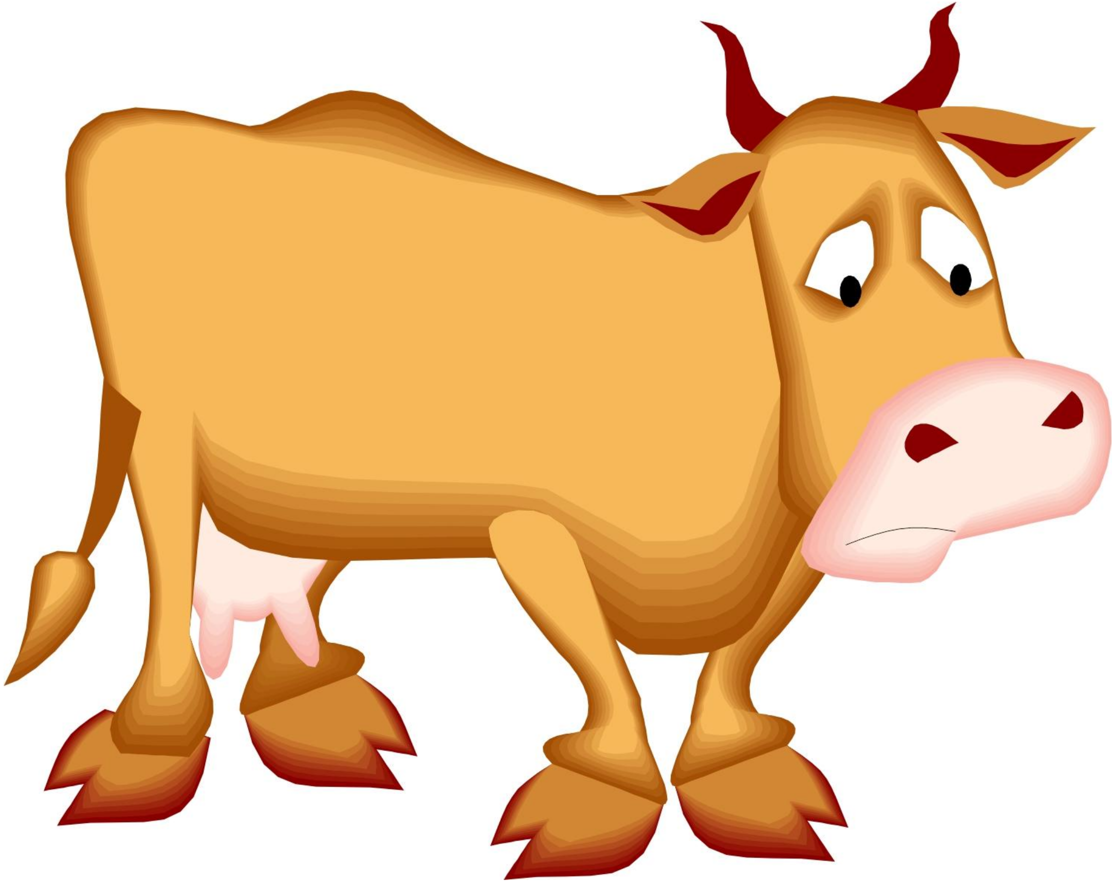
ECE 445 Fall 2017
Livestock Temperature Monitor

Group #4

Cain Benink
Michael Goldstein
Yue Wang

Introduction / The Problem

- The process of identifying sick animals is too manual
- Treatable illnesses generally have already been around for days before external symptoms are visible



