We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

154
Countries delivered to

Our authors are among the

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Middleware-Driven Intelligent Glove for Industrial Applications

Farouq Muhammad Aliyu and Basem Almadani

Additional information is available at the end of the chapter

Abstract

It is estimated that by the year 2020, 700 million wearable technology devices will be sold worldwide. One of the reasons is the industries' need to increase their productivity. Some of the tools welcomed by industries are handheld devices such as tablets, PDAs and mobile phones. However, handheld devices are not ideal for industrial applications because they often subject users to fatigue during their long working hours. A viable solution to this problem is wearable devices. The advantage of wearable devices is that they become part of the user. Hence, they subject the user to less fatigue, thereby increasing their productivity. This chapter presents the development of an intelligent glove, which is designed to control actuators in an industrial environment. This system utilizes RTI connext data distributed service middleware to facilitate communication over WiFi. Our experiments show very promising results with maximum power consumption of 310 mW and latency as low as 23 ms. These results make the proposed system a perfect fit for most industrial applications.

Keywords: RTI, DDS, RTI connext DDS, ubiquitous computing, Internet of Things, embedded systems

1. Introduction

In manufacturing industries, automated industrial systems that are intelligent and cheap are necessary in order to increase efficiency and productivity [1, 2]. Traditionally, such intelligent systems are wired. However, they are now wireless, thanks to the increase in reliability and flexibility of wireless communication systems [3]. Most wireless control systems are controlled through one type of handheld device or another. This is due to their computational

capabilities and energy efficiency. But recent developments in embedded systems have given rise to smaller yet more powerful embedded processors. As such, researchers are now turning their attention toward wearable devices.

In wearable sensor network (WeSN), sensor nodes are designed to gather physiological and kinetic data from the wearers' bodies [4]. These devices extend the users' intelligence and their ability to interact with the environment. They also take advantage of the intimacy between the user, the computer and the environment [5]. Wearable sensors can be helpful in industrial applications that require mobile users [6]. They provide more natural avenues of interacting with the system [7, 8]. In some scenarios handheld devices are inapplicable or too expensive [9, 10]. Furthermore, embedded devices are design specific. Therefore, they are optimized for a given function, unlike their handheld counterparts. This gives them upper hand when it comes to accuracy and reliability [11].

Anliker et al. [12] discussed the properties of a good WeSN. The authors stated that a good WeSN should have: (1)The ability to act in a proactive manner rather than depend on user's commands; (2) the ability to gather complex information regarding the user's interaction with the world; (3) high connectivity and mobility; (4) the ability to process wide variety of data at high speeds with low power consumption; (5) the ability to combine I/O devices and sensors from different parts of the body into one heterogeneous system; and (6) inconspicuous appearance such that it does not change the users' look in an unacceptable way or hinder the user from carrying out his/her normal day-to-day activities.

Sadly, there has been little research in WeSN with focus on industrial applications. This is due to the fact that distributed embedded systems at an industrial scale are not easy to maintain and control. In this chapter, we present the development of an intelligent hand glove for an industrial-based networked control system (NCS). NCS is a distributed control system (DCS) where sensors, actuators and controllers communicate over a network [13]. In order to ensure real-time performance, real time innovations (RTI) connext middleware is used. The middleware is used to manage the heterogeneous distributed system at real time. Also, the project is designed to investigate whether intelligent wearable sensors can replace the present handheld devices used in industrial NCS applications. We do believe that many users are in support and keen to use wearable devices than handheld devices. This is because they are lighter, cheaper, consume less energy and are more convenient to carry around.

2. Industrial application of embedded devices

Researchers were able to apply handheld devices such as smart phones and personal digital assistants to networked control system (NCS) [14]. Görlich et al. [15] developed an NCS where a Profibus-to-Bluetooth connection is used to control and parameterize a flow unit via Nokia 6280 mobile phone. Nonetheless, this system suffers from incompatibility, limited communication range and low bandwidth. Huang et al. [6] developed an oil and gas storage and transportation simulation system (OGSTSS). The proposed OGSTSS system uses wireless local area network (WLAN) and an android tablet PC in order to cover longer range and

obtain high bandwidth. The tablet PC communicates with programmable logic controllers (PLCs) via the PC station. This approach has some problems: first, the workstation between the tablet PC and the PLCs exerts delay overhead even though WLAN is used to speed up communication between the tablet PC and the PLCs and second, albeit tablet PCs are hand-held PCs, their size may discomfort and/or distract the user in some industrial applications.

Nowadays, the developments in embedded systems allow researchers to replace tablets with smaller footprint but equally powerful processing devices. Yang et al. [16] developed a device that traces the wearer's body position and his movements using PIC microcontroller, a tri-axial accelerometer and an RF transmitter. This device is intended for medical applications. It comes in handy for people with the tendency to syncope while they are alone. However, the system lacks enough computation power. Hence it cannot cope with the rigorous data processing of industrial applications. Single board computers (SBCs) like Beagle Bone, Intel Galileo and Raspberry Pi have small footprint yet they have large processing power [17, 18, 19]. This enabled the researchers to replace phones, tablets and computers with SBC in control systems and WeSN [20]. Wang et al. [21] developed a wearable sensor network using Raspberry Pi. The system is used for crowd-sensing. It uses transcoding algorithm to reduce resolution of video files. Khelil et al. [22] developed a body sensor network (BSN) using Raspberry Pi. The system uses cooking hacks e-health shield, which contains an airflow sensor for measuring wearer's respiratory rate, a pulse oximeter for measuring oxygen saturation of hemoglobin and position sensor for motion tracking. Jutila et al. [23] used Intel Galileo as a sink node for a wearable system that helps track children at school.

The aforementioned SBC systems revealed promising results. However, there exists incompatibility between the different types of SBCs. Therefore, a WeSN system can only contain one type of SBC. This problem is a serious setback in industrial applications of WSN and/or WeSN. This incompatibility forces engineers to stick to one type of device, which exposes them to the tendency of discontinued support by the producing company. In order to tackle this problem, [24, 25] proposed the use of middleware. A middleware is a software capable of hiding the complexity and heterogeneity between system components. It increases the ease of managing systems' resources and improves the predictability of the system [26].

Fok et al. [27] developed a fire tracking system which uses mobile agents based on Agilla middleware. This system allows a fire tuple to be injected into the network, where they clone themselves and move from one node to another until they form a perimeter bounding the fire. The system recorded huge success and also proves that a multipurpose and multitask WSN can be developed. Zhai et al. [28] developed a publish/subscribe middleware called wireless message-oriented system (WMOS) on top of TinyOS. The authors were able to reduce overall energy consumption of the network through the use of self-adaptive quality of service (QoS) services and content/topic model. Zug et al. [29] successfully applied a publish/subscribe middleware Family of Adaptive Middleware for autonomOUs Sentient Objects (FAMOUSO) to an industrial automation system using Katana robot. The robot publishes its coordinates and angles and it subscribes to movement, speed and emergency stop commands. This middleware can only work optimally in certain fixed conditions as it does not have quality of service (QoS) parameters. These parameters can be set to fine tune the performance of the middleware thereby improving the performance of the system at large. Furthermore, the aforementioned middlewares are

WSN middlewares. Hence, they cannot be used in industrial environment, where a heterogeneous system consisting of embedded devices and computers is readily found.

The proposed system presented in this chapter is a smart hand glove that accepts hand gesture as input, processes it using an ARM microprocessor (Raspberry Pi) and then publishes it using the RTI connext professional (publish/subscribe middleware) via WiFi. With the proposed architecture, we were able to resolve the data processing constraints, the large hardware footprint problem and the communication delays encountered by the industrial-based wearable systems.

