IoT-type Electric Fan
Remote-controlled by Smart-Phone

Wen-Chung Tsai
Dept. of Information and Communication Engineering
Chaoyang University of Technology
Taichung, Taiwan
e-mail: azongtsai@cyut.edu.tw

You-Jyun Shih
Dept. of Information and Communication Engineering
Chaoyang University of Technology
Taichung, Taiwan
as8730873@yahoo.com.tw

Ting-Ming Tsai
Information and Communications Research Laboratories
Industrial Technology Research Inst.
Hsinchu, Taiwan
tmtsai@itri.org.tw

Abstract—In the network and communication generation, smart-phone is carried by almost every persons. As such, IoT (Internet of Things) device controlled by smart-phone over wireless and telecommunication network such as WiFi and LTE (Long Term Evolution) is a promising technology. Accordingly, this paper implemented a PWM (Pulse Width Modulation) signal generation module and integrated it in a FPGA (Field-Programmable Gate Array) embedded system to realize an IoT-type electric fan that can be remote controlled by smart-phone. Furthermore, the implemented remote controlled framework can be utilized to control other household electric appliances such as refrigerator, washing machine, and air conditioning to achieve the purpose of convenient and smart appliances control.

Keywords—electric appliances; field-programmable gate array; internet of things; pulse width modulation; remote control

I. INTRODUCTION

Recently, smart-phone is popular for high speed internet connectivity. Consequently, applying wireless network and telecommunication technology such as WiFi and LTE (Long Term Evolution) to IoT (Internet of Things) network system is a promising technology. Therefore, IEEE [1] and 3GPP [2] have started to discuss related IoT application issues to face of the rise of different IoT interconnection patterns. The current discussions focus on the functional architecture, service requirement, and application of Machine-type [3] related issues such as M2M (Machine-to-Machine) communication requirements [4]. The mentioned efforts show that the need of further development of electronic appliances to fit various requirements of IoT applications.

Direct Current (DC) inverter motor has been widely applied to air-conditioning products in recent years. Compared to the traditional fixed-frequency Alternating Current (AC) motor, DC inverter technology can avoid repeatedly starting and stopping the operation of motor. So that DC inverter motor can achieve approximately 30% to 35% power saving compared to the transitional fixed-frequency AC motor. Recently, DC inverter motor has extended its application to more and more household electric appliances, such as refrigerator, washing machine, and electric fan.

IoT-type electric appliances have gradually came into peoples’ daily lives. To adapt to the quick evolution of application requirements, the implementation system must be able to integrate multi-microprocessors and various kinds of peripheral controllers. So that programming-able device such as FPGA (Field-Programmable Gate Array) [5] and FPGA-based microprocessor such as ALTERA NIOS II [6] have gradually attracted attentions of market and industry.

Due to huge computing efforts for multimedia applications and quickly evolving standards of electric products, such programming-able device with parallel processing ability has its advantages in the development of IoT-type electric appliances. Developers can conditionally select the resilient features of hardware and software to achieve an optimal tradeoff between product cost and system performance. Wisdom house and smart control are becoming the living style in our daily lives. Integrated system with fully-featured applications has its advantages in increasing performance and decreasing cost for the implementations of consumer electronic products. Accordingly, this paper implemented a FPGA embedded system that can be remote controlled by smart-phone. Furthermore, the implemented key technology can be applied to various household electric appliances, such as refrigerator, washing machine, electric fan and so on.

The rest of this paper is organized as follows. In Section 2, we will first introduce the background about DC inverter motor and its related applications. Section 3 will describe the applied FPGA embedded system, and then the proposed specification of designed modules will be presented. In Section 4, we will provide a demonstration for the implemented IoT-type electric fan. Finally, Section 5 will draw a conclusion.

II. BACKGROUND

Among conventional remote controls, the device figures, configurations, and control methods, are different between each other. Currently, smart-phone has been quite commonly used by people. To control electric appliances by smart-phone it has its rationality and necessity for convenience. Home equipment such as computer, printer, monitor, projector, and television have already provided networking abilities with wireless (WiFi) or wired (Ethernet) connection. Compared to other local and personal wireless technologies (e.g., Bluetooth and ZigBee), WiFi equipments (e.g., Smart-phone and WiFi AP) are commonly equipped in most houses. Therefore, this paper proposed a remote controlled system that can be accessed by smart-phone through a WiFi connection. The details of implements are described as follows.
A. Application Trends of DC Inverter Motor

Traditional fixed-frequency motor requires a large current to start the motor’s operation. When the temperature meets the setting of air-conditioning (or refrigerator), the motor stops running, else restarts the operation. So that, the iteration of startup and shutdown of motor causes inefficient power utilization and unpleasant operation noise.

