****

**Zhejiang Sci-Tech University**



**题 目: Arduino产生周期和占空比可调的方波信号**

姓 名： 吴亚雄

班 级： 机械工程2班

学 号： 201820501050

任课老师： 李晓明

Arduino产生周期和占空比可调的方波信号

摘要：现在需要用Arduino输出方波信号，且方波信号的周期和占空比可调。Arduino里有简单的语句来实现方波信号，但是输出信号的频率不能更改。本文结合网上的文章信息和我的研究结果，解释一下实现频率可调的方波信号的过程。而周期和占空比通过串口与用户进行人机交互来发送指令来进行调整。

关键词：Arduino；方波信号；周期；占空比；人机交互

1. 引言

为了通过Arduino 可以输出方波信号，我们在本篇文章中进行了阐述。Arduino Uno开发板的接口可以通过简单语句来实现一个指定占空比的方波信号，但在实际工程问题中，我们需要这个方波信号对周期和占空比都是可调的，基于这个想法我们通过研究Arduino Uno开发板内部的三个Timer初步实现了这个目的。在这个研究的基础上我们想要通过人机交互调整参数来实现Arduino产生周期和占空比可调的方波信号，所以我们学习了Arduino的串口收发函数，然后基于此函数我们通过Arduino IDE进行了编码，初步实现了Arduino产生周期和占空比可调的方波信号，并且周期和占空比可以通过串口与用户进行人机交互来发送指令来进行调整。

1. Arduino 简介

Arduino是一款便捷灵活、方便上手的开源电子原型平台。包含硬件（各种型号的Arduino板）和软件（Arduino IDE)。它构建于开放原始码simple I/O介面版，并且具有使用类似Java、C语言的Processing/Wiring开发环境。主要包含两个的部分：硬件部分是可以用来做电路连接的Arduino电路板；另外一个则是Arduino IDE，也就是用户的计算机中的程序开发环境。用户只要在IDE中编写程序代码，将程序上传到Arduino电路板后，程序便会告诉Arduino电路板要做些什么了。

Arduino有众多的优点：①.跨平台。Arduino IDE可以在Windows、Macintosh OS X、Linux三大主流操作系统上运行，而其他的大多数控制器只能在Windows上开发。②.简单清晰。Arduino IDE基于processing IDE开发。对于初学者用户来说，极易掌握，同时有着足够的灵活性。Arduino语言基于wiring语言开发，是对 avr-gcc库的二次封装，不需要太多的单片机基础、编程基础，简单学习后，用户可以快速的进行开发。③.开放性。Arduino的硬件原理图、电路图、IDE软件及核心库文件都是开源的，在开源协议范围内里可以任意修改原始设计及相应代码。④.发展迅速。Arduino不仅仅是全球最流行的开源硬件，也是一个优秀的硬件开发平台，更是硬件开发的趋势。Arduino简单的开发方式使得开发者更关注创意与实现，更快的完成自己的项目开发，大大节约了学习的成本，缩短了开发的周期。

1. 软件实验

Arduino Uno开发板的5，6，9，10，3，11接口可以通过简单语analogWrite(pin, dutyCycle)来实现一个指定占空比的PWM。其中pin的值选择（5，6，9，10，3，11），dutyCycle的值在0～255之间，0为占空比0%，255为占空比100%。但是这种方式PWM信号的频率是固定的默认值，大约1000Hz左右（16MHz/64/256）。其次，手动切换高电平和低电平，再在中间加入delay函数，可以实现自定义频率的PWM，Arduino代码如下：

void setup()

{

pinMode(13, OUTPUT);

}

void loop()

{

digitalWrite(13, HIGH);