3. System architecture

Figure 1 shows the proposed system architecture. The system architecture consists of the following four main components: (1) a wearable sensor named Al-yad. The main function of this component is to convert the physical movements into electric signals. The electric signal is then conditioned so that it can be processed by the 3.3 v ARM-6 microprocessor, Raspberry Pi in the case of this research; (2) a real-time innovations connext professional is selected as a middleware to mediate between the wearable sensor and other computing devices of the system. This middleware is selected because of its ability to work on both embedded devices and workstations. It also has a wealth of QoS that can be used to optimize communication as well as energy consumption; (3) a computing device can be any device capable of running
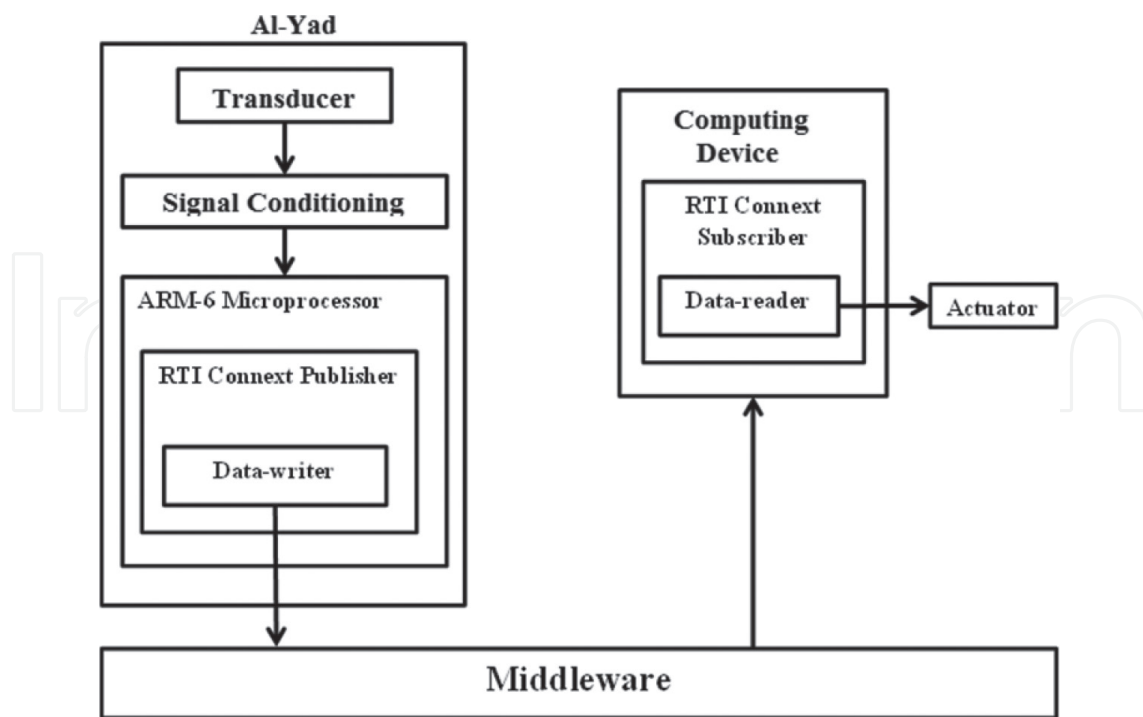

Figure 1. System structure.

connext professional (e.g., PC, Beagle Bone, Intel Galileo or Raspberry Pi); and (4) an actuator represents any industrial machinery and that is required to be controlled remotely.

3.1. Al-yad

Al-Yad is a smart hand glove made up of a combination of light sources (light emitting diodes (LEDs)) and light sinks (light dependent resistors (LDRs)). The 5 mm round red LED is chosen because the LDR is very sensitive to it and it consumes a maximum of 78 mw. A 50 mW, N5 AC-501085-type LDR is chosen for the proposed system because it has small-form factor and consumes small amount of energy.

The LDRs and the LEDs are in line of sight when fingers of the wearer are stretched straight. This gives a high signal (see **Figure 2**). As the user bends his fingers the light stops reaching the LDRs. This raises the resistance of the LDRs, thus, giving low output signal. In order to ensure no ambient light contaminates the system, the LDR is placed in a hollow cylinder of 1.45 cm in depth. Furthermore, the light source is pulsed at a frequency of 5 kHz, which is the highest frequency of PWM based on "wiringpi.h" implementation [30]. The highest frequency available is chosen because the higher the frequency, the smaller the capacitor required, therefore, the lighter the hand glove. Lemma 3.2 shows how the size of the capacitor C2 is derived. The pulsing light and the ambient light add up to form the simulated waveform, as shown in **Figure 3**. The direct current (DC) component of the wave
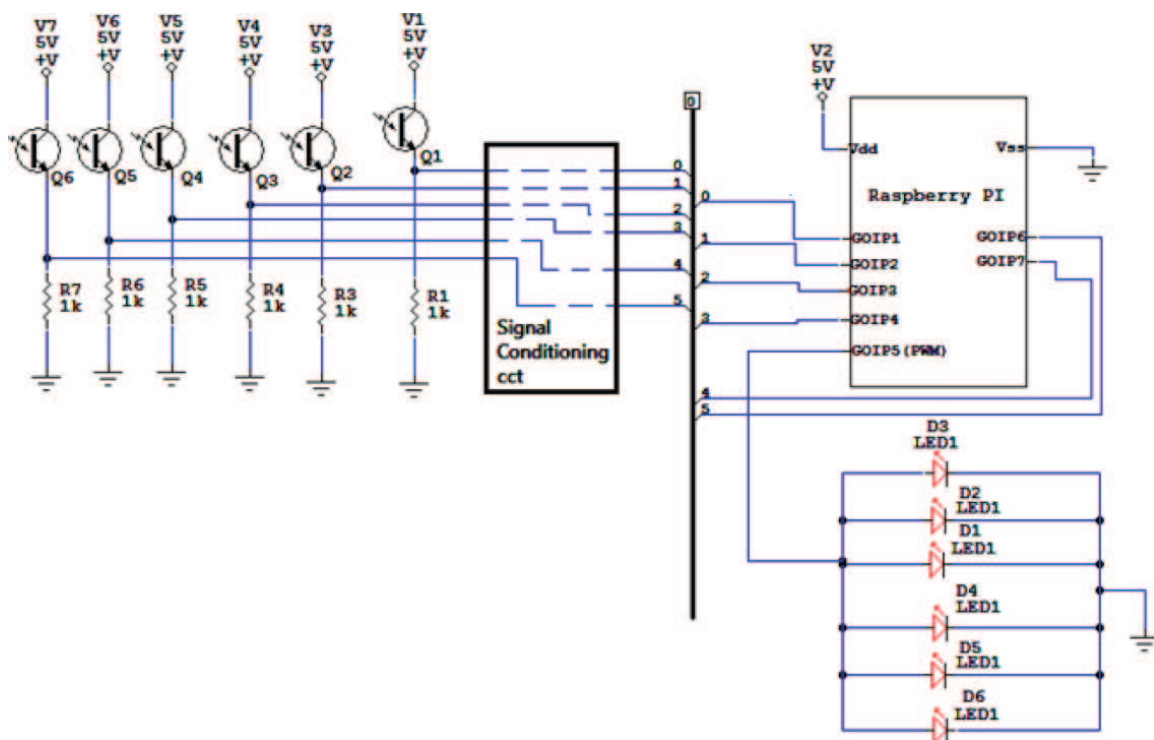