Inverter control technology first applied to AC motor and then to DC motor recently. In general, inverter motor can save more 20% power consumption than fixed-frequency motor. Additionally, DC motor can achieve up to 10% power usage efficiency compared to AC motor. Because of the efficient energy consumption performance of DC inverter motor, thus DC inverter technology is popularly applied to consumer electronic appliances.

B. Principals of Inverter Control

Inverter control for its major purpose is to achieve smoothly adjusting spin speed of motor. When inverter is applied to the motor drive, induction circuit is controlled by the continue variations of alternate current (AC) and frequency to provide different spin speeds (i.e., Revolution Per Minute; RPM) and rotational torque. As a result, by changing voltage and frequency, the motor can react to different speed and torque performance. However, control by analog (i.e., AC) signals tends to drift over time. Besides, analog circuit is vulnerable to noise and interference of environment. Actually, DC inverter control is a better alternative solution.

Compared to analog, digital signal represented by discrete voltage levels. At most conditions, the output can be only two voltage levels of “high” and “low” to represent two states of “on” and “off” (or “true” and “false”) for digital control. And then a PWM (Pulse Width Modulation) sequence of signal is generated by the high-frequency iterations of switching between the two states of “on” and “off” as shown in Figure 1.

\[
\text{CLK} \quad \begin{array}{c} \text{on} \end{array} \quad \begin{array}{c} \text{off} \end{array} \quad \text{T (100%)}
\]

\[
\text{PWM} \quad \begin{array}{c} \text{on} \end{array} \quad \begin{array}{c} \text{off} \end{array} \quad \begin{array}{c} \text{on} \end{array} \quad \begin{array}{c} \text{off} \end{array} \quad \text{T_{on} (25\%)} \quad \text{T_{off} (25\%)}
\]

Figure 1. Signal characters of pulse width modulation.

The major advantage of control by using digital signal is that signals from control master (i.e., processor) to control slave (i.e., motor) are all digital. That means there no further signal conversion is required between formats of digital and analog. Additionally, the anti-jamming capability of digital signal is much better than that of analog. The reason is that an error condition of digital signal incurs only when the strength of interference noise is strong enough to have a substantial impact to cause the signals’ voltage level changing from high to low (or from low to high). PWM is to control the duty cycle of generated pulse of signal. As Figure 1 shows, a cycle length is \( T = T_{on} + T_{off} \). In a motor’s operation cycle, the operation is “on” during \( T_{on} \), else is “off” in \( T_{off} \). \( \delta \) is the percentage of “on” period of one cycle:

\[
\delta = \frac{T_{on}}{T_{on} + T_{off}} \times \% = \frac{T_{on}}{T} \times \%
\] (1)

Then the average voltage \( (V_{avg}) \) can be represented as:

\[
V_{avg} = \frac{T_{on}}{T_{on} + T_{off}} \times V = \frac{T_{on}}{T} \times V = \delta \times V
\] (2)

In which, \( V \) is the voltage of the supplied DC power. It can be observed that when the “on” percentage is adjusted, the average voltage \( (V_{avg}) \) is changed, then the spin speed of the motor is altered accordingly.

C. PWM Control in Traditional Embedded System

Traditional embedded system such as Arduino can support PWM signal output [7]. To take Figure 2 as an example, which applying the PWM signals to a DC motor. When the PWM signal always output as 0V (i.e., 0% Duty Cycle), it is to stop the motor’s spinning. If the 5V output takes an advantage of 25% to a cycle time (i.e., 25% Duty Cycle), it means that the motor operates at a quarter (1/4) of it full speed. In the case of the signal keeps 5V output (i.e., 100% Duty Cycle), the motor spins as fast as it can.

![Figure 2. Duty cycle variations of pulse width modulation [7].](image)

In our studies, the PWM functions that Arduino provided are limited. For example, output frequency is up to 1K Hz, and levels of duty cycle adjustment are restricted and indirectly [7]. Depending on the specification of the adopted motor, these restrictions could lead to a loss of performance. Besides, another insufficiency is for continuous variation control of duty cycle, that could lead to a loss of computing burden in a software-based implementation. For example, following code is to repeatedly make a motor’s spin from stationary to its fastest speed, and then stationary. In other words, the spin speed of the motor is from low to high, then high to low, iteratively. The increasing or decreasing of speed is gradual and the motor’s operation shall be smooth.
```cpp
int speed = 0;
int difference = 1;
int delaytime = 30;

void setup() {
  pinMode(3, OUTPUT);
}

void loop() {
  analogWrite(3, speed);
  speed = speed + difference;
  if (speed <= 0 || speed >= 255) {
    difference = -difference;
  }
  delay(delaytime);
}
```

According to the above program code, inevitably, the microprocessor shall continually make function call each time when the variable of speed is calculated and updated. In fact, to make continuous change of a signal’s duty cycle such as PWM, it is more reasonable to be implemented by hardware in order to reduce such routine processing burdens to the microprocessor. However, to the best of our knowledge, currently, Arduino cannot provide such function (i.e., continuously changing duty cycle of PWM) by hardware. Therefore, this paper implemented a PWM control module and integrated it in a FPGA-based platform to be a remote controlled system.

