UMEÅ UNIVERSITY
Institute of applied physics and electronics
December 2, 2019
Project report

inbsys-ht19

Embedded systems

Speed measuring device with KC-L6 doppler radar

Author Mattias Lindström

E-mail `mali0697@student.umu.se`

Teacher
John Berge

Contents

1 Introduction

The goal of this project was to design a prototype for a handheld speed measuring device utilizing the KC-L6 doppler tranciever together with a STM32 Nucleo-64 development board. The unit was designed from a set of requirements stating that the unit should be able to:

- Be powered from a battery and be used as a handheld device.
- Measure speeds from 0-200 km/h with a resolution of 1 km/h.
- Measure objects from distance of 5 m to 50 m.
- Measure objects with a size similar to a person or a vehicle.
- Continuously do speed measurements as long as a button is held down.
- Display the measurement data on a graphical display.

Additional (optional) requirements included the ability to be able to measure smaller objects with a higher resolution and to be able to store measurements to a USB-drive. The additional requirements was not implemented in this project.

To achieve these goals the development process consisted mainly of work related to: Signal conditioning of the analog tranciever signal, converting the signal to the frequency domain and doing speed calculations, designing and implementing a graphical interface to show measurement data and building a PCB for the prototype.

2 System overview

2.1 The STM32 Nucleo-64 development board

The development board used for this project is a STM32 Nucleo-64, model L476RG. It has a Cortex M4 processor with floating-point and DSP capabilities, which is the main reason it was selected for this project. The development board has a programmer/debugger built in and is connected by a USB-connection to a PC. It can be powered from the USB-connection or from an external source (3.3 V, 5V, 7-12V). [1]

The software *STM32Cube IDE* was used together with the HAL abstraction layer and the ARM math libraries for developing the firmware of the system. All connections to the development board was made to the two ST morpho connectors.

2.2 Block diagram

The K-LC6 tranciever outputs an analog signal with a frequency correlating to the speed of the object in measurement. The analog signal is filtered and amplified and sent to the ADC-input of the microcontroller. Two pushbuttons are connected to the microcontroller, one for starting and stopping measurements and one for entering the settings-menu. The output of the system is sent via SPI to a TFT graphical display. Figure 1 illustrates the signal flow between the different components of the system.

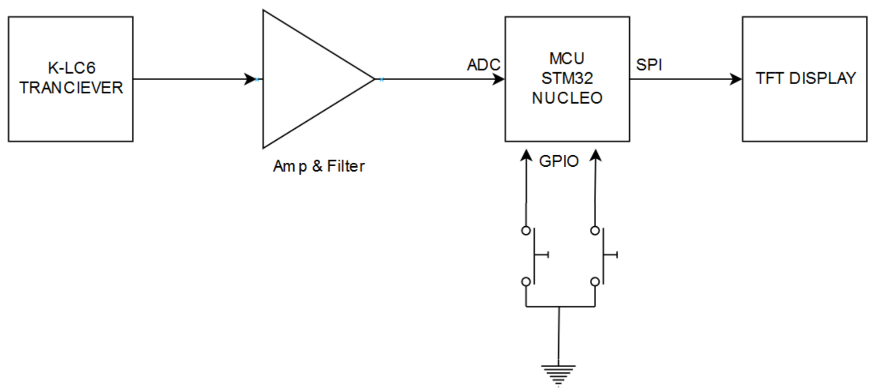


Figure 1: Signal flow of the system.

2.3 The KC-L6 doppler radar

The KC-L6 tranciever outputs a 24GHz frequency and receives the reflection of this signal. The doppler effect creates a difference in frequency between the sent and received signal, and is proportional to the speed of the object reflecting the signal. This difference in frequency is the output of the tranciever and is available on both pin 1 and pin 3 of the component, where pin 3 is the in-phase signal (I) and pin 1 is the quadrature 90-degree phase-shifted signal (Q). The two signals can be used together to determine the direction of movement of the measured object by checking if the phase-difference between them is positive or negative.[2] In this project only the in-phase signal is used to determine the speed of an object. The relationship between the object speed and the output frequency is as follows:

$$f = v * \frac{44Hz}{km/h} * cos\alpha$$

Where f is the doppler frequency, v is the velocity of the object in motion and α is the angle between the object and the radar.