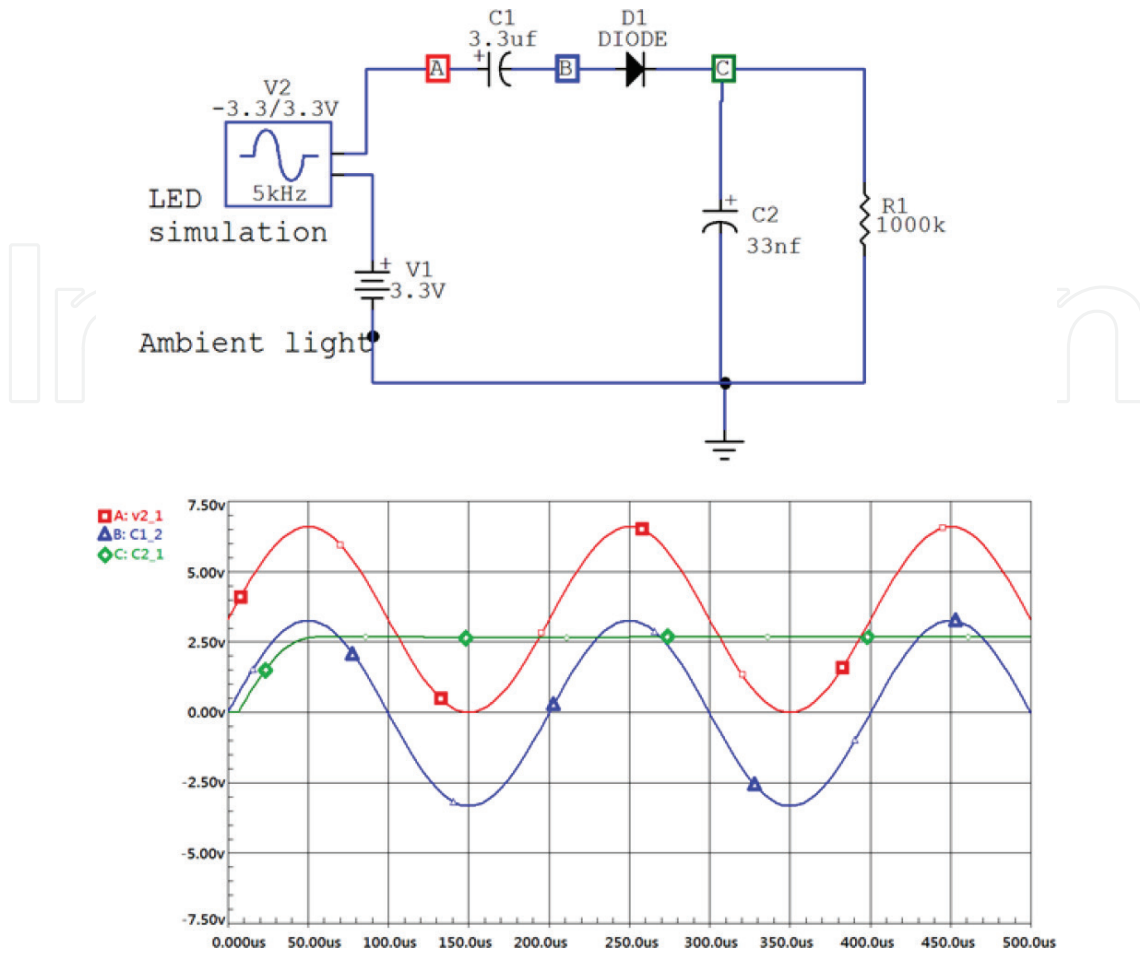**Figure 2.** Proposed systems complete circuit diagram.

Figure 3. Transducer circuit design.

is formed as a result of the ambient light. A coupling capacitor (C1) is added to the circuit in order to remove the DC component. Lemma 3.2 shows how the value of the capacitor is derived.

Lemma 3.1: Let γ = 0.1% be the ripple factor of the half wave DC and f = 5 kHz be the frequency of the pulsing light. Since the input impedance (R_L) of a 7414 Schmitt trigger is very high, let R_L = 1 MΩ. Hence, the capacitance of C2 capacitor can be calculated as:

$$\gamma = \frac{1}{2\pi \times f \times C2 \times R_L}$$

$$0.1\% = \frac{1}{2\pi \times 5000 \times C2 \times 1 \times 10^6} \tag{1}$$

$$C2 = 31.8310 \times 10^{-9}$$

$$C2 \approx 33nf$$

Lemma 3.2: Let X_{C1} be the capacitive reactance of the capacitor C1 and the pulsing light frequency (f) is 40 kHz. Since the input impedance of a 7414 Schmitt trigger is very high, then any value can be chosen for X_{C1}. For simplicity, we choose X_{C1} = 1. Hence, capacitance of C1 capacitor can be calculated as follows:

$$
\begin{aligned}
X_{C1} &= \frac{1}{2\pi \times f \times C1} \\
1 &= \frac{1}{2\pi \times 5000 \times C1} \\
C1 &= \frac{1}{2\pi \times 5000 \times 1} \\
C1 &= 3.1831 \times 10^{-6} \\
C1 &\approx 3.3\mu f
\end{aligned}
\tag{2}
$$

In addition, a rectifying diode (1 N4001) is added to the system in order to prevent negative voltage from reaching the Schmitt trigger (74LS14). 1 N4001 is used because it is the smallest diode that can be used without compromising the reliability of the system. A C2 capacitor is added to the system in order to reduce the ripples down to 0.1% so that the Schmitt trigger receives an almost constant direct current (DC) voltage. Finally, the Schmitt trigger (LM7414) smoothens the signal, making it a perfect DC, which is then passed to the microprocessor without any debouncing effect [31]. In order to ensure accurate data acquisition, the sampling frequency for the hand movement is chosen to be 5 Hz which is a little more than what is recommended by Yuan et al. [32]. Six sensors are placed in three fingers, thereby producing six bits. Two zeros are padded to the upper two bits, as shown in line 3–10 of Algorithm 1. The resulting 8-bit signal is then converted into an integer, which is then published as a command.

The processing unit is made up of a single board computer (SBC) [33]. Raspberry Pi is chosen because it has abundant libraries in Python, C, C++ and Java. The SBC is programmed to scan the glove and convert the gloves input to an integer and publish it through the middleware. Algorithm 1 illustrates the behavior of the SBC in the system.

Quality of service	Value
Durability	kind = VOLATILE
Reliability	kind = RELIABLE
	max blocking time = 100 ms
Liveliness	kind = AUTOMATIC
	lease duration = 1 s
History	kind = KEEP LAST
	depth = 5
Presentation	Access_scope = TOPIC
	Ordered_access = True
	Coherent_access = False
Owner	kind = EXCLUSIVE
Owner_Strength	5
Time_Based_Filter	Minimum separation = 0:2 s

Table 1. QoS used in the proposed systems' middleware.

3.2. Middleware

Research has shown that DDS middleware can achieve up to 75% success rate in mobile node applications [34]. This and the reason that it hides heterogeneity of a system encourages the use of middleware in our proposed solution. The middleware chosen for this task is RTI connext middleware. This middleware is chosen due to its strict compliance with the object management group (OMG) specification for data distribution service middleware [35]. The middleware is implemented according to Algorithm 1 using C++ programming language.

Algorithm 1: Pseudo-code for Publisher side

1: gesture[ ]=0
2: **while** (True) **do**
3:         gesture[0] = GPIO0
4:         gesture[1] = GPIO1
5:         gesture[2] = GPIO2
6:         gesture[3] = GPIO3
7:         gesture[4] = GPIO4
8:         gesture[5] = GPIO5
9:         gesture[6] = 0
10:        gesture[7] = 0
11:        int gestureCommand = 0
12:        **for** (int i = 0; i < 8; i + +) **do**
13:        gestureCommand=gestureCommand + gesture[i] X 2^i
14:        **end for**
15:        Publisher.datawriter.write(gestureCommand)
16:        wait for 0.2s
17: **end while**

The QoS is configured in both the hand glove and the actuator in order to ensure compatibility and efficiency. **Table 1** shows the quality of services used in the proposed system. The first four QoSs in **Table 1** are the recommended QoSs for alarm/event systems by [36]. The remaining QoSs are selected by the authors in order to ensure that the system behaves in a controller-actuator manner. The following is an explanation for the used QoSs:

1. *Durability* is set to "VOLATILE" so that data is not stored for future datareaders (i.e., targeted machine/actuator). This will prevent the targeted machine from executing any command that was issued before it was turned on, hence reducing the possibility of accidents.

