**2020年新工科联盟-Xilinx暑期学校团队项目设计文档**

|  |  |
| --- | --- |
| **设计作品名称** | 摇摇乐 |
| **参赛队员姓名、学号、学校及所在院系** | 孙青楠 2018112814 西南交通大学 信息科学与技术学院  谢麟冰 2018112779 西南交通大学 信息科学与技术学院 |
| **所在班级** | 通信工程2018-01班 |
| **Github链接** | https://github.com/Ankh-dot/Shake\_FPGA.git |

**第一部分**

设计概述 /Design Introduction

1.设计目的：通过本设计，对这次赛灵思暑期学校所学内容的一次检验，充分结合这几天所学的FPGA相关的知识，运用到实际项目中，加深对FPGA的理解，从实践中激发自己的兴趣。

设计特色：此项目是基于SEA-S7开发平台实现。首先由FPGA采集开发平台上自带的IMU传感器数据，经过滤波后得到精确数据，计算得出一分钟开发板被摇动的次数。将最终数据发送给ESP32，通过板载ESP32芯片可以便捷的搭建SEA Board本地传感器与AWS IoT云端服务器连接，由此可搭建FPGA+ESP32+AWS IoT的数据处理与通信平台，实现云数据的功能。

主要功能：预期结果为当开发板上电之后，打开开关，通过摇晃开发板得出摇晃总次数和时间，并计算一分钟晃动平均次数。然后将一分钟晃动次数上传到AWS云端平台显示。

应用场景：摇摇乐可以作为计步器、测振仪和一些简单的游戏项目的基础功能。

知识点：陀螺仪的原理和数据传输方式，ESP32的QSPI通信协议，AWS IOT与ESP32的连接，UART通信协议，IIC通信协议。

2.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 组员 | 完成内容 | | | | | | 贡献百分比 |
| 孙青楠 | 陀螺仪驱动程序 | 摇晃次数算法 | QSPI从机配置 | AWS平台连接 | 设计报告编写 | 顶层模块搭建 | 50% |
| 谢麟冰 | IIC程序编写 | UART驱动编写 | 时钟模块 | AWS平台连接 | 设计报告编写 | 代码改动 | 50% |

**第二部分**

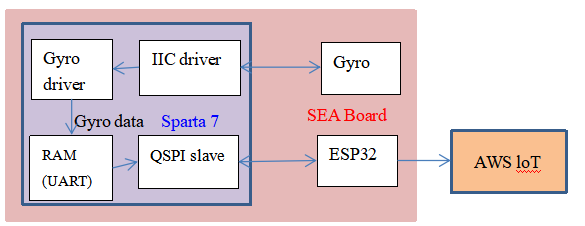
系统组成及功能说明 /System Construction & Function Description

系统组成：由FPGA（基于Sea\_Board开发板）、ESP32、AWS\_IoT云平台三部分组成。

功能实现：开发板上电后，打开开关，通过摇晃开发板得出摇晃总次数和时间，并计算一分钟晃动平均次数。然后将一分钟晃动次数上传到AWS云端平台显示。

功能说明：要实现整个项目设计，整个系统的核心任务就是搭建FPGA+ESP32+AWS IoT的数据处理与通信平台，实现云数据的功能。整个实现可概括为：SEA Board感应——FPGA接受感应并转换成数据——数据从FPGA传输回ESP32——ESP32芯片通过本地传感器将数据上传至AWS。

具体操作实现方式如图所示：



MQTT

通过vivado编写verilog代码，通过搭建数字积木的方式，建立顶层模块和底层模块。

当晃动开发板时，通过传感器感应IIC\_driver模块开始工作，解析IIC信号，并在板载陀螺仪中获取陀螺仪数据，然后Gyro\_diver开始工作解析原始的陀螺仪数据，并判断有效陀螺仪数据，通过运算将陀螺仪数据转化为人手摇开发板的次数，并将处理好的数据串行存放于RAM中。