The tranciever needs a 5 V supply and draws 50 mA typically. To power the tranciever and the associated analog components a linear voltage regulator is used to limit digital noise on the analog circuitry.

2.4 Signal conditioning

The output of the tranciever can be in the sub millivolt range and needs to be amplified and filtered to be used. An amplifying two-pole filter was used as suggested by the application note AN-04. [3]

Figure 2 shows the schematic of the K-LC6 tranciever and the analog signal conditioning. The gain of the circuit is determined by resistors R1, R2, R3 and R4:

$$G = \frac{R_1}{R_3} * \frac{R_2}{R_4} = 6694(76dB)$$

Filtering is done to reduce noise and to prevent aliasing from occuring when sampling the tranciever-signal. The high-pass filtering cutoff frequency is determined by capacitors C5 and C6, and resistors R3 and R4:

$$f_c = \frac{1}{2\pi R_3 C_5} = 15Hz$$

$$f_c = \frac{1}{2\pi R_4 C_6} = 15Hz$$

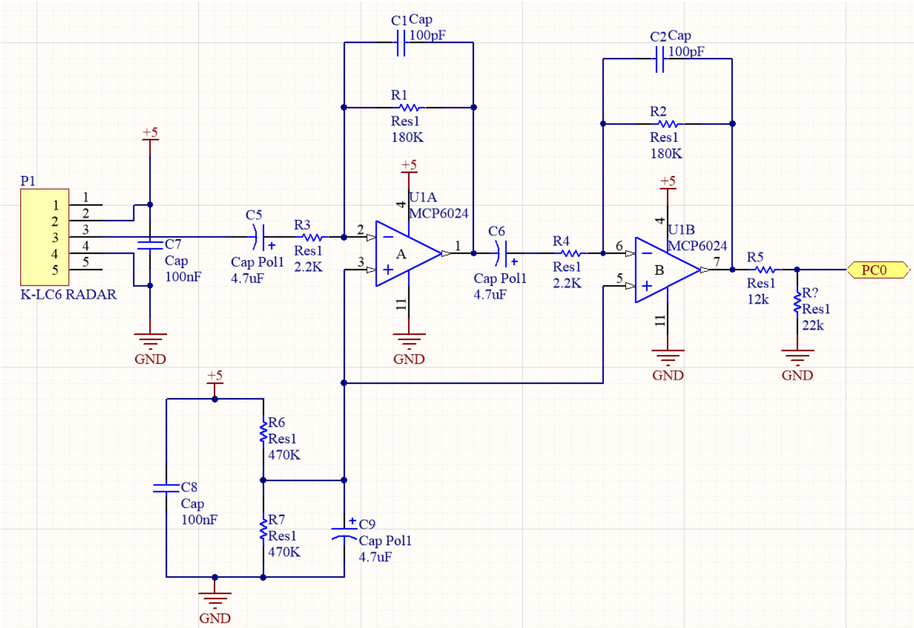


Figure 2: Amplification and filtering of the K-LC6 signal

The low-pass filtering cutoff frequency is determined by capacitors C1 and C2, and resistors R1 and R2:

$$f_c = \frac{1}{2\pi R_1 C_1} = 8841Hz$$

$$f_c = \frac{1}{2\pi R_2 C_2} = 8841Hz$$

The operational amplifier is powered from the 5V regulator, which means it's maximum output voltage is close to 5V. The output is therefore scaled down by a voltage divider to limit the maximum output to 3.3V, which is the reference voltage of the ADC-converter of the microcontroller.