III. FPGA-BASED PWM CONTROL MODULE

This section describes our proposed FPGA-based PWM control module and related efforts for the system integration.

A. System Functional Blocks

In Figure 3, the green block in the upper left corner indicates a PWM control module implemented by this paper. The PWM module is integrated in the FPGA-based SoC (System-on-Chip) via a connection to the system bus (i.e., Avalon System Fabric). On which, there are other functional modules such as NIOS II microprocessor, memory controller, clock generator, and so on.

B. Interface of PWM Module

Interface signals between the PWM module and system bus are listed in Table 1.

<table>
<thead>
<tr>
<th>Name</th>
<th>IO</th>
<th>Width</th>
<th>Subhead</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLK</td>
<td>I</td>
<td>1</td>
<td>Clock a</td>
</tr>
<tr>
<td>RST</td>
<td>I</td>
<td>1</td>
<td>Reset</td>
</tr>
<tr>
<td>ADD</td>
<td>I</td>
<td>2</td>
<td>Address</td>
</tr>
<tr>
<td>WE</td>
<td>I</td>
<td>1</td>
<td>Write Enable</td>
</tr>
<tr>
<td>WD</td>
<td>I</td>
<td>32</td>
<td>Write Data</td>
</tr>
<tr>
<td>RE</td>
<td>I</td>
<td>1</td>
<td>Read Enable</td>
</tr>
<tr>
<td>RD</td>
<td>O</td>
<td>32</td>
<td>Read Data</td>
</tr>
<tr>
<td>PWM</td>
<td>O</td>
<td>1</td>
<td>PWM Signal</td>
</tr>
</tbody>
</table>

a. The CLK is 50MHz in this system.

We used ALTERA QSYS [8] to merge the designed PWM module as one component of the QSYS library.

![PWM module of QSYS component library](image)

In which, the signals of address, write, writedata, read, and readdata are respectively mapping to ADD, WE, WD, RE, and RD in Table 1. CLK and RST are connected to clock and reset of the embedded system. Besides, we allocated address 0x0150~0x0153 in IO space to the registers of the designed PWM module (ref. Table 2).

C. Timing Diagram of System Bus Signals

Operation relationships among signals are displayed as a timing diagram as shown in Figure 5.

![Timing diagram of system bus signals](image)

In the implemented SoC, connections among the PWM controller and other modules are illustrated in Figure 6.
Additionally, a register address map including name and description of each register is shown in Table 2.

TABLE II. REGISTER ADDRESS MAP OF PWM MODULE

<table>
<thead>
<tr>
<th>Name</th>
<th>Address offset</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONTROL</td>
<td>0x0000</td>
<td>System Control</td>
</tr>
<tr>
<td>CYCTIME</td>
<td>0x0001</td>
<td>PWM Cycle Time in unit of clock cycle</td>
</tr>
<tr>
<td>HIPDCYC</td>
<td>0x0002</td>
<td>PWM Hi Period Cycle</td>
</tr>
<tr>
<td>Reserved</td>
<td>0x0003</td>
<td>Reserved</td>
</tr>
</tbody>
</table>

a. The address based is 0x0150~0x0153 in IO space.

IV. DESIGN FRAMEWORK AND DEMONSTRATION

In this section we will provide an introduction to our proposed smart-phone remote controlled system. The related software and hardware architectures are provided as follows.

A. Design Framework

As Figure 7 shows, the implemented system includes a user control interface on the smart-phone. Through a connection to the WiFi AP, user’s control messages can be passed to the FPGA-based development board. The embedded system will process the received messages by a hardware and software co-design and then generate PWM signals to a power module to drive the motor’s operation.

B. Demonstration

As Figure 8 shows, we integrated the designed FPGA-based remote controlled system into a commercial electric fan to implement a smart-phone remote controlled IoT-type electric fan. User can connect to the IP-based web-server in the electric fan to access the control page. The control settings include power control, speed adjustment (supposed 100 levels), and a wisdom mode that is hardware-implemented to make the spin’s speed from low to high, then high to low, iteratively. A video can be demonstrated on YouTube [9].