2. *Reliability* of communication is set to "RELIABLE" and the maximum blocking time is left at the default value, which is 100 ms. The setting ensures that all commands published reach the targeted subscriber.

3. *Liveliness* allows datareader to know when the datawriter (i.e., the wearable sensor) is dead or disconnected. It is set to *AUTOMATIC* and the lease time is set to 1 s so that the datareader automatically checks the presence of the datawriter every second.

4. *History* QoS allows the datawriter to cache some samples. Here, the datawriter is allowed to queue only five samples.

5. *Presentation* is set such that the middleware will ensure that the samples are accessed in the order they were sent by the operator.

6. *Owner* is set to *EXCLUSIVE* so that only the hand glove with the highest owner strength can control the target machine.

7. *Owner_Strength* is set to five. This is the maximum number of gloves that can be used in the system. Therefore, only when a glove with the strength of five is down the glove with strength four takes over and so on.

8. *Time_Based_Filter's* minimum separation is the minimum time period before a new command is sent to the datareader. Since the sampling rate is 5 Hz, then the minimum separation = 1/(5 Hz) = 0.2 s.

3.3. Processing device

When a user wants to control a machine, he/she makes a gesture equivalent to the required command. The gesture is converted into a byte of command and is transmitted to a processing device. The processing device is connected to the machine to be controlled. It is responsible for subscribing and converting the sent signal into commands usable by the target machine. Any device capable of using RTI middleware can be used as the processing device. In the case of our experiment a PC workstation with Intel Core i5 microprocessor and Atheros AR8161/8165 Gigabit Ethernet Controller is used.

4. Experiments

The sensor casings are designed using Blender 2.6 3D modeling software and they are printed with Replicator2 3D printer. LEDs and LDRs are placed in the 3D printed casings. They are connected in a manner, as shown in **Figure 4**. The conditioning circuit is then connected to the SBC/Raspberry Pi, where the data is processed and transmitted to the actuator. In our experiments, a PC is used as the actuator. A RTI subscriber is installed on a PC and publisher on the Raspberry Pi. The gesture commands sent by the Raspberry Pi are received by the PC and displayed on the monitor. It is with this setup that all experiments are carried out.

In this research throughput, latency, power consumption and comfort of the system are evaluated. The setup in **Figure 5** is used to measure both latency and throughput. RTI's performance toolkit, "Perftest," is used [37]. To test latency of the system, packets are sequenced and time stamped before they are transmitted to the datareader (receiver). The packets are immediately re-routed (i.e., bounced) back to the transmitter where the round trip time (RTT) is calculated. Similarly, the receiver counts the number of packets received per second in order to calculate the throughput. Finally, the transmitter reports the latency of the system while the receiver reports its throughput. As illustrated in the setup in **Figure 5**, Perftest was executed as a publisher on the Raspberry Pi and as a subscriber on the PC. The latency and
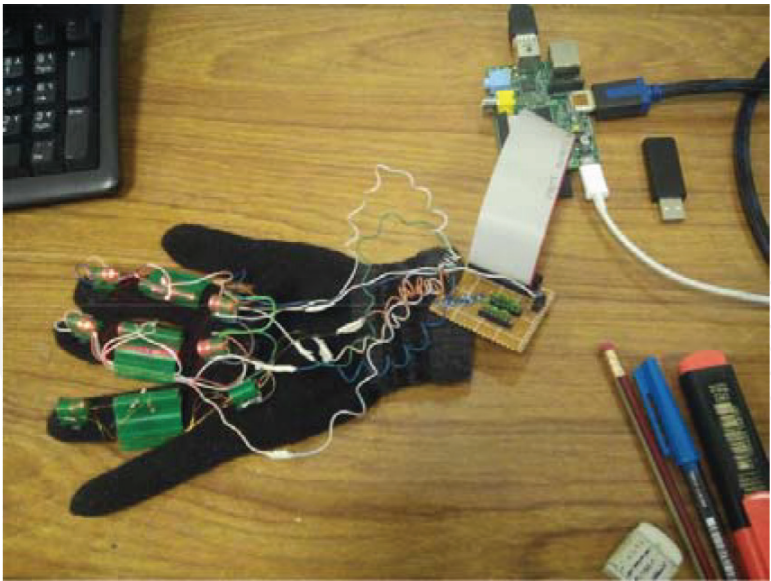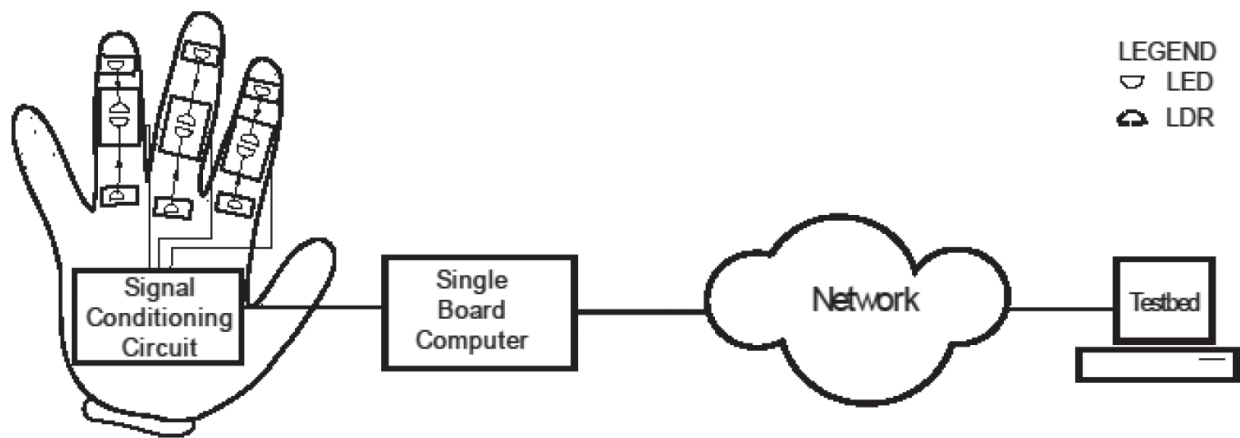
Figure 4. Proposed system.

Figure 5. Experiment setup for measuring latency and throughput.

throughput of the system is then measured. The payload size for the experiment is 28 bytes. This packet size is chosen because the proposed system only sends integer values between 0 and 63 and the minimum allowable payload size in RTI connext is 28 bytes. Furthermore, the experiment is carried out using QoS settings and without QoS, so as to investigate the cost of applying the quality of service. **Figures 6** and **7** show the results obtained for latency and throughput of the system, respectively.

We performed an experiment to ascertain the response time of the transducer. This is necessary because LDRs naturally have a very slow response. Therefore, it is important to know what quota of the total delay is contributed to it. In this experiment, the hand glove is connected to an oscilloscope and the output voltage (Vo) is fed to the Raspberry Pi and to the oscilloscope. The light is blocked and the response time is obtained through the oscilloscope, as shown in **Figure 8**.