2.5 Sampling of the signal

To be able to measure a maximum speed of 200 km/h without any problems arising due to the Nyquist sampling theorem the sampling frequency needs to be atleast:

$$fs = 200km/h * 44Hz * 2 = 17.6kHz$$

And is set to 20 kHz. The number of samples decides our frequency- and hence out speed resolution and is set to 512 samples. This gives us a frequency resolution of:

$$\Delta f = \frac{20kHz}{512} = 39.0625Hz$$

Which satisfies our speed resolution requirement of 1 km/h since a 44 Hz signal is equivalent to an object moving at 1 km/h.

The sampling of the signal is done with the 12-bit ADC of the microcontroller and is controlled by timer 2. The timer is set to generate interrupts with a frequency of 20 kHz and sampling is done in the interrupt-handler by calling HAL_ADC_Start_DMA() for every interrupt. This means conversion is handled by the DMA-unit but is not done continuously since every sample is individually called for in the interrupt-handler.

The timer is controlled by a pushbutton connected to PA6 and starts counting when the button is pressed down. The timer stops counting when the button is released, this means that sampling is only done as long as the button is held down.

The 512 samples are stored in an integer array. When the buffer is filled the ADC conversion complete callback function is called, where the buffer is copied over to a float array of the same size. This is because the math and FFT functions needs to be able to write to the buffer while it's performing calculations.

2.6 Calculating speed

The *HAL_ ADC_ ConvCpltCallback* function is called every time the buffer has been filled with new samples. This makes it a useful function for placing the the speed calculation routine which can be summarized as followed:

1. Copy the integer buffer to a float buffer.
2. Perform an FFT on the buffer.
3. Remove the DC-component from the buffer.
4. Get the magnitude of the real part of the FFT buffer.
5. Find the strongest frequency component in the magnitude buffer.
6. Divide the strongest frequency with 44 to get the speed in km/h.

The ARM math library is used for the FFT function, the magnitude function and the max value function. The arm_rfft_fast_f32 function is used because we only need the real part of the FFT to calculate speed since we are not interested in the direction of the object in motion.

2.7 User Interface

The user interface consists of two pushbuttons and a 1.8″ TFT-display. the *triggerbutton* is set up to generate interrupts on both rising and falling edges and controls the speed-measuring process. The *menubutton* is for entering a settings menu where the user can change parameters affecting the measurements. Figure 3 shows the schematic view of the user interface. The buttons are connected to PA6 and PA7 of the Nucleo board and have internal pullups enabled. Both buttons are polled in software but the triggerbutton is also interrupt-driven when controlling speed measurements.

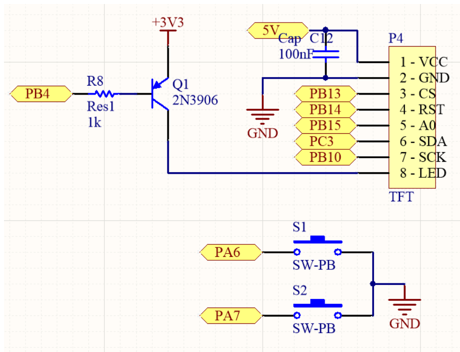
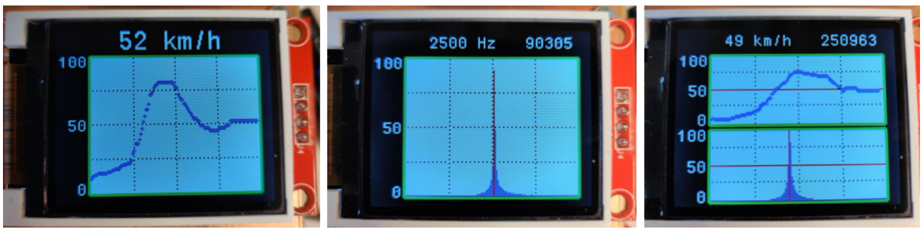


Figure 3: User interface schematic

The TFT-display is used for displaying measurement plots and for drawing the settings menu. It's connected to SPI2 of the microcontroller. The backlight of the display is controlled by a PNP-transistor switch connected to PB4 of the Nucleo board and is lit by setting the pin low.

There are two different plots implemented, one for displaying speed-values over time and one for diplaying an fft magnitude plot. The plot windows can be resized and placed freely on the display, the combined speed and fft plots shown in figure 4 demonstrates the plot functionality of the system.