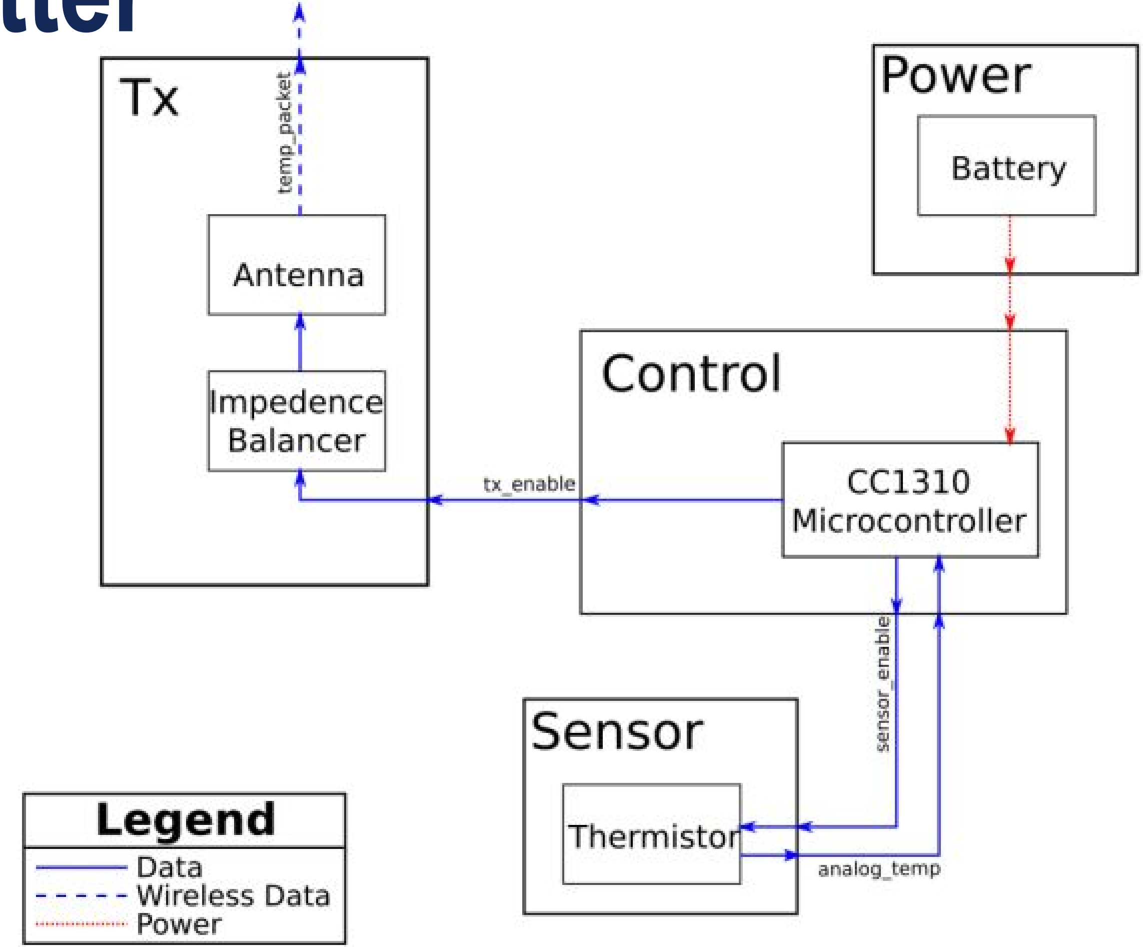
The Solution / Our Project

- Design a circuit to measure cow temperatures
- Put the circuit inside existing cow ear tags
- Periodically transmit temperature data to a receiver

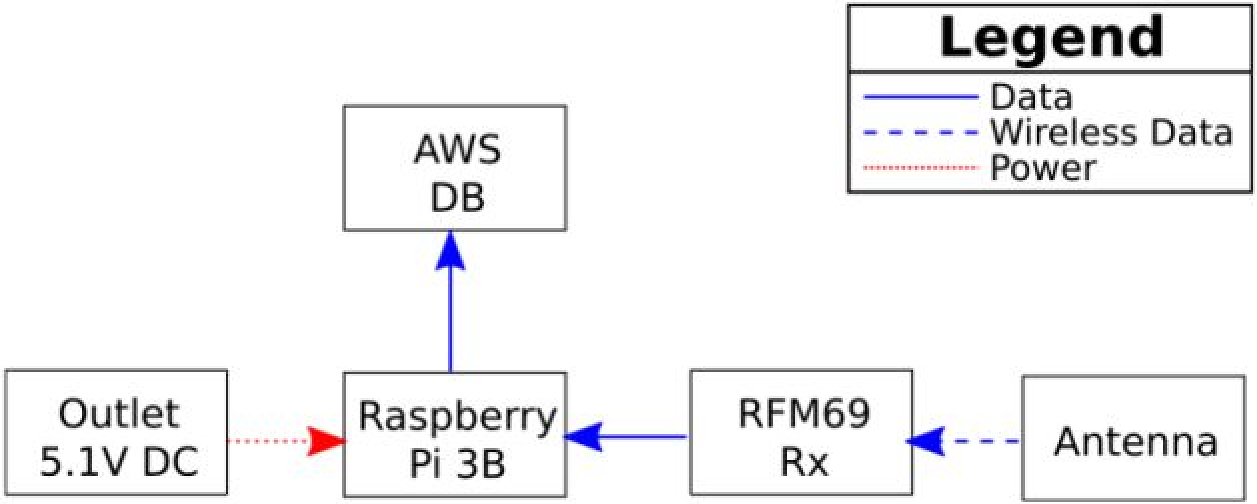
High Level Requirements

- Must last 18 months (lifespan of beef cattle)
- Read temperatures with enough resolution to detect fevers (within 0.2 degrees Celsius)
- Transmit frequently enough to detect fevers in a short amount of time (12 hours)
- Have a bulk unit cost of less than $10

