnosis of neuromuscular disorder measurement was achieved, with high speed data processing for storage compartment in the gait monitoring system. This research has been able to develop software that can handle qualitative gait analysis; the software was tested to handle the diagnosis of neuromuscular disorder. The result also shows that visual qualitative observation was unable to detect any improvement on the gait of the pathological patients sampled after a period of one week but quantitative observation with the developed software gives improvement or otherwise, though minute, on the gait of the subjects. The slight increase in the peak voltage of the patients' gaits supports this claim. Adequacy of this research work in monitoring and analysis of human gait is justified.[3–17] Implementation of this research has shown that the developed application software can be use to improve and monitor gaits of stroke patients, military amputee, accident victims and physically impaired people in hospitals and sports facilities.

Acknowledgements

None.

Conflict of interest

Author declares that there is no conflict of interest.

References

1. Jacobs NA, Skorecki J, Chanley J. Analysis of Vertical Components of Force in Normal and Pathological Gait. *J Biomech.* 1972;5(1):11–34.
2. Miyazaki S. A Four-Channel PDM/FM Telemetry System For Continuous Measurement of Foot-Force During Walking. *Tokyo Ika Shika Daigaku Iyo Kizai Kenkyusho Hokoku.* 1977;11:77–81.
3. Abu-Faraj ZO, Harris GF, Abler JH, et al. A Holter Type Microprocessor based, Rehabilitation Instrument For Acquisition And Storage of Plantar pressure Data. *J Rehabil Res Dev.* 1997;34(2):187–194.
4. Alade MO, Ajani AS. Performance evaluation of UHF wireless digital gait monitoring instrumentation for long range data transmission. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*. 2013;2(1):696–704.
5. Amin M, Cohen L, Williams WJ. Methods and Applications for Time Frequency Analysis. University of Michigan; 1993.
6. Brunnstornm Sigrie. *Clinical Kinesiology*. 2nd ed. USA: Davis Company; 1996:18–72.
7. Buraimoh-Igbo LA, Olaogun MO, Jubril AM. Measurement of Ground Foot Reaction Force to Determine Gait Asymmetry Using Computer-based telemetry System. *Rehabilitation Research and development reports*. 1997;34:53–55.
8. Chapman AE, Morrison JB, Hannah RE. Kinematic Symmetry of the Lower Limb. *Arch Phys Med Rehabil*. 1984;65(4):155–158.
9. Hodgins D. Human Motion Analysis, Medical Device Technology. Codicote, UK: European Technology for Business Ltd; 2008.
10. Jubril AM. Development of a Computer-Based Telemetry System to Determine Gait Asymmetry. Nigeria: M.Sc. Thesis, Obafemi Awolowo University; 2001.
11. Microchip Technology Inc. *PIC 16F87XA Data Sheet*. USA; 2001:1–17.
12. Morris SJ. Shoe-Integrated Sensor System for Wireless Gait Analysis and Real-Time Feedback. PhD Thesis Proposal, Harvard/MIT Division of Health Sciences and Technology and MIT Media Laboratory; 2002:10–32.
13. Oyebisi TO. Development of a Computer-Based telemetry System. Ph.D Thesis Obafemi Awolowo University; 1991.
14. Paradiso JA, Hsiao K, Benbasat AY, et al. Design and implementation of expressive footwear. *IBM Systems Journal.* 2000;39(3–4):511–519.
15. Porter D, Roberts VC. Review of Gait Assessment in the Lower Limb Amputee Part 1: Temporal and kinematic Analysis. *Clinical Rehabilitation*. 1989;3:65–74.
16. Simkin A, Stokes IAF. Characterization of the Dynamic Vertical Force Distribution Under the Foot. *Med Biol Eng Comput.* 1982;20(1):12–18.
17. Smidt GL, Wardsworth JB. Floor Reaction Force During Gait: Comparison of Patients with Hip Disease and Normal Subjects. *Phys Ther.* 1973;53(10):1056–1062.

Citation: Ajani AS, Ogunbode OA, Adewumi AS, et al. Development of digital gait monitoring software for diagnosis of neuromuscular disorder. *Int J Biosen Bioelectron.* 2018;4(6):249–258. DOI: 10.15406/ijbsbe.2018.04.00135