Speed plot FFT plot Speed and FFT plots

Figure 4: Implemented plot types

3 Firmware

When starting the system the processor initializes clocks and peripherals then enters the typical while(1) loop. If the triggerbutton is pressed an interrupt occurs that starts timer 2 which begins the AD-conversion and speed calculation process.

In the main program loop the menubutton is polled. If the button is pressed the program enters the menu state by setting the menu variable high and disabling the triggerbutton-interrupt to stop from new measurements happening while in the menu state. Both buttons are polled in the menu loop. The menu button increases the value of the menuitem variable which selects which menu item is currently selected. The trigger button changes the value related to the item by switching the menuitem variable and then increasing the value of the variable coupled to that item. The last item is the exit item which resets the menu variable to exit the menu. Figure 4 illustrates the general program flow of the microprocessor, with the ADC-Callback and speed calculations excluded.

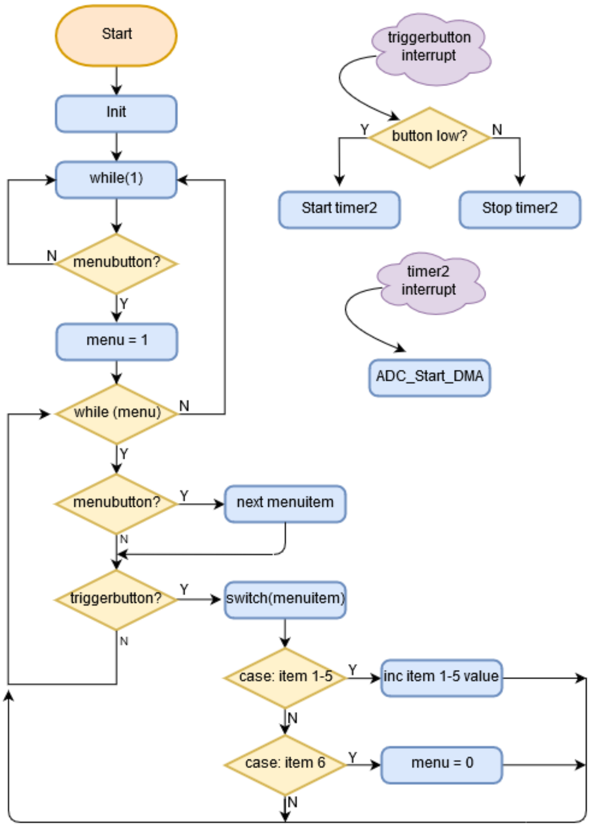


Figure 5: Program flow

4 PCB

A double layer PCB-design was made in Altium Designer. The design was made to be used as a shield for the Nucleo-64 development board. The top side of the PCB houses all components for the signal conditioning ciruitry aswell as the TFT-display and pushbuttons. The KC-L6 doppler tranciever was placed on the bottom side. The Nucleo-64 board plugs directly into the morpho headers on the top side of the PCB. Figure 6 shows a 3D model of the design.

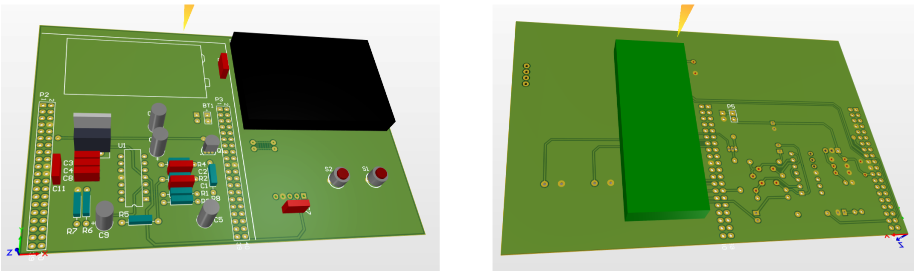


Figure 6: 3D model of the PCB.

To simplify construction and modifiability all components used were through-hole components. The number of vias was kept low and some of the component-legs were used as vias to reduce buildtime of the board.

