Performance Analysis of RTOS with GPOS for Heart Rate and Temperature Measurement Wireless System Using Raspberry Pi

N P Raju Mandapaka¹, V V Subhash²

¹PG Student, Dept of ECE, Kakinada Institute of Engineering and Technology-II, JNTUK, E-mail: mnprawin@gmail.com
²Assistant Professor, Dept of ECE, Kakinada Institute of Engineering and Technology-II, JNTUK, E-mail: veerendrasubhash@kietgroup.com

Abstract: The objective of the project is to analyze the General Purpose Operation System (GPOS) performance with Real Time Operating System (RTOS) using a Heartbeat sensor and temperature sensor as a real time application. Methods-Statistical Analysis: The method used to calculate response is real feel technique which uses dedicated timer and interrupt to calculate the response. Findings: The responses of sensing the temperature, heartbeat are continuously sensing and after its values are sending through bluetooth is calculated RTOS having an average of 22ms where as GPOS has an average of 450ms of response time in real time environment. Conclusion: The results show that RTOS has better response time than GPOS. In any real time applications RTOS was mostly preferable to accurate response.

Keywords: GPOS, Latency Calculation, Raspberry pi, Finger tip sensor, Bluetooth module, RTOS, GPOS.

1. Introduction

Timing is the most important factor in any real time applications where some applications should react in very small amount of time in those situation applications needs a platform where the timing constrain was meet. In real-time systems, all real-time tasks are differentiated based on their timing, such as sporadic, response time, deadline etc. Real time systems classified in to two types’ soft real time systems and hard real-time systems. Hard real time system means it should complete the task within the deadline period otherwise its computation is useless. The damages caused by the hard real time systems are irreparable. The system builder’s should responsible to choose an operating system that can support and schedule these jobs with respect to their timing criteria so that no deadline will be missed. Soft real time systems require performance assurances from the operating system[1][2].

Some applications such as video/audio visual gaming require performance assurance and also acceptable jitter in timing these applications comes under soft real-time applications. The general architecture of RTOS is shown in Figure 1.

![Figure 1.1. General Architecture of RTOS](image-url)

Any RTOS basically have the characteristics like Multitasking and Preemptibility, Task Priority, Priority Inheritance, Short Latencies, Reliable and Sufficient Inter Task Communication Mechanisms, Control of Memory Management[3][4].

The GPOS for the proposed research work is based on Linux flavor. Many researches from the past few years used Linux as their platform Figure 2 it is inferred that, use of Linux is increasing in developing countries where technology increases[5].

![Figure 1.2. Use of Linux and Internet in Developed and Developing Countries](image-url)

Increasing complexity of devices and systems, huge competition, shorter time to Market and to reduce the cost of development requirement makes to use high-end software. Acate began to develop OXO systems. OXO belongs to hard real...
time systems where it must respond to the event in predefined period whatever the load stress; this makes them to think that existing real time kernel is inadequate to support all the expected services.[6] Development of the OXO systems was divided into three classes: non real time and soft real time Linux processes in user space and hard real time processes in kernel space and RTLinux shares the kernel space as well. A comparison of windows patched with RT kernel and Linux patched with RTKernel is also carried out in this development on services like voice services, Data pluse internet services, Software development environment, cost and support these results are shown in Table 1.1

Table 1.1 comparison various services of RTOS and GPOS[14]

<table>
<thead>
<tr>
<th>SERVICES</th>
<th>WINDOWS + RT EXTENSION</th>
<th>LINUX + RT EXTENSION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voice Services</td>
<td>Sufficient</td>
<td>Good</td>
</tr>
<tr>
<td>Datapluse Internet Services</td>
<td>Very Good</td>
<td>Very Good</td>
</tr>
<tr>
<td>Software Development Environment</td>
<td>Good</td>
<td>Sufficient</td>
</tr>
<tr>
<td>Support</td>
<td>Good</td>
<td>Good</td>
</tr>
<tr>
<td>Cost</td>
<td>High</td>
<td>Low</td>
</tr>
</tbody>
</table>

This approach also improves real time interrupt latencies of hybrid O.S with two level hardware interrupts. Real time operating system comparison by different real time operating system and few General purpose operating system comparisons on parameters like worst case response time, Interrupt latency, Latency jitter. Worst case response time was calculated by using method proposed by ISA[8].