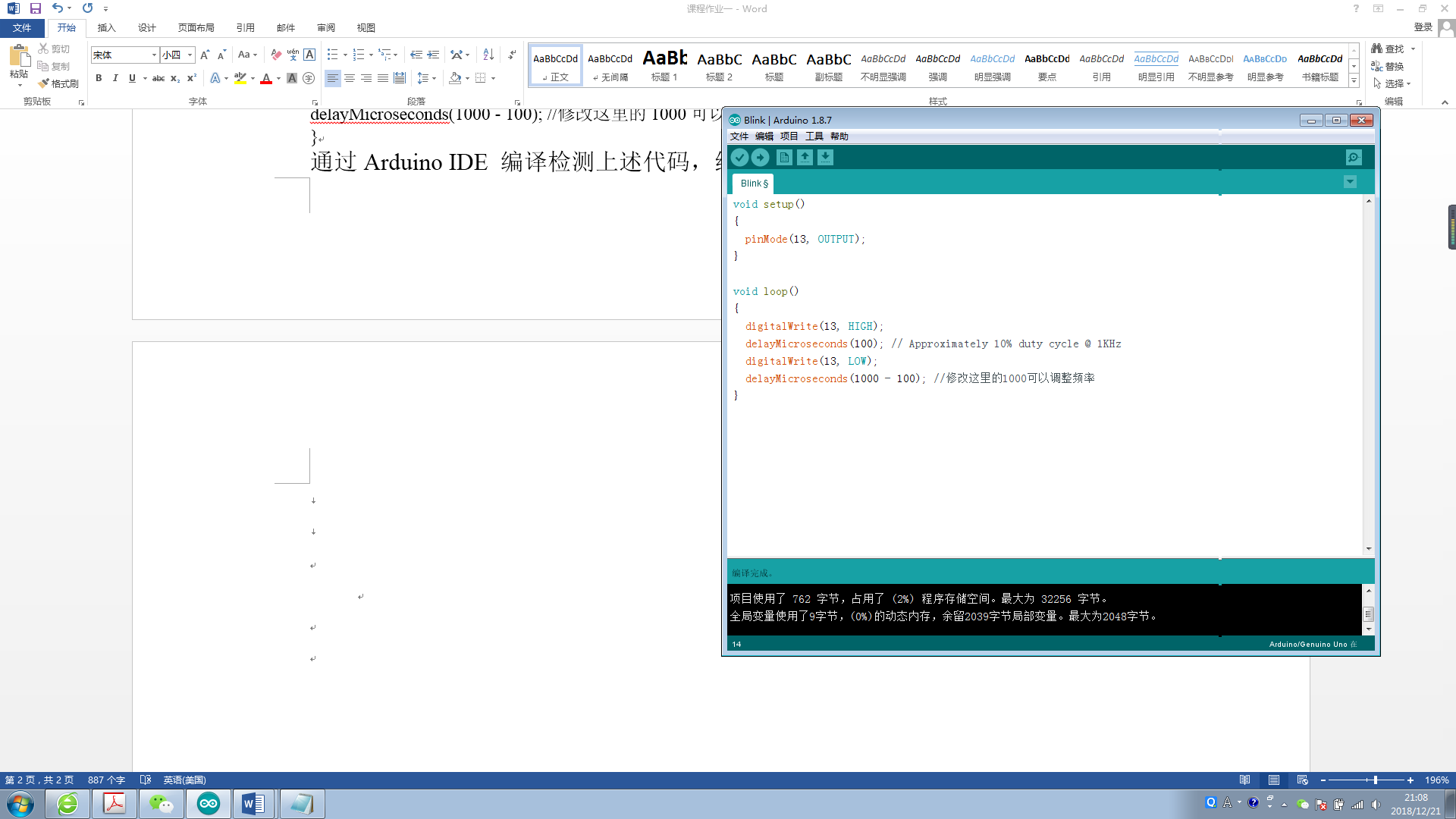
delayMicroseconds(100); // Approximately 10% duty cycle @ 1KHz

digitalWrite(13, LOW);

delayMicroseconds(1000 - 100); //修改这里的1000可以调整频率

}

通过Arduino IDE 编译检测上述代码，结果显示没有问题，检测如下：



但是，这种操作需要CPU全神贯注的查数，也就是说对CPU的要求较高，任何其他的进程的干扰会导致输出的信号频率不准。为了避免上述问题，我们需要底层的手段来控制Arduino实现PWM的频率调节。