The PCB was etched inhouse with a small etching laboratory setup. The legs of the larger components are bent so that the components can fit under the Nucleo-64 board. Figure 7 shows the build after etching and soldering.

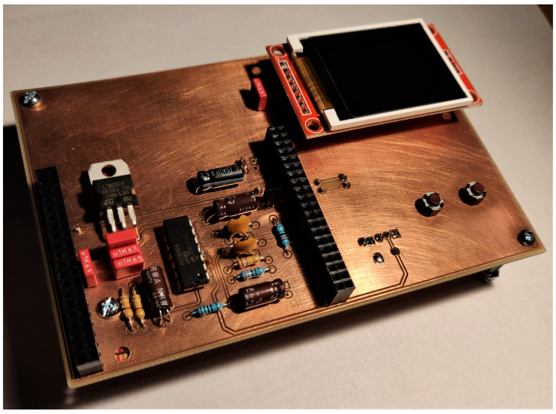


Figure 7: PCB after etching and soldering.

5 Tests and limitations

5.1 Tests with signal generator

To ensure that the speed calculation and display routines were correct the ADC was fed with a sinewave from a signal generator. Figure 8 shows speed and FFT plots with different y-axis scaling.

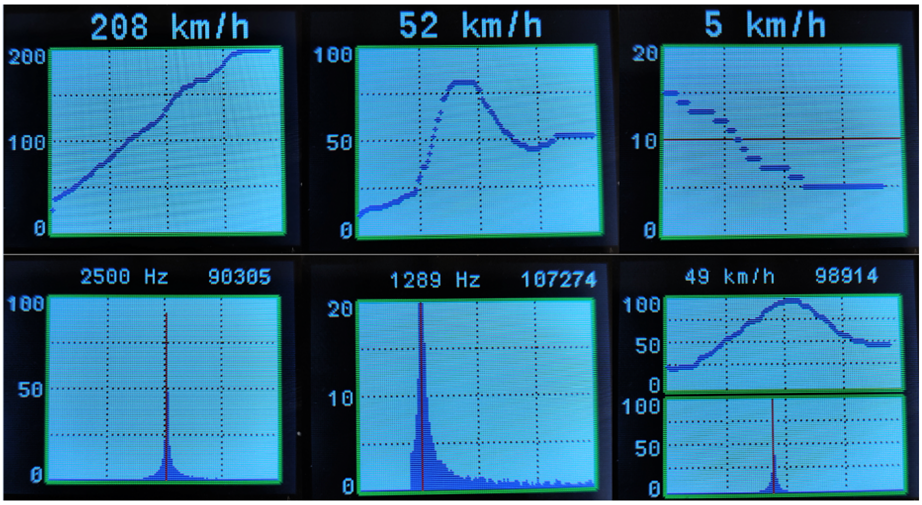


Figure 8: Speed plot tests.

5.2 Field testing on traffic

Figure 9 shows the location and setup for the field testing. The speed limit of the road was 60 km/h and measurements were taken approximately 50 m from a roundabout with the vehicles passing and moving away from the unit towards the roundabout.

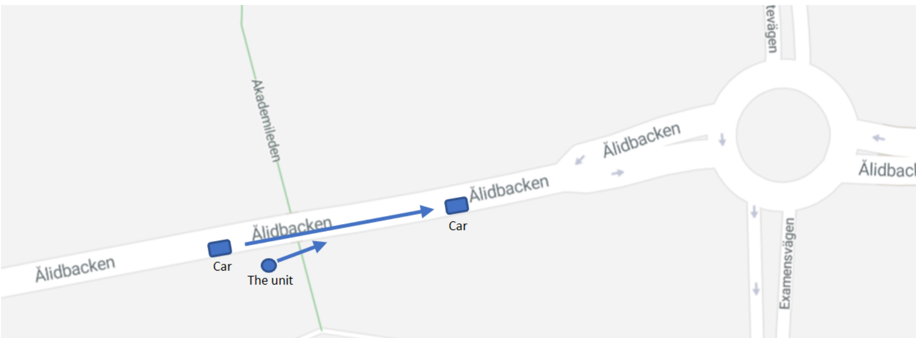


Figure 9: Test location and setup

Figure 10 shows speed measurements with a FFT-magnitude threshold value of 3000. This threshold is set to ignore noise and only capture reliable signals.