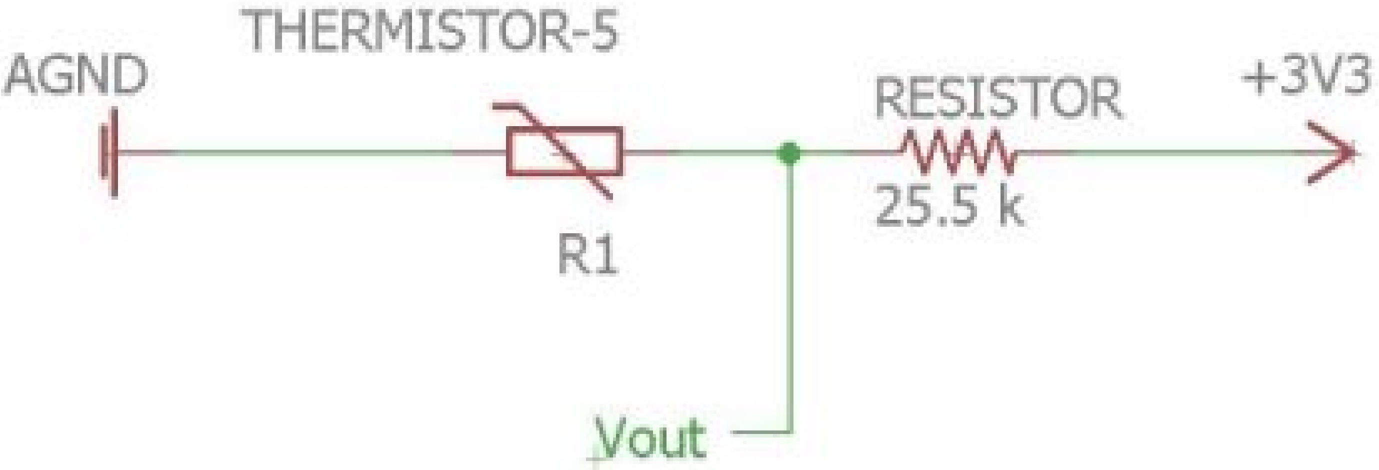
Transmitter Block Diagram



Receiver Block Diagram



Sensor Block

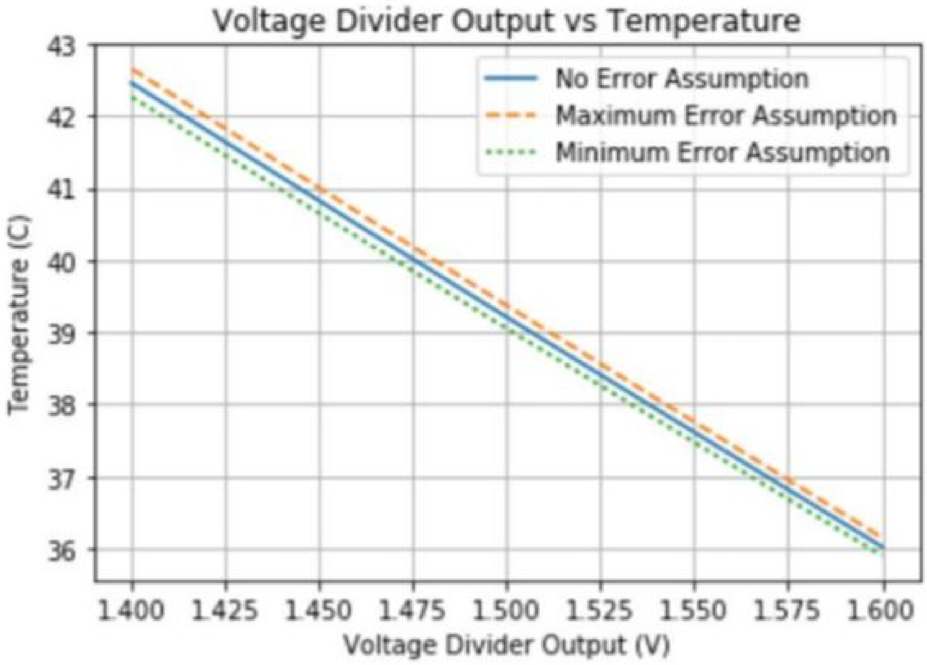


Sensor Calculation

$$R = R_o e^{-B/T_o} e^{B/T}$$

$$T = \frac{B}{ln(R/R_o) - B/T_o}$$

$$V_{out} = \frac{R}{R+25500}$$

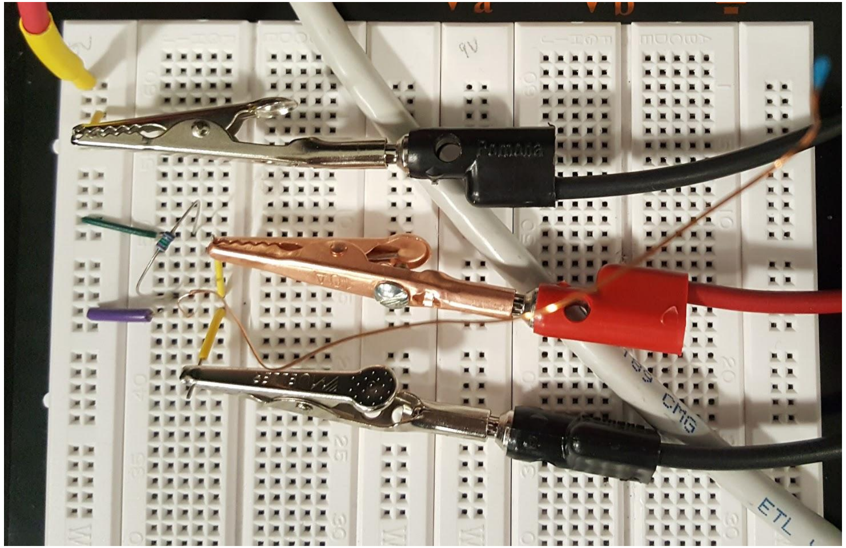


Sensor R&V

- Sensor shall be accurate to within 0.2 deg C between 38.6 and 40 C

- Sensor shall be long enough to reach into a cow’s ear canal, 150 mm +- 10 mm in length

Sensor Testing



Power Consumption

Microcontroller Power Specifications:

- Standby: **0.7 µA** (RTC Running, RAM & Register Retention)
- TX at +10 dBm 868 MHz: **13.4 mA**

12 byte packet @ 50kbps => about **2ms per message**

5 minute send intervals => total power usage of around **14.3 mAh** over 2 years
Our CR2025 battery has a capacity of 165mAh, so it is more than enough.