Arduino Uno有三个Timer：Timer0，Timer1，Timer2。 三个Timer都可以自定义调整频率，但是各有特点。Timer0负责控制delay等函数，动了Timer0的频率会导致计时函数不准；Timer1的计数器是16位的，和Timer0，Timer2的8位计数器不太一样；Timer2的频率可调的档位更多，它有7档预除数，这里选择Timer2进行调节操作，代码如下：

void setup() {

// put your setup code here, to run once:

pinMode(3, OUTPUT);

pinMode(11, OUTPUT);

TCCR2A = \_BV(COM2A0) | \_BV(COM2B1) | \_BV(WGM21) | \_BV(WGM20); //Set Timer2 to varying top limit fast PWM mode

TCCR2B = \_BV(WGM22) | \_BV(CS22) | \_BV(CS21) | \_BV(CS20);//another way to set prescaler CS2=fff

OCR2A = 155; //Top value A

OCR2B = 30; //Toggle value B, Output at pin 3

//CS2 Divisor Frequency

//001 1 31372.55

//010 8 3921.16

//011 32 980.39

//100 64 490.20 <--DEFAULT

//101 128 245.10

//110 256 122.55

//111 1024 30.64

}

void loop() {

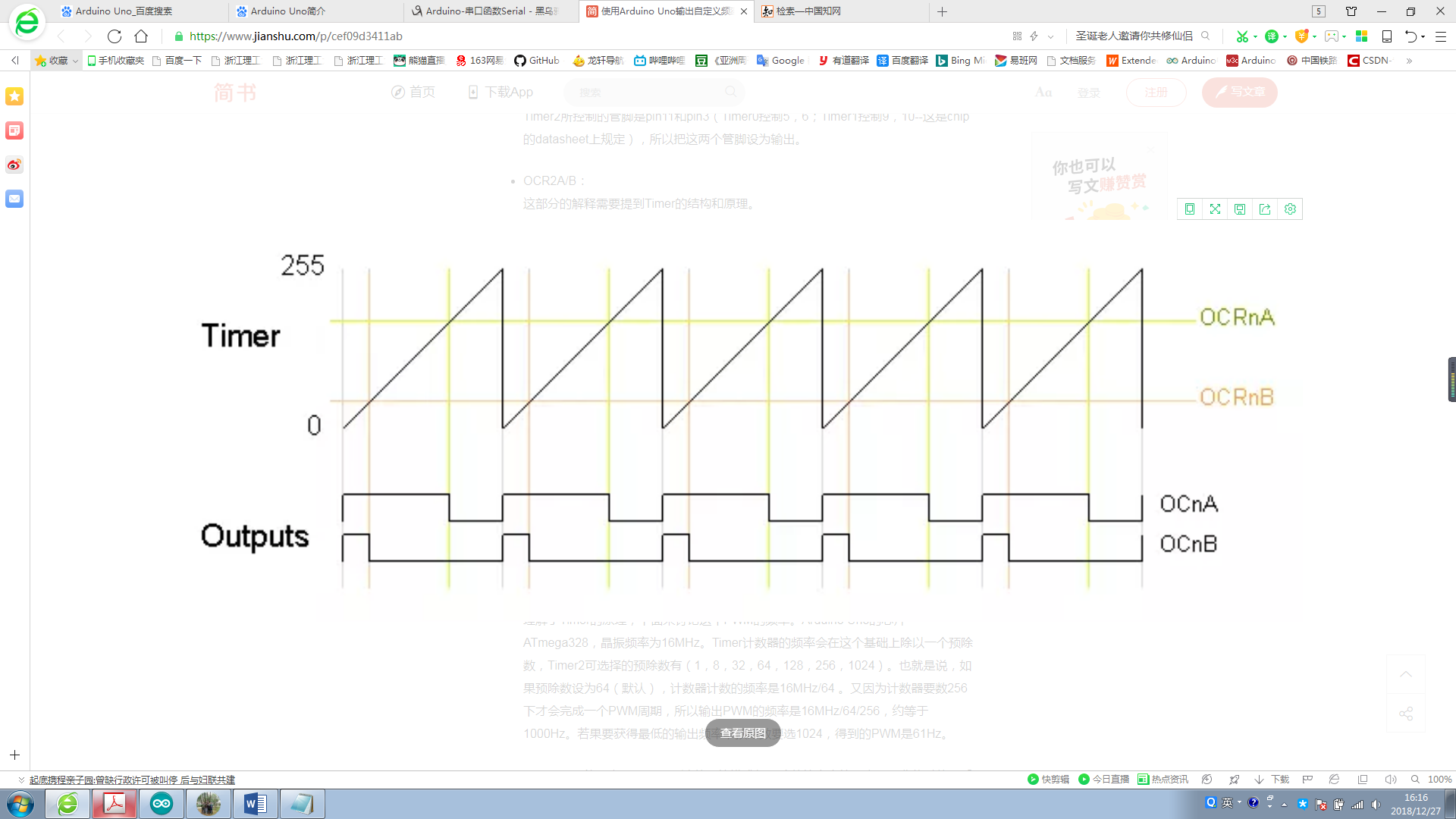
// put your main code here, to run repeatedly:

}

由以上代码可见，需要的设置分为三个部分：pinMode，TCCR2A/B，OCR2A/B（这里的2是因为选择了Timer2）。我们下面对这三部分进行一一阐述。

pinMode：Timer2所控制的管脚是pin11和pin3（Timer0控制5，6；Timer1控制9，10--这是chip的datasheet上规定），所以把这两个管脚设为输出。

OCR2A/B：这部分的解释需要提到Timer的结构和原理。每个Timer里都有一个计数器和两个比较寄存器。Timer2里计数器从0数到255（8位）然后归0继续从头数；Timer2的两个比较寄存器分别为OCR2A和OCR2B。比较寄存器就是你设置一个小于255的数，比如155。当计数器数到0时输出为高电平，数到155的时候改变输出为低电平。这样就实现了占空比的调节。在普通模式下（Fast PWM），OCR2A控制pin11的占空比，OCR2B控制pin3的占空比。如下图所示。



TCCR2A/B：理解了Timer的原理，下面来讨论这个PWM的频率。Arduino Uno的芯片ATmega328，晶振频率为16MHz。Timer计数器的频率会在这个基础上除以一个预除数，Timer2可选择的预除数有（1，8，32，64，128，256，1024）。也就是说，如果预除数设为64（默认），计数器计数的频率是16MHz/64 。又因为计数器要数256下才会完成一个PWM周期，所以输出PWM的频率是16MHz/64/256，约等于1000Hz。若果要获得最低的输出频率，预除数要选1024，得到的PWM是61Hz。TCCR2A/B就是来控制Timer2计数器的模式与预除数的大小的，由于是分位赋值，看起来怪怪的，我来解释一下。先说CS2位，这个就是来控制Timer2计数器预除数的：\_BV(CS22) | \_BV(CS21) | \_BV(CS20)的三部分由逻辑按位或“|”链接；每个BV是按位赋注（bit value）的意思；\_BV(CS22 )= 在CS2里，1<<2(把1左移2位) = 00000100；得到三部分分别是00000100，00000010，00000001；按位或最终得到0111；查代码里的表得到对应的预除数是1024。

一般将字符通过串口打印到电脑(用USB串口)已经够用了，但是当我们想把字符发送到另一个串口的时候，或是想用一块手头上已有的开发板发送数据给Arduino，然后让Arduino显示，问题顿时变得麻烦起来。为了实现从一个单片机的串口发送数据到Arduino上，然后用Arduino显示。我们先了解一下Arduino的串口收发函数(以下实例基于Arduino UNO)。