The setup in **Figure 6** is used to measure the energy consumption of the whole system. A 1 resistor is connected in series with the proposed system so that the current passing through the system and the resistor is the same. As such, the voltage drop across the resistor follows
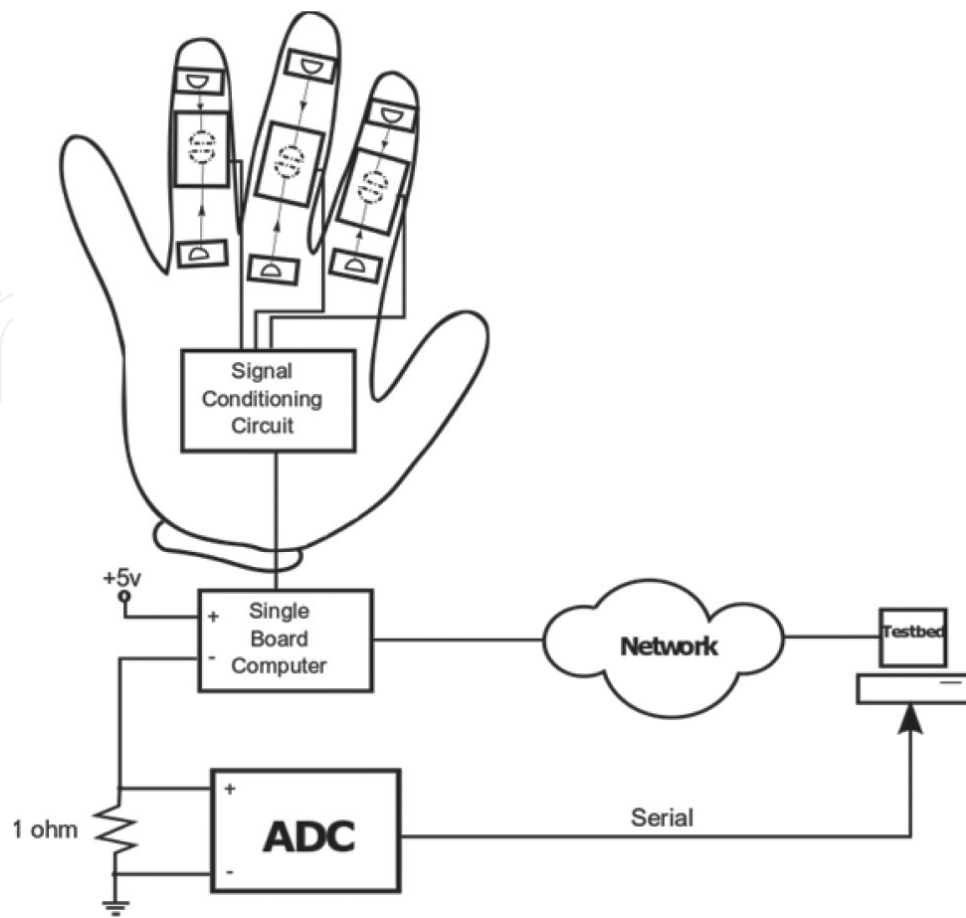
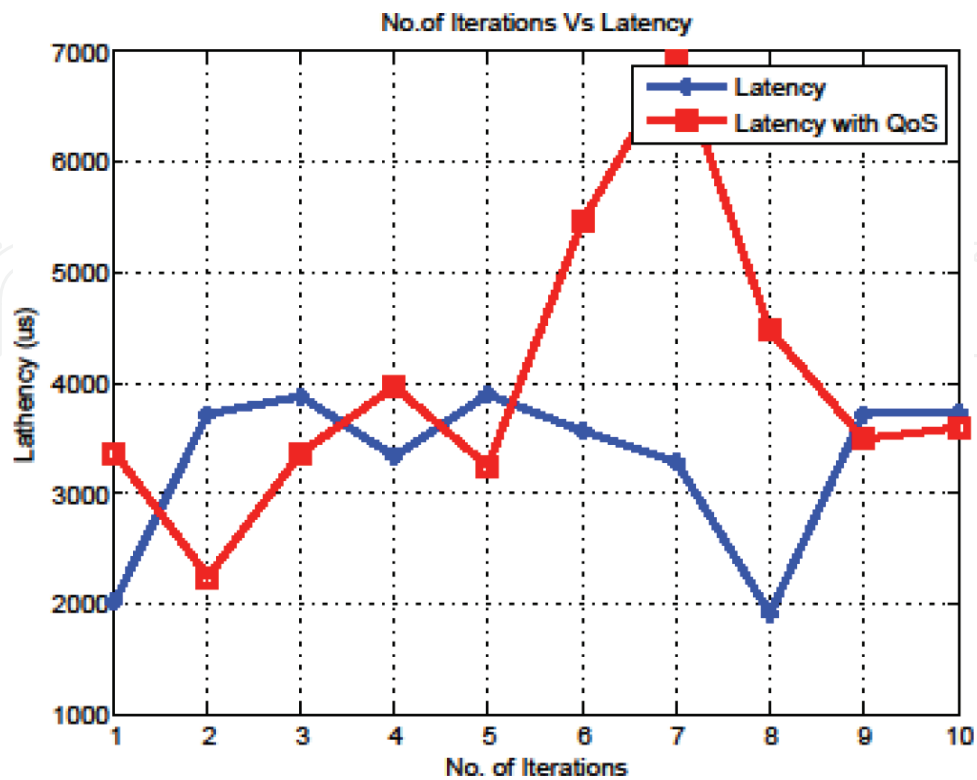**Figure 6.** Experiment setup for measuring energy consumed by the system.

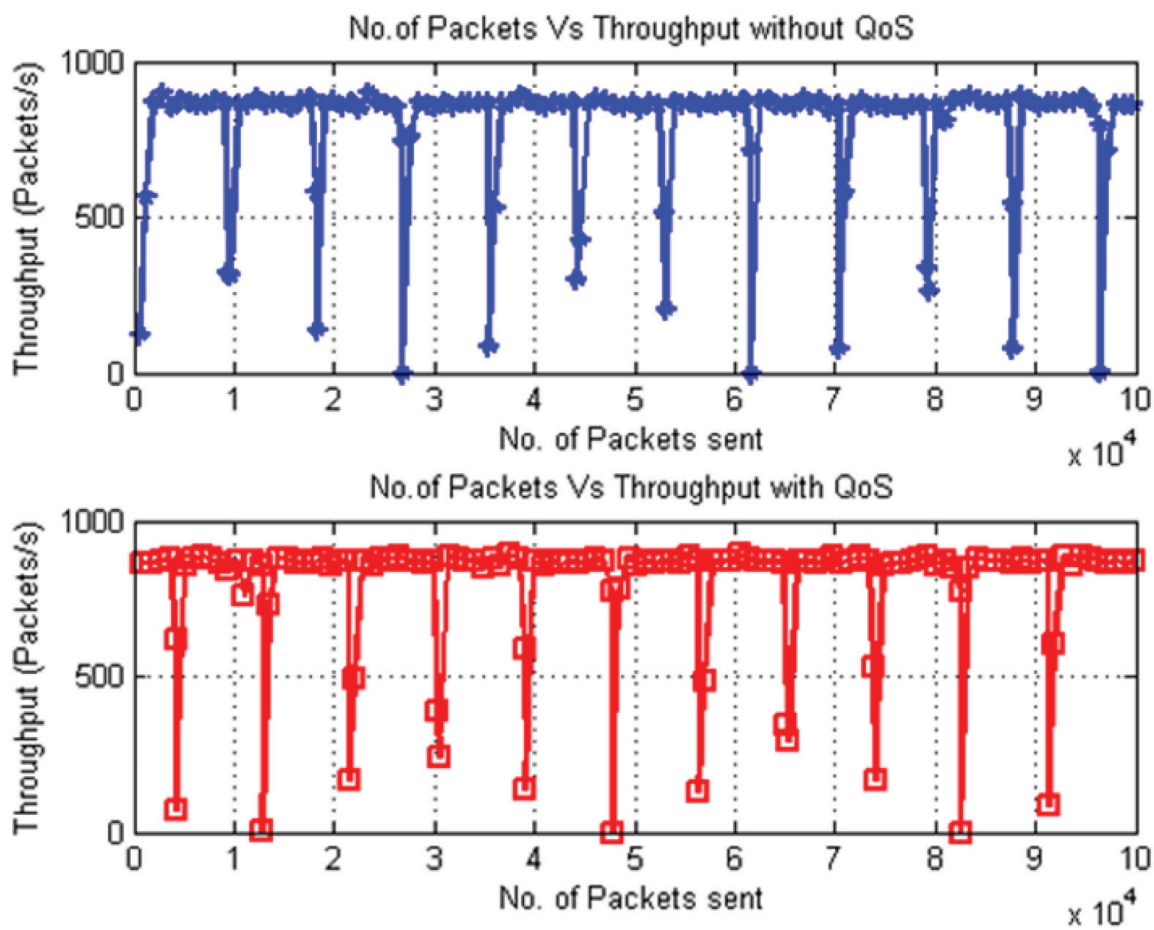**Figure 7.** Latency of the proposed system.