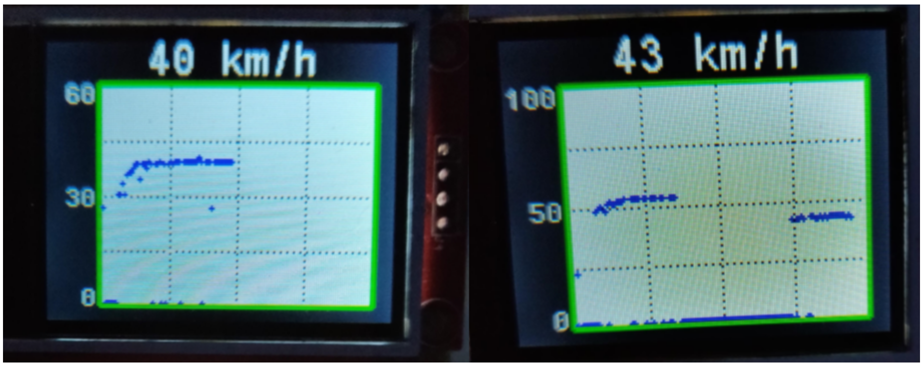


Figure 10: Speed plot tests.

Figure 11 shows measurements with the threshold value set to 500. With a lower threshold value noise becomes apparent but the range of the unit is increased. The rightmost picture also demonstrates the issue arising from a too large angle between the unit and the vehicle. It looks like the cars were accelerating when passing the unit but they were driving at an almost constant speed. To attain accurate measurements the unit has to be directly behind or in front of the object in measurement.

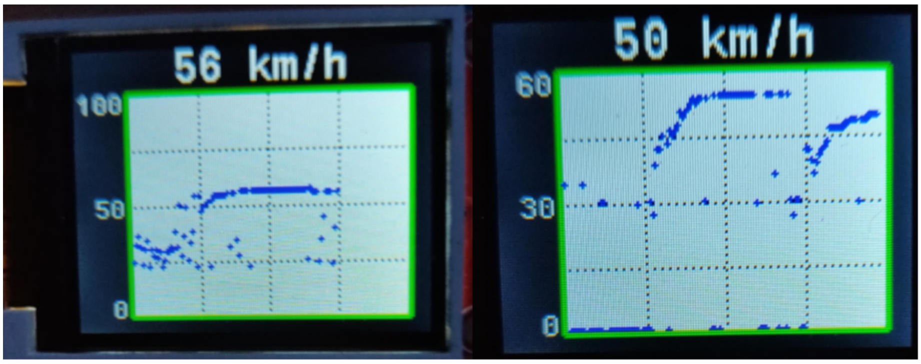


Figure 11: Speed plot tests.

Figure 12 shows the FFT and dual modes in field testing. The FFT-plot shows that noise are greater in lower frequencies and has a magnitude of about 2000. The red line is representing the greatest frequency-component of the signal which comes from the object in measurement, its signal magnitude is 4655. The rightmost picture shows the dual-mode when measuring a vehicle. The usefulness of this mode is limited due to the slower samplerate which arises from the increased time it takes to update both plots between each sample.

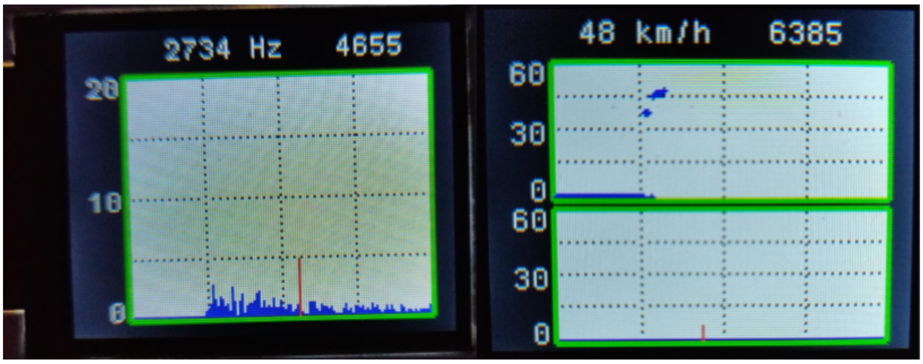


Figure 12: FFT and dual-mode tests.

