A Non-invasive Appliance and Application for Remote Washing Machine Monitoring.

Bret Marshall
Earlham College
801 National Road West
Richmond, Indiana 47374
bdmarsh13@earlham.edu

ABSTRACT

Non-invasive Appliances are devices that do not require the opening of a machine for installation. This project involves the creation of a non-invasive appliance to be used on washing machines to determine if and when the machines are in use. This appliance and its associated web application will remove the problem of students in dorms having to repeatedly carry their dirty clothes to various washing machines to see if they are available. Existing solutions require expensive pieces of hardware and specific applications that do not integrate easily with other applications and platforms. Data was collected, an algorithm for detecting a running machine was implemented, and a device based on the Arduino platform was utilized to collect and send data to a database through an API implemented in this project. Lastly, a web application was designed to display information about the monitored washing machines. In this paper, an algorithm, appliance, and application based on collected data are presented to solve the aforementioned problem. **Keywords:** *Non-invasive Appliance, Embedded Hardware, Arduino, ammeter*

1. INTRODUCTION

Energy Monitoring systems are vital technologies in all fields. The ability to know the efficiency and power usage of machines in this day and age is highly sought after and essential. As energy prices increase and shortages affect more of the world, being the appliance or the system with the lowest power usage will become the goal for all. To determine this energy usage, extensive energy monitoring systems are being developed and used in all industries.

The motivation behind this project stems from a problem I have noticed while living in residence halls. With washing machines on only the basement floor, one would have to walk down many flights of stairs with their laundry only to discover the machines were taken. This process would be repeated multiple times until an open machine appeared. For myself and other busy students, this problem is both an aggravation and huge waste of time.

To solve this problem, a universal non-invasive appliance was designed using an Arduino microcontroller to monitor current being drawn by a specific washing machine. The appliance constantly takes measurements and sends that data to a database via an API, implemented in this project for receiving information. Lastly, a web application that displays the status of the washing machine being monitored was created.

The rest of this paper is organized as follows. Section 2 reviews related work on this subject. Section 3 describes a experiment with it related results and conclusions that influenced the design of the appliance and web application. Section 4 introduces and provides details on the design of the appliance, API, and web application. Finally, conclusion and future work are stated in Section 5.

2. RELATED WORK

In this section, some papers on the subject of energy monitoring systems are reviewed and compared. Based on metrics for energy monitoring systems, they are divided into the following categories:

2.1 Non-invasive hardware

Non-invasive hardware is a common approach to energy monitoring in the industry today. In Non-invasive hardware architectures, the system to be monitored need not be dissected and the structure of the monitored system does not need to be known, meaning the non-invasive hardware is universal. Non-invasive hardware is a necessity because of the great amount of time and money needed to pay people to get into every machine and know the internal structure so that the appliance can be monitored. Thus a non-invasive architecture saves time, money, and hassle.

Hart investigated a load monitoring technique for energy monitoring [2]. The non-intrusive appliance covered in this article would be used in homes to provide information on energy consumption on major electrical appliances, although the author does note that uses in the commercial and industrial sectors are possible. The article went into depth on how the use of this appliance in homes will aid in monitoring of the energy crisis.

Gomez et al. described and introduced the Energino, a standalone plug-load meter based on the Arduino platform [1]. The researchers stated that the driving factor behind the Energino's development was the lack of affordable and reliable energy consumption tools, which researchers need for the development of simulators. The article went in depth

ACM ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

on the hardware, software, and networking involved in the Energino. One of the main goals described by the article is to create an easily usable device.

Shajahan and Anand investigated a possible use of the Arduino platform for the development of a smart plug, a non-invasive universal plug that can be used remotely to monitor energy consumption [5]. This article also went into detail on the device's connection to an android device for remote monitoring. However, the article does not give a prediction of when this device will be on the market and their website does not either.

Jiang et al. went in depth on a hardware design of an energy monitoring appliance using their presented SPOT architecture [3]. SPOT stands for scalable power observation tool. Their design featured nodes which are used to monitor a range of devices over a period of time. This article detailed the hardware level design and dissected each part of the design to describe its use.

2.2 Software/Networking

Software engineering and network engineering of these embedded systems used for energy monitoring are vital parts of the non-invasive and universal architecture. Due to the wireless nature of non-invasive hardware for energy monitoring, software and networking for these systems are very important to get the collected data to useful machines for the monitoring to occur.

Lu and Gungor went into detail on the process of transmission of data from non-intrusive energy monitoring devices across wireless sensor networks [4]. The article also described their lab experiments, tests, and results.

In addition Gomez et al. went into detail on the software and networking involved in Energino described in the above subsection [1]. The article analyzed the C/C++ programmed Arduino hardware and Wiring library used to create the Energino. In addition, the article covered the wireless network setup used and how it connected to the client used for monitoring. Lastly, Shajahan and Anand covered the software and networking involved in connecting the Arduino platform smart plug to a server housing an android application [5].

3. PRELIMINARY EXPERIMENTS

To gain an understanding of a washing machine, what changes during a cycle and by how much, the following experiment was performed. Then the collected data was analyzed to influence the design of the hardware and appliance.

3.1 Methods

To determine what current ranges indicate a washing machine in use, an experiment was set up to watch the current flow over time on various washing machines with various cycle types. To measure the current being drawn by a machine, a Watts Up Pro meter was used. The meter was plugged into the wall and the machine to be monitored was plugged into the Watts Up Pro. Once plugged in, the Watts Up displayed instantaneous current being drawn by the machine that was plugged into it. Following are the data and results collected from the experiments

3.2 Data

Data was collected from various runs of the washing machine on different cycles in the form of time vs current, with the knowledge that voltage over the runs stayed constant at 121 V. Figure 1 displays a graph of a data run on a single cycle in the form, Time vs Log(current).

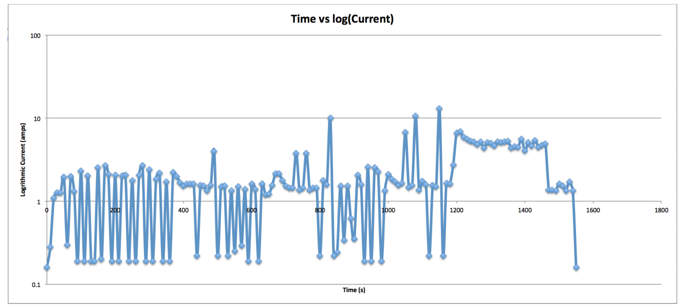

Figure 1: Time vs Log(Current) of single cycle

3.3 Results

From the data show in Figure 1, three critical points were determined. First, the current that indicates that a machine is on, indicated by the first and last data points in Figure 1, 0.16 amps. Second, the smallest current that occurred during a cycle, 0.19 amps. Third, the largest current that occurred during a cycle. 14.8 amps. To design a circuit that can handle these high values of current as well as have the ability to distinguish between the critical values, data was analyzed to determine the following:

1. Internal resistance of the Washing Machine at the critical values.
2. Possible different resistor values to be used in the added component and their associated voltage drop and power dissipation at the critical values.

Equation 1 was used to calculate the resistance of the washing machine at the critical values.

$$V = IR \quad (1)$$

Where V is the potential difference between two points which include a resistance R and I is the current flowing through the resistance.

Knowing the inputted voltage was steady during the cycle at 121 V, the following results were calculated.

Table 1: Resistance in the Washing Machine at the critical points

Current(amps)	Resistance(ohms)
0.16	756
0.19	637
14.8	8

In its simplest form, the circuit acts as two resistors in series where one resistor is the washing machine (the load) and the other resistor will be used to have just enough voltage drop across the resistor to obtain a reading of the current using the equation 1, at the Arduino's sensitivity of 0.00488758553 V. This was determined by the fact that an Arduino can detect voltage differences between 0 and 5 volts and displays them as an analog value between 0 and 1023.

Also, the input can handle up to 30 V without causing damage to the entire Arduino board. Equation 2 was used to determine the voltage drop across the added resistor.

$$V_{\mathrm{x}} = (R_{\mathrm{x}} * V_{\mathrm{s}})/(R_{\mathrm{t}}) \quad (2)$$

Where V_{x} is the voltage being investigated, R $_{\mathrm{x}}$ is the resistor value being looked at, V $_{\mathrm{s}}$ is the supplied voltage, and R $_{\mathrm{t}}$ is the total series voltage.

In addition, resistors dissipate power and at high enough voltages this can damage them. Power dissipation of a resistor is governed by equation 3.

$$P = V^2/R \quad (3)$$

Where V is the voltage being investigated, R is the resistance value of the resistor being looked at, and P is the power dissipation.

To choose the appropriate resistor value the following was considered: The fact that the relative voltage drop across the added resistor component needed to be large enough that the Arduino's sensitivity could register a value at the various critical points, yet small enough that the resistor did not draw away too much voltage from the washing machine at any of critical points. Also considered was that the resistor value needed to be small enough that power dissipation across that resistor at the various critical points was not large enough to cause damage to the resistor. Tables 2 and 3 were created to determine the necessary value of the resistor across which the current flowing through the system was calculated.

Table 2: Voltage drop across the possible resistances at critical points

	Voltage at		
Resistance (ohms)	0.16 amps	0.19 amps	14.8 amps
1	0.159	0.189	13.187
0.5	0.079	0.094	6.973
0.25	0.039	0.047	3.590
0.01	0.00159	0.0019	0.147

Table 3: Power dissipation across the possible resistances at critical points

	Power Dissipation at		
Resistance (ohms)	0.16 amps	0.19 amps	14.8 amps
1	0.025	0.036	173.89
0.5	0.0127	0.018	97.26
0.25	0.0064	0.009	51.56
0.01	0.003	0.0036	21.378

Influenced by the results in tables 2 and 3, an ideal resistance of 0.25 ohms was determined. By placing four 1 ohm resistors in parallel, the ideal resistance was achieved to give the voltage drop needed to distinguish between the critical values. As well, with four resistors the power dissipation of each of them is low enough to not cause damage to the resistors at the highest critical value.

4. DESIGN

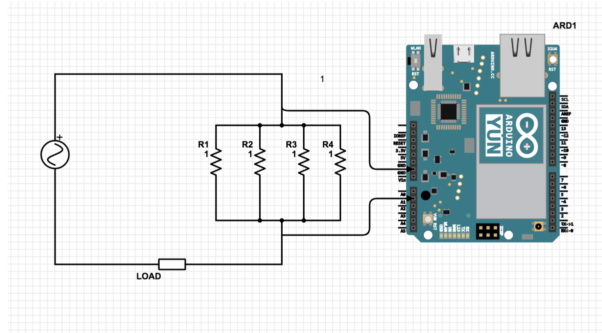

Figure 2: Circuit design of Non-invasive Appliance

Influenced by the data and results described in section 3 of this paper, a non-invasive piece of hardware was developed using the Arduino platform to solve the problem described in the introduction and abstract. The hardware determines the current being supplied to a washing machine, then wirelessly sends a signal to an API that was setup to edit a database of washing machines, communicating to the API whether the machine is being used. Lastly, a web application was developed to pull information on the machines from the API and display that information to users of the app. This allows users to determine whether a specific machine is in use at any moment. In the following subsections, a circuit, hardware component, algorithm, API and web application are described.

4.1 Circuit/Hardware

Figure 2, displays the layout of the circuit board. The load resistor in the diagram refers to that of the washing machine. An extension cord was used as the base of the appliance to achieve the non-invasive nature of the project. The use of an extension cord allowed for the wires to be opened on the extension cord, to avoid having to cut the wires of the specific washing machine. Thus, my device was inserted into the system by cutting the wire of the extension cord and soldering each end of the wire to an end of the circuit. Attention was paid to the wires used in the circuit to ensure they could handle the current levels. Figure 3 demonstrates this.

4.2 Algorithm

A cutoff voltage of 0.20 amps average over a 15 second period of time was calculated to be the cutoff for determining if a washing machine was on. This value allowed enough space to ensure that all machines and all cycles that were analyzed could be determined to be on or off. In addition, taking an average of the readings ensured that random spikes did not cause errors in the reading, as well as limited the number of times the Arduino had to access the API through its internet connection. Algorithm 1 displays the algorithm used in the Arduino, programmed in C, over the Arduino IDE, to take readings and send them to the API.

4.3 API

API stand for Application Program Interface. An API was developed to interact with the database that stores the state of the washing machines. The API was developed using the PHP programming language and a PostgreSQL

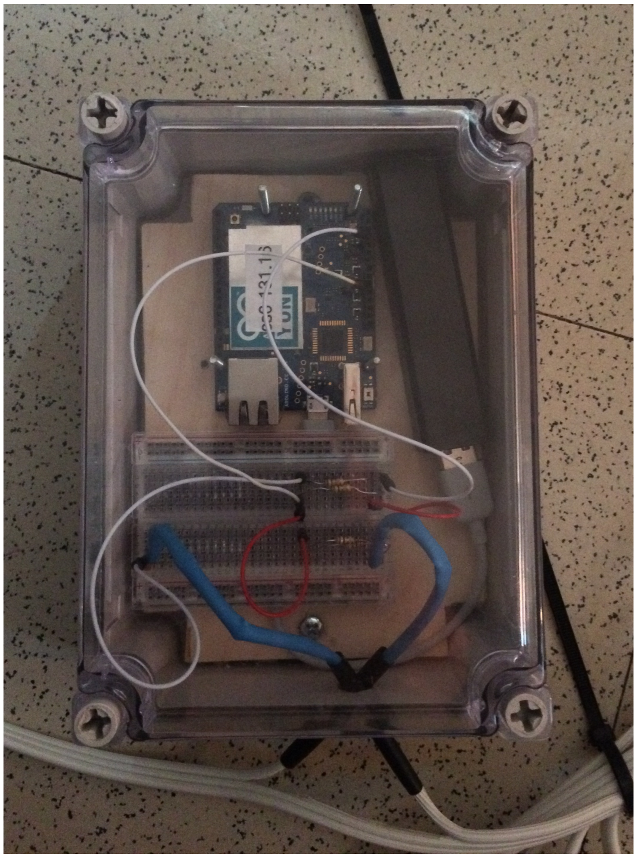

Figure 3: Circuit and Appliance

```
state = False;
machineId = 1;
cutoff = 0.20;
while True do
    count = 0;
    total = 0;
    avg = 0;
    while count < 150 do
        total += (analog read of port *
          0.00488758553)/(0.25)
        wait(100) //0.1 second pause
        count ++;
    end
    avg = total/150
    if avg >= cutoff and state==False then
        visit the api to change the state of the machine
          in the database to on
        state=True;
    end
    if avg <cutoff and state==True then
        visit the api to change the state of the machine
          in the database to off
        state=False;
    end
end
```

Algorithm 1: Algorithm running on the Arduino

database. When the API is visited on its update section (apiURL\update), it is given both the Id of the machine and the state of the machine, on or off. Using PHP's get method, these variables are obtained from the URL and passed to a function to interact with a PostgreSQL database using an Update statement from the SQL language. The API works as a middle man between the Arduino and the database because the Arduino is unable to directly interact with the database.

4.4 Web Application

A web application was developed to display the state information of the machines being monitored with the non-invasive appliance described in this paper[1]. The web application is written in the PHP programming language and in the same manner as the API, connects to the PostgreSQL database. The site currently is hosted on an Apache server on the Hopper cluster machine.

5. CONCLUSION AND FUTURE WORK

In this paper, we have presented a solution to a problem many college student face on a weekly basis. Not knowing when a washing machine is available is a waste and this machine will solve that problem. The solution came in the form of a non-invasive appliance that the washing machine is plugged into, as well as a web application that users can visit to be informed about the state of the washing machines. In addition, this paper described in depth the design of the appliance, API, and Web application.

To advance this project further in the future, the following are planned to be worked on: Add advanced functionality to the web application and API. An example of an advanced functionality is the sending of text messages and emails to alert users when machines are available upon request. Develop the appliance with a less expensive microcontroller. The Arduino Yun was used because of its ease of use and availability in the Earlham College Computer Science Department. However, it is not the least expensive option on the market now. This would cut the price of a machine down significantly. Produce enough appliances to outfit the entire campus with the remote monitoring system discussed in this paper. Develop software to automate the setup of a new device. New devices need to be manually programmed with the machine id and also need to be manually added to the database. In the future, automating this process to have the machine request a new id and new setup and then keep track of those pieces of information would cut down on setup time.

6. ACKNOWLEDGMENTS

This work was supported by the Earlham College Computer Science Department as part of the Senior Seminar Class. Special thanks goes to my adviser Charlie Peck who lead me in the right direction and provided his assistance and knowledge throughout the process. Also, a thank you to Kyle Zilic, a professor of physics at Earlham college, who assisted me in the design and safety measures of my circuit. Lastly, a thank you to Craig Early, a fellow senior computer science major at Earlham College, who assisted me in soldering my appliance.

[1]hopper.cluster.earlham.edu/bret

7. REFERENCES

[1] Karina Gomez, Roberto Riggio, Tinku Rasheed, Daniele Miorandi, and Fabrizio Granelli. Energino: A hardware and software solution for energy consumption monitoring. In *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt), 2012 10th International Symposium on*, pages 311–317. IEEE, 2012.

[2] George W Hart. Residential energy monitoring and computerized surveillance via utility power flows. *IEEE Technology and Society Magazine*, 8(2):12–16, 1989.

[3] Xiaofan Jiang, Prabal Dutta, David Culler, and Ion Stoica. Micro power meter for energy monitoring of wireless sensor networks at scale. In *Proceedings of the 6th international conference on Information processing in sensor networks*, pages 186–195. ACM, 2007.

[4] Bin Lu and Vehbi C Gungor. Online and remote motor energy monitoring and fault diagnostics using wireless sensor networks. *IEEE Transactions on Industrial Electronics*, 56(11):4651–4659, 2009.

[5] Altaf Hamed Shajahan and A Anand. Data acquisition and control using arduino-android platform: Smart plug. In *Energy Efficient Technologies for Sustainability (ICEETS), 2013 International Conference on*, pages 241–244. IEEE, 2013.