Figure 8. Throughput of the proposed system.

the same behavior as the current flowing through the circuit. To measure the actual current (I) flowing through the system, we used Eq. (3). The assumption in Eq. (3) is safe due to the fact that the Thevenin's resistance of the system is far greater than 1. An analogue to digital converter (ADC) is connected across the 1 resistor. It samples the voltage drop across the resistor every 200 ms. The acquired data is then transmitted to a computer via USB, where it is stored. The current flow through the system is recorded for two transmissions after which the WiFi dongle is immediately removed so as to record the overhead of the dongle itself.

$$I = \frac{V}{R}$$
$$I = \frac{V}{1} \therefore I = V$$

(3)

Finally, a survey is contacted to investigate the ergonomics of the proposed system. Questionnaires are given to eight people. The questionnaire investigated the six attributes of comfort proposed by James and Chris [38]. Their comfort scale consists of the following attributes: emotion, attachment, harm, perceived change, movement and anxiety. The participants are asked to use the hand glove and then they are asked to answer 12 questions. The questions

require them to score the attributes between 0 and 10. The score 0 represents the least while 10 represents most severe discomfort on the user.

5. Discussion of results

The latency of the system with and without QoS also showed a minor difference of 1 ms. This can be seen from **Figure 7**. The performance test toolkit calculates an approximation of the latency by dividing round trip time (RTT) by 2. Since this is not a reliable answer, the software also reports the average and the standard deviation, as shown in **Table 2**. The transmission's throughput and latency results obtained show that QoS has little effect on the performance of the proposed system. This is attributed to the small size of the packets transmitted.

Figure 8 shows that there is a little difference in the throughput of the system with or without using QoS. It is found that the average throughput of the system with QoS is 769 and 778 Mbps without QoS. In both cases no packets are lost during the 100,000 packet transmission. Given the fact that we are only sending commands in the form of integers over the network, this result is highly acceptable.

Moreover, the total latency of the system is the sum of the latency due to transmission and the inherent delay of the electric circuit that reads the hand gesture and places it on a bus for the SBC to read. **Figure 9** shows us that both the rising time of the signal and its falling time are approximately 20 ms. Therefore, the total delay of the system is 23 ms, as calculated using Eq. (4).

$$t_{delay} = t_{glove} + t_{trans} \quad t_{delay} = 20 + 3.297 \quad t_{delay} = 23.3 \text{ ms} \tag{4}$$

Figure 10 shows the current (top) and the moving average of the current (bottom) flowing through the system during and after data transmission. Immediately after the last transmission, the WiFi dongle is removed, hence, the spike at 375th s. However, no change in the

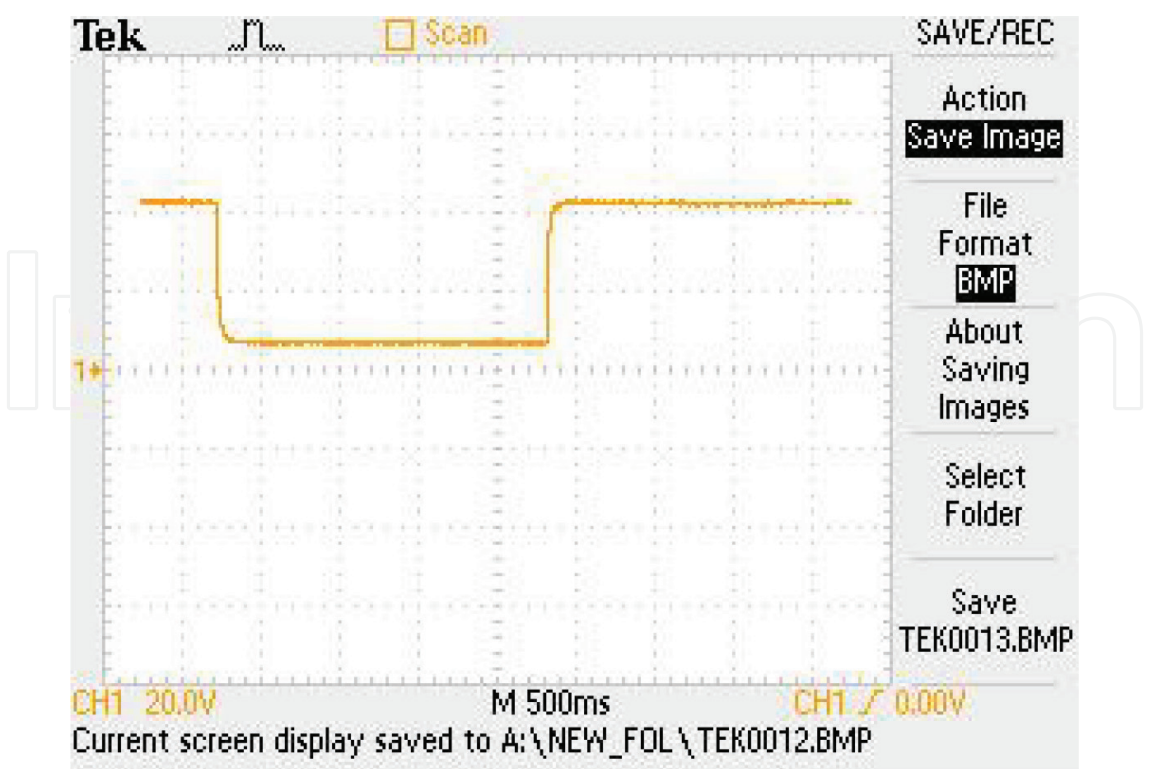| Parameter	Value obtained
Ave latency	3.297 ms
Std. deviation	0.6983 ms
Min. latency	1.905 ms
Max. latency	3.881 ms
50th percentile	3.707 ms
90th percentile	3.881 ms
99th percentile	3.881 ms
99.99th percentile	3.881 ms

Table 2. Latency results.

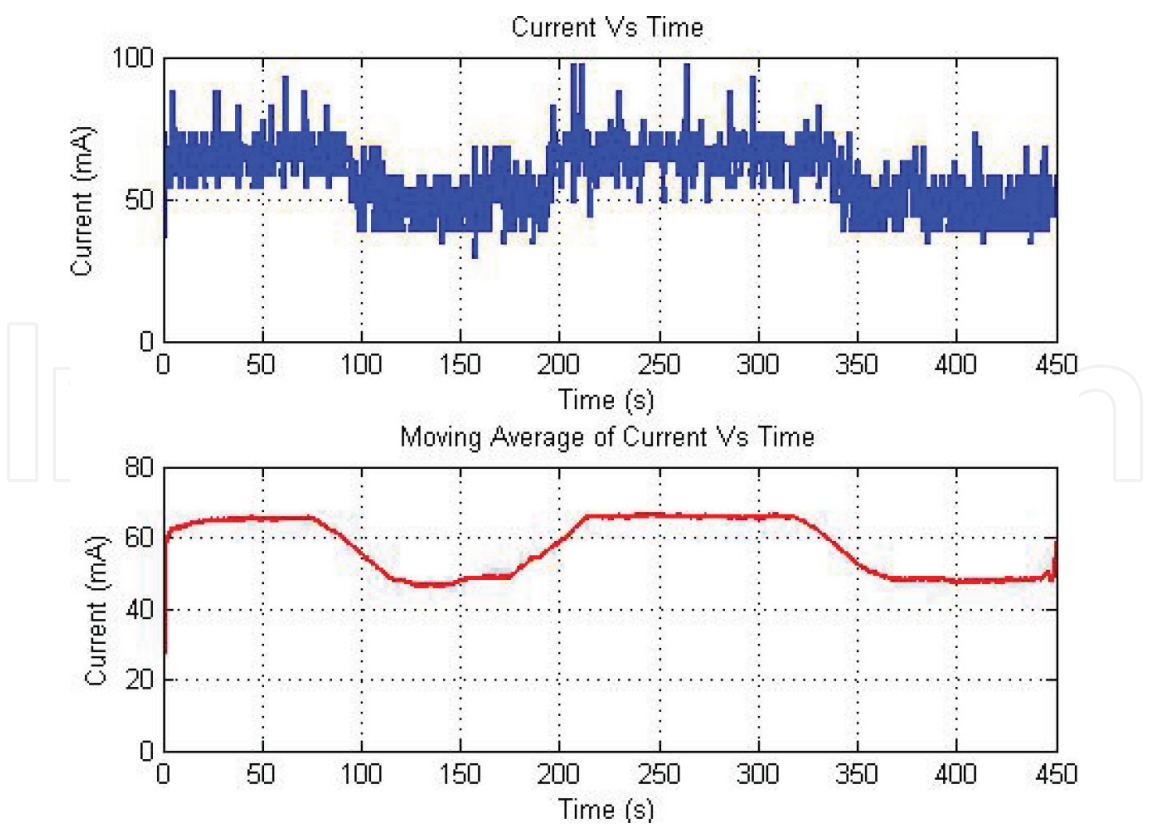**Figure 9.** Time response of Al-Yad.