This research describes those designs of very low-cost devices which measures the heart rate and temperature of the subject by clipping sensors on one of the fingers and tap our finger to temperature sensor and then displaying the result on Android Mobile or any bluetooth device. The device has the advantage that it is Raspberry pi based and thus can be programmed to display various quantities, such as heart rate, temperature and latency.

2. System Implementation

Fingertip sensor contains an IR LED and IR photo detector receiver. Using this device the heart rate signal can found. After getting the signal, it must be amplified, because the signal amplitude is very low. This is done using amplifier circuit. Then it is transmitted to the Bluetooth terminal through raspberry pi. The temperature sensor sense the human temperature and it also transmitted to Bluetooth terminal through raspberry pi. Finally it calculated the latency. This process repeats both RTOS and GPOS. The bellow figures shows RTOS and GPOS system setups[9].

A new low cost approach for course and laboratory designed to allow students to design robotics, embedded devices that feature IOT, networking, a Real Time Operating System (RTOS) with different languages support like C, C++, python java and others. Power consumption in mobile and handheld devices is the key embedded design issue; by varying the cache parameters in RTOS can reduce the consumption of the power by the embedded system[7]. The xenomai RTOS can perform better than RT patched Linux on overload conditions and RT patched Linux can provide high throughput than Xenomai.

In some cases, this performance benefit is so small that the use of a specific scheduling policy for soft tasks is questionable.

Performance comparison of Vxworks, Linux, RTAI, xenomai in hard real time applications presented in and the experimental results shows that open sources software is suitable for real time application. The underlying hardware should be shared by Linux and with other additional component this can be achieved by Xenomai and RTAI by using ADEOS nano-kernel which acts as communicator between hardware functionality. A hybrid O.S
In this section, we discuss the system overview like pulse detection, signal extraction, pulse amplification, and physical properties of our propose HRM device.

2.1 PULSE DETECTION AND AMPLIFICATION

It consists of the two parts: a heart rate displaying unit and pulse sensing unit. The device uses two red LEDs and a photo-sensor to measure heart rate through the change of blood reflectivity on the index finger. The power transmitted by the LEDs is matched with the photo sensor in such a way that the resistance will vary within the range of the photo sensor after attenuations through the index finger. The attenuations will vary depending upon the person, our specifications assume that the attenuation is, 80 percent, on average, of the light transmitted. The resistance network is used with the sensor to detect the changes in resistance to the changes in voltage. The voltage varies between 0 and 10 mV with respect to each heart pulse. Below shows a clip sensor which consists of two LEDs which gives the light and a Light Detective Resistor is used to detect the changes of resistance according to the amount of light transmitted from the tissue [10][11].

These signals are passed by an amplifier which detects the peak of each pulse and creates a corresponding pulse of high amplitude. To detect signal amplification an LM358 is used and the heart signal is amplified twice after passing through band pass filter (shows in below Fig.).

2.2 TEMPERATURE SENSOR

The DS18B20 has a 1-Wire interface, which means that one of its pin is used for serial communications. The other two pins need to be connected to 3.3V and 0V respectively. It contains a small circuit which generates a serial output [13].

The below figure is shows connection diagram of DS18B20. Pin 1 is connected to either of the 3.3V pins on the GPIO connector. Pin 3 is connected to one of the ground pins on the GPIO connector. Pin 2 is connected to GPIO pin 4. Pin 2 must not be allowed to float, hence a pull up resistor 4.7kOhm must be used to connect pin 2 to 3.3V.
Each DS18B20 has a 64-bit serial code, which allows multiple DS18B20s to function on the same 1-Wire bus.