1. 接收函数Serial.read()

int Serial.read(void)

如果串行数据缓冲区有数据，这个函数会读取串行数据缓冲区的第一个字节，数据读取位置移动到下一个数据缓冲区，也就是说如果继续读取的话会读取下一个数据缓冲区的第一个字节。如果数据缓冲区没有数据，将返回-1。

1. 发送函数Serial.print()

Serial.print(val)

首先注意print是一个打印函数，主要用于在上位机上显示，让数据可视化，它会将数据用字符的形式(ASCII码)逐个发送到串口，一般不用这个函数来进行数据传输。既然print是一个打印函数，那么自然，参数val的类型是任意的。

Serial.print(val, format)

将val的值转换成format指定的形式发送，可为BIN，OCT，HEX，DEC。

1. 发送函数Serial.write

Serial.write(str)

str为字符串的首地址，buf为一个用来存放数据的数组的首地址。这个函数会先发送字符串，然后返回字符串的长度。

Serial.write(val)

val的数据类型为unsigned char，这个函数每次只发送一个字节的数据，所以val的值在0-255之间。

Serial.write(buf, len)

buf为要发送的数组的首地址(注意数组的类型为unsigned char) ，len为数组的长度。

回到开头的问题，如何从一个单片机的串口发送数据到Arduino上，然后用Arduino显示呢?我们可以先用单片机的print函数把数据打印到串口，然后用Serial1.readBytes把数据从串口读取回来(以ASCII码的形式) ，然后把ASCII码转换成字符，然后发送。通过以上的分析，我们可以写出如下代码：

int sinputPin = 10; /\*Change according to the hardware configure\*/

int soutputPin = 8;/\*Change according to the hardware configure\*/

int buzzPin = 7;/\*Change according to the hardware configure\*/

int wait;

void setup()

{

//set the IO working mode

pinMode(sinputPin, INPUT);

pinMode(soutputPin, OUTPUT);

pinMode(buzzPin, OUTPUT);

//Setup the serial port

Serial.begin(9600);

}

void loop()

{

//the next 5 lines are used to generate a pulse

//to trig the ultrasonic ranging unit

digitalWrite(soutputPin, LOW);

delay(100);

digitalWrite(soutputPin, HIGH);

delayMicroseconds(15);

digitalWrite(soutputPin, LOW);

//get response from the ranging unit

unsigned int distance = pulseIn(sinputPin, HIGH);

distance = distance/58; //in cm

if(distance < 20){

tone(buzzPin, 2000, 100);

wait = 200;

}

else if(distance > 100)

{

tone(buzzPin, 800, 100);

wait = 2000;

}

else{

tone(buzzPin, 1000, 100);

wait = 1000;

}

//display the results to the serial monitor

Serial.print("[");

Serial.print(millis());

Serial.print("]");

Serial.print(" ");