Power Consumption, Part 2

However…

- We were unable to put the MCU into standby mode.
- We found our average current draw to be about 5.5mA

  => a battery life of just 30 hours, not > 18 months

Reasons:

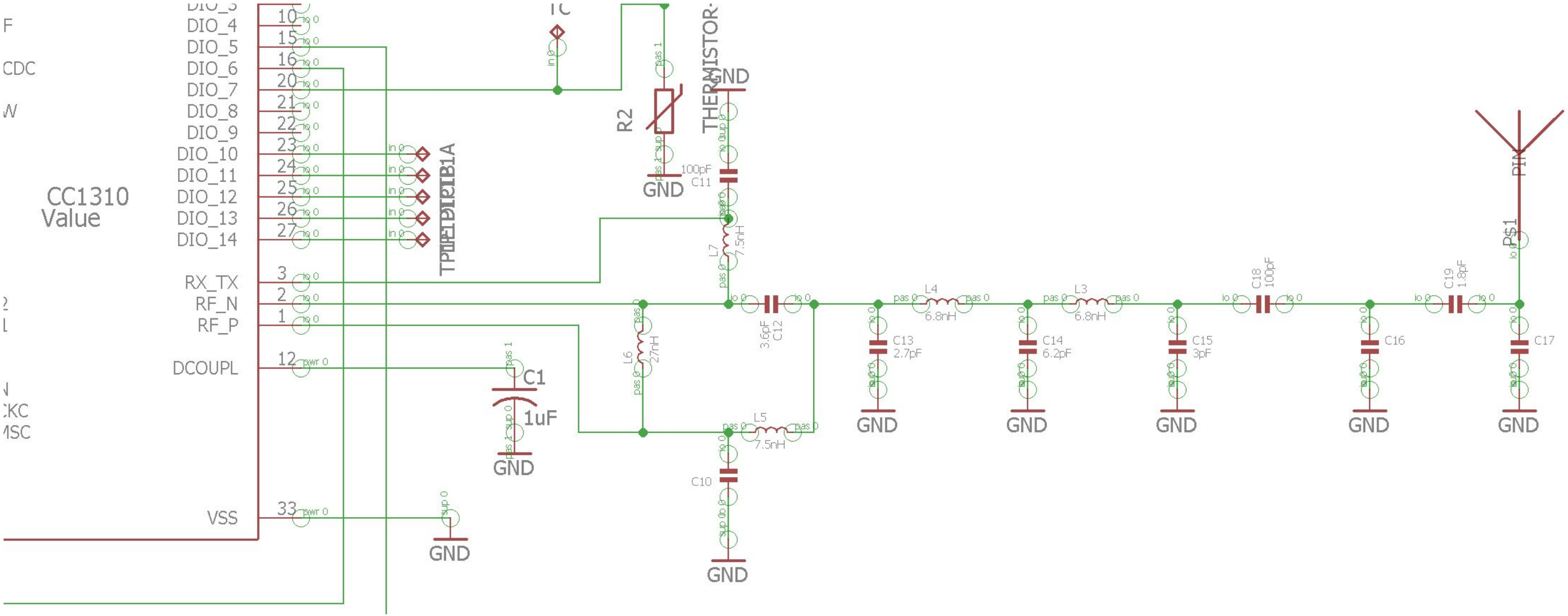
- It’s possible that the LaunchPad peripherals were drawing most of that power
- With a programmed MCU on our PCB, we could more accurately estimate power usage

Control Block

Control R&V

- Will only transmit once every 5 minutes
- Converts voltage reading into 4 bytes of data
- Has a saved unique ID