Table 2.1. Temperature/Data Relationship

<table>
<thead>
<tr>
<th>TEMPERATURE</th>
<th>DIGITAL OUTPUT (BINARY)</th>
<th>DIGITAL OUTPUT (HEX)</th>
</tr>
</thead>
<tbody>
<tr>
<td>+125</td>
<td>0000 0111 1101 0000</td>
<td>07D0h</td>
</tr>
<tr>
<td>+85°</td>
<td>0000 0101 0101 0000</td>
<td>0550h</td>
</tr>
<tr>
<td>+25.0625</td>
<td>0000 0001 1001 0001</td>
<td>0191h</td>
</tr>
<tr>
<td>+10.125</td>
<td>0000 0000 1010 0010</td>
<td>00A2h</td>
</tr>
<tr>
<td>+0.5</td>
<td>0000 0000 0000 1000</td>
<td>0008h</td>
</tr>
<tr>
<td>0</td>
<td>0000 0000 0000 0000</td>
<td>0000h</td>
</tr>
<tr>
<td>-0.5</td>
<td>1111 1111 1111 1000</td>
<td>FFF8h</td>
</tr>
<tr>
<td>-10.125</td>
<td>1111 1111 0101 1110</td>
<td>FF5Eh</td>
</tr>
<tr>
<td>-25.0625</td>
<td>1111 1110 0110 1111</td>
<td>FE6Fh</td>
</tr>
<tr>
<td>-55</td>
<td>1111 1100 1001 0000</td>
<td>FC90h</td>
</tr>
</tbody>
</table>

2.3 BLUETOOTH TRANSMITTER AND RECEIVER

The HC-05 is the Bluetooth Transceiver, it is the latest Bluetooth wireless serial cable. These modems work as a serial (RX/TX) pipe. Any serial stream from 9600 to 115200bps can be passed seamlessly from our computer to our target[15].

The remote unit can be powered from 3.3V up to 6V for easy battery attachment. All signal pins on the remote unit are 3V-6V tolerant. No level shifting is required. Do not attach this device directly to a serial port. We can either solder a 6-pin header or individual wires. Unit comes without a connector[16][17][18].

2.3.1 KEY FEATURES:

- CSR Bluetooth Chip Solution
- Bluetooth Spec v2.0 EDR Compliant
- Enhanced Data Rate (EDR) compliant with V2.0.E.2 of specification for both 2Mbps and 3Mbps modulation modes
- Full Speed Bluetooth Operation with Full Pico net Support and Scatter net Support
- Incredible small size with 3.3V input, and RoHS Compliant
- UART interface and with baud rate setup function
- Support for 8Mbit External Flash On-board
- Support for 802.11Co-Existence
- Bluetooth devices can communicate at ranges of up to 10 meters.
- Bluetooth devices do not need to be in direct sight of each other.

2.4. FLOWCHART FOR THE SYSTEM

Figure 2.6 Flowchart for System Design

When switch ON the raspberry pi board first it is booted and then initialize the temperature sensor and Heartbeat sensor and it starts their functioning they are ready to sense and ready to transmitting first it is tested on GPOS and then RTOS the response time from After initialising the sensors and After sending the values. It calculates the latency time for both RTOS and GPOs. It proves the RTOS was mostly suited for Real time applications when compare to GPOS.

2.5 ALGORITHM

Step1: Start
Step2: While switch on the raspberry pi board it boots
Step3: Initial the serial ports
Step4: Executes the commands
Step5: Start the time and while intercept is occurs and read the heart beat pulses and temperature through its sensors continuously
Step6: Transfer the data if 1 minute is complete and save it to log file for reference purpose and calculate the end time. Send these data to connected bluetooth device.
Step 7: Checks for interrupt if interrupt was not occurring repeat the steps 3 to step 4 continuously until the stop the execution process.

Step 8: Stops

Step 9: Insert the GPOS chip

Step 10: Repeat the steps from 1 to 8

Step 11: Observe the result

3. Experimental Results
   The GUI of the presented system, which is opened the desktop of the system.

Step 1: Open MobaXterm Personal Edition

**Figure 3.1 MobaXterm Window**

Step 2: Enter the lxsession this command is used to enter the o.s

**Figure 3.2 RTOS Desktop Window**

Step 3: Open root terminal

**Figure 3.3 Root Terminal Window**

Step 4: Set the path first (enter cd Desktop)