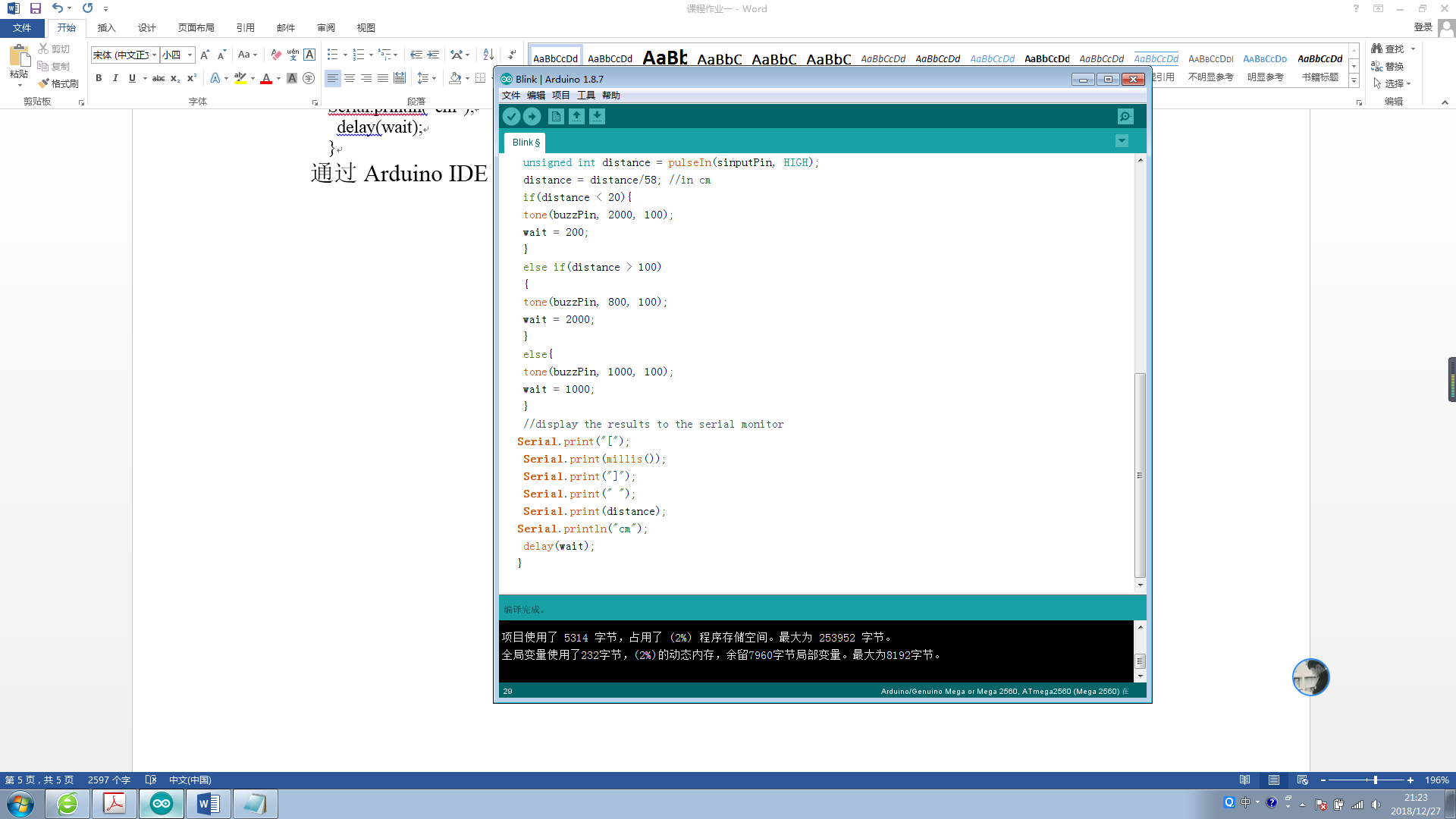
Serial.print(distance);

Serial.println("cm");

delay(wait);

}

通过Arduino IDE 编译检测上述代码，结果显示没有问题，检测如下：



1. 总结

在本篇文章中，我们对Arduino进行了简单介绍，在此基础上，明白了Arduino UNO的一些基本知识，最后在Arduino IDE 软件上进行编程，实现了在Arduino UNO指定的引脚上产生一个方波信号，并且可以通过串口与用户进行人机交互，通过发送指令调整该方波信号的周期和占空比可调。

1. 参考文献

[1]桂淮濛.基于Arduino的电子产品设计[J].电脑编程技巧与维护,2018(09):59-60+88.

[2]林广峰.用单片机实现频率可调的PWM控制信号[J].科技传播,2010(12):220-221.

[3]李秀山.周期占空比大范围可调方波信号源[J].辽宁师专学报(自然科学版),2004(03):12.

[4]陈丛浩, 张正中.Arduino的应用及发展前景[J] .轻工科技, 2018,(05).