最后ESP32芯片将数据从RAM中取出，通过QSPI\_Slave模块将数据通过PC端传输到Aws云平台显示，至此整个系统的运作完成。

操作流程：

摇晃

Arduino烧录代码

AWS平台点击测试并订阅

Bit文件加载

**第三部分**

完成情况及性能参数 /Final Design & Performance Parameters

完成情况：分模块完成作品（已实现的功能）：

一、FPGA方面可分为四个模块：

1.IIC diver:主要负责解析IIC信号，并在板载陀螺仪中获取陀螺仪数据。

主要参数为：

从机地址：Slave\_Addr,

寄存器地址：Reg\_Addr\_h,Reg\_Addr\_l

传入数据：Data,

判断依据（写入和读出）： Write,Read,

需要传输的数据：i\_data\_w

需要读取的数据：i\_data\_r

判断依据（IIC驱动模块工作还是空闲）：Busy

IIC时钟：SCL

判断数据是否输入：i\_SDA,

控制数据方向，高电平数据输出：SDA\_Dir

输出数据： o\_SDA

2.UART0\_ctrl模块：将传输的数据在串行通信与并行通信之间加以转换。将陀螺仪数据串行输出并存入RAM中。由时钟模块，数据打包传输模块和驱动模块组成。

其中UART0\_Driver模块还可细分为：UART时钟 Clk\_UART、数据发送 UART\_Tx、数据接收UART\_Rx三个模块。

主要参数为：

系统时钟： clk

复位信号： rst

数据发送使能信号：Tx\_en

数据接收使能信号：Rx\_en

设置波特率：Set\_BaudRate

波特率:BaudRate

数据接收：Rx

数据发送：Tx

传入数据：Tx\_Data

接收数据：Rx\_Data

数据接收应答信号： Rx\_ACK

数据发送应答信号：Tx\_ACK

UART时钟：clk\_UART

3.Gyro\_diver0模块:负责通过IIC读取原始的陀螺仪数据，并将陀螺仪数据存放于RAM中。

首先初始化陀螺仪，定义初始状态，设置暂存态。以时钟信号为频率，判定IIC信号，检测到的数据实时存储并串行发送。数据主要包括磁力计三轴数据，角度三轴数据，温度。

1. RAM模块：该模块分为两大部分，一是RAM的IP核添加，二是陀螺仪数据上传到RAM中。data\_in是RAM的数据输入口，data\_out是数据输出口。其中mem为寄存器，暂存陀螺仪检测到的数据。

5.QSPI\_slave模块：搭建QSPI的从机配置，添加QSPI交互模块。 Arduino侧通过QSPI利用SeaTrans函数读取输入数据。

1. Arduino方面：

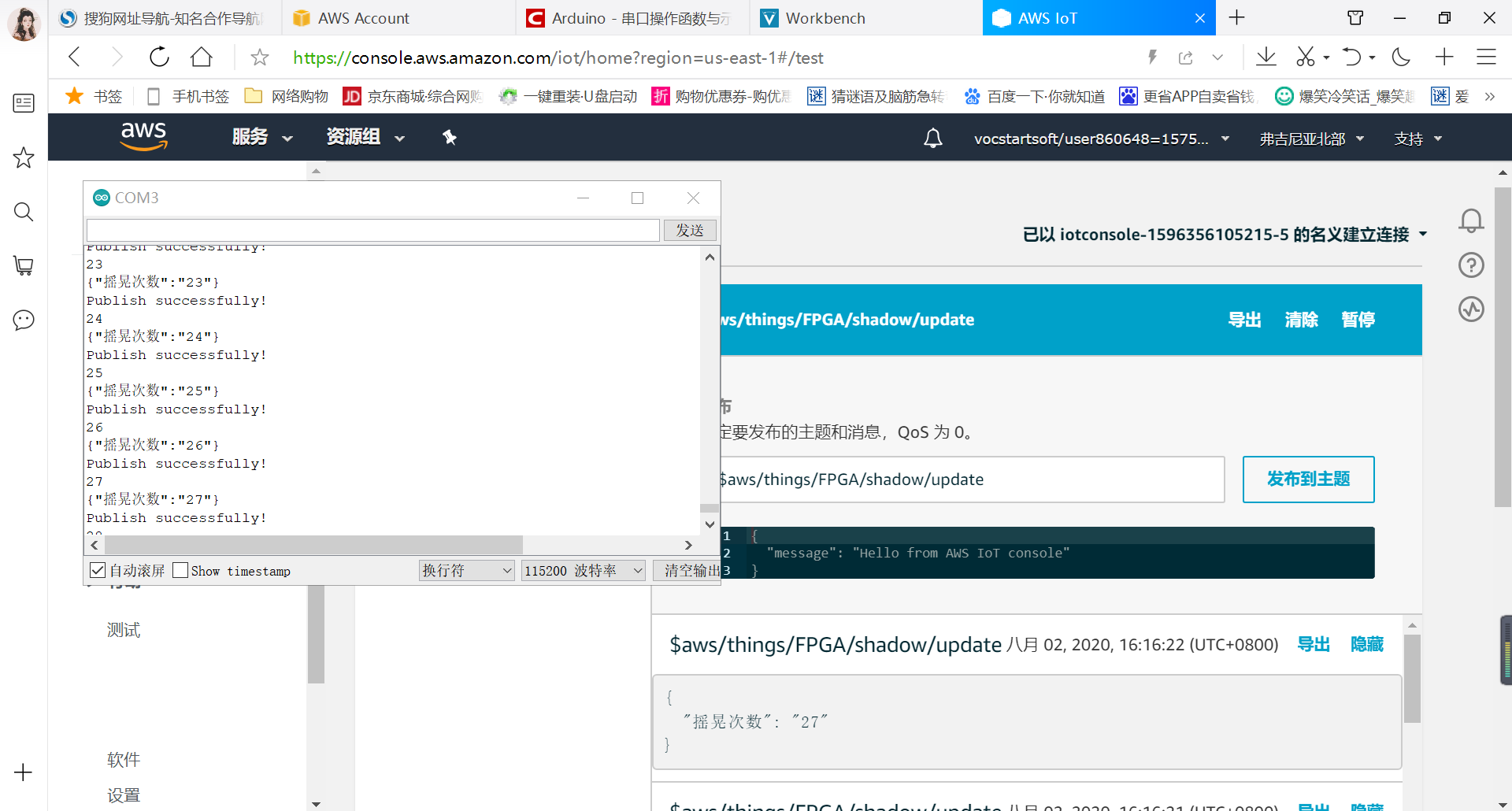
将数据处理代码写于Arduino中，通过QSPI，将RAM中的数据读出后，与设定值进行比较判断，并根据判断结果对摇晃次数进行累加或者不加，实时输出数据，显示在串口和AWS云端。

1. 云端方面：

首先在AWS平台建立物品和证书，并添加策略，其次在MQTT上设置主题，然后在AWS上订阅主题，可以看到MQTT端发布的消息，AWS上可以实时接收。下图为MQTT端发送的消息显示在AWS：

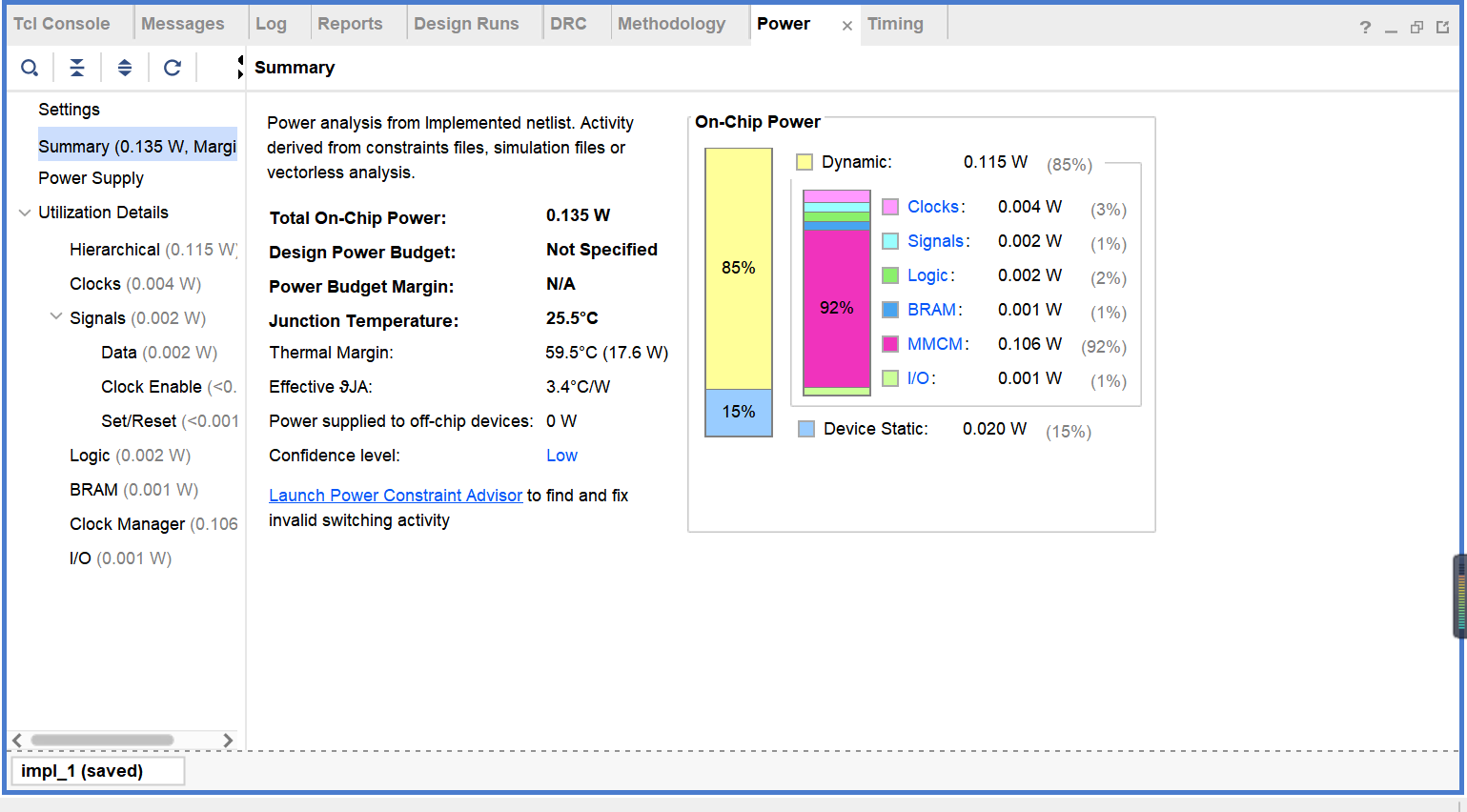


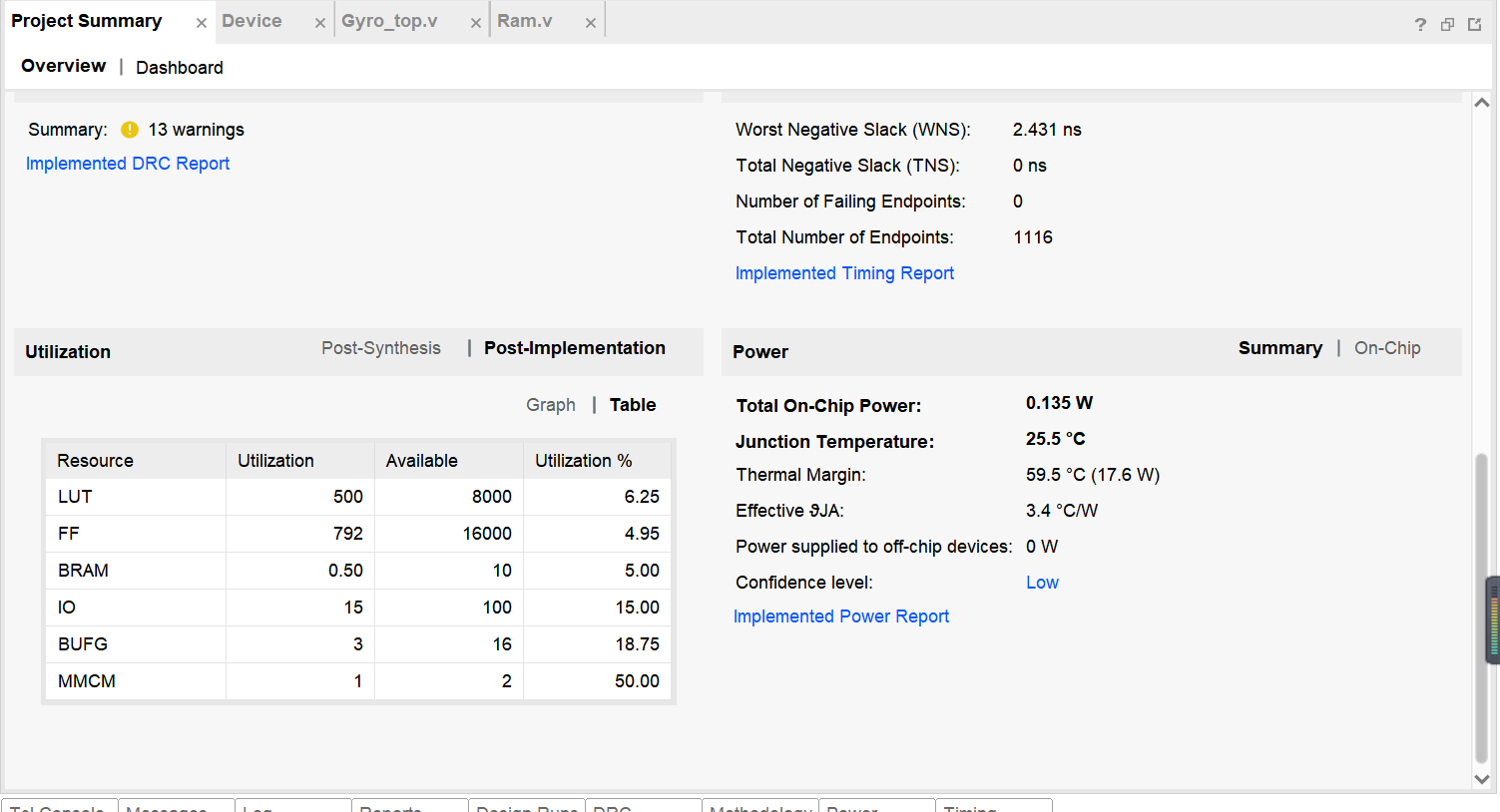
对PubTest代码进行相应的修改，改写证书和密钥文件内容，实现ESP32wifi和AWS云端的连接。截图如下：



演示视频会发布在GitHub上

性能参数：





未完成部分/问题描述：

滤波算法未实现，检测摇晃次数尚不精确，偶有延迟。在进行实验过程中，AWS平台不稳定，抑或是网络原因，导致ESP32偶尔会与AWS平台断开。其次，次数的统计不够灵敏，或许是设定的判断条件还不够精确完整，这是未来开发再进一步精确的地方。

**第四部分**

总结 /Conclusions

谢麟冰：

此次暑期学校的学校，带我走进了FPGA这一新的领域，通过这几天的学习，我掌握了FPGA方面的知识，虽然我现在还是处于初学者阶段，但是对FPGA的兴趣已经成功地激发。

通过本次项目设计，充分锻炼了我们的能力，通过自己的努力，查阅资料，理解题意，将这几天在暑期学校所学的知识充分运用到项目设计中，对触及到还没来得及学习的知识，能够通过查阅资料、询问老师等途径解决，提高了自学能力。总的来说，本次设计不仅是对我们关于FPGA知识理解的考验，也考验了我们的自学能力、团队协作能力。尽管最后项目设计实现的功能可能十分的简单，但也是像我们这种初学者尽全力得到的结果，这将是打开我们对FPGA这扇大门的钥匙，因为对FPGA的兴趣已经成功地激发，相信未来我们能做出更好的项目设计。

孙青楠：

六天的项目设计转眼就到了结束的时候。在这段时间里，遇到了很多困难。在这之前，FPGA对于我而言是陌生的东西，所有相关知识只停留在数电课程上。问题从刷固件，烧录bit文件开始，断断续续持续到项目结束。我发现有很多细枝末节的小问题，难以发现，却影响甚大。常常一个人反反复复过流程都发现不了错误关键所在。每每这种时候，我都会求助同学，换一双眼睛就能看出正确流程下，那一丝差错。

虽说项目整体的完成不是很顺利，但是实实在在学到了很多东西，也借助了同学伙伴们的力量，最终完成了项目设计。就像一张白纸一样摊开在我的面前，我不断努力，寻找，想尽方法填满这张纸。我学到了vivado软件的使用，学到了ESP32的一些通信协议，学到了AWS IOT平台的搭建，对FPGA又多认识了一些。在这几天里，我充分认识到自学能力的重要性，也体会到个人能力水平的不足，还感受到协作的重要性。我又想起当初选择工科的初衷，就是为了通过自己的设计，能够实现一些功能。比如小灯的亮灭，游戏的运行，或大或小的功能都是像我们这种开发者最本质的快乐吧。我感叹越来越多功能强大的开发板的出现，它们成为了我们实现心中所想的一大助力。