Transmitter Block

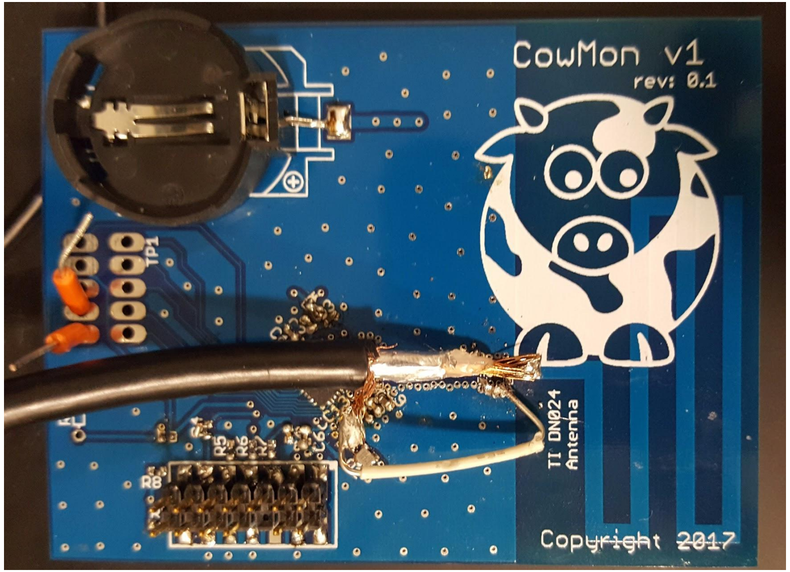


Transmitter R&V

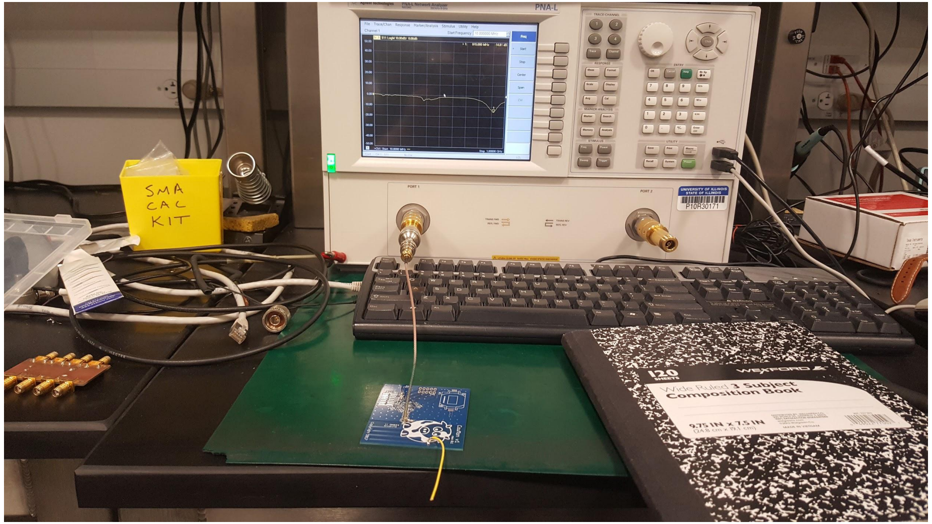
- Transmits at 915 MHz
- At least 100 meters of range

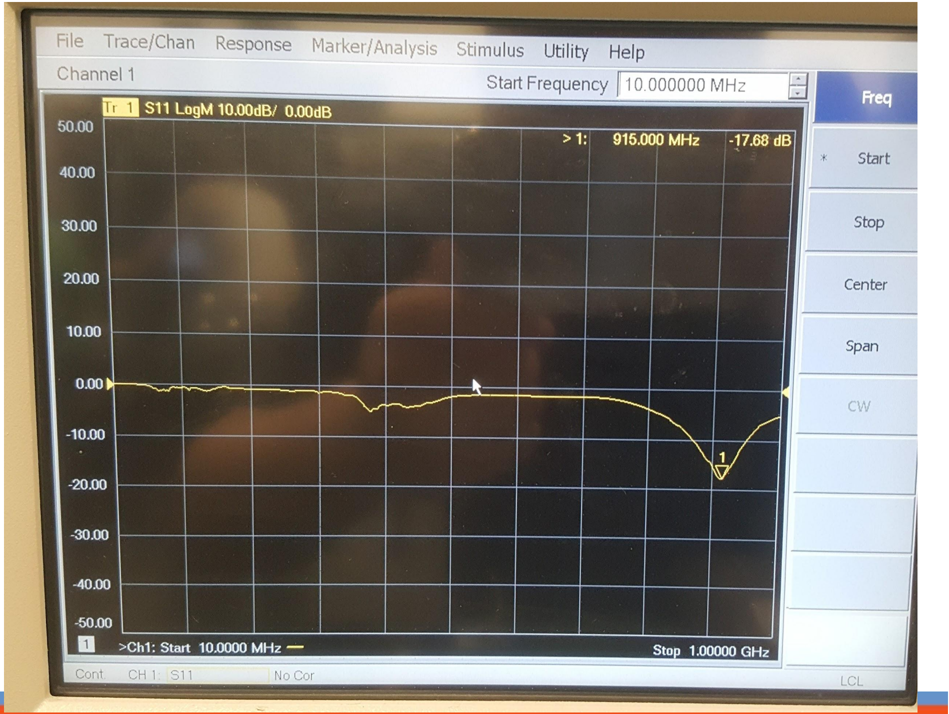
Transmitter Antenna Matching

- Reduces antenna reflection as much as possible
- Stops signals from bouncing back