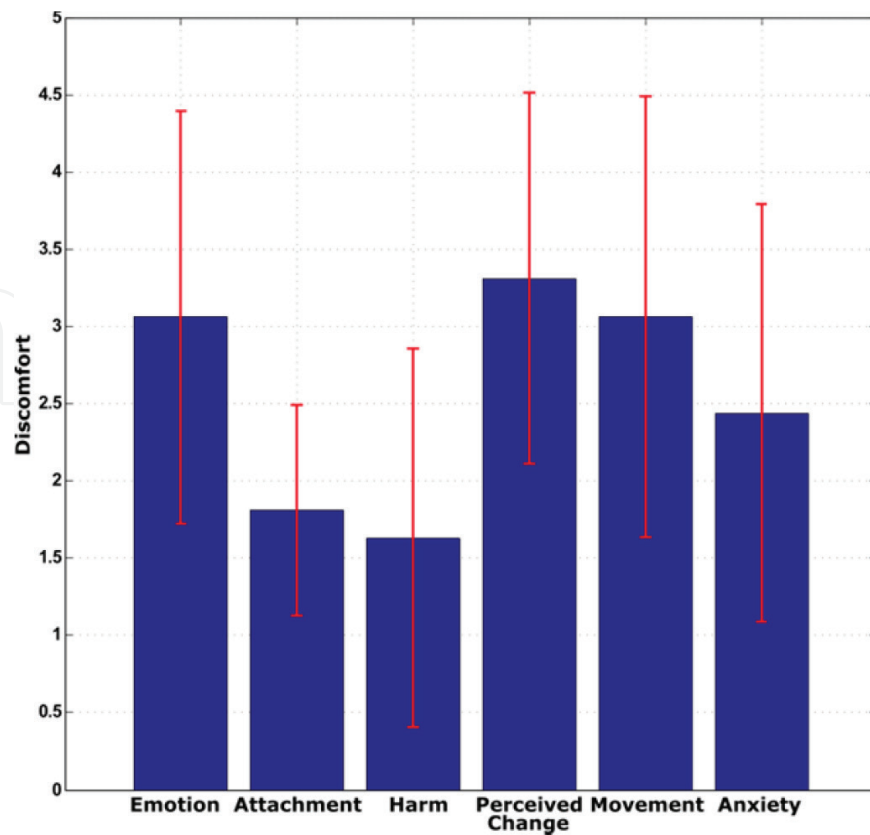**Figure 10.** Current consumed by system while running.

Figure 11. Survey result for comfort-ability of the hand glove.

	Emotion	Attachment	Harm	Perceived change	Movement	Anxiety
Emotion	1.0000	0.6436	0.8205	0.6207	0.5691	0.6397
Attachment		1.0000	0.6517	0.7298	0.7685	0.5202
Harm			1.0000	0.6726	0.7477	0.5358
Perceived change				1.0000	0.7273	0.0309
Movement					1.0000	0.4969
Anxiety						1.0000

Table 3. Correlation-coefficient matrix for comfort-ability of the hand glove.

flow of current is recorded after the dongle is removed. This shows that the energy overhead comes from data transmission itself.

Power dissipated by the system can be calculated by multiplying the current in **Figure 10** by the voltage across the system (i.e., 5 v). This gives an average of 310 mW during transmission and 220 mW when the system is not transmitting.

Finally, a survey was carried out to access the comfort of the hand glove. **Figure 11** shows the result of the survey and **Table 3** shows the correlation between the different attributes inves-

tigated. Eight participants were asked about the comfort of the hand glove. The participants were asked about their experience based on: emotion, attachment, harm, perceived change, movement and anxiety [38].

Emotion is whether the user is worried about his looks or on the edge because of wearing the glove. Attachment is the feeling the user gets when the glove is moving around—that constant feeling the user gets of wearing something. Harm is when the user fears being harmed by the glove. Perceived change is when the user feels physically different as opposed to emotion which is a psychological difference. Movement attribute is the measure of the glove which affects the user's movement. Anxiety is the worry a user gets about the glove's safety and reliability.

The attributes were tested on a scale of 0–10, with 0 being least negative and 10 being the most negative experience. The results show that the overall discomfort of the glove is well below the scale of five (see **Figure 11**).

However, perceived change, movement and emotion show higher readings. To investigate further, correlation coefficient between the different attributes is computed. **Table 3** shows the correlation matrix between all the attributes. It can be seen that there is a very strong relationship between emotion and harm, which suggests that the user's emotion toward the device is due to the user feeling some kind of mild harm. This may be due to the flying wires and the printed circuit boards (PCBs) that were glued to the glove. Also there is strong correlation between attachment, movement and perceived change. This tells us that the user could not move well or he is afraid of moving because he feels that the glove is loose, which leads to the user feeling physically different. Harm is strongly correlated with movement restriction because the user feels mild pain, in addition to moving parts mentioned earlier.

In conclusion, the users show emotion, perceived change and movement restriction the most because of the way the glove was packaged. To improve the glove, Raspberry Pi may be replaced with a smaller SBC like Beagle Bone. PCBs should be covered with slim plastic casing before attaching it firmly to the glove. This will help prevent parts from falling over and reduce harm due to protruding electronic components and heat. Flying wires can be covered by a layer of fabric. This will also reduce the movement of the components on the glove.

6. Summary

This chapter discussed our proposed intelligent glove designed to control actuators in an industrial environment setting. The glove is constructed using Raspberry Pi, a WiFi dongle and an assembly of light sources and light sensors. It was found that the system has a delay of approximately 23 ms, which is fast enough for controlling motors, pneumatic actuators and other mechanical actuators. Furthermore, the maximum power dissipated by the system is 310 mW, which showed that the system could be powered by a mobile phone's Lithium ion battery (3100 mAh) for up to 10 h of continuous data transmission. Moreover, a survey was carried out to investigate the comfort ability of the glove: It was found that the glove was acceptable but users have safety concerns and movement restrictions. This problem can be solved by repackaging the device.