The results show that the unit is performing at an acceptable level when doing speed measurements of vehicles moving at around 60 km/h. The maximum range of the unit is approximately 30-40 m, which is lower than the specified requirement of 50 m. This is mainly due to the poor signal-to-noise performance of the unit.

6 Conclusion

The prototype shows that it is possible to do real-time plotting of the speed of traffic with the KC-L6 doppler tranciever together with the Cortex-M4 processor in a handheld format. Improvements can be made in regards to noise performance and power efficiency. The DSP-capabilites of the Cortex M4 should be further utilized by incorporating digital filters to reduce signal noise. The current layout of the PCB is also thought to be a contributory factor to the poor noise perfomance. The Nucleo-64 board sits directly above the analog circuitry which means that digital noise is likely to be induced into the analog signal. The PCB should be redesigned with signal-integrity as a primary focus instead of size or compactness of the build.

No sleep functionality is currently incorporated in the firmware of the unit. To be viable as a handheld battery driven device the current-draw of the build needs to be reduced substantially. The most power hungry part of the build is the KC-L6 tranciever, which draws around 50 mA. Unfortunately the tranciever has a start-up delay of a few seconds which makes it hard to optimize for power, it is therefore suggested to look for alternative trancievers when doing a battery driven build. The backlight of the TFT-display should also be disabled on inactivity to improve battery life.

This project serves as a good base for exploring the possibilites of the KC-L6 doppler tranciever. The plotting functionality is useful for doing testing outside of a laboratory environment and to gain an intuitive understanding of speed measuring with a doppler radar.

