Smart Tag Project

# Introduction

Hi I am making a worksheet that needs to be updated by the members of this project, I have formatted it to display the date the page was written in the header. Professor wants me to share our progress per week. I have Input my share of the progress, please update whatever you have worked on in the previous week. This worksheet has been divided between the three of us, The sheet starts with the Name of the member . under the name is the time stamp of the day you are writing it. Once you have written the progress of the previous week, for the next week just press enter till you go to a new page(or add a page break ) and put the time stamp on the new page, use the format of the time stamp as **heading 2** (so that we can make a content list out of it). The update does not need to be perfect just a rough draft of what you did and how you did it and what results you gout out of it.

NOTE: I have added the section break right below the --------X-------X------- line, so just input your work above this line under your name. and do not delete this line or anything under this line.(or add anything under this line).

**Project Members:**

1.Vidur Nayyar

2.Young Lee

3.Raghavi Raghuraman

# Vidur Nayyar

## Monday, August 17, 15

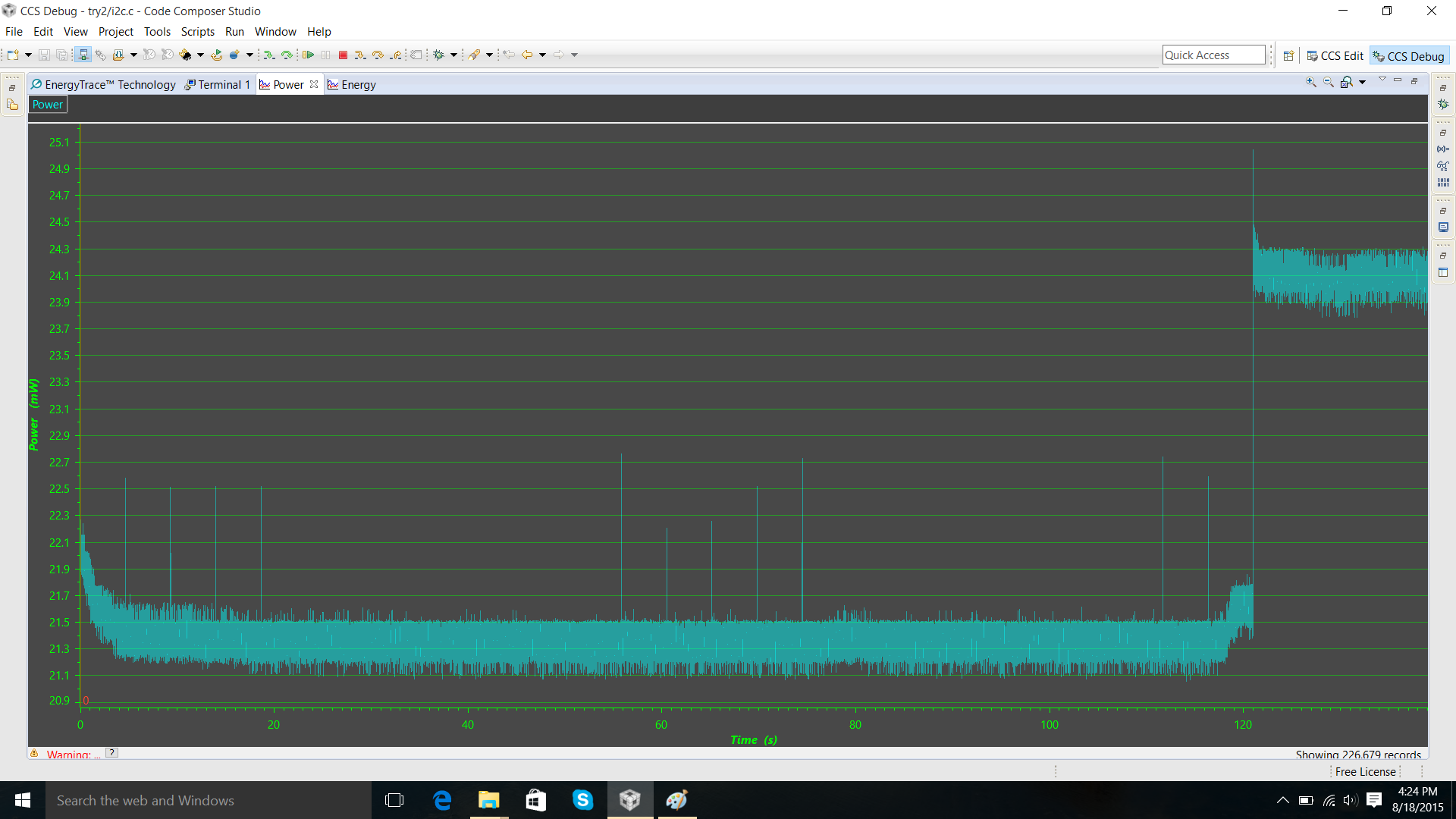
While checking the power consumption, I found out that the code performing the USART is consuming a lot of power.

The capacitive touch uses little power, but needs to be modeled better depending on for how long we want to keep it on and what should be the interval when we should check for touch. Greater the interval = lower the battery consumption.

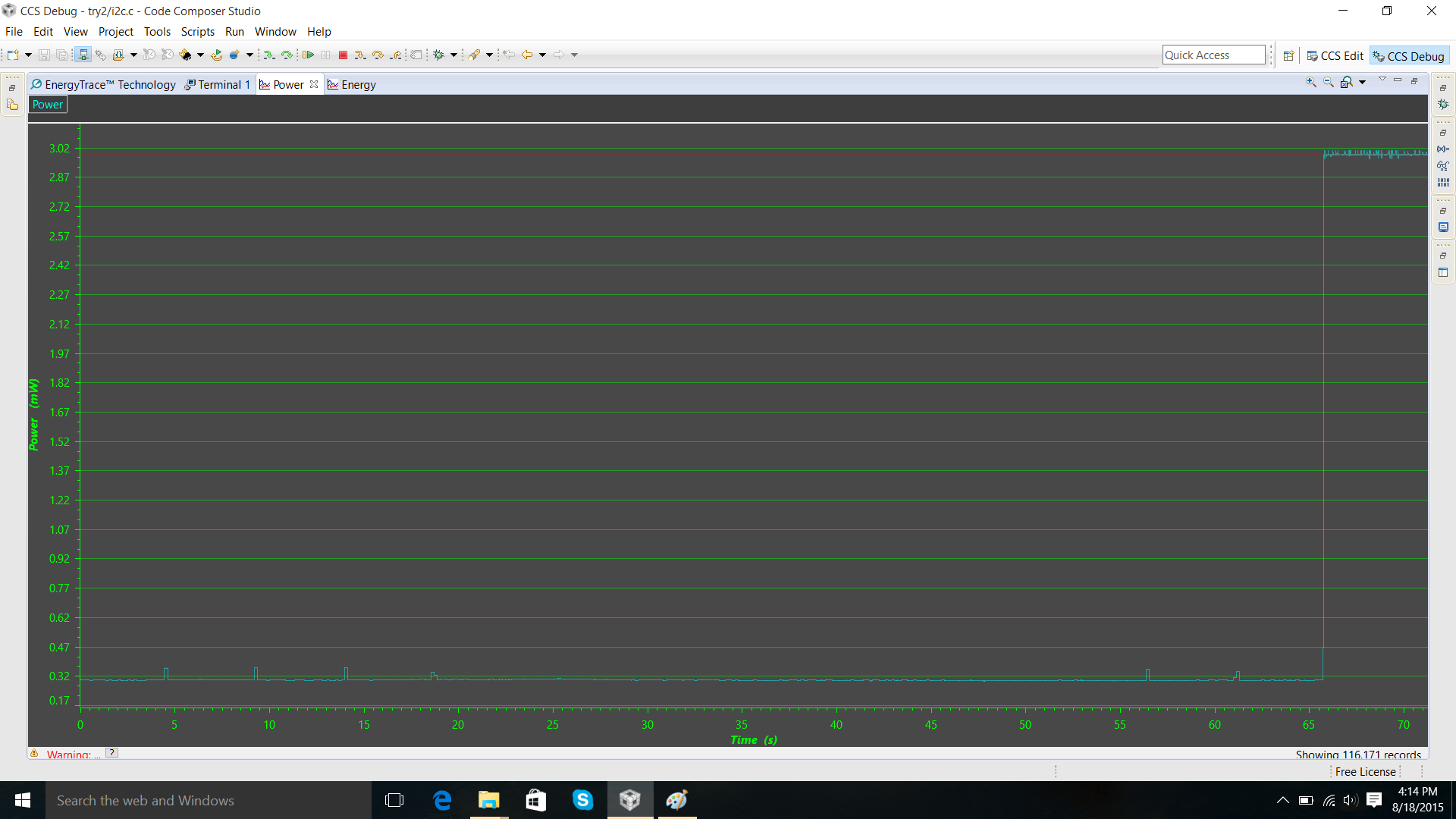
I also found that the RFID transmission takes up a lot of power, So we Either need to keep this to the minimum or try to find a more efficient way of doing so. Even just connecting the RFID tag to the controller sucks out a lot of power, so I feel that we should provide power to the RFID tag via the controllers pins so that we can give it power when the we want to, rest of the times we don’t give it any power. This might have 2 disadvantages, which are unclear at this point, I predict this might draw extra current from the pins frying the pins of the controller (which can be corrected by using a transistor or fast switching mosfet to act like a switch to provide the RFID tag with power when the controller wants to.). The other issue is that by not providing the tag with any power, the tag might not use its passive RFID capability to bounce back the RFID information to the receiver, this issue can be dealt with by going through the worksheet of the RFID chip or checking out the other RFID chips which can be used for similar things and don’t need power to act as a passive RFID tag.

The Power consumption was checked under different situations and found out that the power was wasted due to connecting the RFID tag to the Controller. Nearly 20mW of power is drained by simply connecting the RFID to the controller.

The figure below shows the power consumption when the RFID chip was connected to the controller and then touched at 120th second.



I also noticed that the power consumed to write the data onto the RFID chip’s EEPROM is not much, it is just a jump of 2.7mW(this was verified by checking the jump when the RFID tag’s power was connected to the controller and when it wasn’t connected to the controller.)



The power consumed while the controller was in sleep mode 3 is 0.314mW and the power consumed while checking the capacitive touch is 0.386mW. The capacitive touch is very quick and the controller stays at at the 0.386mW level for a very small period of time. See the figure below.



In such a condition the battery will last for 80 days if its not touched. The issue of power drain occurs when the RFID chip is given Power via the controller, this can be seen below. The figure below is the power consumption of the circuit when a simple led blink code is running on the controller which is not even remotely linked to the RFID chip, still the power consumption is around 20mW as this is the power absorbed by the RFID tag the moment it is connected to the power of the controller. We need to find a way to avoid this draining of power.

If we only power the rfid to write the code, at the receiver the rssi value will peak for that point. Coz when supply power, it requires less power from rfid.

## Friday, August 21, 15

I think the issue is arriving when the RF reader is reading the memory of the chip and at the same time you make a I2C write command , which is ignored. because this command is ignored so there is no ack from the chip to the controller. And because the controller waits for an ack in the while loop so it gets stuck.

I am attaching the part of the data sheet that explains the conflict scenario.

RF interface/I2C interface arbitration

The UCODE I2C needs to arbitrate the EEPROM access between the RF and the I2C

interface.

The arbitration is implemented as following:

• First come, first serve strategy - the interface which provides data by having a first

valid preamble on RF envelope (begin of a command) or a start condition and a valid

I2C device address on the I2C interface will be favored.

• I2C access to the chip memory is possible regardless if it is in the EPC Gen2 secured

state or not

• During an I2C command, starting with an I2C start followed by valid I2C device

address and ending with an I2C stop condition, any RF command is ignored.

• During any EPC Gen2 command any I2C command is ignored

## Wednesday, September 9, 15

-------------------------------------------X-------------------------X--------------------------------

# Raghavi Raghuraman

## Monday, August 17, 15

-------------------------------------------X-------------------------X--------------------------------

# Young Lee

## Monday, August 17, 15

-------------------------------------------X-------------------------X--------------------------------