**Figure 3.4 Set Path**

Step 5: Enter sudo modprobe w1-gpio command

**Figure 3.5 Initial The Real Time Commands**

Step 5: Enter sudo modprobe w1-therm command

**Figure 3.6 Run The Code**

Step 6: Enter python heart_modified.pyc command to execute the code

**Figure 3.7 Output Response Of RTOS**
Step 7:
Enter CTRL+C to stop the execution

Figure 3.8 Exit Window

Step 8:
To Enter poweroff for exit the o.s

Figure 3.9 Reconnecting Window

Step 9:
Enter r to reconnect the terminal

Figure 3.10 Run The Code

Step 10:
Enter lxsession to enter the GPOS

Figure 3.11 GPOS Window

Step 12:
Enter cd Desktop to enter the desktop

Figure 3.12 Root Terminal Window

Step 13:
Enter python heart.pyc command to run the code

Figure 3.13 Run The Code

Step 14:
Interrupt window

Figure 3.14 Run The Sensors Code

Step 15:
Output appears on window

Figure 3.15 Output Response
Step16:-
To stop the process ( ctrl+c)

Step17:-
To power off the system enter sudo power
to

4. Applications & Advantages

4.1 Applications:
We can use medical applications
• We can easily monitor the patient health
• Quick response
• Better and accurate method of measuring

4.2 Advantages:
• Accuracy determination of location
• Maintenance cost is low
• Easily replaceable
• More security in Army

5. Conclusion & Future Scope

5.1 Conclusion:
The raspberry based heart beat and
temperature monitoring system using fingertip
and temperature sensors was developed to make the
portable device and cheaper. A doctor can use this
technology from any remote place like villages.
Any non professional educated person can also
operate that device.

The overall research shows that heart beat
sensing and transmitting and human temperature
sensing and sending the data through all this
happens in nearly 140ms second in GPOS and it is
nearly 40ms in RTOS. The results show that real
time operating system have better response time
compared to the general purpose operating system.
The response time is important in many real time
applications. This research can be implemented in
many hospitals and industries need low response
reactions, in industries it helps fast operations and
advance areas.

5.2 Future Scope:
✓ We can use the WI- FI dongle instead of
LAN
✓ We can also transmit the message to the
mail of the person who is monitor.
✓ We can increase the micro SD CARD
storage size.
✓ Maintain a log register for future purpose.

References
1. Kshetri N. Economics of Linux adoption in
developing countries. IEEE Softwares. 2004
2. Marchesin A. Using Linux for Real Time
3. Marieska MD, Kistijantoroi AI, Subair M.
Analysis and benchmarking performance of real
time patch Linux and Xenomai in serving a real
time application. IEEE Electrical Engineering
4. Burdalo L, Terrasa A, Espinosa A, Garcia-
Fornes A. Analyzing the effect of gain time on
soft-task scheduling policies in real-time
system. IEEE on Software Engineering. 2012
5. Barbalace A, Luchetta A, Manduchi G, Moro
M, Soppelsa A, Taliercio C. Performance
comparison of VxWorks, Linux, RTAI and
Xenomai in a hard real-time application. IEEE
Transactions on Nuclear Science.2008 Feb;
http://www. adeos.org
7. Weinberg B. Uniting mobile Linux application
improving real-time interrupt latencies of
hybrid operating systems with two-level
hardware interrupts. IEEE Transactions on
composing soft real-time schedulers. The 22nd
10. http://www.instructables.com/id/Direct-
Network-Connection-between-Windows-PC-
andR/
11. http://www.engineeringgarage.com/embedded/ras-
pberry-pi/how-access-raspberry-pi-directly-
through-ethernet-port-pc
Available:http://www.webmd.com/heart-
disease/guide/how-heartworks
13. https://electrosome.com/ds18b20-sensor-
raspberry-pi-python/http://apk-
dl.com/bluetooth-terminal/Qwerty.Blu
Sao Carlos – SP – Brasil.
Embedded Software Systems and Real-time
Operating Systems And Component Integration
Considerations In Integrated Modular Avionics
Department of Transportation.
Overview of Real-Time Operating System.
Journal of the Association for Laboratory
Automation, 40–44.
Benchmarking Real-time Determinism. Texas:
Automation & Control Systems Committee.