Author details

Farouq Muhammad Aliyu* and Basem Almadani

*Address all correspondence to: g201303650@kfupm.edu.sa

Department of Computer Engineering, King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia

References

[1] Waris MM, Sanin C, Szczerbicki E. Smart innovation engineering: Toward intelligent industries of the future. Cybernetics and Systems. [Online]. 2018:1-16. Available from: https://doi.org/10.1080/01969722.2017.1418708

[2] Gungor V, Hancke G. Industrial wireless sensor networks: Challenges, design principles, and technical approaches. IEEE Transactions on Industrial Electronics. 2009;**56**(10): 4258-4265

[3] Aliyu F, Sheltami T. Development of an energy-harvesting toxic and combustible gas sensor for oil and gas industries. Sensors and Actuators B: Chemical. 2016;**231**:265-275

[4] Elsarnagawy T, Farrag M, Haueisen J, Abulaal M, Mahmoud K, Fouad H, Ansari S. A wearable wireless respiration rate monitoring system based on fiber optic sensors. Sensor Letters. 2014;**12**(9):1331-1336

[5] Billinghurst M, Starner T. Wearable devices: New ways to manage information. Computer. 1999;**32**(1):57-64

[6] Huang Z, Lu Y. Wireless monitoring and control system via android tablet pc. In: International Symposium on Computer, Communication, Control and Automation (3CA 2013). Atlantis Press; 2013. pp. 449-452

[7] Aleksy M, Rissanen MJ. Utilizing wearable computing in industrial service applications. Journal of Ambient Intelligence and Humanized Computing [Online]. 2012;**5**(4):443-454 Available from: http://dx.doi.org/10.1007/s12652-012-0114-2

[8] Benavente-Peces C, Ahrens A, Filipe J. Advances in technologies and techniques for ambient intelligence. Journal of Ambient Intelligence and Humanized Computing. 2014;**5**(5):621-622

[9] Westelaken R, Hu J, Liu H, Rauterberg M. Embedding gesture recognition into airplane seats for in-flight entertainment. Journal of Ambient Intelligence and Humanized Computing [Online]. 2010;**2**(2):103-112 Available from: http://dx.doi.org/10.1007/s12652-010-0032-0

[10] Kikhia B, Stavropoulos TG, Meditskos G, Kompatsiaris I, Hallberg J, Savenstedt S, Melander C. Utilizing ambient and wearable sensors to monitor sleep and stress for people with bpsd in nursing homes. Journal of Ambient Intelligence and Humanized Computing [Online]. 2015:1-13 Available from: http://dx.doi.org/10.1007/s12652-015-0331-6

[11] Rachael R. Wearables vs. smartphone apps: Which are better to count steps? LiveScience [Online]. 2015. Web 15 May 2018. Available from: http://www.livescience.com/49756-smartphone-apps-wearables-step-counts.html

[12] Anliker U, Beutel J, Dyer M, Enzler R, Lukowicz P, Thiele L, Troster G. A systematic approach to the design of distributed wearable systems. IEEE Transactions on Computers. 2004;**53**(8):1017-1033

[13] Zhao Y-B, Liu G-P, Kang Y, Yu L. A brief tutorial of networked control systems. In: Packet-Based Control for Networked Control Systems. Singapore: Springer; 2018. pp. 1-11

[14] Jianjun Z, Min'Gang C, Man Z, Su L. Remote monitoring and automatic navigation system for agricultural vehicles based on wlan. In: 4th International Conference on Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08; October 2008. pp. 1-4

[15] Gorlich D, Stephan P, Quadflieg J. Demonstrating remote operation of industrial devices using mobile phones. In: Proceedings of the 4th International Conference on Mobile Technology, Applications, and Systems and the 1st International Symposium on Computer Human Interaction in Mobile Technology. ACM; 2007. pp. 474-477

[16] Yang C-C, Hsu Y-L. Development of a portable system for physical activity assessment in a home environment. In: Proceedings of International Computer Symposium; Taipei Taiwan; 2006. pp. 1339-1344

[17] Coley G. Beaglebone Black System Reference Manual; Beaglebone.org [Online]. 2013. Available from: https://cdn-shop.adafruit.com/datasheets/BBB_SRM.pdf

[18] Ramon MC. Intel galileo and intel galileo gen 2. In: Intel R Galileo and Intel R Galileo Gen 2. Springer [Online]; 2014. pp. 1-33. Available from: https://link.springer.com/book/10.1007/978-1-4302-6838-3

[19] Pi R. Raspberry pi 3 model b (8 April 2017), Retrieved April. Vol. 9; 2017. p. 2017

[20] Shah J, Patel VP. Real time interfacing & control techniques using an open source. International Journal of Science, Engineering and Technology Research (IJSETR). 2014;**3**(3)

[21] Wang S-T, Fan C-L, Huang Y-Y, Hsu C-H. Transcoding on wearable devices: Technical report; 2015

[22] Khelil A, Shaikh F, Sheikh A, Felemban E, Bojan H. Digiaid: A wearable health platform for automated self-tagging in emergency cases. In: 2014 EAI 4th International Conference on Wireless Mobile Communication and Healthcare (Mobihealth)[Online]; November 2014. pp. 296-299. Available from: http://ieeexplore.ieee.org/abstract/document/7015969/

[23] Jutila M, Strmmer E, Ervasti M, Hillukkala M, Karhula P, Laitakari J. Safety services for children: a wearable sensor vest with wireless charging. Personal and Ubiquitous Computing [Online]. 2015;**19**(1):1-13 Available from: http://dx.doi.org/10.1007/s00779-015-0838-z

[24] Park S, Hosen AS, Cho G. An dds based architecture in supporting of data centric wireless sensor network environments. International Journal of Control and Automation. 2014;**7**(1):251-258

[25] Boonma P, Suzuki J. Tinydds: An interoperable and configurable publish/subscribe middleware for wireless sensor networks. Principles and Applications of Distributed Event-Based Systems [Online]; 2009. pp. 206-231. Available from: http://www.cs.umb.edu/~jxs/pub/pernets09.pdf

[26] Mascolo C, Hailes S, Lymberopoulos L, Picco GP, Costa P, Blair G, Okanda P, Sivaharan T, Fritsche W, Karl M, et al. Survey of middleware for networked embedded systems. Project Report: http://www.ist-runes.org/docs/deliverables/D5 01.pdf; 2005

[27] Fok C-L, Roman G, Lu C. Mobile agent middleware for sensor networks: An application case study. In: Fourth International Symposium on Information Processing in Sensor Networks, 2005. IPSN 2005 [Online]; April 2005. pp. 382-387. Available from: https://dl.acm.org/citation.cfm?id=1147747

[28] Zhai L, Li C, Sun L. Research on the message-oriented middleware for wireless sensor networks. Journal of computers. 2011;**6**(5):1040-1046

[29] Zug S, Schulze M, Dietrich A, Kaiser J. Programming abstractions and middleware for building control systems as networks of smart sensors and actuators. In: 2010 IEEE Conference on Emerging Technologies and Factory Automation (ETFA)[Online]; September 2010. pp. 1-8. Available from: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5641341&isnumber=5640954

[30] Weaver WP. Wiringpi. Software tone library [Online]. 2018. Available from: http://wiringpi.com/reference/software-tone-library/

[31] Instrument T. Hex schmitt-trigger inverters datasheet; 1988. pp. 1-11

[32] Yuan H, Perdoni C, He B. Relationship between speed and eeg activity during imagined and executed hand movements. Journal of Neural Engineering. 2010;**7**(2):026001

[33] Langdon GG. Single board computer. Computer Design. San Jose: Computeach Press; 1982

[34] Shen H, Li T. Data management of wireless sensor networks. In: Consumer Communications and Networking Conference, 2009. CCNC 2009. 6th IEEE [Online]; January 2009. pp. 1-2. Avalable from: https://ieeexplore.ieee.org/abstract/document/4784851/

[35] OMG. Data distribution service for real-time systems version 1.2. Object Management Group [Online]; 2015 . Available from: https://www.omg.org/spec/DDS/1.4/

[36] Hunt G. Is your data secured? RTI [Online]. July 2014. Available from: http://www.slideshare.net/RealTimeInnovations/ is-your-datasecure. pp. 20-21

[37] RTI. RTI Connext DDS Performance Test. Data distribution service (DDS) community RTI connext users. RTI [Online]; 2018. Available from: https://community.rti.com/kb/example-performance-test-rti-message-service

[38] James FK, Chris B. A tool to assess the comfort of wearable computers. Human Factors. 2005;**47**(1):77-91