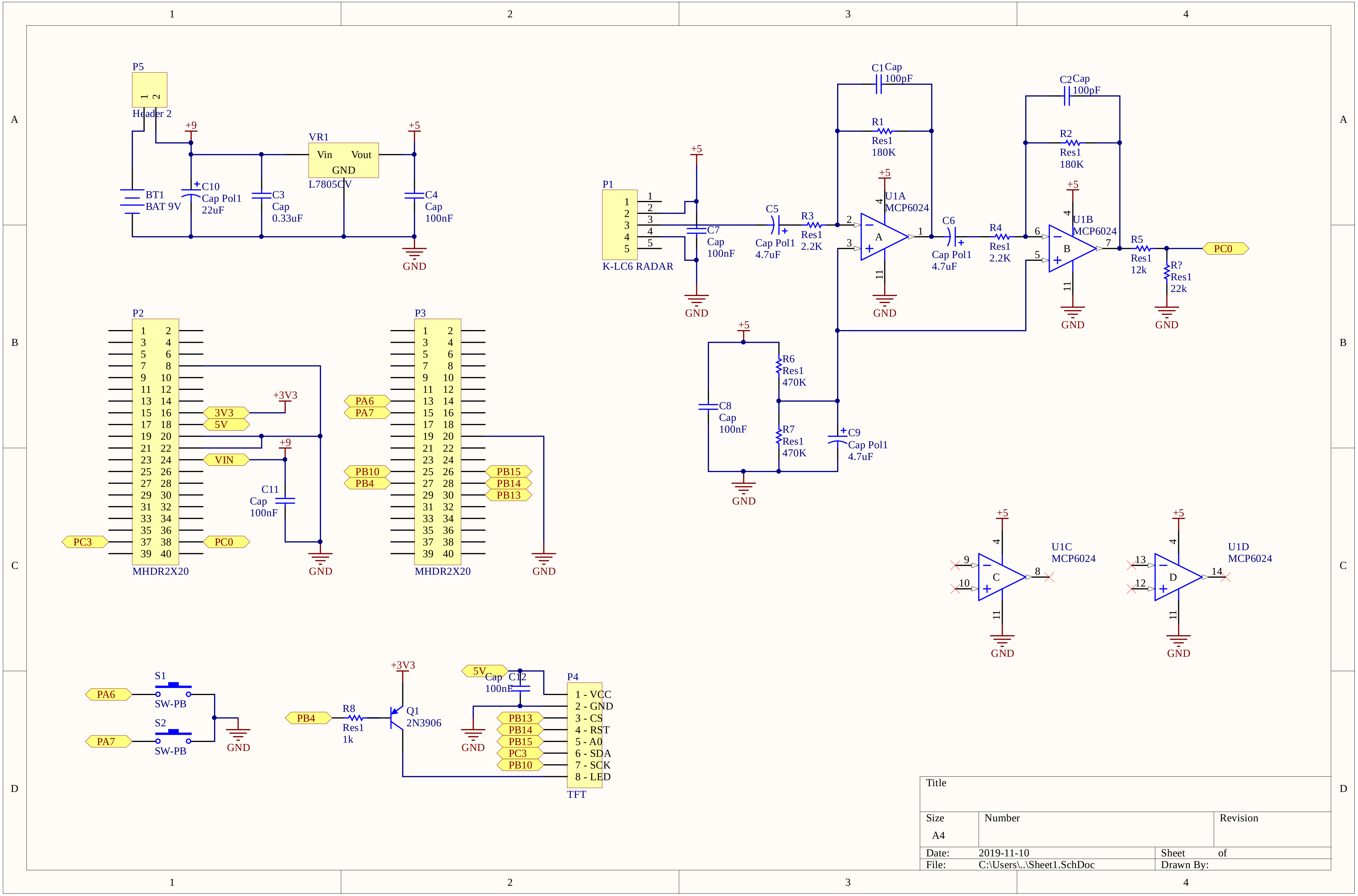
References

[1] ST Microelectronics. "UM1724 User Manual STM32 Nucleo-64 boards"
https://www.st.com/resource/en/user_manual/dm00105823.pdf

[2] RFbeam Microwave. "K-LC6 Radar Tranciever datasheet"
https://www.rfbeam.ch/files/products/12/downloads/Datasheet_K-LC6.pdf

[3] RFbeam Microwave. "Typical Dopper Signal Amplifier Application Note AN-04"
https://logosfoundation.org/ii/Radar_2017/AN-04%20TypicalSignalAmp.pdf

7 Appendix



P5
Header 2
BT1
BAT 9V
+9
C10
Cap Pol1
22uF
C3
Cap
0.33uF
VR1
Vin
Vout
GND
L7805CV
+5
C4
Cap
100nF
GND
P1
K-LC6 RADAR
C7
Cap
100nF
C5
Cap Pol1
4.7uF
R3
Res1
2.2K
C1
Cap
100pF
R1
Res1
180K
U1A
MCP6024
C6
Cap Pol1
4.7uF
R4
Res1
2.2K
C2
Cap
100pF
R2
Res1
180K
U1B
MCP6024
R5
Res1
12k
R?
Res1
22k
PC0
R6
Res1
470K
R7
Res1
470K
C8
Cap
100nF
C9
Cap Pol1
4.7uF
P2
MHDR2X20
3V3
5V
VIN
+3V3
C11
Cap
100nF
PC3
P3
MHDR2X20
PA6
PA7
PB10
PB4
PB15
PB14
PB13
U1C
MCP6024
U1D
MCP6024
S1
SW-PB
S2
SW-PB
R8
Res1
1k
Q1
2N3906
C12
Cap
100nF
P4
1 - VCC
2 - GND
3 - CS
4 - RST
5 - A0
6 - SDA
7 - SCK
8 - LED
TFT
Title
Size
A4
Number
Revision
Date:
2019-11-10
Sheet
of
File:
C:\Users\..\Sheet1.SchDoc
Drawn By: