A Simplified Design of Automatic Energy Consumption Measuring Platform for Embedded Systems

Huanjie Wu, Chun Chen and Kai Weng
College of Computer Science and Technology, Zhejiang University, 310027, Hangzhou, Zhejiang, China
Email: 3100100294@zju.edu.cn

Abstract. In this paper, a simplified design of automatic energy consumption measuring platform for embedded systems is proposed. A straightforward device named USB-TTL replaces the intermediate device in the original design and becomes the link between the host computer and the target board. The simplification makes the design of host computer software simpler and the whole platform more reliable. It also eliminates the need to design software for the intermediate device. In general, it has features similar to the original design. A series of test sets are also designed to verify its capabilities and accuracy. Compared to the original design, the simplified one has almost consistent accuracy, is easier to deploy, and requires less wiring and devices.

1. Introduction

With the development of System-on-Chips (SoC), embedded systems are deployed everywhere in our daily life. Energy consumption has significant impacts on not only battery-life but also hardware thermal design as well as device security and reliability. Many methods have been proposed to figure out how much energy an embedded system consumes. Last year, we introduced our design of automatic energy consumption measuring platform for embedded systems. In practice, we found that the wiring of the intermediate device used in the original design was complicated, and additional programming development work was needed for this device. At the same time, due to the limitation of the sampling frequency of the digital power supply and the digital multimeter, the ability of the intermediate device to respond to changes in GPIO levels quickly cannot be fully utilized. Therefore, we tried to replace and simplify the intermediate equipment based on the original design. We also designed a series of test sets and verified the new simplified design.

2. Related Works

There are many ways that we can measure the energy consumed by embedded systems. Mesa-Martinez F J proposed a method of measuring energy consumption for Android smartphones with a battery voltage characteristic curve [1]. It is fast and straightforward; Nevertheless, it is inaccuracy, and the resolution is low. McCullough J C used built-in energy counters on the Intel Calpella platform to measure actual power consumption on subsystems directly [2]. David H and Hähnel M used RAPL (Running Average Power Limit) to measure power consumption, and the resolution is as high as 0.2W [3], [4]. However, neither of these two methods is universal. They depend heavily on some built-in circuit components and operating system functions. Energy consumed by the chip can also be calculated by infrared cameras deployed on the surface of the board, which capture pictures of heat distribution of the chip [1], [3]. It is very visual and intuitive, but it is not easy to quantify. Mesa-Martinez F J used multimeter [1], Shye A used current clamps [5], and Bircher W used power probes...
to measure energy consumption. Some researchers, like Chang N and Gilberto C, also used oscilloscopes for a higher sampling rate [7], [8].

These measurement methods have their own advantages and disadvantages, but we found that there are always two problems to be solved: one is how to establish the connection between the power or energy consumption curve and the code running on the target board; the other is how to automatically complete multiple measurement tasks to reduce unnecessary manual work.

We introduced the design of an automatic energy consumption measurement platform in a previous paper. In that design, we introduced an intermediate device with GPIO interfaces, which can output high and low levels on GPIO pins, and can also sense the changes of levels. This intermediate device runs a real-time operating system. We developed a program for it to output different levels on the GPIO pins according to the control instructions from the host computer. It can also respond to the rising and falling edges on the GPIO pins and record the timestamp. At the same time, it communicates with high-precision digital power/multimeter to obtain real-time power data, such as voltage and current and records the corresponding timestamp as well. Using this intermediate device, we established the connection between the power parameter curve and the code running on the target board. We also developed software on the host computer to control the entire hardware system and complete the automated measurement.

3. Our Idea and Design
As we all know, energy consumption is the time integral of power. To measure energy consumption, we need to understand both the runtime power consumed by the target device and the time it takes to execute the program. In our original design, we used an STM32F7 board as the intermediate device, and its GPIO interfaces can respond to level flip frequency up to 54MHz [9]. However, the sampling frequency of most digital power supplies/multimeters is much lower than this value [10]. In general, the sampling interval is hundreds of microseconds when the sampling accuracy is four and a half digits; and the sampling interval may be several milliseconds when the sampling accuracy is six and a half digits. Besides, since the time required for the serial port and Ethernet communication is much longer than the response time of the GPIO, using such a 'high GPIO performance' intermediate board can hardly contribute anything to improve the measurement accuracy.

In practice, we found that USB-TTL modules have many of the features we need. The USB-TTL module has the ability of serial communication. In addition to the pins required for data transmission and reception, it also has several pins for flow control, such as DTR, DSR, RTS, and CTS. These flow control pins can be multiplexed as GPIO input and output pins.

Figure 1 shows the wiring of our simplified measurement platform. The target board's GPIO pins are connected to the module's flow control pins. To program an image file to the target board, the target board's serial port pins are connected to the module's serial transceiver pins. Also, flow control
pins can be connected to the RST and BOOT pins of the target board to guide the target board to restart and enter different working modes.

In the simplified design, the digital power supply or digital multimeter is directly connected to the host computer via Ethernet or serial port as well. Thus, the parameter configuration instructions, data instructions, and return values are directly transmitted without intermediate devices.

3.1. Automatic Measurement Software
To achieve automatic and unattended measurement, the host computer software is designed to finish a series of jobs automatically. Figure 2 shows the workflow of measuring energy consumption. The host computer software configures power supply output parameters such as output voltage and maximum output current before each test case starts so that the designers and researchers don't have to reconfigure power supply manually every time. It will also query and log real-time power supply output data. As there will be large quantities of test cases to be tested, automatic cross-compilation and image flashing are also needed. The software will perform cross-compilation for each test case source code files and then perform the program task. These tasks were done manually and one by one in the past, but with this automatic host computer software, they can be completed automatically. In our design, the host software can reboot target board into different working modes by controlling the flow control pin level of the USB-TTL module (connected to the RST and BOOT pins of the target board), such as ISP programming mode and normal operation mode. The image programming can be done through the serial port, or it can be done through JTAG by calling external programs.

3.2. Digital Power Supply or Digital Multimeter
Digital power supply or digital multimeter provides readings of real-time voltage and current data. The instrument is required to be equipped with at least an Ethernet or serial interface and support queries using commands like SCPI. The instrument is directly connected to the host computer via ethernet or serial port, and it adjusts output voltage and maximum current according to host computer software’s configure instructions and replies real-time output data when receiving query instructions. Some types may support output RTC timestamp of the device, and this can be used for time synchronization between the host computer and the device itself.

Figure 2. The flow of measuring energy consumption

For unattended measurement, all real-time power supply output data will be captured and recorded in the logfile for each test. For fault recovery, when the software does not receive data after the timeout, we assume that the power supply or target board is down. In this circumstance, a hard reset will be performed by the software, and both the power supply and the target board will be automatically restarted.

3.2. Digital Power Supply or Digital Multimeter
Digital power supply or digital multimeter provides readings of real-time voltage and current data. The instrument is required to be equipped with at least an Ethernet or serial interface and support queries using commands like SCPI. The instrument is directly connected to the host computer via ethernet or serial port, and it adjusts output voltage and maximum current according to host computer software’s configure instructions and replies real-time output data when receiving query instructions. Some types may support output RTC timestamp of the device, and this can be used for time synchronization between the host computer and the device itself.
3.3. Modification in Target Board Code
Application code running on the target board also needs some modification. A GPIO call is added to produce a rising edge just before the beginning of the code segment to be tested, as well as a GPIO call to produce a falling edge right after the ending of the code segment. These begin-end signals make it possible to establish the relationship between the power supply curve and codes running on the target board.

4. Tests and Results
KEITHLEY 2280S-32-6 type digital power supply is used in this paper, and in practice, its sampling rate can reach about 1KHz. Figure 3 is the user interface of real-time data logging. The host computer software of our platform can draw a real-time chart of power supply voltage, current, begin-end signals, and other event signals from received data. The red line is for power supply voltage, blue for current, vertical green dash line for begin-end signals, and vertical pink dash line for other event signals. The software also records the data received and store them in files for future use.

![Figure 3. Software user interface showing curve and signal](image)

To verify the capability of our platform, we designed several test sets, and an STM32F4 core only board is used as the target board.

4.1. Ability to Measure Time Intervals
Energy consumption is based on both power rate and duration, and the power rate can be calculated from power supply output voltage and output current. In our design, the period is timed by GPIO rising and falling edges. Thus, we’ve got to verify our design’s ability to measure time intervals. For this test, we’ve designed several programs with different execution time, and we also compared the readings with those from the original design. The test results are shown in figure 4, and we can see that in terms of time measurement accuracy, the simplified design has almost the same readings as those of original design. The maximum difference between their readings is about 1%.
4.2. STM32F4 in Different Power Modes

The STM32F4 is designed with three different power-saving modes, two of which, sleep and stop modes, can also use different regulators. We designed six groups of test cases to measure the operating current of the STM32F4 in different power states. They are: normal working mode, sleep and stop modes using the main regulator, sleep and stop modes using low power regulator, as well as the standby mode. Figure 5 shows the energy consumed by the STM32F4 core only kit-board in different power states: normal, sleep, stop, and standby. For this test, because there is no time factor, we only compare the working current readings of the two designs. The test results showed that our simplified design still has similar accuracy as the original design, with the maximum difference of about 3% and average of absolute deviations 1.02%.

4.3. STM32F4 Processor Running with Different Frequency

According to the STM32F4 datasheet and user manual, the STM32F4 processor has a maximum frequency of 168MHz, which can be tuned by modifying PLL parameters. We designed a series of test cases that have 40 sets of different PLL parameter combinations, setting the processor to run on different frequencies. Table 1 shows the trend of power supply output current with different processor frequency by setting different PLL parameters on the STM32F4 processor. Due to space limitations, we only show the test results of some frequency points here. In this test, not only the power rate but execution time are also variables. We made a comprehensive comparison of the readings of the simplified design with those of the original design.

| Test Item | Original design readings (μJ) | Simplified design readings (μJ) | Difference (%) |
|-----------|-------------------------------|---------------------------------|----------------|
| 1MHz      | 8.2894                        | 8.3197                          | -0.366         |
| 2MHz      | 8.326911                      | 8.5526                          | -2.710         |
| 4MHz      | 8.877227                      | 9.0001                          | -1.384         |
| 8MHz      | 9.948424                      | 10.1476                         | -2.002         |
| 16MHz     | 12.053634                     | 11.7096                         | 2.854          |
| 32MHz     | 17.795274                     | 17.7466                         | 0.276          |
| 64MHz     | 24.290318                     | 24.4644                         | -0.717         |
| 128MHz    | 39.492187                     | 39.0114                         | 1.217          |

In this test, we also performed measurements using our original design as control groups. Table 1 shows the test results. The difference between the readings of the two measuring methods is around 1.35%. Different designs do have impacts on the measurement results, however, according to all our three groups of tests, we believe that the difference in readings is acceptable. The results showed that
our automatic measuring platform has similar accuracy to the original design of the measurement platform. We’ve designed more than 1,200 test cases and 40 different hardware environment configurations. A total of over 8,000 tests were conducted, and it took about a whole week to finish them all. With the pride of our automatic measuring platform, all the tests listed above are executed automatically. The platform will skip to the next test case if it reaches a maximum of 3 failures in the same test case. This automated and unattended measuring platform allows us to liberate from the repetitive compilation, programming, measuring, and recording tasks so that we can focus more on the analysis of measurement results.

5. Conclusion
The most significant advantage of our simplified design is its simplicity, both in hardware and software. It has more straightforward and more intuitive wiring, and it eliminates the need for software development and design on intermediate devices. The simplified design does have its shortcomings. It does not have as many GPIO interfaces as the original design, and only supports two pairs of GPIO pins. Despite its flaws, the simplified design still meets the requirements of automatic energy consumption measurement in most cases, and its measurement accuracy is still comparable to the original design, without losing accuracy due to simplification. The new simplified design makes it possible for our automated energy consumption measurement platform to be widely used. We believe this design can help more embedded developers and researchers.

6. References
[1] Mesa-Martinez F J, Brown M, Nayfach-Battilana J and Renau J 2007 Measuring performance, power, and temperature from real processors Proceedings of the 2007 workshop on Experimental computer science pp 16-es
[2] McCullough J C, Agarwal Y, Chandrashekar J, Kuppuswamy S, Snoeren A C and Gupta R K 2011 Evaluating the effectiveness of model-based power characterization USENIX Annual Technical Conf Vol 20
[3] David H, Gorbatov E, Hanebutte U R, Khanna R, and Le C 2010 RAPL: memory power estimation and capping 2010 ACM/IEEE International Symposium on Low-Power Electronics and Design (ISLPED) pp 189-194
[4] Hännel M, Döbel B, Völz M and Härtig H 2012 Measuring energy consumption for short code paths using RAPL ACM SIGMETRICS Performance Evaluation Review 40(3) 13-17
[5] Shye A, Scholbrock B and Memik G 2009 Into the wild: studying real user activity patterns to guide power optimizations for mobile architectures Proceedings of the 42nd Annual IEEE/ACM International Symposium on Microarchitecture pp 168-178
[6] Bircher W L, Valluri M, Law J and John L K 2005 Runtime identification of microprocessor energy saving opportunities Proceedings of the 2005 international symposium on Low power electronics and design pp 275-280
[7] Chang N, Kim K and Lee H G 2000 Cycle-accurate energy consumption measurement and analysis: Case study of ARM7TDMI Proceedings of the 2000 international symposium on Low power electronics and design pp 185-190
[8] Gilberto C and Margaret M 2005 Power prediction for intel xscale processors using performance monitoring unit events power prediction for intel xscale processors using performance monitoring unit events ISLPED Vol 5 pp 8-10
[9] STMicroelectronics STM32F746ZG Datasheet - production data https://www.st.com/resource/en/datasheet/stm32f746zg.pdf
[10] Tektronix Series 2280S Datasheet https://www.tek.com/datasheet/series-2280s-precision-measurement-low-noise-programmable-dc-power-supplies