Transmitter Antenna Matching





Channel 1

Start Frequency 10.000000 MHz

Tr 1 S11 LogM 10.00dB/ 0.00dB

> 1: 915.000 MHz -17.68 dB

>Ch1: Start 10.0000 MHz

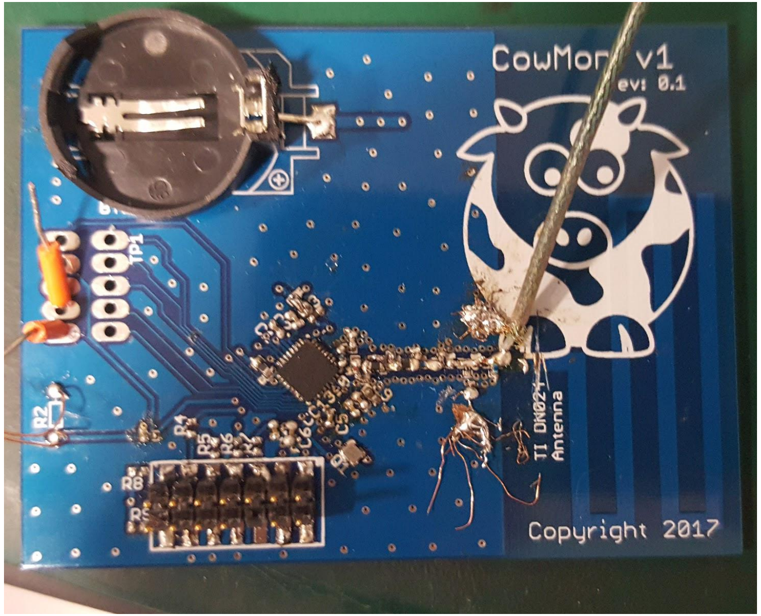
Stop 1.00000 GHz

Cont. CH 1: S11 No Cor

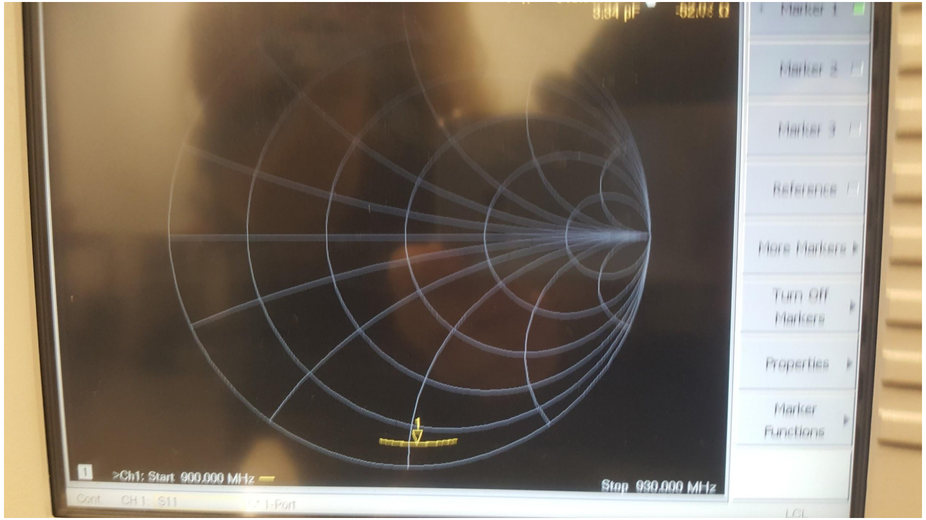
Transmitter Component Matching

- Reduces power loss from transmission

- Ensures no signal reflection



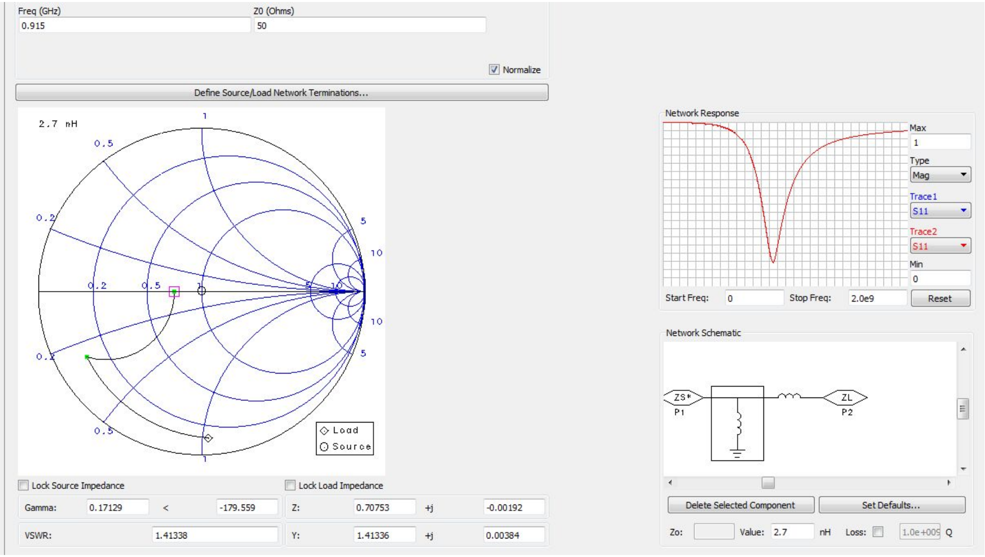
Component Matching Results



>Ch1: Start 900.000 MHz

Stop 930.000 MHz

Component Matching Results



Freq (GHz): 0.915

Z0 (Ohms): 50

☑ Normalize

□ Lock Source Impedance

□ Lock Load Impedance

Gamma: 0.17129 < -179.559

Z: 0.70753 +j -0.00192

VSWR: 1.41338

Y: 1.41336 +j 0.00384

Max: 1

Type: Mag

Trace1: S11

Trace2: S11

Min: 0

Start Freq: 0

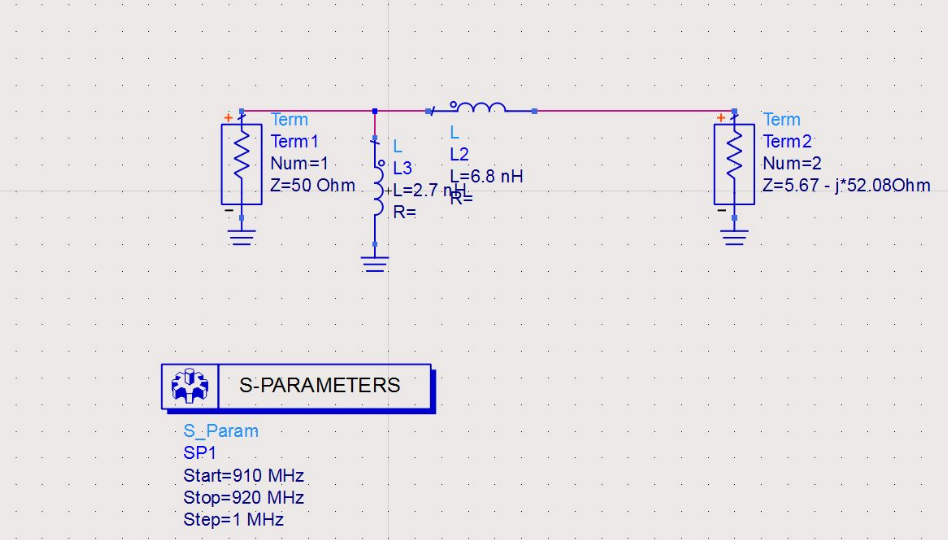
Stop Freq: 2.0e9

Zo:

Value: 2.7 nH

Loss: □ 1.0e+009 Q

Component Matching Results



Term
Term1
Num=1
Z=50 Ohm

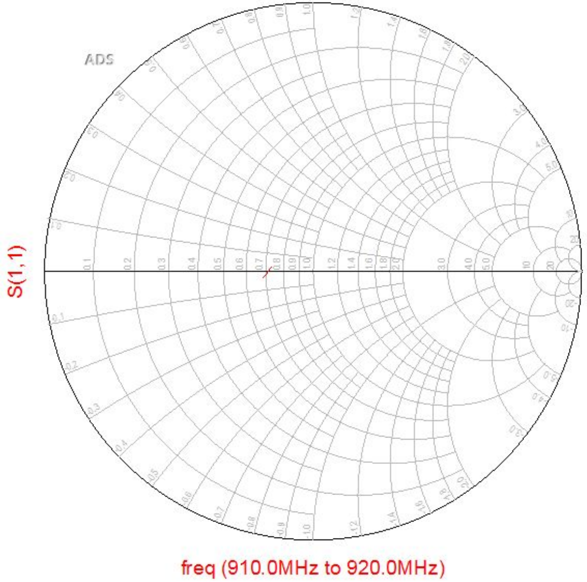
L
L3
L=2.7 nH
R=

L
L2
L=6.8 nH
R=

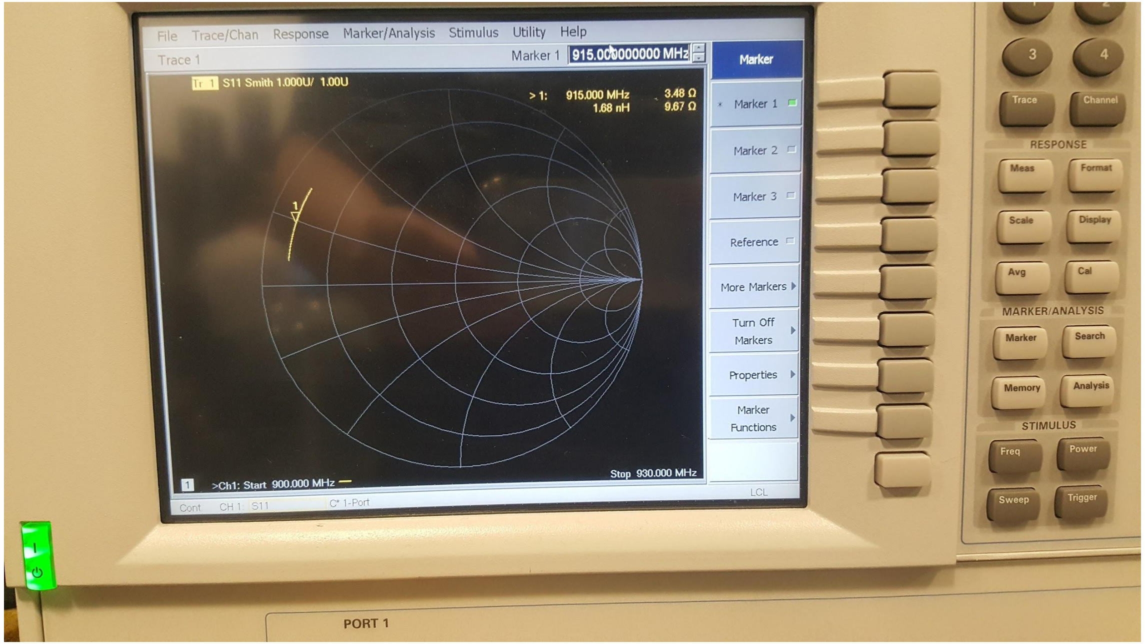
Term
Term2
Num=2
Z=5.67 - j*52.08Ohm

S-PARAMETERS

S_Param
SP1
Start=910 MHz
Stop=920 MHz
Step=1 MHz



Component Matching Results



Marker 1 915.000000000 MHz

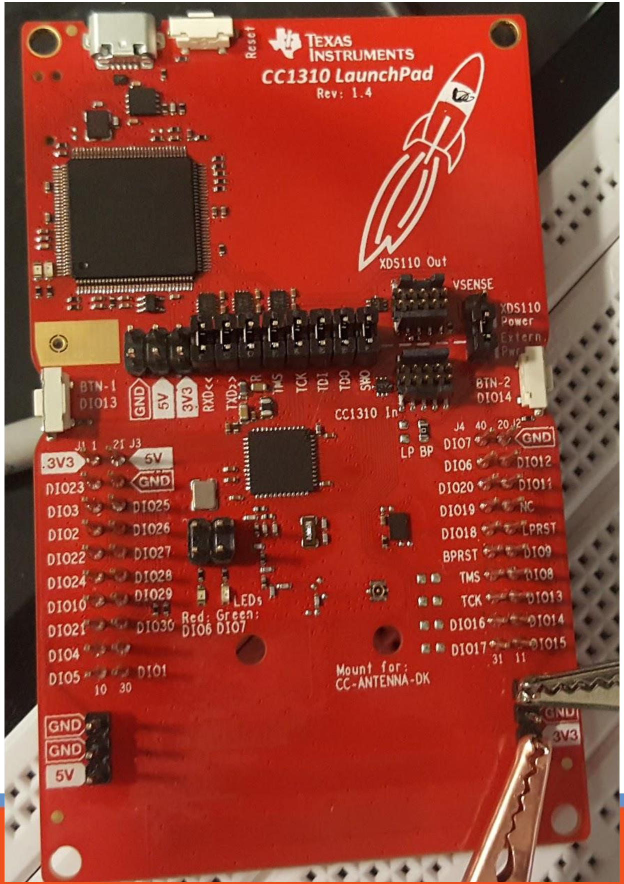
Tr 1 S11 Smith 1.000U/ 1.00U

> 1: 915.000 MHz 3.48 Ω
1.68 nH 9.67 Ω

>Ch1: Start 900.000 MHz

Stop 930.000 MHz

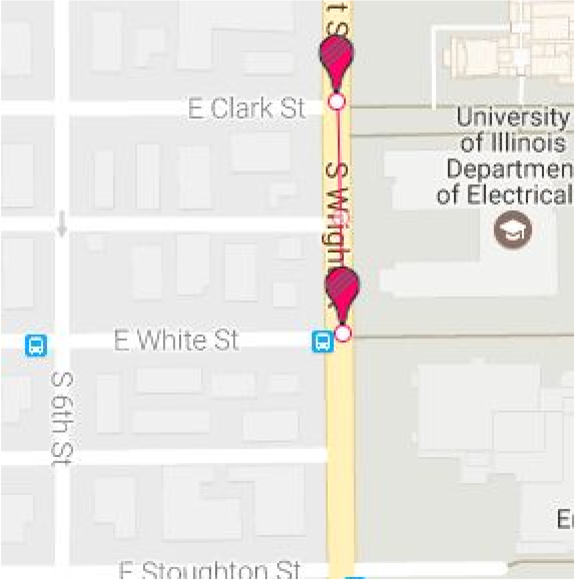
Receiver


Texas Instruments
CC1310 LaunchPad
Rev: 1.4
Reset
XDS110 Out
VSENSE
XDS110 Power
Extern Pwr
BTN-1 DIO13
BTN-2 DIO14
GND
5V
3V3
RXD<<
TXD>>
TMS
TCK
TDI
TDO
SWO
CC1310 In
LP BP
3V3
5V
GND
DIO23
DIO3
DIO2
DIO22
DIO24
DIO10
DIO21
DIO4
DIO5
DIO25
DIO26
DIO27
DIO28
DIO29
DIO30
DIO1
LEDs
Red: Green:
DIO6 DIO7
DIO7
DIO6
DIO20
DIO19
DIO18
BPRST
TMS
TCK
DIO16
DIO17
GND
DIO12
DIO11
NC
LPRST
DIO9
DIO8
DIO13
DIO14
DIO15
Mount for:
CC-ANTENNA-DK
GND
GND
5V
GND
3V3

Receiver R&V

- Receives packets at 915 MHz at a range of at least 100m
- Receiver shall process packet into 2 bytes magic number, 2 bytes ID number, and 4 bytes of voltage data
- Receiver shall convert 4 bytes of voltage data into temperature

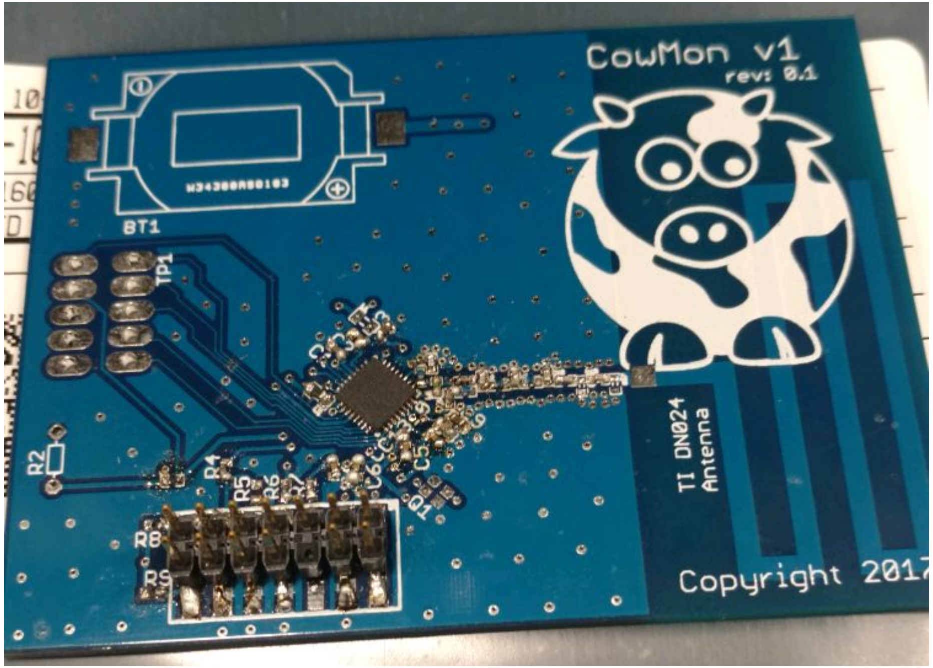
Receiver Testing



Conclusion and Future Work

- Successfully program the MCU on the fabricated PCB
- Successfully put MCU into standby mode to save power
- Create a case
- Make receiver software upload to AWS DB
- Create a user interface to display data and identify sick cattle
- Add waterproofing
- Add shielding

Questions?


CowMon v1
rev: 0.1
BT1
TP1
R2
R4
R5
R6
R7
R8
R9
C5
D1
TI DN024
Antenna
Copyright 201