INDIRES

Information Driven Incident Response

User Manual
LF/VLF Mesh Communication System

3rd November 2020

This page is intentionally left blank.

TABLE OF CONTENTS

1 INTENDED AUDIENCE

An output of the INDIRES project is user documentation for each of the products produced during the project. However, the status of the various products differs and, for this reason, the intended audience for the *User Documentation* will be different for each product. It is necessary, therefore, to indicate the type of user who will benefit from this particular *User Document*.

As work on the through-the-earth (TTE) mesh communication system progressed, it became clear that decisions on so many factors would be dependent on aspects of the mine including its size and layout and the geology. In particular, for optimal performance at minimal cost, it would be necessary to analyse the characteristics of the particular mine in order to select the type of antenna, the frequency of operation, the transmit power, the physical spacing of the nodes, and the network and routing algorithms. For this reason, a degree of flexibility was built into the electronic and software aspects of the sensor. However, it was not possible to offer such flexibility is all aspects of the design, and while the current prototype allows the operating frequency and the routing algorithms to be altered, the type of antenna and the maximum transmitter power are fixed. This led to the inevitable conclusion that the field tests that were to be carried should be considered as a proof-of-concept, in the sense that they would only prove the correct operation of a TTE system, including the fixed sensor unit, in a particular mine.

Unfortunately, the Covid-19 pandemic and its effect on industrial production caused problems with the manufacturing of the TTE node housing, resulting in its very late and incomplete delivery. As a result of this delay, and bearing in mind that it became increasingly unlikely that it would be possible, due to Covid-19, to test the unit in a mine environment, those aspects of the design that would require adaptation to the specific mine were not completed. Therefore, some additional development would be needed before the prototype developed in INDIRES would be in a suitable form for testing in a particular mine. Furthermore, following such a successful test, further research and development would be needed to establish a technique whereby the fixed sensor unit can readily be adapted to meet the requirements of a particular mine operator.

Accordingly, this *User Document* provides technical details, including ample background information and details on tests carried out in the project, about the TTE mesh node, as opposed to operational instructions. It is, therefore, anticipated that it will mostly be relevant to organisations who intend to further develop and test the resilient, survivable environmental sensor unit.

Developers who want to know more, and perhaps to discuss possible collaborative opportunities, are referred to *Section 11*.

This page is intentionally left blank.

2 BACKGROUND AND OVERALL REQUIREMENTS

2.1 Intended Use – Overall Requirements

The Wireless Mesh Node is intended to be used in the resilient fixed sensor unit, as well as part of a through-the-earth communication network. As such, its hardware contains not only the wireless or radio stages and a microcomputer, but also interfaces and power supply circuits for connecting all sensors in the fixed sensor. It also includes audio interface circuitry (microphone and loudspeaker). The microphone interface is used in the fixed sensor, since it is the primary sensor for detecting "signs of life".

The use in the fixed sensor has also influenced the form factor of the overall design. Considering the information collected from industrial partners, and bearing in mind the need to install the mesh node in a borehole in a gallery wall, a long and narrow design was considered to be desirable, not more than 130mm wide, with no special limitation in length. The dimensions of the Printed Circuit Board (PCB) is 90mm × 240mm.

Finally, the power supply voltage and current consumption was made to be as low as possible, in order to make ATEX certification easier in the future. Moreover, the power supply design allowed selective disconnection of parts of the circuit and very low power standby (or sleep) operation.

2.2 Overall Design

The result of the design work is a device named XCVR-HF-01A. It was designed around a microcomputer module, with adequate I/O, memory and processing power. Most sensors can be connected directly – or with minimal interface circuitry – to it through a dedicated connector. Interfaces for a display and keyboard are also included.

A second building block comprises the radio (reception and transmission) interfaces, which allow great flexibility in the possible modes of operation, be they analogue or digital, data or voice. Analogue circuitry for interfacing with an external microphone and a loudspeaker or a headset are included in this block.

An elaborated power supply circuitry, with low standby power and selective disconnection capacity, completes the design.

Full circuit diagrams are included as *Appendix B* to this Document *(Section 12*).

Details on the electronic design are provided in the next sections (*Sections* 3 to *6*). In *Section 9* data on the actual implementation is presented, including renderings of the PCB (full details are included as *Appendix C* in *Section 14*), and pictures of the prototype and testing waveforms.

This page is intentionally left blank.

3 CORE MICROCOMPUTER

In the first stages of development, a member of the Arduino MKR family – MKR Zero – was selected as the core microcomputer, as it had adequate performance and I/O pins for the intended work. However, when implementing the system, it was considered convenient to have some means of debugging (the available software development tool lacks debugging features), in the form of a color LCD screen, a simple keyboard and encoder interfaces. But implementing these features required more I/O pins than available in the MKR. Moreover, it was desirable to have more memory and processing power than MKR's SOC, while keeping its small form factor.

After an extensive search, a module in the Teensy family was selected. Teensy family boards are complete USB-based microcontroller development systems – all programming is done via the USB port – in very small footprints. Among the available modules, the 3.5 version (*Figure 1*) was selected. It features a 32-bit 120MHz ARM Cortex-M4 processor with floating point unit. All digital pins are 5V tolerant (this is the most powerful one with 5V tolerant I/O pins, a feature convenient in this application). It has up to 58 digital I/O pins, up to 27 analogue inputs and 2 analogue outputs (some shared), 512KB of flash memory (a type of in-circuit programmable read only memory which is nonvolatile so keeps its contents after power-off and is used to store programs), 256KB of RAM and 4096 bytes in EEPROM. Full data is available at www.pjrc.com/store/teensy35.html and www.pjrc.com/teensy/techspecs.html.

Figure 1 – Teensy 3.5Microcomputer Module

Moreover, a full schematic diagram of the module is available, and is included in *Appendix A* (*Section 12*), which makes the path to future ATEX certification much easier.

The microcontroller module I/O is used almost to its full capacity, including some pins on the bottom side which are not-normally available. The module's pin-out is also included in *Appendix A* (*Section 12*).

The microcomputer is complemented by a 3.5" QVGA Display (DISP1), a sensors connector J7, and a serial interface J9 which is intended to connect an external keyboard. The display is connected according to the usual conventions in ARDUINO graphics libraries to facilitate software development. Finally, signals PADDLE_TIP and PADDLE_RING are routed to an external 3.5mm stereo connector. They can be used as an external serial port (Ser6) or for any other purpose.

I/O pin connections (of the computing module) can be found in Sheet 5 of the schematic diagram of the WMN, included as *Appendix B* (*Section 13*). Its function is detailed in*Table 1*.

Signal Name (Net name)	TEENSY 3.5 Pin	Description
VIN	-	Power input, Direct+H36H2:H2:H37
12V	-	Power input, after diode
+3V3	-	Radio 3V3 Digital
+3V3ANA	-	Radio 3V3 Analogue
+3V3_SHT	-	Sensor 3V3
+5V	-	Radio 5V Digital
+5VANA	-	Radio 5V Analogue
+5VS	-	Air Sensor+I9 analogue reading+H2:H37
+5VTEENSY	Vin	TEENSY Power supply
+VANA	-	Voltage for analogue front-end (can be 3V3 or 5V)
DVDD	-	Internal Digital Voltage for ADAR7251 (1.8V)
GND	GND	Digital Ground
AGND	-	Analogue Ground
ADC1_RST	Reset	ADC1 (ADAR7251) Reset
ADC_BCLK	D25	ADC1 Bit Clock. Tied only to ADC1 (ADAR 7251) (Master In I2S mode. Requires tweaking. Default pin is D9)
ADC_LRCLK	D23P	ADC1 Left/Right (ACD1.1/ADC1.2) Clock. Tied only to ADAR 7251. (Master In I2S mode, Default pin)
ADC_OUTA	D27	ADC1 Data Out. Tied only to ADAR 7251 (Master In I2S mode. Requires tweaking. Default pin is D13)
ADC_OUTB	N/C	ADC1.2 (ADAR 7251) Data Out. Not Connected (data flows from Out A)
ADC_SPI_CLK	D46-SCK2	ADC SPI Clock. Tied to both ADC1 (ADAR 7251) and ADC2 (AD7380)
ADC_SPI_CS	D43-CS2	ADC SPI Chip Select. Tied to both ADC1 (ADAR 7251) and ADC2 (AD7380)
ADC_SPI_MISO	D45-MISO2	ADC SPI Master In Slave Out. Tied to both ADC1 (ADAR 7251) MISO and ADC2 (AD7380) OUT_A
ADC_SPI_MOSI	D44-MOSI2	ADC SPI Master Out slave In. Tied to both ADC1 (ADAR 7251) MOSI and ADC2 SDI (AD7380)
ADC2_OUTB	N/C	ADC2.2 (AD7380) Data Out. Not Connected (data flows from Out A)
AIR_S_AOUT	A22	Air Sensor analogue reading
AUDIO_MON_1	A7	Audio monitoring tap 1 (Right after Q-DeMod)
AUDIO_MON_2	A23	Audio monitoring tap 2 (Loudspeaker/earpiece)
CH4_S_AOUT	A17	CH4 Sensor analogue reading
CLK0	-	SI5351 CLK0 Output. Used to generate receive LO (4X) I and Q signals by dividing in IC15
CLK1	-	SI5351 CLK1 Output. Used as Transmit frequency for binary 0 code
CLK2	-	SI5351 CLK2 Output. Used as Transmit frequency for binary 1 code
CO_S_AOUT	A6	CO Sensor analogue reading

Signal Name (Net name)	TEENSY 3.5 Pin	Description
DEMOD	-	Demodulated output (analogue circuitry) to Audio amplifier
I_0_RAW	ADC2.1+	Down-converted signal, 0 Degrees phase (I), amplified (200x, +23dB)
I_0_RAW_NA	ADC1.1+	Down-converted signal, 0 Degrees phase (I), Raw (NOT amplified)
I_180_RAW	ADC2.1-	Down-converted signal, 180 Degrees phase (-I), amplified (200x, +23dB)
I_180_RAW_NA	ADC1.1-	Down-converted signal, 180 Degrees phase (-I), Raw (NOT amplified)
I_RAW	A8	Down-converted I signal, 0 Degrees phase, amplified (1200x, +30,8 dB), Shifted by V = Vana/2
KBD_CLK	D32-SCK1	Keyboard interface CLK
KBD_ENC_A\	D53-MISO2	Encoder A input, or part of a synchronous serial output interface
KBD_ENC_B\	D52-MOSI2	Encoder B input, or part of a synchronous serial output interface
KBD_IN	D1-Tx1	Data from Keyboard interface
KBD_PAR_LD	D51-MISO2	Keyboard interface parallel load. Really not needed. Can be generated y quite simple HW in the Keyboard PCB
KBD_S_L\	D31	Keyboard interface Shift (High) or Load (Low) control
LCD_CS	D17-A3	LCD Display Chip select. Most Display signals are according to Graphics Libraries conventions
LCD_D0	D8P	LCD Display Parallel data D0
LCD_D1	D9P	LCD Display Parallel data D1
LCD_D2	D2P	LCD Display Parallel data D2
LCD_D3	D3P	LCD Display Parallel data D3
LCD_D4	D4P	LCD Display Parallel data D4
LCD_D5	D5P	LCD Display Parallel data D5
LCD_D6	D6P	LCD Display Parallel data D6
LCD_D7	D7P	LCD Display Parallel data D7
LCD_RD	D14P-A0	LCD Display Read
LCD_RS	D16-A2	LCD Display Register Select
LCD_RST	D18-A4	LCD Display Reset
LCD_WR	D15-A1	LCD Display Write
LO_I	-	Local Oscillator I signal (to Quadrature mixer)
LO_Q	-	Local Oscillator Q signal (shifted 90° to I signal, to Quadrature mixer)
PADDLE_RING	D48-Tx6	Originally Morse key or Iambic key ring. General purpose or auxiliary Serial Out
PADDLE_TIP	D47-Rx6	Originally Morse Key or Iambic Key tip. General purpose or auxiliary Serial In
P_CLK	D57-SCL3	Pressure Sensor Clock

Signal Name (Net name)	TEENSY 3.5 Pin	Description
P_RX	D55	Pressure Sensor Rx (to µP Tx)
P_TX	D56-SDA3	Pressure Sensor Tx (to µP Rx)
Q_90_RAW	ADC2.2+	Down-converted signal, 90 Degrees phase (Q), amplified (200x, +23dB)
Q_90_RAW_NA	ADC1.2+	Down-converted signal, 90 Degrees phase (Q), Raw (NOT amplified)
Q_270_RAW	ADC2.2-	Down-converted signal, 270 Degrees phase (-Q), amplified (200x, +23dB)
Q_270_RAW_NA	ADC1.2-	Down-converted signal, 270 Degrees phase (-Q), Raw (NOT amplified)
Q_RAW	A20	Down-converted Q signal, 90 Degrees phase, amplified (1200x, +30,8 dB), Shifted by V = Vana/2
RADIO_PWR_ENA	D24	Radio Power enable
RADIO_RX_ENA	D40	Radio Receiver enable
RADIO_RX_IN	-	RF Direct in to tuned Rx input transformer
RADIO_TX_CLK	D29P	Square wave. Edges signalling phase alignment between CLK1 and CLK2. Used as Data Tx Clock (Keying shall be aligned with RADIO_TX_CLK to maintain phase coherence)
RADIO_TX_DAT	D28	Tx Data. Selects between CLK1 and CLK2 to feed the RF PA to implement SFSK. Change in rising and falling edges of RADIO_TX_CLK
RADIO_TX_ENA_1	D41	Tx Enable, Level 1 Power (1 x BST170 Active)
RADIO_TX_ENA_2	D42	Tx Enable, Level 2 Power (2 x BST170 Active. If TX_ENA_1+TX_ENA_2 active, Level 3 Power)
RADIO_TX_KEYING	D30P	Keyer (Power supply to PA) control.
RADIO_TX_PWR	A10	Measure of Tx Power
RX_SEL	D34	Serial data selection (O2 or CO2) from sensor board
RX_SENSOR	D0-Rx1	Serial data from sensor board
SI_SCL_0	D37P-SCL1	SI5351 Serial Clock
SI_SDA_0	D38P-SDA1	SI5351 Serial Data
SENSOR_PWR_ENA	D24	Sensor Power enable
SHT_S_CLK	D33	SHT75 (Humidity and Temperature sensor) Clock
SHT_S_D	D35	SHT75 (Humidity and Temperature sensor) Data
S_TONE	DAC0-A21	Sidetone in the original design. Connected to DAC0, so can be used to inject digital voice to the audio amplifier.
VMON	A11	Power supply voltage monitoring

Table 1 – TEENSY Module Signals

4 RECEIVER DESIGN

The wireless receiver is loosely based on an open CW transceiver design, freely available from the amateur radio community, with significant changes and improvements. The main ones are:

1. A change in the RF input chain, making the use of independent Tx and Rx antennas possible.
2. Changed local oscillator synthesis method, to allow operation down to the VLF range.
3. A change in the first amplifier (used an LT6232 instead of two LM4562s), with better noise figure and output voltage swing.
4. The addition of two independent high-speed A/D converters (ADAR7251 and AD7380, not intended to be mounted simultaneously). Each has some interesting features; and deserve some testing. More details are provided below.
5. Added some interface circuitry to the transmitter stage, intended to make a Synchronous Frequency Shift Keying (SFSK) modulation scheme easier to implement (otherwise known as Minimum Shift Keying, MSK and Fast Frequency Shift Keying, FFSK. Data rate: 2400bps, possibly 4800bps with some modifications).
6. Added a form of output power control to the RF Power Amplifier, switching off one or two of the three BST170 output MOSFET.
7. The original microcontroller and user interface components, which had very low processing power, were replaced by the TEENSY 3.5 module and ancillary elements.

4.1 RF Front End

The receiver is a Zero-IF design, that can be wired to use the same or a different antenna than the transmitter.

The main input to the receiver is done through transformer TR1 (*Figure 2*), which has a tuned primary and an un-tuned symmetrical secondary.

When used with the same antenna as the transmitter, jumper PS2 is installed, and PS1 and C3 should be removed.

When used independently, there are two possible antenna connections: directly from J1, or through a low-pass filter from J2. The exact configuration will require some experimenting and will be decided at a later stage.

The secondary of TR1 is connected to the Quadrature (otherwise known as Tayloe) mixer, IC2, driven by the quadrature signals generated by the Local Oscillator (LO). The quadrature baseband signals are recovered at the terminals of C8 to C11, superimposed on a DC component injected in the mixer for fixing the DC operating point of the amplifying chain.

The baseband signals are fully differential, having 0°, 90°, 180° and 270° relative phase. They are amplified by a low noise differential stage built around IC1 (Gain 200 for each signal, 400 differential gain), and summed by an adder built around IC3. This adder is asymmetric (+input gain is 2 and –input gain is -1), for a total gain of 3. Asymmetry is not important in this case but can be reduced by increasing the gain of this stage.

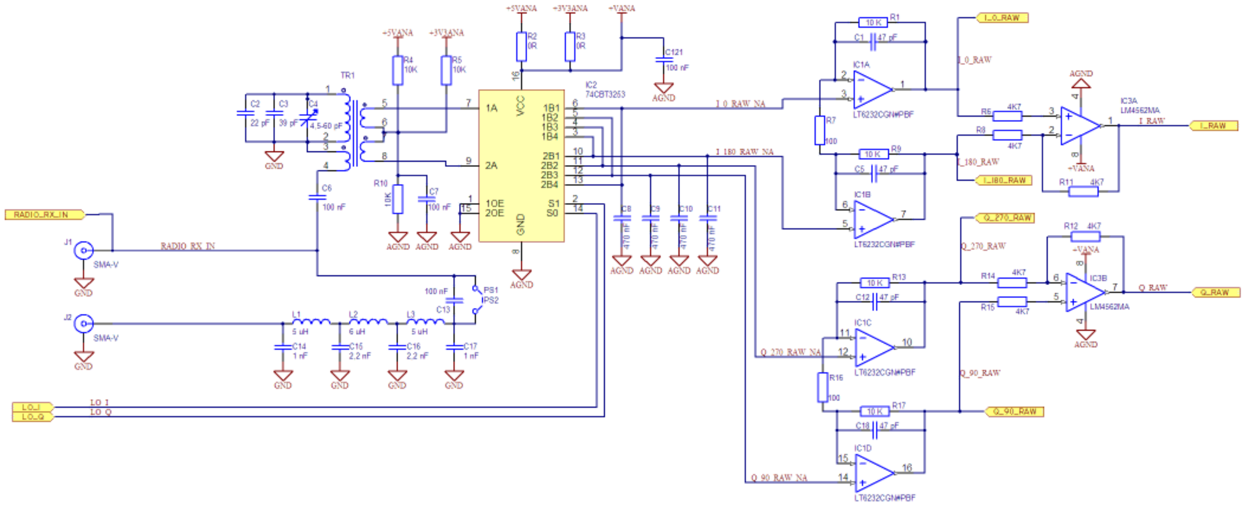


Figure 2 – XCVR-HF-01A WMN: RF Front End

At the outputs of IC3, therefore, the baseband I and Q components of the modulating signal are available.

In summary, I and Q signals are available in three levels (always with an added DC component equal to Vana/2):

1. raw as down-converted, not amplified, 4 phases or differential (I_0_RAW_NA, Q_90_RAW_NA, I_180_RAW_NA, Q_270_RAW_NA),
2. same but amplified (I_0_RAW, Q_90_RAW, I_180_RAW, Q_270_RAW),
3. as two quadrature single ended signals I and Q (I_RAW, Q_RAW).

The above signals are routed to different processing paths:

1. I_0_RAW_NA, Q_90_RAW_NA, I_180_RAW_NA andQ_270_RAW_NA are sent to IC6, ADAR725, which is a quite complex signal processing device (see more details below),
2. I_0_RAW, Q_90_RAW, I_180_RAW, Q_270_RAW are sent to IC7 (AD7380, a 4MS/s differential 16-bit, two simultaneous channels, SAR A/D converter),
3. I_RAW, Q_RAW are sent to an analogue processing chain and to two of the analogue inputs of the TEENSY 3.5. This module is based on a MK64FX512VMD12 Cortex-M4 Processor running at 120MHz, which has two internal A/D converters. The inputs for connecting I_RAW and Q_RAW were selected to belong to different A/D; so simultaneous or synchronous conversion is possible (This is essential for SDR processing).

4.2 Analogue Detection

I and Q signal are passed through two parallel unity gain (*Figure 3*) networks designed to shift the phase of inputs by angles whose difference is 90° (over a certain bandwidth, I signal phase is shifted by φ and Q signal by φ+90°). Phase shifting is frequency-dependent, and in the ends of the pass-band can be trimmed using P3 and P4 (it will be necessary changing values of some components for shifting the pass-band to 2400/ 4800Hz).

The result is two signals: I(φ) and Q (φ+90°). Addition or subtraction of these will cancel one sideband and double the output for the other, resulting in CW or SSB detection.

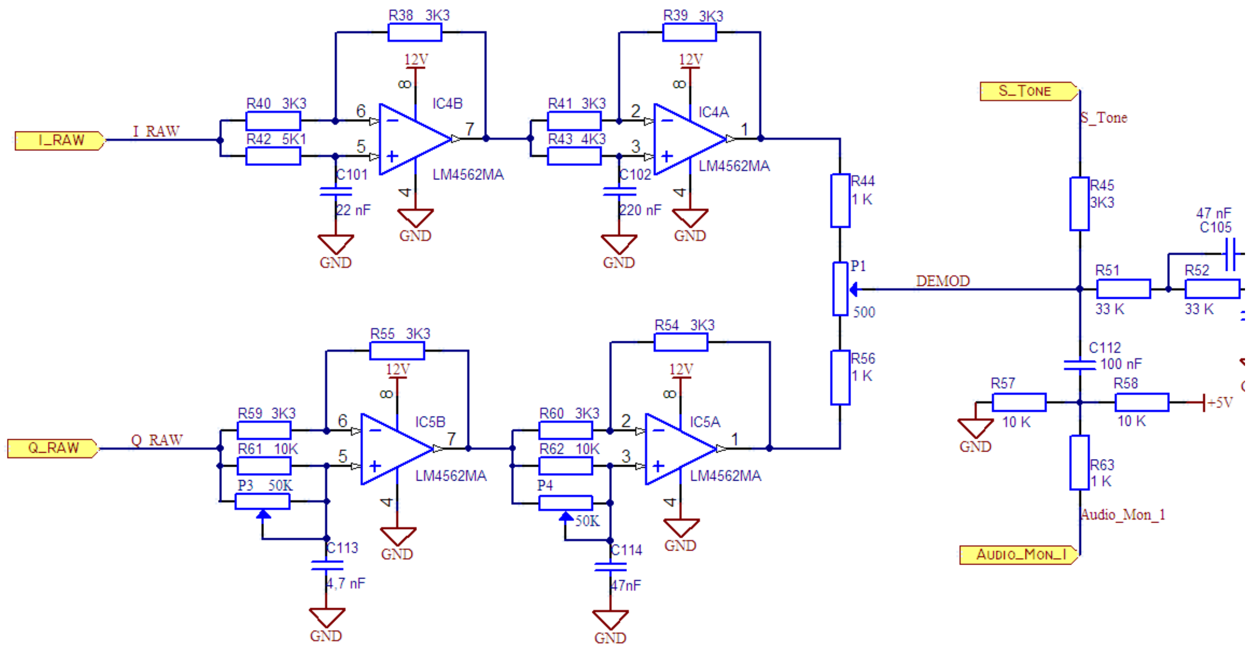


Figure 3 – XCVR-HF-01A WMN: Analogue Detector

4.3 Audio Amplification

The detected signal is feed in parallel to a pass-band filter (IC8 & IC9, *Figure 4*) and to another of the analogue inputs of the TEENSY as Audio_Mon_1. Most possibly synchronous detection of the SFSK signal will be possible at this point, but some tests are needed. Estimation is that between 2400 and 4800bps will be feasible.

Also, a side-tone can be injected here, or, indeed, any digital audio.

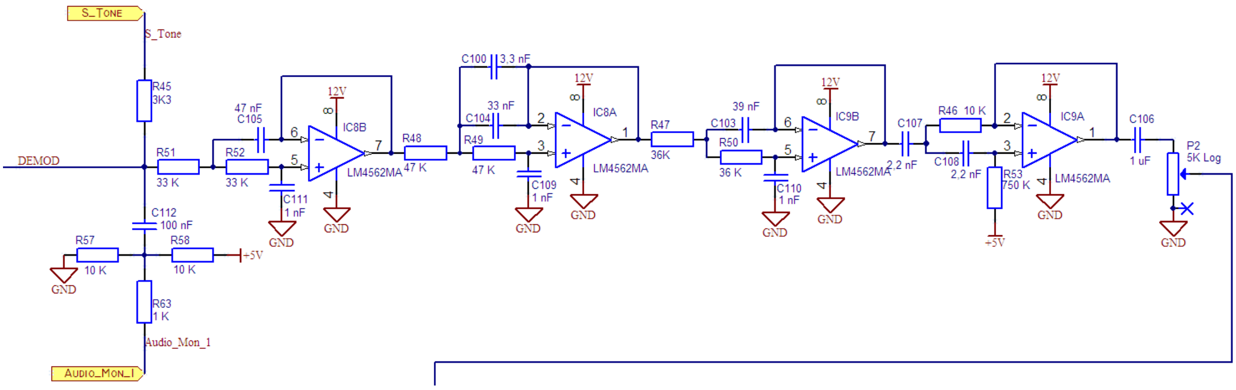


Figure 4 – XCVR-HF-01A WMN: Band-pass Filter

Finally, the filtered signal is sent through a volume potentiometer to the final audio stage (*Figure 5*). Audio can be silenced by the receiver enable – RADIO_RX_ENA – signal. The amplified audio signal is available at a stereo 3.5mm standard socket and routed as AUDIO_MON_2 to yet another analogue input of the TEENSY for monitoring or any other purpose. The purpose of including an audio stage allows testing and debugging.

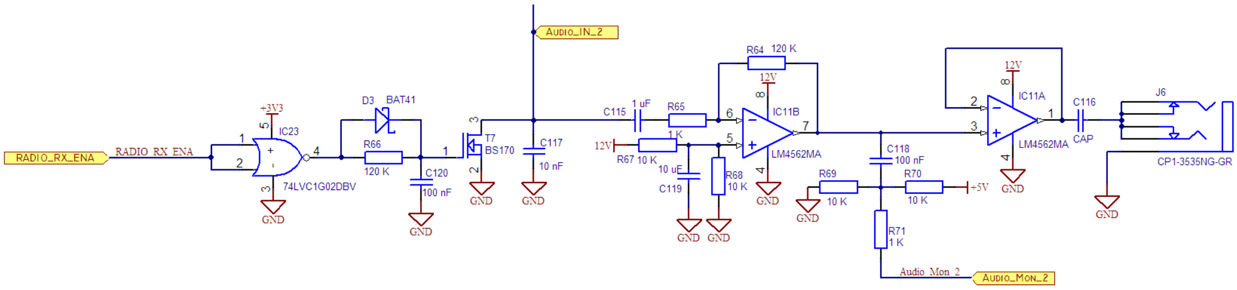


Figure 5 – XCVR-HF-01A WMN: Audio Amplifier

4.4 Local Oscillator & RF Generation

The Local Oscillator (LO) and transmission RF generator are built around a Si5351A, a 3-output flexible clock generator (*Figure 6*). CLK0 is used for generating the receiving LO. It runs at 4 times the receiving frequency, and its output is divided by 4 in IC15, which generates 2 square wave outputs with a 90° phase difference (I LO and Q LO). This method is used to allow operation in the LF range during reception .Other designs use CLK0 for generating the I LO signal and CLK1 for generating the Q LO signal, but this last approach does not allow for an LO -or reception – frequency below 3.5MHz, due to limitations in the internal architecture of SI5351.

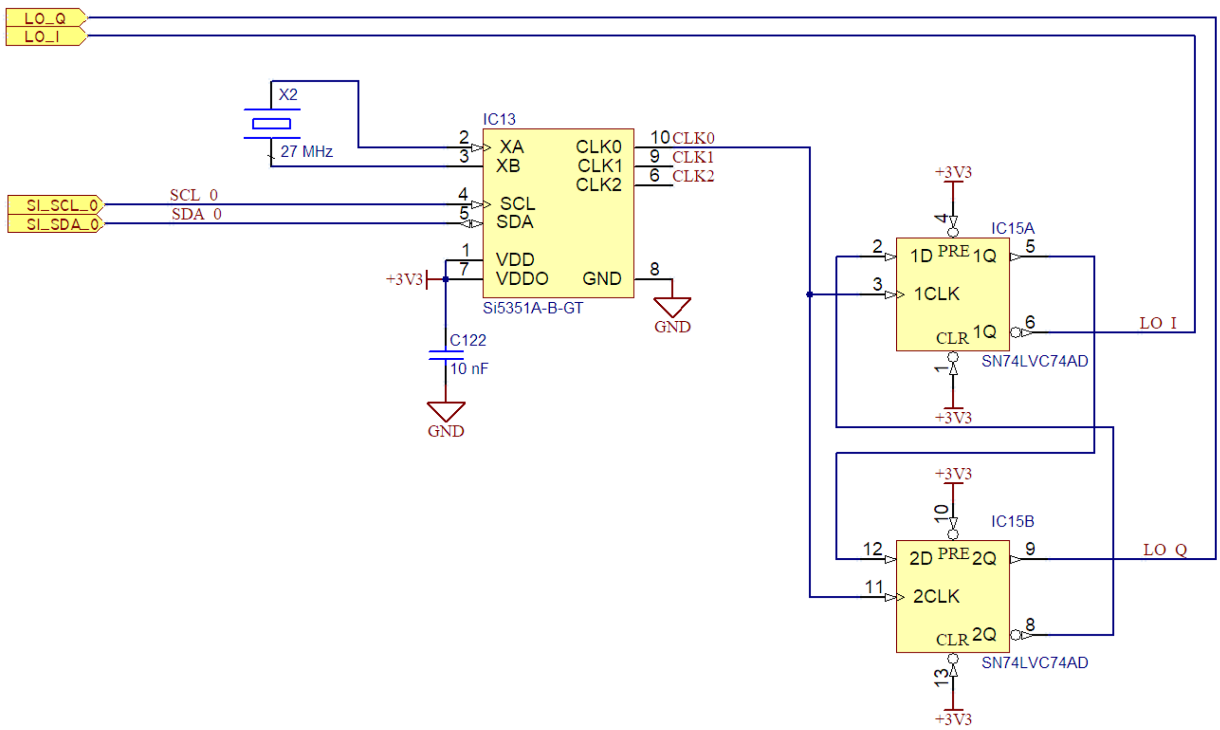


Figure 6 – XCVR-HF-01A WMN: ReceiverClock Generation

CLK1 and CLK2 are used for generating transmitting signals. More details are provided below.

4.5 Receiver Extra Features (Experimental)

Two extra devices were added for experimental purposes (ADAR7251 and AD7380, not intended to be mounted simultaneously). Both include at least two independent high-speed A/D Converters. Each has some interesting features, and deserve some testing. More details are provided below.

An ADAR7251 (the internal architecture of which appears in *Figure 7*) is a 16-bit, 4-channel, simultaneous sampling differential analog-to-digital converter (ADC) designed especially for applications such as automotive LSR-FMCW or FSK-FMCW radar systems. Each of the four channels contains a low noise amplifier (LNA), a programmable gain amplifier (PGA), an equalizer, a multi-bit Σ-Δ ADC, and a decimation filter.

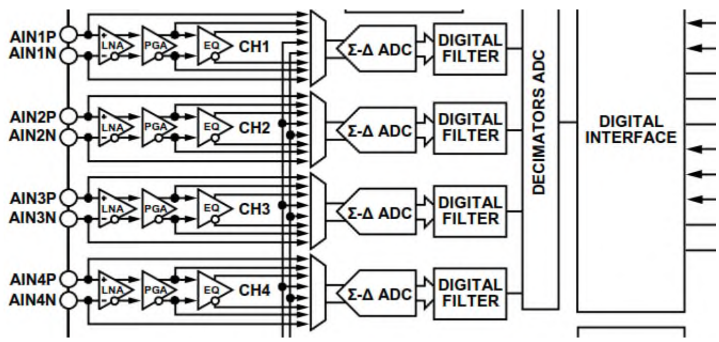


Figure 7 – Internal Architecture of ADAR7251

Only two channels are used, as shown in *Figure 8*. Gain and filtering can be programmed using its SPI interface, allowing a flexible signal processing.

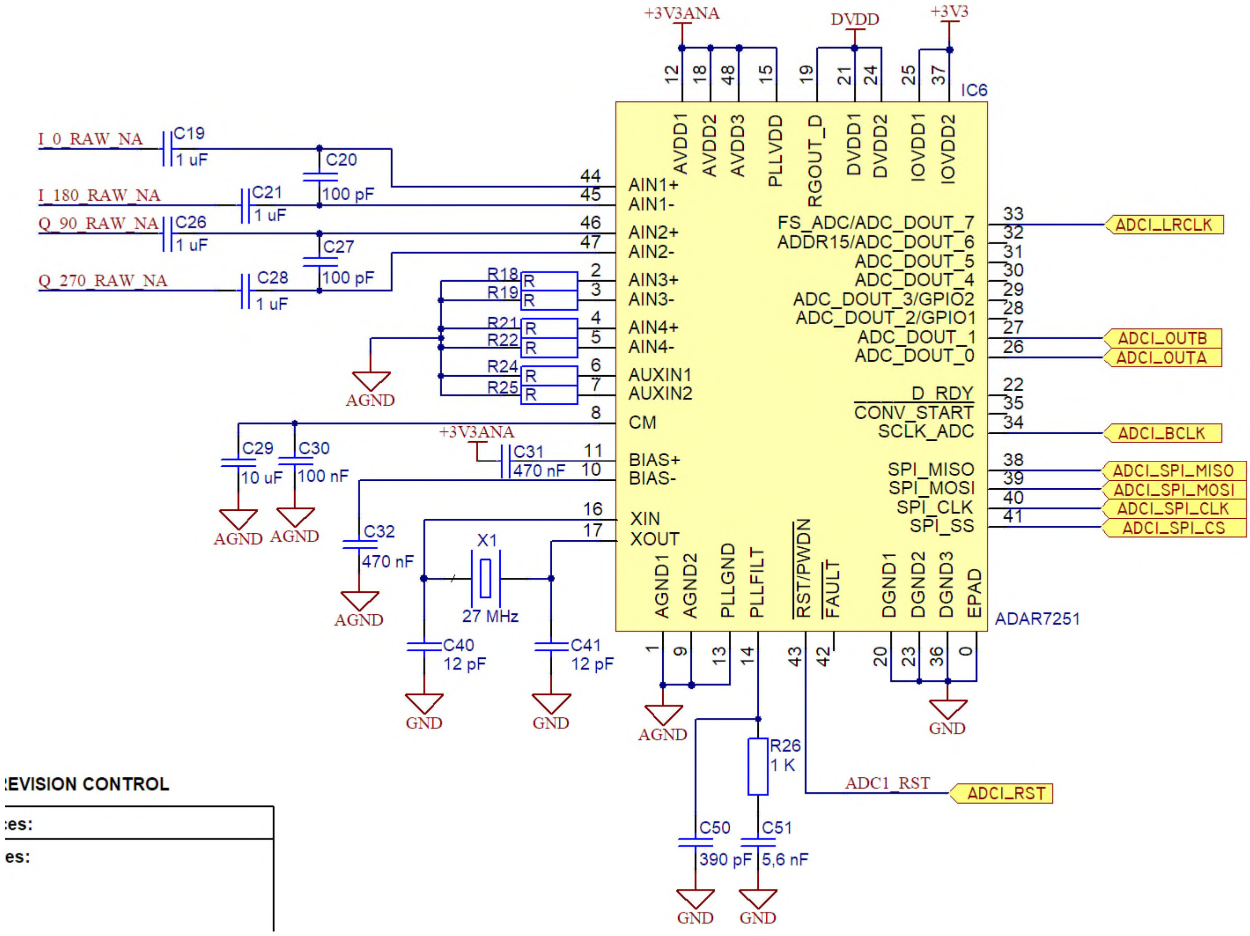


Figure 8 – XCVR-HF-01A WMN: ADAR7251 Connections

The AD7380 (*Figure 9*) is just a two-channel, synchronous, 16-bit fully differential 4 MS/s successive approximation register (SAR) A/D converter. Integrated on-chip oversampling blocks improve dynamic range and reduce noise at lower bandwidths. A buffered internal 2.5V reference is included. Alternatively, an external reference up to 3.3V can be used. It is intended to replace the internal A/D of the TEENSY, which can offer, at most, single-ended 14-bit resolution.

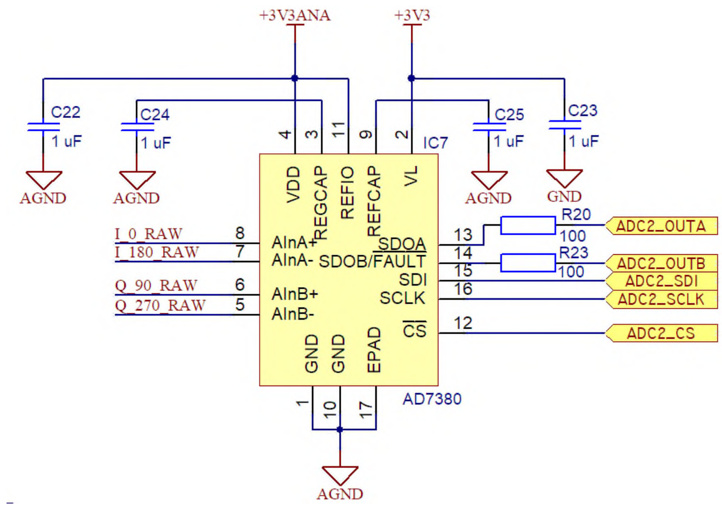


Figure 9 – XCVR-HF-01A WMN: AD7380 Connections

This page is intentionally left blank.

5 TRANSMITTER DESIGN

5.1 RF Power Amplifier

The transmitter section is much simpler than the receiving one. The original design on which XCVR-HF-01 is based is a CW transceiver and, therefore, a linear output stage is not needed, a more efficient Class E one being sufficient (*Figure 10*). A square wave is used to switch three MOSFETs in parallel (T4 to T6), which then generate RF power at their drains. This signal is sent to the antenna through a Low Pass Filter (LPF) to eliminate higher order harmonics and obtain a pure (or almost pure) tone.

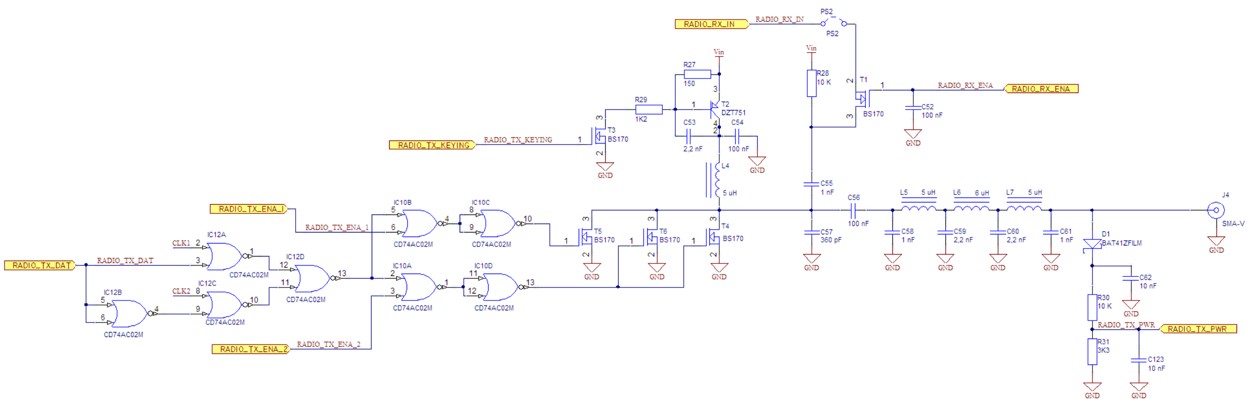


Figure 10 – XCVR-HF-01A WMN: RF Power Amplifier

Moreover, the input or the LPF is routed through T1 (controlled by radio Rx enable signal RADIO_RX_ENA) and PS2 to the receiver input stage, when a single antenna is used.

5.2 Synchronous Data Interface

However, the radio stage “as is” is not appropriate for higher than Morse keying transmission speeds, as it is keyed using a power PNP transistor (T2) with some “shaping” circuitry that slows its ON-OFF switching.

Therefore, it was modified to allow for higher data rates. SI5351’s CLK1 and CLK2 are used for generating transmitted signals. They are programmed to generate two signals 2400 and 4800Hz above CLK0/4, which is the nominal carrier frequency. Frequency shift keying is done routing either CLK1 or CLK2 to the PA, which is equivalent to an SSB (USB) modulation with 2400 or 4800Hz. Selection between CLK1 or CXLK2 is done by the gates in IC12, according to the value of RADIO_TX_DAT.

The main issue with this method is that keying must be done at “0 phase” in the baseband. Translated to RF terms, this means that switching between CLK1 and CLK2 must be done when they are “in phase”. To detect this point, CLK1 is routed to the 1D input of IC14 (*Figure 11*) and CLK2 to its 1CLK input. This arrangement is a phase detector that switches 1Q output at 0° and 180° phase difference between CLK1 and CLK2. The switching frequency of the signal at 1Q (RADIO_TX_CLK) is the difference between CLK1 and CLK2, or 2400Hz.

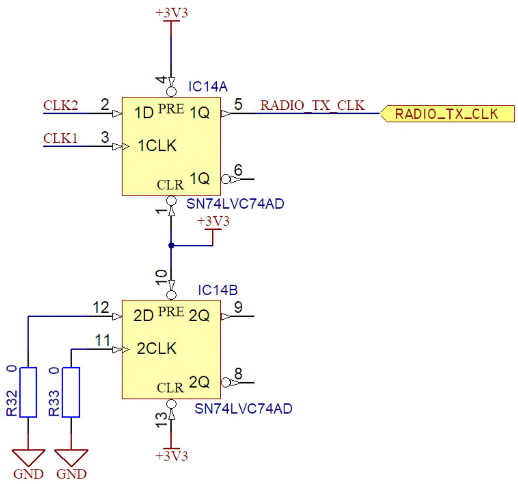


Figure 11 – XCVR-HF-01A WMN: Synchronous Data Transmission Clock Generator

Therefore, on the rising edge of RADIO_TX_CLK (through an Interrupt Service Routine, ISR), RADIO_TX_DAT is switched according to the value of the next bit to be transmitted. In baseband terms, this means that the modulating signal is switched from 2400 to 4800Hz at 2400bps and 0° phase, and therefore one or two full signal cycles fit in the bit transmission time.

5.3 Microphone Interface

An external microphone can be connected through the keyboard connector J9 (*Figure 12*). Two of the pins in this connector are routed to analogue inputs (A12 and A13) and can be used to convert the microphone signal to digital form for life sign monitoring or for voice transmission. Depending on the type of microphone used, the connection can be made directly (three terminal microphones) or will need a few external components (for using two terminal electret microphones, for example). Any other pin in the connector can be used for implementing a Push-to-Talk (PTT) functionality.

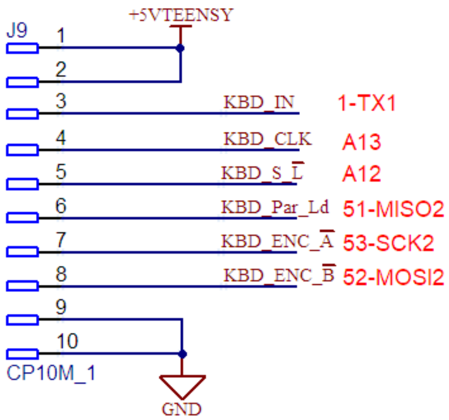


Figure 12 – XCVR-HF-01A WMN: Keyboard/Microphone Interface

6 POWER SUPPLY

The power supply is designed to accept an input coming from an external battery or power supply which, in future versions, can be an intrinsically safe one.

It shall allow selective enable / shutting down of the radio and sensor interfaces. Other design requirements were having idle and shutdown currents as low as possible, and low output noise. An LP2989 regulator was found to be the best compromise. It can deliver up to 500mA, has a maximum input voltage of 16V, has a low noise figure, further improved by adding a bypass capacitor, and a typical shutdown current of 50nA.

Seven integrated circuits are used, although only one regulator is active at all times, the one supplying 5V to the TEENSY module (*Figure 13*).

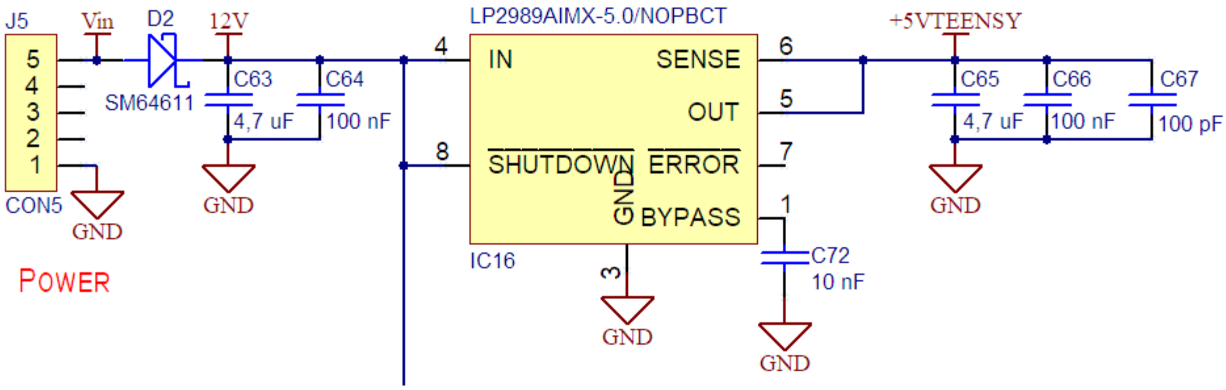


Figure 13 – XCVR-HF-01A WMN TEENSY Module Regulator

The remaining regulators are controlled by the TEENSY module, allowing a selective shutdown (or enabling) of the radio and sensors circuit sections.

For further reducing the shutdown current, the regulators are organized in three “chains”:

1. power for the digital circuits of the radio,
2. power of the analogue circuits of the radio, and
3. power for sensors.

The first regulator in each chain (*Figure 14*) provides a 5V output. The input of the second, providing a 3.3V output, is connected to the output of the first. So, it is only necessary to control the first regulator in the chain, reducing the power consumption of the second one to zero when in shutdown state.

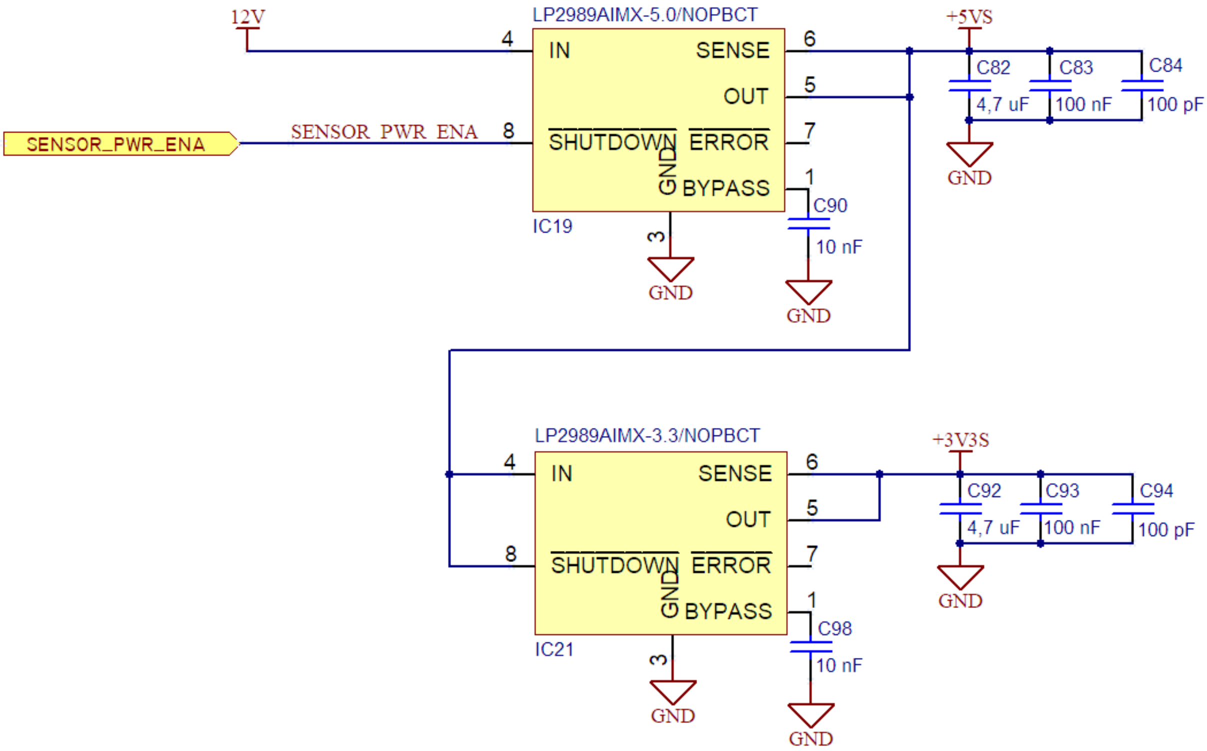


Figure 14 – XCVR-HF-01A WMN Regulator Chain (Sensors)

7 ANTENNAS

7.1 Introduction

In this section, we present some information on the prototype antennas (not all of which were sized to fit the TTE node enclosure) that were produced and their testing, plus design guidelines. However, it is important to recognise that a very significant amount of background work was also carried out, investigating a broad range of other antenna options. Given that many aspect of a real-world TTE system would have to be adapted to a particular mine environment, much of this background work would be relevant to anyone engaging in a project aimed at migrating the proof-of-concept developed in INDIRES into a form in which it could be tested and deployed in a specific mine. The full report on antenna options is available to interested parties – see *Section 11*.

7.2 Air-cored Loops

7.2.1 Purpose

The main objective of these tests was to characterize some types of air-cored loops (or antenna coils), measuring their electric parameters, to:

1. Evaluate the actual efficiency of this type of antennas (In the literature there are only theoretical analysis, but little or no actual performance data)

2. Select the types of antennas most adequate for the mesh-nodes and sensors be developed.

3. Use that data in the design of the transmitting and receiving stages of the radio interface of these mesh-nodes.

7.2.2 Construction and Characteristics – Initial Screening

A number of loop (or coil) antennas were built and tested in the laboratory, wound with 1mm solid magnet wire on forms with diameters of 61, 73, 130, 168mm. Coils of single and dual layers were tested, as well as tight and loose coiling pitches. The forms, made of PVC, were grooved in the lathe for guaranteeing a constant coiling pitch.

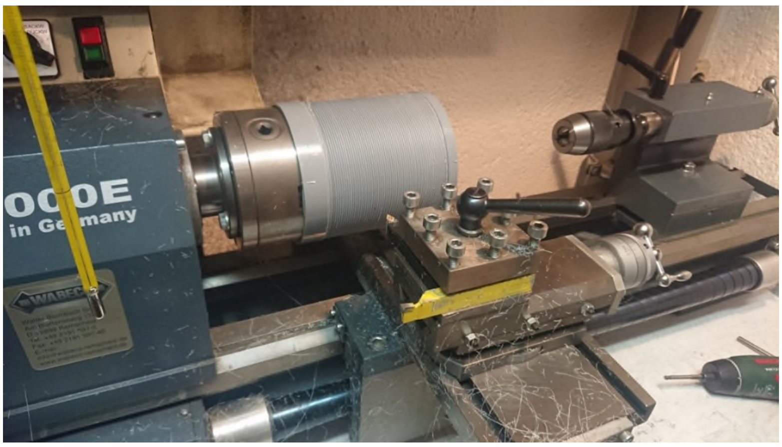


Figure 15 – Grooving in the Lathe a 130mm Coil Form

Also, big coils with larger diameters (500 and 1200mm), wound with copper tubing, of 6.3mm diameter, were built.

Figure 16 – Winding a 74mm Diameter Dual Layer Coil

For all antennas, Self-Inductance was measured using an L/C Meter IIB from AAD Electronics. For some small antennas, inductance was measured with both air and ferrite cores, while for the bigger ones only the air-cored configuration was measured (*Figure 17*).

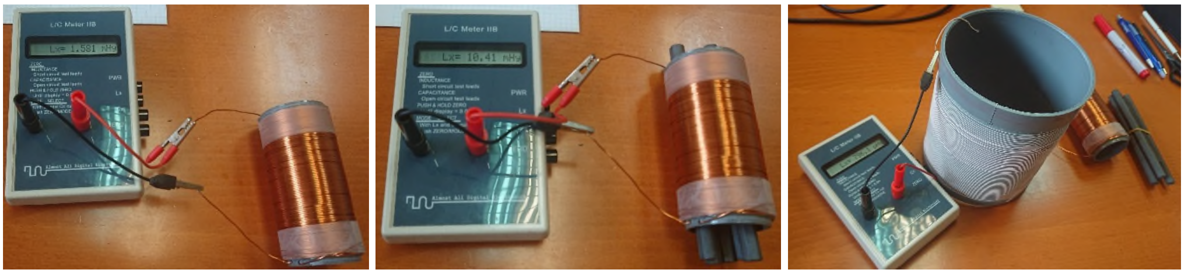


Figure 17 – Measuring Self-Inductance of a Coil Antenna

Initial performance tests were carried out with the above coils, using a low power signal generator as source, serial tuning with a variable air capacitor, and an SDR (Software Defined Radio) receiver for monitoring the level of received signal at 25m.

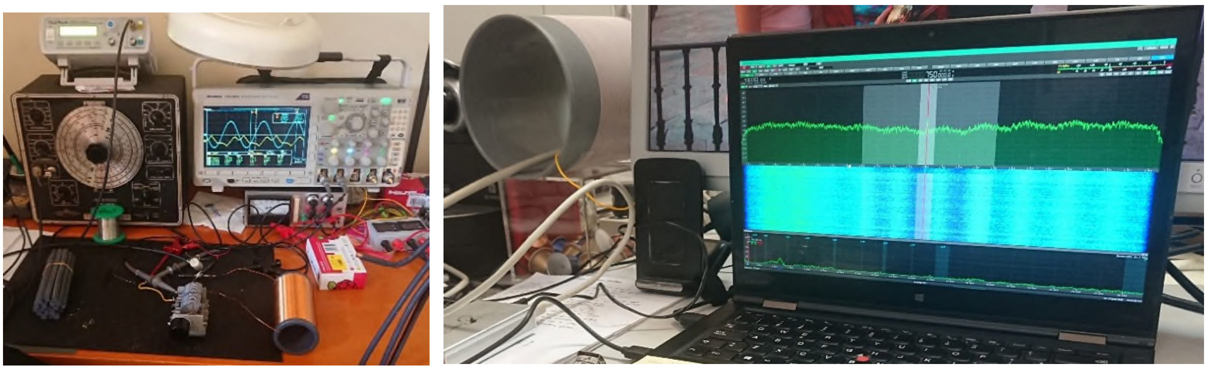


Figure 18 – Testing a 73mm coil.
Left: ferrite core, tuning capacitor and coil. Right: SDR receiver display

In these initial tests, as predicted by the theory, it was noted the difference in performance between coils of lower and higher diameters. It was also noted that dual layer coils had lower performance than single layer ones, and that sparsely-spaced pitches were preferable to tight coiling. Also, that the use of ferrite cores in the lower diameter coils did not improve significantly their performance,

Following this first screening, the use of smaller diameter coils was discarded, at least for fixed sensors, for the performance was much lower than that of the bigger ones. It was also discarded the use of multilayer coils. Consequently, detailed tests were carried out mainly on 168mm single-layer, air cored, coils. However, for completeness sake, one 74mm diameter and both copper-tubing coils were tested.

Figure 19 – Some Antennas Tested in the Laboratory
Center: 1200mm coil. Right: 500mm coil.

Table 2 summarises the coil antennas that were object of parametric testing in the laboratory.

Coil #	Coil Dia. (mm)	Turns	Pitch (mm)	Length (mm)	Length /Dia	Wire Dia. (mm)	Wire length (m)	DC R (Ω)	L (µH)	C (pF)	SFR (Hz)	Wire Type
1*	74	70	2	140	1.89	1	16.27	0.34	151.6	5.50	5.511.746	Solid magnet wire
2*	168	27	2	54	0.32	1	14.25	0.30	164.0	8.40	4.288.037	Solid magnet wire
3*	168	60	2	120	0.71	1	31.67	0.66	517.0	7.20	2.608.607	Solid magnet wire
4*	168	60	2	120	0.71	175x 0.04	31.67	2.50	519.0	8.40	2.410.443	Litz wire ***
5**	168	80	2	160	0.95	1	42.22	0.88	755.0	9.00	1.930.747	Solid magnet wire
6**	168	140	2	280	1.67	1	73.89	1.55	1475	10.80	1.260.992	Solid magnet wire
7**	168	200	2	400	2.38	1	105.56	2.21	2200	13.50	923.511	Solid magnet wire
8*	500	9	15	135	0.27	6.35	14.14	0.01	37.6	37.00	4.267.027	Bare copper tubing
9*	1200	4	30	120	0.10	6.35	15.08	0.01	37.6	40.50	4.078.483	Bare copper tubing

Notes: * Tests carried out using a switched capacitor bank. Nominal capacitance used in calculations.

** Tests made using a 10-100 pF high voltage vacuum capacitor.

*** 175 strands of AWG #46 wire. 0.079 Ω/m in DC, equivalent to 0.2 mm^2 solid copper

Table 2 – Summary of Small Loop Antennas Object of Impedance Testing

7.2.3 Testing Methodology (Electrical Parameters)

For the initial tests, variable capacitors of the type installed in old broadcast receivers were used for tuning the coil antennas. These provided a convenient and quick way for serial-tuning them, and for expedite performance testing, but had several issues. The first was the difficult repeatability, which required measuring its capacitance in each test. The second was related with the test results themselves. The values for *Rs* where higher than predicted by the models and the literature. It is a known fact that the wiper in these devices can have significant resistance. This is not too important for its normal use in receivers but would introduce significant errors when carrying out resistance measurements.

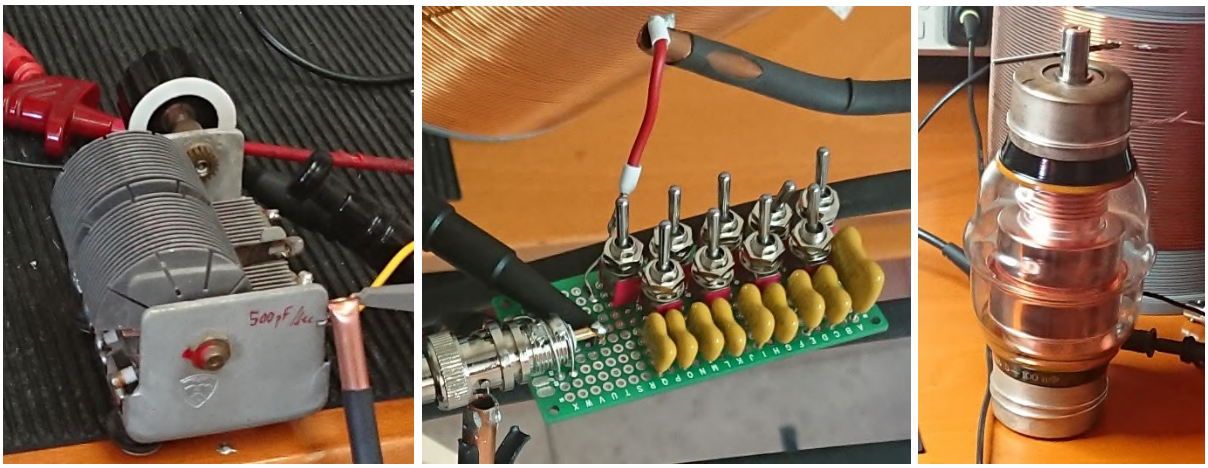


Figure 20 – Capacitors for serial tuning used in tests. Left to right: Standard variable capacitor, switched bank of silver mica capacitors and vacuum trimmer.

Therefore, in the second round of tests, it was used a switched capacitor bank, built with silver-mica capacitors. These had satisfactory performance for parametric testing and when applying low power. However, they could not withstand the current and voltages appearing when applying moderate powers (tens of wats). Consequently, in the last rounds of tests, a high power-high voltage vacuum trimmer was used for tuning the antennas.

For measuring the electrical parameters -other than self-inductance- of the antennas, it was used a Mini VNA-Pro Vector Network Analyzer (VNA, http://miniradiosolutions.com/minivna-pro/). A VNA is an instrument used to characterize electronic components and devices in high frequency. Mini VNA-Pro is an USB device specifically designed to characterize antennas, which operates from 100kHz to 200MHz. Together with VNA/J (https://vnaj.dl2sba.com/) software, it can measure and plot the most interesting electrical parameters as a function of the signal frequency.

The first step when using it is performing a calibration of the measuring jig. Three frequency sweeps are performed with the testing connection first open, then short-circuited and finally connecting a 50Ω load. Then, a range of interest in frequencies is defined, and the antenna scan can be carried out. The output of the instrument is presented in VNA/J window (*Figure 21*)

The pair Mini VNA Pro + VNA/J measures the following parameters:

- Return Loss and Phase (***RL*** and ***RP***). In telecommunications, return loss is the loss of power in the signal returned/reflected by a discontinuity in a transmission line. In this case, the discontinuity is a mismatch with the terminating load, the antenna.
- Impedance magnitude ($|\boldsymbol{Z}|$, modulus of the complex reactance Z = Rs + j Xs)
- Ohmic resistance and pure reactance (***Rs*** and ***Xs***)
- Angle of ***Z*** in the complex plane
- Standing Wave Ratio (***SWR***), a measure of the impedance mismatch of the antenna with an ideal generator of a given (usually $(50 + 0\ j\ )\ \Omega$) output impedance. For a perfect match, $SWR = 1$, for a total mismatch $SWR = \infty$.

Any pair of these parameters can be plotted simultaneously, which allows a quick inspection of the behaviour of the Device Under Test (DUT). During the tests, usually Ohmic Resistance and pure Reactance were selected, but sometimes return loss, or phase proved to be more useful.

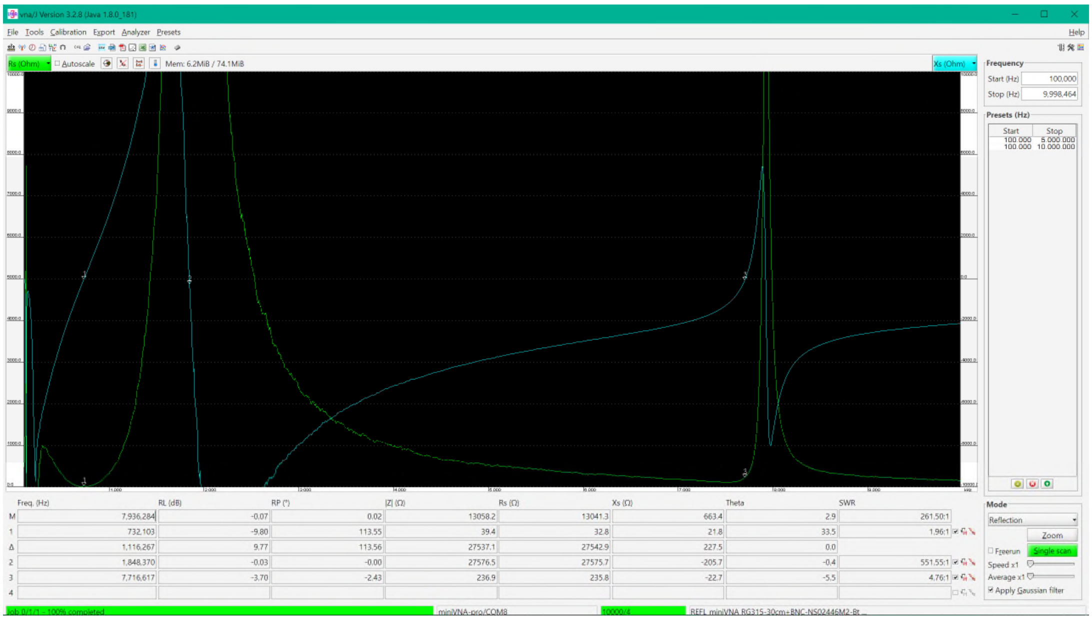


Figure 21 – Measuring the Parameters of an Antenna: VNA/J Screen Capture

VNA/J offers the possibility of placing up to three markers in the display window. For each marker, it is displayed the value of the Frequency and of the above parameters. During tests, marker #1 was set approximately on the first serial resonance frequency (where *Xs* changed the sign, *Figure 22*). Ideally, at this point Xs = 0, but, due to quantization in the frequency sweep, sometimes it had a small, either positive or negative, value.

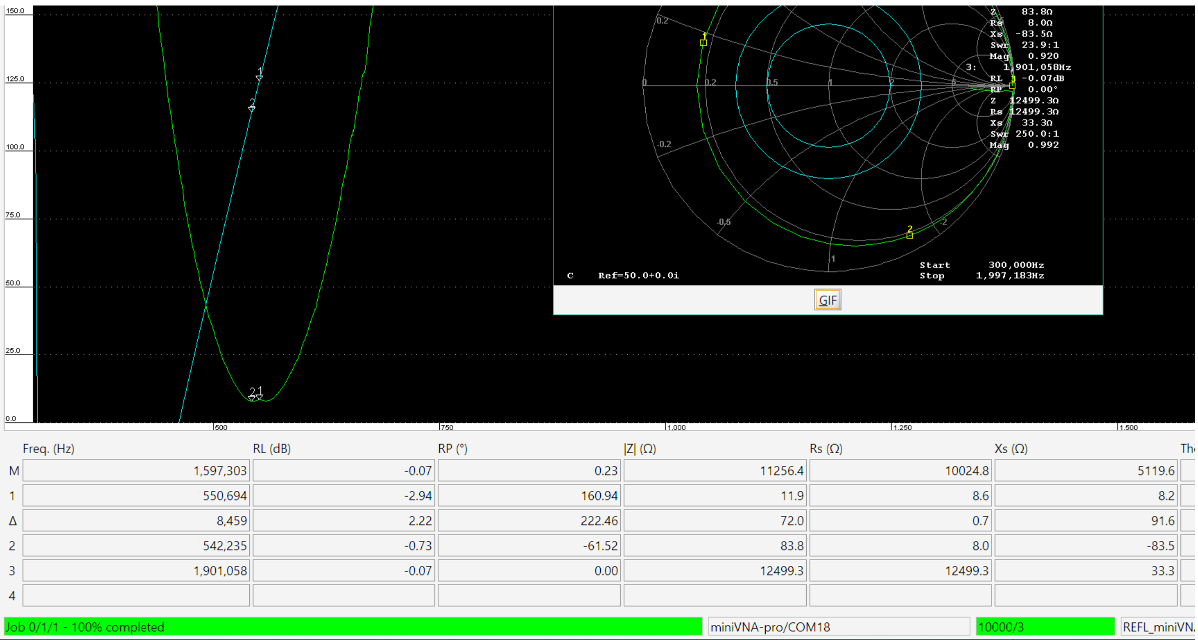


Figure 22 – Measuring the Parameters of an Antenna: Detail of VNA/J Screen Capture

In the above picture (*Figure 22*), the resonant frequency was 550.694kHz, *Rs* = 8.6Ω and *Xs* = 8.2Ω. These values were recorded and tabulated in a Test Sheet for each tested coil, together with the value of the tuning capacitor.

The test sheet contains two main sections. The first summarizes the geometric and electric data for each coil: Diameter, length, number of turns, Self-inductance and capacitance, etc.

Coil Test sheet | **Coil # 1**

Coil data			Wire data			
Diam. (m)	Turns	l_{Cu}	Diam. (mm)	Equiv. h_{SQ}	ρ_W	d/s
0,074	70	16,27	1,024	0,907	0,02095	0,454

Jig Data	Coil Electrical Parameters					
C_{PS} (pF)	R_{DC}	L (µH)	C_P(pF)		SRF (Calc.)	SRF (Meas.)
7	0,341	151,6	5,5		5.511.746	

Figure 23 – Test Sheet Heading for Coil #1: General Coil Data

The second contains actual test data: Frequency f, Wavelength λ (m), Skin depth δ, the ratio between AC and DC resistance at that frequency (Factor R_{SE+PE}/R_{DC}), Serial tuning capacitance, Cs, the value of $1\text{-}\omega^2LC$, the value of the factor $D = (1-\omega^2LC)^2 + (\omega RC)^2$, the expected value of Rs according to the Lumped Parameter model ($R'_{SE+PE}= R_{SE+PE}/D$) and the actual measured value of Rs (R_M). The difference among these is also computed. In another column there is the product of this difference by D ((R_M - R'_{SE+PE}) * D). This value should be the radiation resistance. For reference, it is also included the value of the calculated radiation resistance. Finally, the estimated value of the resonance frequency, according to the Lumped Parameters model, is computed, as well as its percentual difference to the actual resonant frequency.

Moreover, as a double check of the measuring methodology and data reliability, in some cases an additional set of measurements was taken, inserting a 50Ω RF resistor in the middle of the coil. It was expected that this resistor would increase the value of Rs by 50/D Ω. Any discrepancy would indicate either an issue with the readings or with the models / methodology.

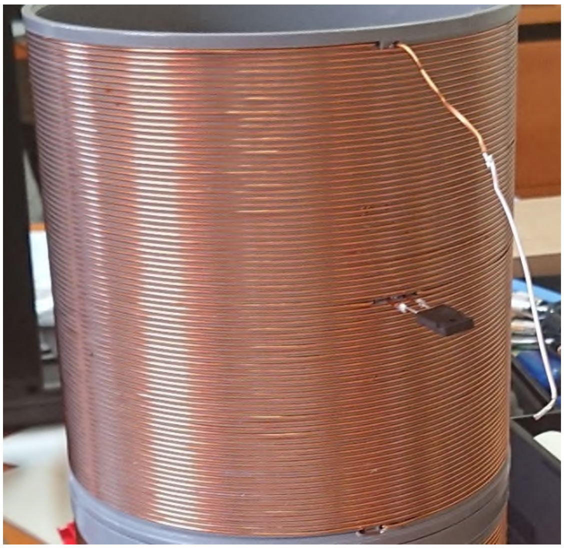


Figure 24 – 50 Ω RF Resistor Inserted in a Coil Antenna

When these measurements were carried out, two additional columns were added: R'_M, the measured value, and the product $(R_M – R'_M) * D$. The value of this product was expected to be 50 Ω.

7.2.4 Test Results

Test results are in general in good agreement with the predictions of the literature for the value of self-capacitance, and with the Lumped Parameters model for the resonance frequencies. Also, in the cases in which the 50Ω resistor was inserted in the coil, the change in the measured resistance was very close to the expected value.

However, from the first measurements, an outstanding fact was apparent: the value of the measured Radiation Resistance, column $(R_M - R'_{SE+PE}) * D$, computed as difference of the measured value for the Ohmic resistance at resonance and the value predicted by the model (considering the skin and proximity effects) was much higher than anticipated.

This fact was unexpected. The experts consulted, and literature search did not provide any clear explanation. One possibility could be that the formulas used were out of its field of validity. This issue was observed for frequencies such that the wire length was higher than 0.15-0.2 λ. But, when deriving the formula used for computing the radiation resistance of loop antennas, some assumptions on current distribution were made, which are valid only for lengths of wire lower than 0.1 λ. Therefore, one plausible explanation may be that, in these conditions (wire length higher than 0.15-0.2 λ), the usual formula for computing the radiation resistance of small loop antenna is no longer valid, and they are much more efficient than expected. However, it was not found any reference in the literature supporting this hypothesis. A second possibility is that the increase in AC resistance (and therefore ohmic losses) were higher than predicted by the literature.

On the other hand, even if reasons for that behavior are not totally understood, it was considered favorable for the implementation of the wireless system. An apparent impedance in the ranges measured fits very well with the output characteristics of standard High Frequency Power Amplifiers (HF PA). This fact allows injecting appreciable currents in a tuned antenna without adding impedance matching circuitry.

Nevertheless, it was considered interesting to try to find some explanation for the unexpected increase in the value of the apparent ohmic resistance of the serially-tuned antenna. For this purpose, injecting significant amounts of power to a tuned antenna could provide some clues. As said in section, the apparent ohmic component of the impedance of any antenna is composed of two terms: The radiation resistance, representing the energy going into the surrounding space, and the actual AC resistance, representing the energy transformed into heath by the Joule effect. Therefore, if, when injecting high power in the antenna, it heated-up, the responsible for the increase in the apparent resistance would be the loss component. On the other hand, if there is no appreciable heating, it can be assumed that the increase in the apparent resistance is due to radiation of energy.

7.2.5 Medium Power Transmission Tests

Therefore, the next step was building a medium power wideband PA (20W nominal). The PA had a 500kHz - 30MHz operating range.

After testing the PA with a dummy load, it was connected to a serially tuned antenna. First trials were made using a silver-mica capacitor for tuning, but it was blown in few seconds. Silver-Ceramic trimmers didn't behave better, heating and having general bad performance. Standard (500V) air trimmers did show arcing, due to the very high voltages appearing in them (*Figure 25*, this was predicted by the theory). It has to be noted that capacitors with air as dielectric can be used, but they would need more inter-plate spacing, and were not readily available (they had to be custom built).

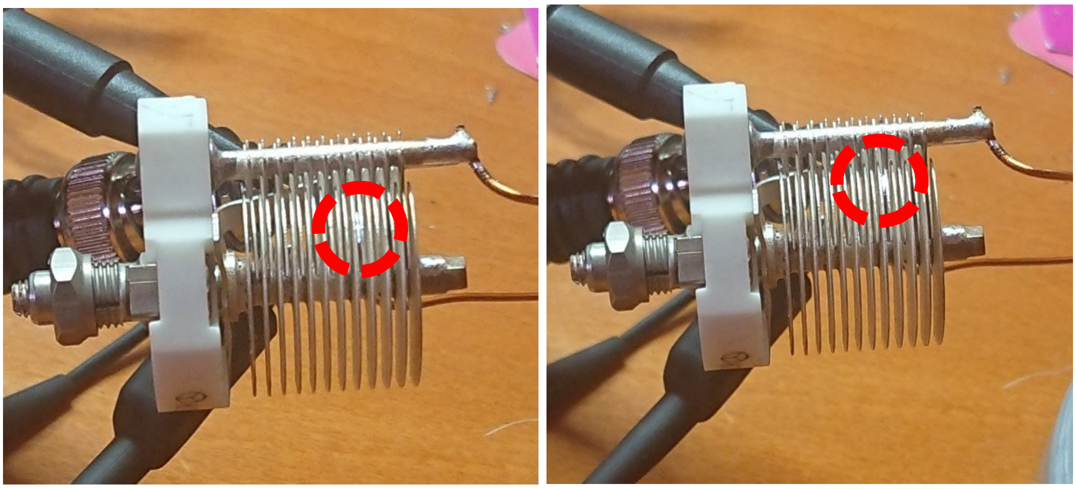


Figure 25 – Arcing in an Air Trimmer Capacitor

Therefore, for the subsequent tests, a vacuum trimmer (15kV, 10-100pF) was used. This trimmer had a good repeatability and did not show any arcing issue. The good tuning and efficient power transfer to the coil was demonstrated visually in a quite spectacular way (*Figure 26*). When the antenna was properly tuned, a fluorescent lamp was lighted by the electric field in the proximity of the coil and capacitor, with no contact or external power. Glow and internal discharge were apparent in gas discharge displays (Nixie tubes), again with no apparent power supply, also due to the intense field.

Measurements taken with an oscilloscope did show a good matching between the PA and the antenna system, that is, the impedance seen by the PA was high enough for it to transfer its nominal power. It was interesting to notice that there was no apparent temperature elevation in the coil, fact that suggests that the energy is transferred to the external medium. This does not mean necessary that energy was radiated, but that it was somehow coupled to the surroundings.

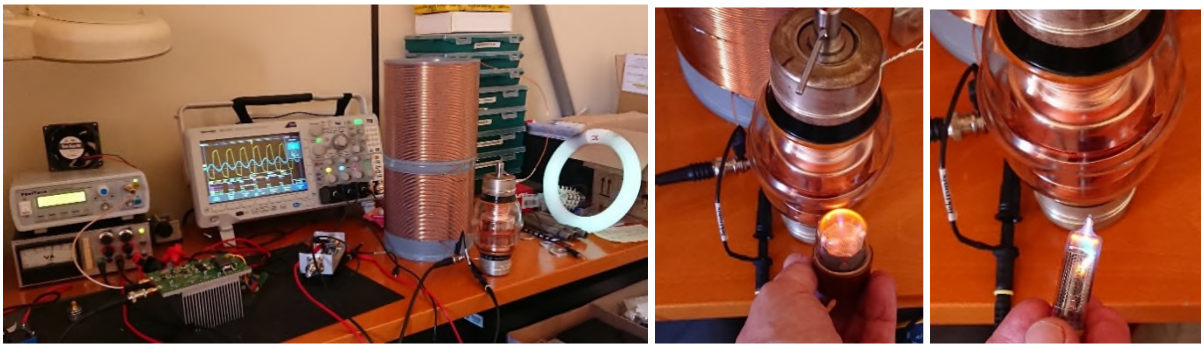


Figure 26 – Injecting RF Power to a Coil Antenna

Measurements taken with a handheld broadband scanner (AOR-AR8200) indicated that radiation was not efficient (signal was lost at 50m), although these indications should be taken cautiously, as the conditions in the laboratory did not allow taking proper signal strength measurements. However, high signal levels were detected in the proximity of electric wiring, distribution boards and devices connected to mains power (for example, a vending machine located at the equivalent of 5 floors in vertical and 50m in horizontal from the antennas, *Figure 27*)

Figure 27 – Signal Level Close to Power Wiring (Vending Machine)

This fact suggested that a good coupling to nearby conductors was achieved, and this was considered a very positive fact, as, usually, in the mines' roadways there are power lines and other wiring, which can increase the operating range of wireless transceivers significantly. A question regarding this coupling was if it was radiated or conducted, that is, if signal went through the cables in the power supply (conducted) or directly from the antenna to the mains wiring. To elucidate this issue, it was built a power signal generator (*Figure 28*), composed of a Direct Digital Synthetizer (DDS), a microcomputer for controlling it and a PA. The microcomputer was programmed to show on a screen the main operating parameters of the generator, and to accept control inputs from the encoders in its front panel. The device was designed to be powered either from an external power supply or from batteries, isolating it from the mains network, and, as an added benefit, allowing its use in field testing in mines. It was able to deliver up to 20W of RF power on a 50Ω load, 80 times more than the 0.25W that a standard generator can supply when powered form a 24V Power Supply. When powered from a 12V battery, its output was halved to 10W, enough for testing purposes.

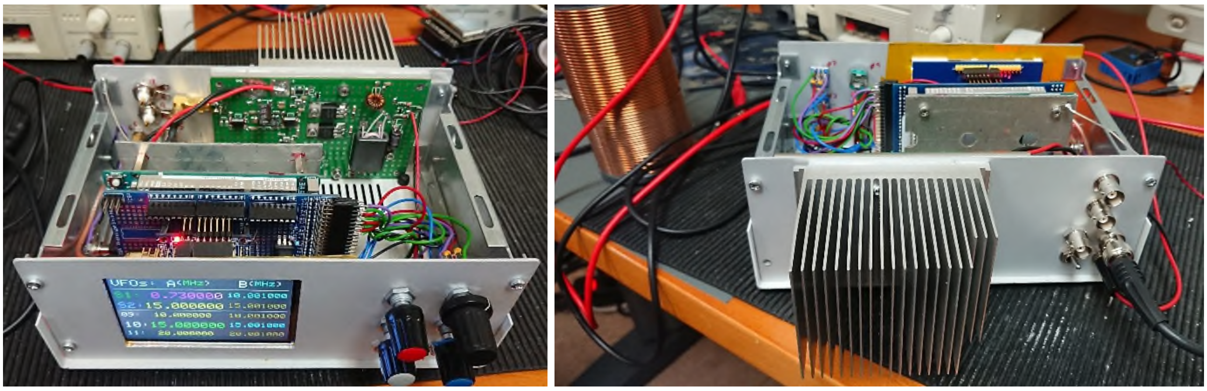


Figure 28 – Power Signal Generator

When powering it from batteries, a slight decrease in signal level was detected in the vicinity of power lines -reflecting the decrease in output power-, but the signal was still strong, indicating a good coupling to said lines. At this point, it was decided to move to field tests for obtaining more insights on the performance of coil antennas.

7.2.6 Computer Simulations

To obtain an additional validation of the theoretical models, and to provide additional validation of the experimental results, numerical simulations were carried out, using both the Lumped Model and the Transmission Line models. For the Lumped model, the equivalent circuit was simulated with the P-SPICE engine included in Tina Pro (*Figure 29*), while for the Transmission Line model, MATLAB was used.

In the lumped model case, the result of interest was the frequency of the first serial resonance. In the model output, this is identified by a step in the phase plots and a peak in the gain plot. Other resonances (parallel ones, at higher frequencies) have no interest for our purposes. In all cases the resonance frequencies were in good agreement with the experimental results. For example, in the case of the model of the *Figure 29*, representing a 0.168m diameter – 200 turn antenna, tuned with a 100pF serial capacitor, the simulation result is presented *Figure 30*. The serial resonance is located around 325kHz, while the measured value was around 330kHz. These values were also in good agreement with the values estimated (319kHz).

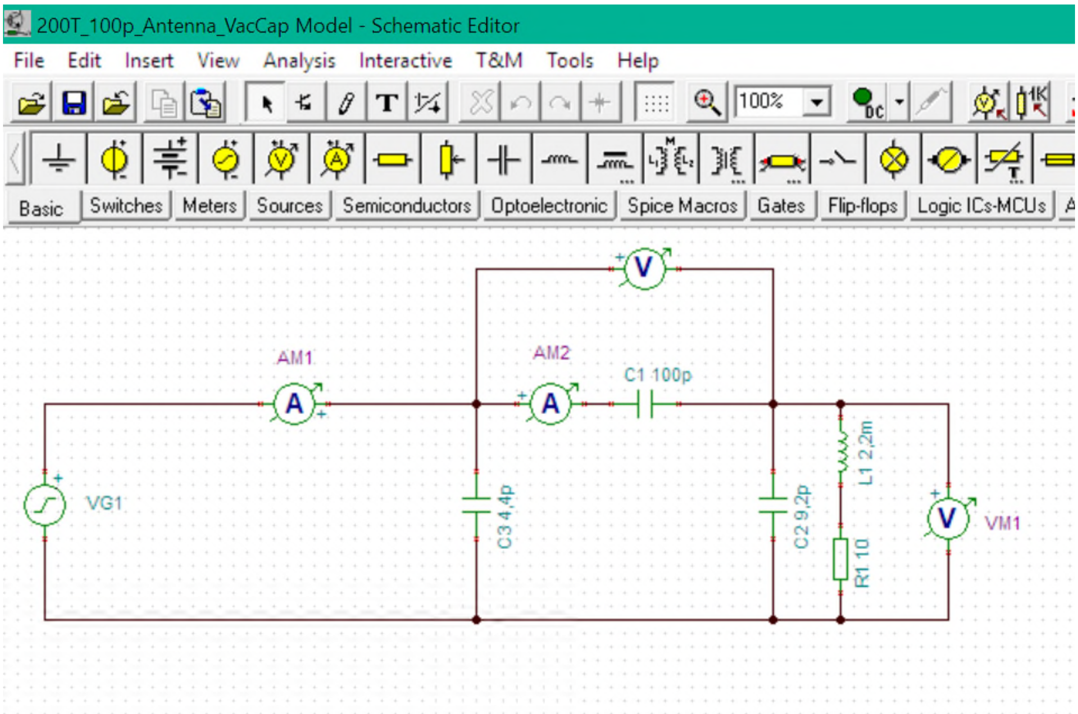


Figure 29 – Lumped Parameters Model of a 200-turn Coil Antenna (tuning with 100pF)

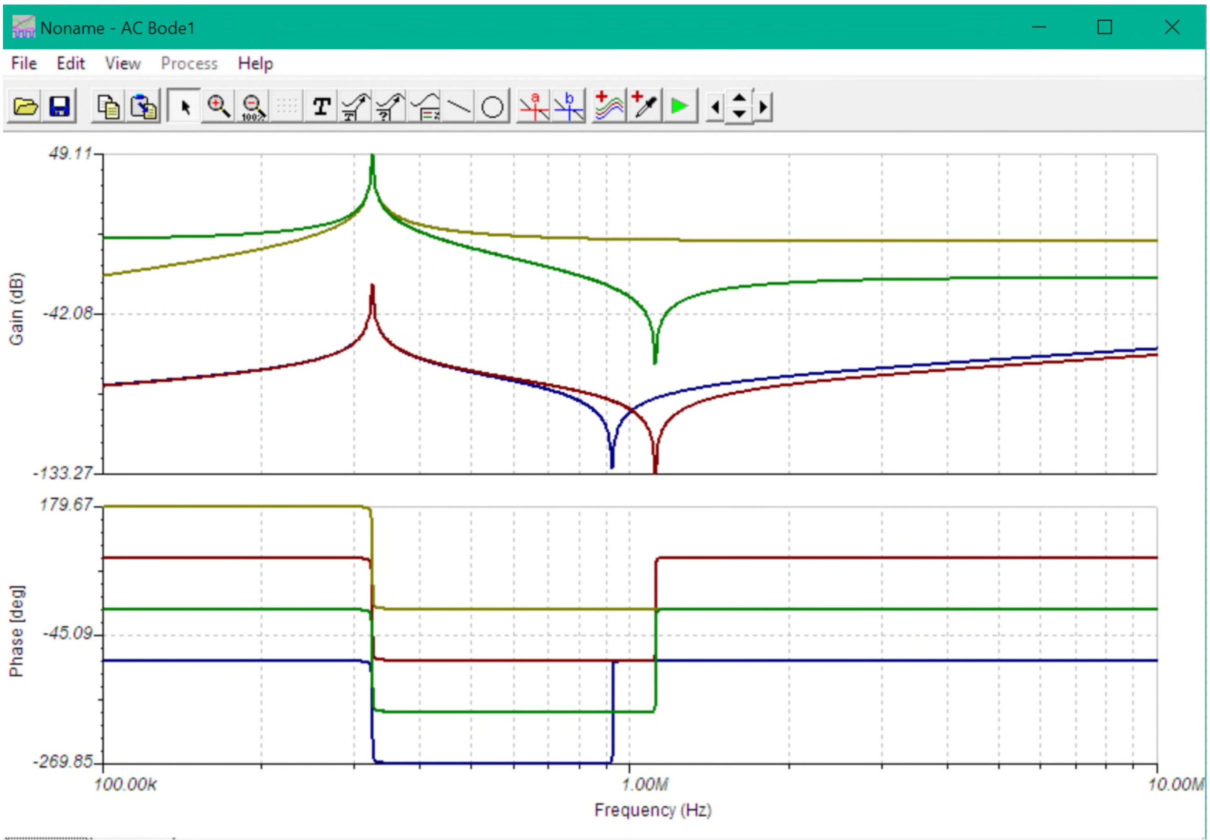


Figure 30 – Lumped Parameters Simulation of a 200-turn Coil Antenna (tuning with 100pF)

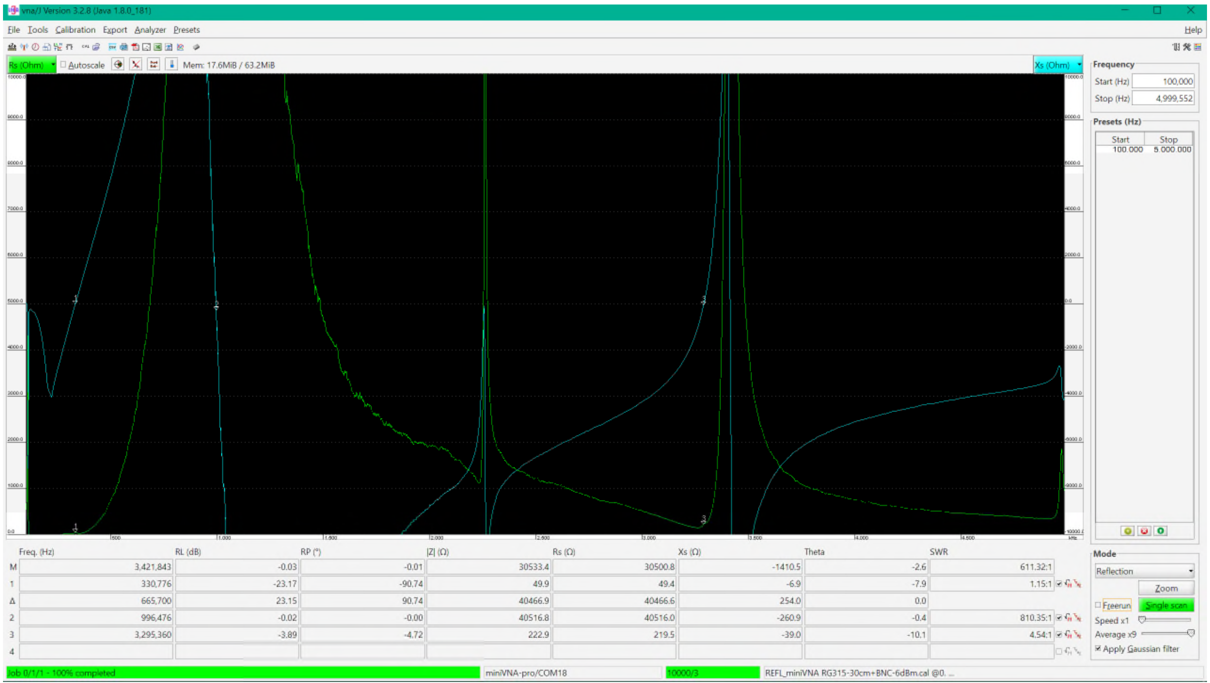


Figure 31 – Actual Measurements on a 200-turn Antenna (tuning with 100pF)

In the case of the Transmission Line model, the simulation results are also in good agreement with the experimental values. For example, in the Figure 32, it is shown a resistance and reactance plot for a 0,168m diameter - 80 turn antenna, tuned with a 100 pF capacitor. Measured values are presented in the Figure 33, which is almost identical. These figures show clearly the capacity of this model for predicting the behavior of such coil above the self-resonance frequency.

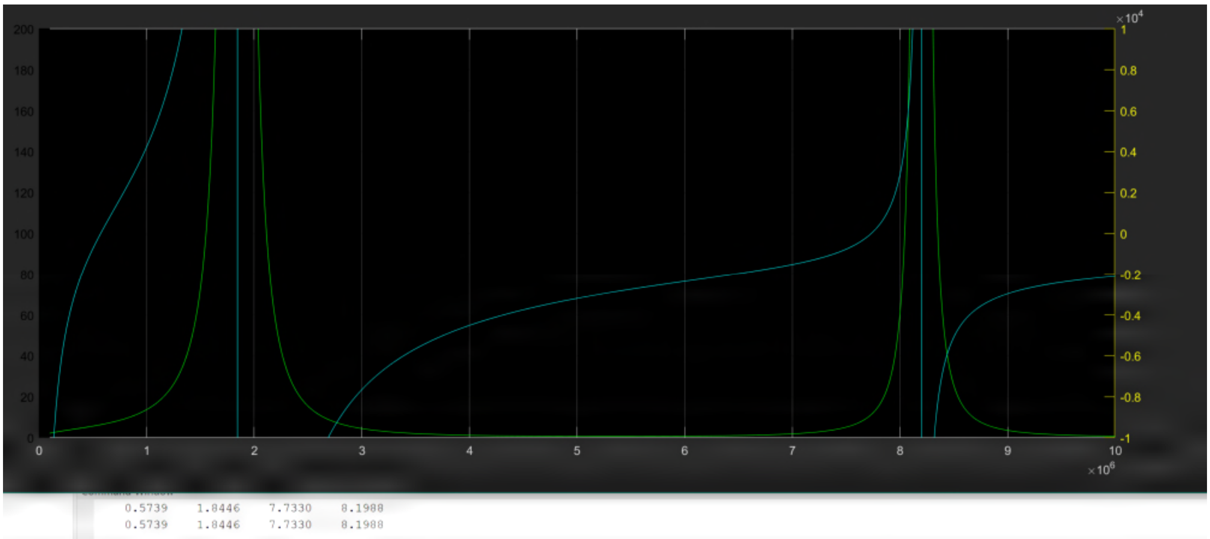


Figure 32 – Simulation of an 80-turn Coil Antenna using the Transmission Line Model

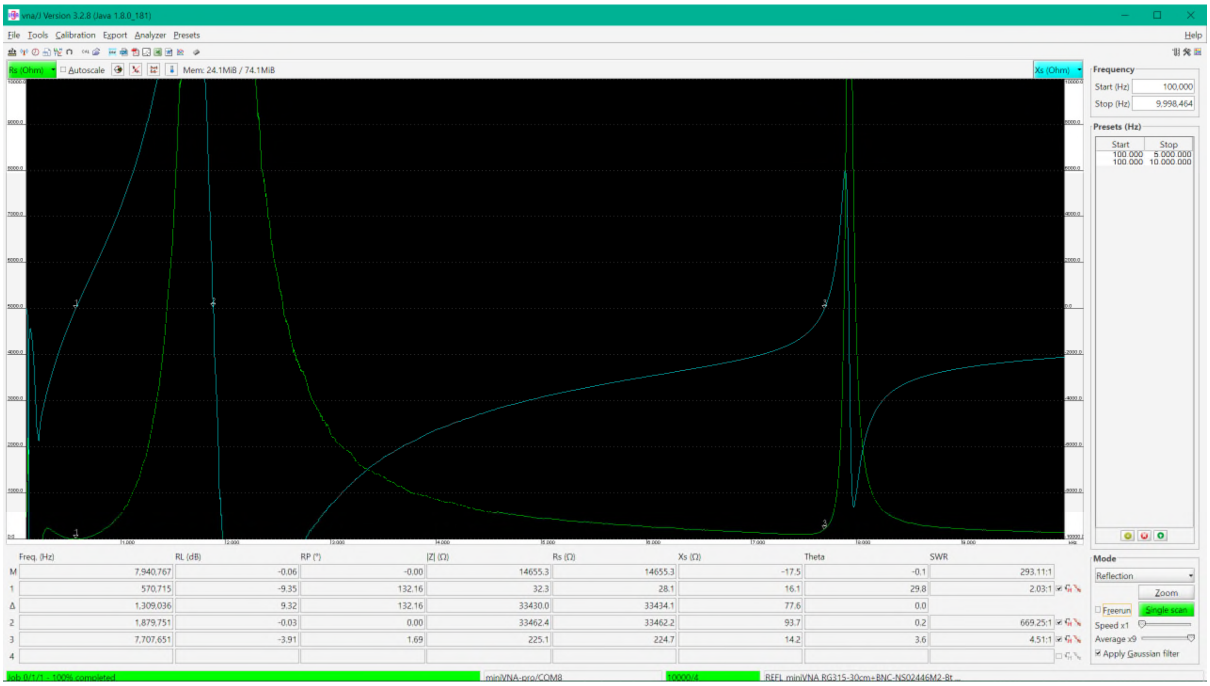


Figure 33 – Actual Measurements on an 80-turn Coil Antenna (tuning with 100pF)

7.3 Ferrite-cored Solenoid and Bobbin

Ferrite loopsticks are commonly used in broadcast receivers for reception of signals up to 1.5 MHz, and therefore are an option as LF/MF receiving antennas. But they are also used as transmitting antennas in LF (125kHz) Radio Frequency Identification (RFID), and therefore some attention as paid to the possibility of using them as serially-tuned transmitting antennas.

For this purpose, a couple of loopsticks were built. Litz wire was used, and the winding was tight, as is the usual practice in broadcast receivers. Serial tuning capacitors were calculated for obtaining resonance at 130kHz. However, characterization and impedance measurement were not possible, for they were out of the limits of the available Vector Network Analyzer.

Tests in the lab were not encouraging. An operating range of just one metre was obtained when driving them with 10V_{PP} (a realistic value for a transmitter intended for underground use), and therefore work in this field was discontinued.

7.4 Air-cored Loop Design Guidelines

7.4.1 Introduction – Design Targets

As result of the theoretical and experimental activities, some design rules can be established for designing coil antennas to be used in underground fixed and/ or portable applications the LF / MF / HF frequency ranges. It is assumed that the efficiency will not be high, being the design targets:

1. Achieving the highest possible efficiency for a given size
2. Achieving an apparent resistance, when tuned, in a range suitable for direct connection to Power Amplifiers (PA) of standard design, that is, somewhere between 25 and 75Ω, with DC resistance as low as possible.

Design input data are:

1. Carrier (central operation) frequency *f*
2. Peak power W_P.
3. Desired bandwidth

7.4.2 Design Procedure

For designing the coil antenna, follow the following steps:

1. Select a suitable diameter *D* for the coil. It shall be the maximum one compatible with the operational requirements.
2. Select coil wire diameter / gauge *d*. It shall be the maximum one compatible with the application. 1mm diameter should work well in most cases. 0,5 mm should be considered a minimum for a transmitting antenna.
3. Calculate wavelength $\lambda = 300.000 / f$, with λ in m and *f* in kHz.
4. Wire length l_w shall be between 0.18 and 0.2 λ for an apparent ohmic resistance at resonance in the range of 50Ω. Can be lower if a lower apparent ohmic resistance is desired.
5. Calculate the number of turns *N* as $N = l_w / (\pi D)$.
6. Select the coiling pitch *s*. As initial approach can be between 1,5 and 2 times the wire diameter *d*.
7. Total coil length will be $N * s$.
8. Build a prototype and measure its self-inductance *L*.
9. Estimate the self or parasitic capacitance as $C_P = 0{,}5 * D$ (C_P in pF, *D* in cm)
10. Make a first estimation of the self-resonant frequency *SRF*, as $SRF = 1 / (2\,\pi\sqrt{L\,C_P}\,)$. It shall (and will) be higher than the design frequency *f*.
11. Calculate the apparent *L'* inductance value at the design frequency as
$$L' = \frac{L}{\left(1 - \frac{f^2}{SRF^2}\right)}$$
12. Calculate the value of the serial tuning capacitor C_S as $Cs = 1 / (4\,\pi^2\,L'\,f^2)$.
13. Test the tuned antenna with a Vector Network Analyzer (VNA). Note the serially-tuned resonant frequency f_{SR} and the value of the ohmic component Rs of its apparent resistance (for tests with VNA, a 500V silver-mica or Ceramic NP0 capacitor will suffice).
14. If the value of f_{SR} differs from *f*, correct the value of C_S and repeat step 13.

15. If W_P was specified, the RMS voltage needed for transferring a power of W_P Watts to the antenna will be $V_{RMS} = \sqrt{Rs\ W_P}$. The peak to peak voltage will be $V_{PP} = 2\sqrt{2}\ V_{RMS}$. OR, if the output V_{PP} of the PA was specified, this value shall be taken directly.
16. At resonance, the peak to peak voltage V_{PPC} appearing in the capacitor will be $V_{PPC} = \frac{V_{PP}}{2\pi f R_S C_S} = \frac{2\pi f\ L'\ V_{PP}}{R_S} = Q\ V_{PP}$. The rated voltage of the tuning capacitor shall be at least 0.75 times V_{PPC}.

7.4.3 A Case Study

Let's consider the following design requirements to implement an example of the design procedure.

- Operating frequency f: 1.35 MHz = 1.350kHz
- Design apparent resistance at resonance: Around 50Ω.
- Power Amplifier supplying 20V_{PP}, with 1Ω output impedance.
- Maximum antenna diameter: 180mm.

The closest form diameter is 168mm, so this diameter is selected for the coil. 1mm diameter magnet wire will be used, with a coiling pitch equal to 2mm.

λ = 300,000 / 1350 = 222.22m. Let's start with a wire length of 0.18 λ = 40m.

For this wire length, the number of turns is 40m / (π * 0.18 m) ≈ 70.

Measured self-inductance L is 638μH, and a first estimation of self-capacitance C_P is 8.6pF.

Therefore *SRF* = 2149kHz

The apparent inductance at 1350kHz is L' = 638μH/(1-(1350/2149)^2) = 1054μH.

To resonate this coil at 1350kHz, it would be needed a capacitor of 1/(4 π^2 * 0,001054 H * 1350000^2 Hz^2) = 13 * 10^{-12} F = 13pF.

Testing with a Mini-VNA Pro leads to the graph shown as *Figure 34*.

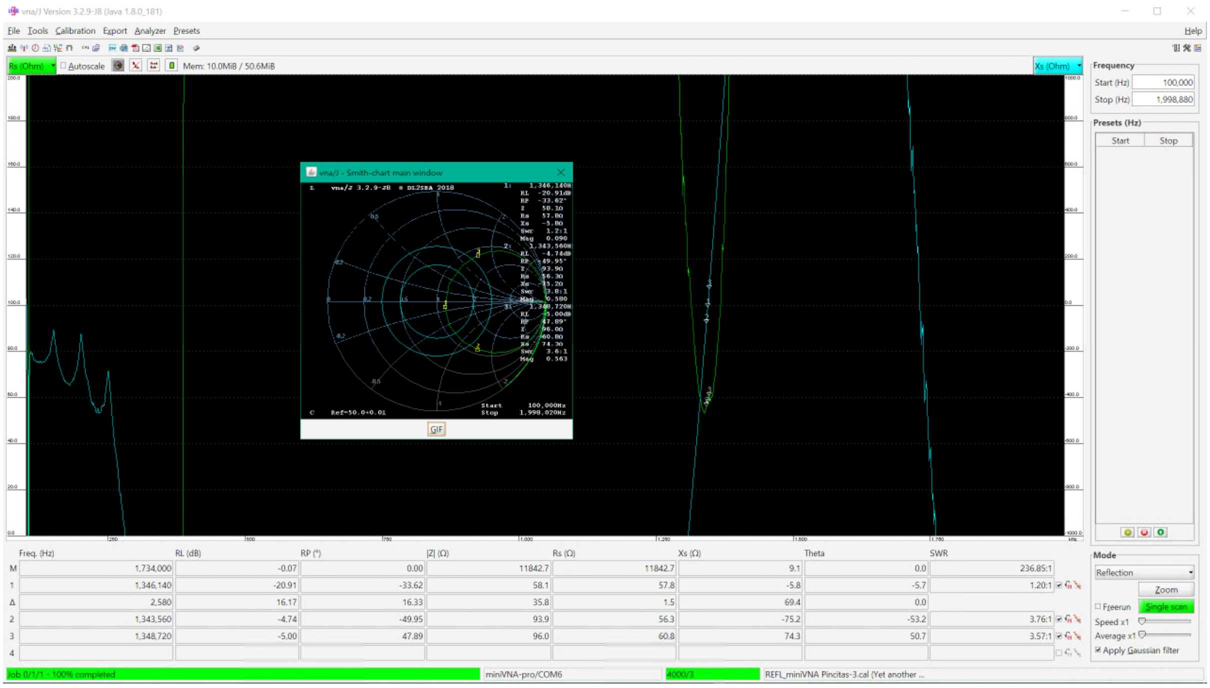


Figure 34 – Testing the Coil Antenna with a VNA

From it, the resonance frequency is (Marker #1) f =1346kHz, and the apparent Ohmic resistance is Rs = 57.8Ω. These values are considered acceptable.

The peak to peak voltage V_{PPC} appearing in the capacitor will be $V_{PPC} \approx 20\ V_{PP} / (2\ \pi * 1350000\ \text{Hz} * 13 * 10^{-12}\ \text{F} * 57.8\Omega) = 3138V_{PP}$. Or, the peak voltage in the capacitor will be half of this value, 1569V. Therefore, the rating for the serial tuning capacitor shall be at least 2000V_{DC} (2KV).

Also, from the VNA measurements, the bandwidth (Marker #2 – Marker #3) is around 5kHz, sufficient for most mesh sensor applications. However, it must be noted that in practice, observed bandwidth is an order of magnitude higher.

This page is intentionally left blank.

8 NETWORK AND ROUTING

Routing algorithms are only one aspect of packet data transmission, and so protocols for the latter must be considered prior to an investigation of routing. Protocols for packet data transmission on the Internet are well established, and generally operate using what is known as the *four-layer DARPA model*, or *Internet Protocol Suite*. However, our network is not attached to the Internet (except, perhaps, at some specified gateway) so we are free to re-specify its operation to suit our requirements.

As will be discussed below, the mesh network might have a lower dimensionality than originally expected, and this will be dependent on the particular installation (e.g. with respect to mine layout and geology). This has a bearing on whether the radio communications should be designed to operate in the near field or transition zone, which in turn governs the design of the node equipment, which then has a bearing on the data protocols. Additionally, within a development system, some network latency is acceptable so that packets can be transmitted in a human-readable format. Taking all this into account, it is clear that there can be no 'one size fits all' system.

The layered communications model is especially useful during development as it allows the work modules to be designed – and to a large extent, tested – in isolation from one another. This is particularly useful when it comes to evaluating the network routing algorithms because, not only do these depend on the particular application (as noted above), but to test them in a physical environment would require the construction (and installation in a suitable underground environment) of a large constellation of nodes, which is not practical.

The physical prototype of the mesh nodes demonstrates the transmission and acknowledgement of data between two nodes. However, without a complete constellation of nodes, the higher layers of the communications model – as described in this section of the report – cannot be adequately demonstrated in the physical environment. For this reason a simulator was built for evaluation of the Application and Routing Layer protocols, which is also described in this section of the documentation.

8.1 The Logical Network

8.1.1 Packet Addressing

It was originally envisaged that the network being developed would have much in common with Internet network protocols, in terms of packet transmission and the methods of packet assembly. However, as the project developed it became clear that not only was such an approach not necessary, but it could actually be disadvantageous. The reasoning was…

- In many installations it is likely that our network will not connect to a WAN at all. If it does, it is likely that this will be via a single gateway node. Therefore, our internal data transfer protocols do not have to follow any Internet conventions or specifications. This situation is not unlike that of local area networks, which do not use IP addresses for routing, but MAC addresses – in other words, LANs behave differently to WANs, and there is no need for our network to behave like either.
- For ease of debugging and development it would be easier if all data control messages were human-readable, in a similar way to Application Layer messages. For example, instead of a

packet header containing a block of bytes that describes the packet, the description can be conveyed in human-readable ascii code. This additional overhead would reduce the information transmission speed but, as our equipment is only a demonstration of a development system, this was not considered a disadvantage.

- A further important difference to the Internet is in how data packets should be addressed. In a WAN, packets are addressed using their IP addresses, but this is not true of a local network (LAN). Although a device on a LAN may use an IP address to identify a device (by broadcasting a message saying "which of you has this IP address?" once it know the identity of the device it uses its MAC address to route local packets. For the INDIRES system, we decided to use neither IP addresses nor MAC addresses, but 'node' addresses from, say, 0x000 to 0xFFF, corresponding to the physical INDIRES network nodes. It was considered that it might be useful to retain the concept of a port number, to route data packets to different applications.

8.1.2 Application Layer Protocol

This protocol is also discussed in *Section 8.1.3*, on the *Communications Layer Model*. However, the Application Layer has a special significant to this project, because the routing and transport operations are deferred to this layer and, for this reason, it is introduced here.

The Application Layer is the first layer in the public specification of our layer model. Further layers may exist on top of the Application Layer, but these are the responsibility of individual applications. It was determined that we should use a single stateless protocol, somewhat similar to HTTP. It was further decided that, in our system, packet re-assembly be done at the Application level. Clearly, this is entirely feasible, if rather unusual; it just means that the rigid concept of layers has been relaxed somewhat, because packet re-assembly is no longer invisible at the Application Layer. There is also the inference that these Application Layer packets are variable in length, although possibly the only example of long or multiple packets in a transmission would be when sending encoded speech.

Users of the Internet will be familiar with a number of standard protocols, including SMTP, POP3, FTP, and HTTP. The first three of these are **stateful** protocols that involve dialogue between client and server that is undertaken in human-readable English. By contrast, the HTTP protocol is **stateless** – the client issues human-readable data (a set of HTTP headers and possibly a body containing POST data), and the server then responds, usually by issuing a set of response headers and then serving an HTML page. The server then 'forgets' all that has happened. A significant question to be answered in this project was therefore whether the INDIRES Application Layer protocols should be stateful or stateless. From a development point of view, a stateless protocol is much simpler. It is easier to debug, although a criticism could be that it simply defers the problem. However, our applications are simple ones and can undoubtedly be achieved with a stateless protocol.

The main applications are a) the transmission of sensor data and text messages, and b) the transmission of low-bandwidth encoded speech data. For both of those it is straightforward to devise a stateless protocol to do the task. For development purposes, it is useful if this protocol is similar to HTTP, so that it can easily be simulated.

Speech Transmission

One of the uses of the INDIRES mesh network was intended to be digital speech transmission. However, even in a high speed network (such as Zigbee) there are problems due to latency and the lack of deterministic behaviour, which can make it difficult to achieve a high-enough enough quality. It was therefore decided that if a speech system is desirable, it should be in the form of a messaging service – that is, a user can record an audio message as an alternative to typing a text message, but the delivery of the message will be subject to network performance. Provided users understand the nature of this audio messaging, this is not considered to be a problem.

Text Transmission

At the Application level, the default situation will be that text messages are provided as plain text, and any source coding will be applied by the lower layers of the network model. However, we have also considered simple methods of text compression that could be applied at the Application level. But, as this demonstration system will have a high overhead, there is probably little to be gained by, for example, Huffman encoding of the text.

8.1.3 Communications Layer Model

A communications system can be represented by a model that partitions it into a number of so-called abstraction layers. One model – known as the OSI model – partitions a system into seven layers. The Internet is sometimes modelled in a simpler way, using fewer layers. During the development of the INDIRES project it became clear that the INDIRES mesh network could benefit from a variation on the Internet model. The proposed model, and a comparison with the standard model, are shown diagrammatically in the figures below.

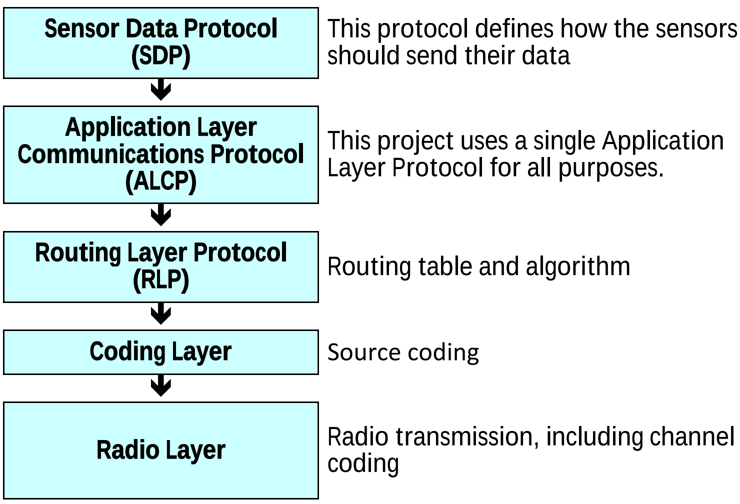


Figure 35 – The INDIRES Communications Model

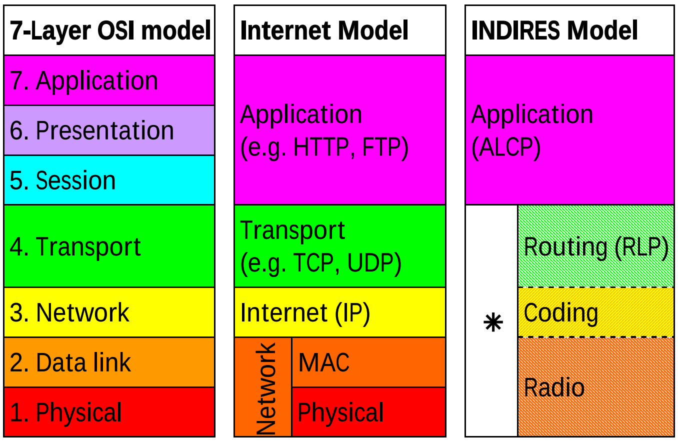


Figure 36 – A Comparison between the Internet and INDIRES Layer Models

⁎ These layers do not have a precise 1:1 correspondence with the OSI layers

Our layer model usefully separates the work modules. For example, a physical 'box' with digital data i/o could contain the Coding and Radio Layers, and a separate control module could contain the Application and Routing Layers. These two elements are essentially separate within the project, and do not necessarily need to be combined in order to test a prototype system.

The communications layers of the figures above are summarised in the table below, and then the individual layers are described in more detail.

Layer	Description
Application	Human-readable character strings are used to convey data at an Application level. We will use a single stateless protocol, deferring Application data processing to a higher (and undefined) level in the layer model. In other words: all applications communicate using the same protocol, and process their own data. Another difference to the standard model is that applications will be responsible for controlling packet size and for assembling their own data packets.
Routing	The packets provided by the Application Layer will have human-readable routing information added, in a similar fashion to how email messages accumulate header information; although a better comparison might be to the Usenet 'path' header. The Routing Layer also handles the routing table.
Coding	This layer accepts a digital data stream for transmission; and adds the required source coding (e.g. parity, CRC). Data packets are not broken up for transmission, because the packet size is controlled at the Applications level. (Although, clearly, the Apps need to be aware of the limitations of the Data Coding Layer). At the receiver, the process works in reverse and produces a digital data stream at the receiver.
Radio	This covers the modulation and channel coding of the data. Decisions here are based on a through-rock propagation analysis and the design of the antenna and associated transmission and receiver front-end circuitry.

Table 3 – A Description of the Layers in the INDIRES Communications Model

Application Layer

The Application Layer was outlined in the previous section. An example of a dialogue is shown in *Table 4*, below. The similarity with HTTP should be evident. The client headers in this example suggest that it is asking for a response page from the destination. This is, essentially, the application's acknowledgement that the message has been delivered

– although it could contain other data, since the structure of this dialogue is very like that of an HTTP POST dialogue. We could specify that if no response page is requested, then none is required, and the transmission then becomes more like a UDP operation than TCP. There are obviously overheads in sending human-readable data, but the benefits – in terms of debugging – are significant, and we are, after all, only building a demonstration model. A commercial system may well require a re-examination of many aspects of the network protocols described in this report.

```
POST 0FA:23/ack.txt                          // POST to node 0x0FA port 23. Expect responseack.txt
DATE Thu, 07 Feb 2019 15:57:24 +0000         // Human-readable date, for debugging purposes
FROM 005:23                                  // Human-readable origin, for debugging purposes
SUBJ voice                                   // Optional Subject, for debugging purposes
TIME 1549552874                              // Unix timestamp
SEQ 123                                      // Millisecond timestamp, used with TIME as packet ident.
PREV 1549552873:994                          // Previous timestamp, allows packet re-assembly
APP speech_LPC100                            // A “content type” indicator: use App speech_LPC100
ENC base64                                   // A “content type” indicator: use base64 encoding
LEN 354                                      // Length of the POST data
                                             // Blank line separating headers from body
TG9yZW0gaXBzdW0gZG9sb3Igc2l0IGFt             // Message body contains 354 char’s incl. EOL chars.
ZXQsIGNvbnNlY3RldHVyIGFkaXBpc2Np             // This decodes to 256 bytes.
bmcgZWxpdC4gRHVpcyBzZW1wZXIgdmlO
YWUgbWFnbmEgYXQgaGVuZHJlcmlOLiBQ
cmFlc2VudCBzaXQgYW1ldCBudWxsYSBs
aWd1bGEuIE1hZWNlbmFzIGxhb3JlZXQg
c2FnaXR0aXMgZmVsaXMsIGEgYXVjdG9y
IG1hdXJpcyBibGFuZGl0IHF1aXMuIFBy
YWVzZW50IHZpdmVycmEgYmxhbmRpdCB2
ZXN0aWJ1bHVtLiBWaXZhbXVzIGF0IG9k
aW8gdmlOYWUgcmIzdXMgYQ==
```

Table 4 –Example Dialogue in
Application Layer Communications Protocol (ALCP)

Routing Layer

The Application Layer packets have to be routed around the network, so there must be some type of routing table that is propagated around the system. A salient point of the packet routing is that we have decided that packets will be defined at the Applications level, and that all data will be routed with human-readable headers for ease of debugging. In other words, the concept of a Routing Layer has been weakened. An example of a dialogue is shown in *Table 5*, below

```
DEST 0FA                                     // This packet is destined for node 0FA
GOTO 0E3                                     // Routing table indicates send now, to node 0E3
PATH 006 007 0AB                             // Path taken so far (origin at 006; I am node 0AB)
PAYL516                                      // size of payload
                                             // blank line
<payload of Application Layer data>
```

Table 5 –Example of Routing Data Added by the Routing Layer Protocol (RLP)

Coding Layer

The data needs to be encoded for transmission. This could amount to simply adding parity bits at a byte level, or CRC checksums. One of the salient points of Internet networks is that the error rate is low, so error checking and correction does not need to take place at a low layer in the model. However, our network is likely to behave significantly differently, with a higher error rate, so it could be advantageous to apply error detection, correction, packet acknowledgement and possible re-transmission requests at a lower level in the layer model.

Radio Layer

The Radio Layer applies modulation, channel coding and all the provisions of a radio link. It is possible that developments beyond this project might usefully consider several modulation methods – for example, a carrier could include a low speed DPSK channel as well as a higher speed channel.

8.2 The Physical Network

8.2.1 Wide Area Mesh v. Semi-linear Chain

An idealized representation of the physical network was originally depicted as shown in *Figure 37* below. Although it is shown as two-dimensional it is, of course, three-dimensional in nature.

As the project developed, it became clear that it was somewhat unrealistic to expect the network to function as a 'wide area' mesh as depicted below. In practice, the nodes are likely to follow the mine roadways, which may result in less scope for re-routing around failed areas. A more realistic depiction of the network might be as shown in *Figure 38*, below, the salient points being...

- Each node normally communicates only with the one next to it.
- After an incident, a wide area of the mine might be affected but the nodes will still be able to operate, by transmitting through debris and roof-falls.
- If a particular node does fail, then its adjacent nodes will be able to bridge the gap, because they will have (must have) sufficient power reserve in place.

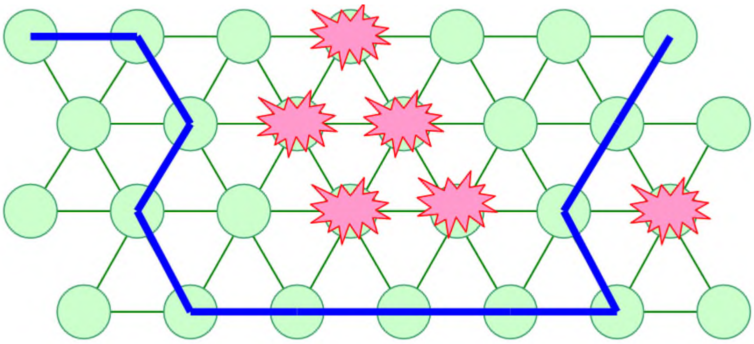


Figure 37 – Original Depiction of Idealised Mesh Network

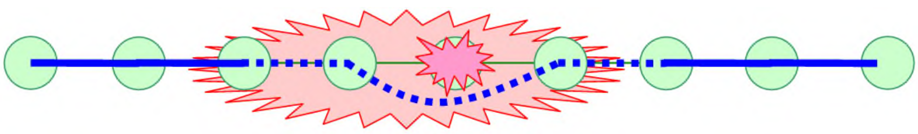


Figure 38 – A More Likely Representation of the Network

8.2.2 Bandwidth

A conventional mesh network operates by transmitting a packet of data from one node to another, sequentially. Thus, the overall data transfer speed (or bandwidth) is given by dividing the intrinsic bandwidth by the number of nodes. For our low-frequency system, this could represent a large latency. The question of bandwidth is one that relates closely to the number of nodes used, as the cost analysis below indicates.

A possible solution to the problem of latency would be to replace the traditional 'store-and-forward' operation of network nodes with a 'pass-thru' operation where a node will start to retransmit the data packet whilst it is still being received. This would be aided by the extreme localisation of a near-field system. A pass-thru operation raises questions on how error detection and correction are applied. It is not planned to investigate a pass-thru mode as part of the INDIRES project, but this would be a valuable subject to study in a future project.

8.2.3 Choosing the Number of Network Nodes

General Discussion

One of the key considerations is how dense our mesh network should be. We know that with only a few widely-spaced nodes, each transmitter would have to be very powerful and would be large and expensive. The entire purpose of the mesh network is to reduce the overall cost by using smaller, lower-power transmitters. The reason this works is that the power needed to transmit over a given distance rises at least as the square of the distance and, for a near-field system, it rises as the sixth power of distance.

However, when considering the overall cost, there is a law of diminishing returns, because the overall installation cost will rise with the number of nodes, even though the nodes themselves are smaller and much cheaper, overall. A mathematical analysis shows that there is an optimum way to split the cost between the installation components, power components and antenna components; and it dictates the ideal spacing of the nodes. It is not easy to allocate a precise cost to each parameter but one conclusion we were able to draw was that, for some systems, the large 'sensor nodes' could usefully be supplemented by a potentially large number of smaller 'repeater nodes'. The repeater nodes would be small, and have a range of perhaps only 20–50 m.

A mesh with a large spacing of 200 m or more would conceivably involve direct transmission through the rock, from one roadway to another. However, a mesh with a smaller spacing would only operate along the passages (and through any rock falls). The network is more linear than mesh-like and so, in the event of a node failure, it must be designed to be able to skip a node. This means that the nodes must be spaced with plenty of reserve. Detailed arguments to do with frequency and rock conductivity then lead to the conclusion that ***the nodes should operate at a low frequency well within the near field.*** This has an interesting side effect because, in the near field, the effect of the conductivity of the rock becomes vanishingly small. This makes the system's performance easier to predict, as well as making it easier to design and test; notably because it does not require testing underground. However, in practice, it is not possible to guarantee that the operation *is* well within the near field, except for closely-spaced nodes.

Quantitative Determination of the Optimum Number of Nodes

An analysis of how the system cost varies with the number of nodes was determined, which described how the cost was related to the mass of the equipment M (mostly battery and antenna), the power requirement P, and the cost of installing the equipment I. It was shown that if there were separate, quantifiable costs associated with M (e.g. volume of enclosure, as well as antenna cost, specifically) and P (e.g. battery size, component ratings) then the system could be characterised by the product MP and the minimum cost would occur when it was divided equally between the M and P elements of the system. In other words, there is no point in using a large battery with a small antenna or a small battery with a large antenna.

It was further demonstrated that, for a near-field system the total number of nodes should be arranged such that the infrastructure cost is twice the total (matched) cost of the M and P elements whilst, for a transition-zone system the infrastructure cost should be the same as the total (matched) cost of the M and P elements. In other words, there is no point in using a very large number of small nodes, or a very small number of large nodes.

Clearly, a significant point is whether the network is mesh-like and so operating in the transition zone; or linear, and operating in the near field. In other words, the design is dependent on the dimensionality of the network.

8.2.4 Alternative Antennas

Clearly, any constraint on the dimensionality of the network will have a bearing not only on the routing algorithms and the node spacing but also on decisions related to choice of antennas. This is only summarised here. It should be noted that these matters cannot be fully resolved within this project, which therefore became more of a ‘proof of concept’ than a finished system.

The possibility of using less powerful transmitters in nodes that are more closely spaced calls into question whether we can build a single ‘one size fits all’ system. Installation costs and the conductivity of the rock might dictate the design of substantially different systems for different applications. So, although, for this project, we have undertaken practical tests using the air-cored squat cylinders described above, we suggest that other antenna systems merit further study.

Antenna research was the subject early work in the project but, insofar as the antenna design impacts on node spacing and therefore cost, it will be briefly summarised here.

One attractive option would be to use an untuned loop antenna. Wideband untuned loops have seen success in sub-sea applications where video has been transmitted over short distances. The characteristics of an untuned loop mean that it can be designed to avoid skin, proximity and self-resonance effects in its winding so, in these respects, it is a more attractive option than the tuned antenna. Additionally, there are no design dilemmas concerning the number of turns to use, and we can use a single turn of wire or tape. With a tuned antenna, using a few turns of thick wire instead of a larger number of thinner turns can cause problems due to the low resistance of the winding, which then requires a very high current and can result in an inefficient power amplifier. But the impedance of an untuned antenna is largely governed by the inductance, not the resistance. It is true that there can be problems in driving an untuned antenna, but these are largely related to the use of an analogue power amplifier and do not apply to a digital system, although that is an added complexity to be dealt with. It is also interesting to note that if the limiting factor in the design of the antenna is the reactive voltage, then the performance when it is tuned (that is, the magnetic moment it produces for a specified power consumption) is identical to the performance when it is untuned. A node based on a small untuned loop might use under a watt of power and provide a range of 20–50 m in a box about the size of a large paperback book. However, the R&D costs of implementing an un-tuned antenna design should not be under-estimated.

Another possible antenna uses a small ceramic tile. This was considered and it was decided that, for pragmatic reasons, a more traditional antenna was better suited to this project. Such an antenna is expensive and requires specialised electronics but it does have some potential benefits – especially in a low-power repeater node. This is also suggested as something that could be pursued in a further project. A ceramic tile might not be suitable as a magnetic field receiver but there is no reason why the *same* type of antenna *has* to be used for the transmitter *and* the receiver; and a ferrite rod is an example of this ‘dual antenna’ concept, with it being more suited for use in a receiver than a transmitter.

Lastly, grounded dipoles are known to be highly efficient, but they may run into problems with ATEX certification. As noted above, there is no ‘one size fits all' solution and a particular application – whether for reasons of cost or geology – might require a substantially different antenna system.

Summary

- **Main finding.** It seems clear that one result of the project will be that there is not a ‘one size fits all’ solution in terms of antenna design
- **Low node spacing.** The ideal node spacing cannot be stated without a knowledge of the costs involved, which will likely vary between applications. However, we do know that the semi-linear nature of the network means that the nodes should operate well within the near field, so we will not require large, powerful transmitters.
- **Use of repeater nodes.** The likely close spacing of the nodes suggests that we could utilise basic *repeater* nodes as well as the more sophisticated *sensor* nodes.
- **More sensitive receivers.** Near field operation implies less signal dispersion and so a greater noise tolerance, which may result in a more sensitive receiver.
- **Non-radio design platform.** Near field operation implies that power radiation is not relevant, and that power matching is also irrelevant.
- **Further research.** There are opportunities for further research into untuned antennas, ceramic tiles and grounded dipoles.

8.3 Development of Routing Algorithms

8.3.1 Summary of Layer Architecture

The layer model, outlined above, shows a Routing Layer as existing below the Application Layer. In a commercial product, optimised for speed, this would probably be true. However, as also noted above, we decided that, for this project, packets will be defined at the Applications level, and that all data will be routed with human-readable headers, so the Routing Layer is less significant in our prototype design.

The data payload at the Applications level will therefore accrue routing information in a similar way to how Usenet accrues its Path: header or email accrues its Received: headers. The reason for this approach is that, in a development system, especially one where the routing is the subject of research, it is an advantage to have human-readable headers.

8.3.2 Routing Algorithms

From the discussion of the physical network, it is clear that the routing algorithm will depend on the precise nature of the network (i.e. to what degree it is linear or mesh-like), a fact which can only be determined for a specific installation (e.g. with respect to mine layout and geology). This means that a prototype system will undoubtedly need to work with a relatively unsophisticated algorithm if it is not too specialised to be of general use. However, for a successful practical demonstration we only need to demonstrate end-to-end communication through the system. That is, we can broadcast a data packet to all nodes and then demonstrate that the packet of data arrives at all the nodes within listening distance.

In the INDIRES system, there are three distinct stages to the operation of routing, referred to as mapping, routing and linking.

Mapping

Mapping is the process of determining the structure of the mesh network and distributing the information to the individual nodes. For the INDIRES project, we consider this to be a separate operation to the determination of a route, but this distinction is not usually evident in network design.

A typical example of ‘traditional’ routing within a mesh network is the protocol used with Zigbee network devices, which is known as *Ad-hoc On-demand Distance Vector* (AODV) Routing. In the AODV protocol, a node broadcasts a routing ‘ping’ to all devices on the network. Those that receive it repeat the ping, and the process continues until the destination is reached. The destination then responds by transmitting a message back to the source via the lowest cost route. The source then updates its routing table accordingly.

In our case, the above operation is slightly modified. Initially, a node broadcasts a routing ‘ping’. The nodes that receive it do not rebroadcast it immediately, but wait until an allotted time frame to do so. For example if node 37 receives a ping from node 9, it will wait for 28 frames before re-transmitting it, with the additional path information. This avoids the need to process data clashes and, because this is an initial mapping exercise, speed is not important. Before re-broadcasting the ping, the node examines the path information to ensure that it is not repeating a transmission. In this matter – and with some additional network control information – a complete map of the network, containing ‘reliability’ audit data for communication between each pair of nodes, will arrive back at the originating node. Note that a node pair need not be symmetrical – it could be that transmission from node 23 to node 17 has a different reliability to communication from node 17 to 23.

The information that is gathered by this exercise comprises the *network map*.

Routing

Once the originating node has built a map of the network it can apply a routing algorithm to the data, in order to construct a routing table. That provides the optimum, or lowest cost route between any two nodes. In practice, end-to-end communication is likely to be between a designated gateway note and any one of the other satellite nodes. The routing table is simply a matrix showing, for each possible pair of nodes in an end-to-end link, which is the initial node to be contacted. Once the routing table has been determined, it is broadcast to all nodes.

A traditional routing algorithm (which predates computer networks) known as Dijkstra's algorithm is suitable for this task. This and related algorithms are significantly complex but an emergency communications system that spends most of its time dormant certainly has time to execute some quite complicated and lengthy mapping and routing algorithms.

As indicated in *Figure 38*, above, our network is likely to be 'sparse', which simplifies both the mapping and the routing. Once the routing has been established it would be possible to revise the node numbering in order to optimise the time frames used during the mapping. However this is seen as beyond the scope of this project but it could be an interesting task for future investigation.

Unlike some mesh networks, where node dropouts are rare, our system is expecting nodes to cease to function. One function of the routing table would be to identify any points of failure that would prevent the network from functioning, and to prepare additional routes to be used if a primary route were to fail. It is considered that this analysis goes beyond the remit of the project but would certainly be an important point to investigate for any commercial system.

An added complication in the INDIRES system is the need to save power during an incident, when the nodes might need to rely on their internal power supply. Although it was originally considered that the nodes might have a standby mode and need to synchronise their wake-up, it is now considered that a better scheme might be for the receivers to operate continuously, and to trigger a wakeup that is essentially similar to the 'wake on LAN' mode of an Internet-connected computer.

An important aspect of routing is the possibility of allowing 'pass-thru- operation, as discussed in *Section 8.2.2* above.

Linking

Linking is the name we have given, within this project, to the notion of sending pre-routed data across the network. Instead of each node determining the subsequent node by looking at its own copy of the routing table, it is possible for the originating node to specify the complete route in advance, and to transmit this information along with the data payload. This is primarily a convenience during development as it allows the three aspects of data transmission – mapping, routing and linking – to be developed separately.

8.4 Proof-of-Concept Demonstration and Simulation

A simulator to investigate the Applications and Routing Layers of the network was developed; the premise being that we could simulate the network by using a number of

Raspberry Pi boxes and a simple network switch. The salient point is that requesting a web page is the same operation as 'sending some data and obtaining an acknowledgement', which is the reasoning behind the design decision that our Application Layer should operate in a similar fashion to HTTP POST. The equipment is shown in the photo and block diagram below.

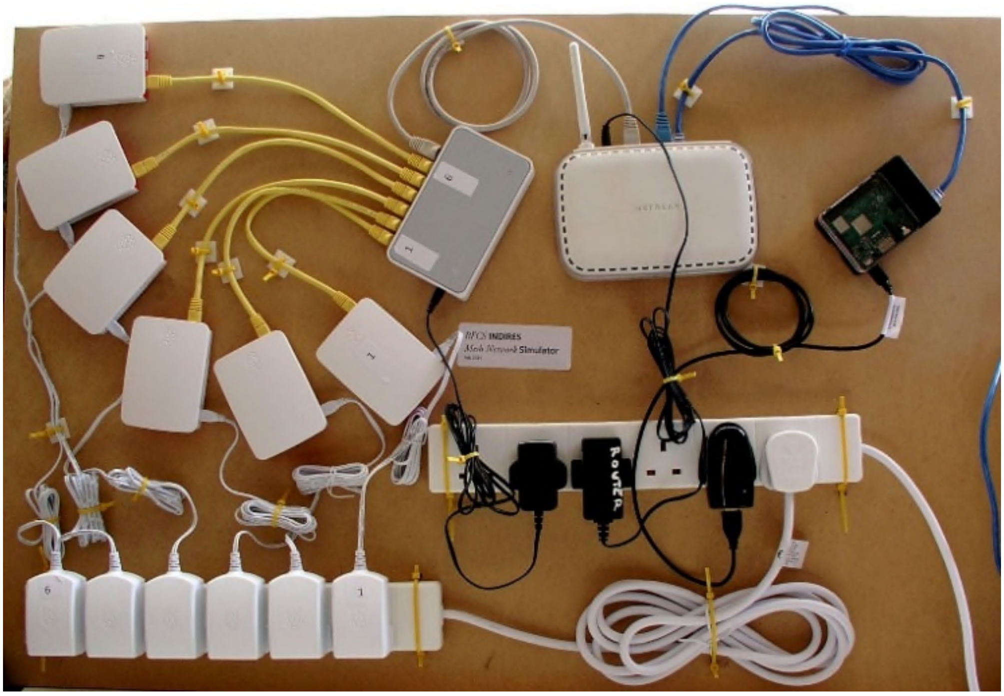


Figure 39 – Mesh Simulator using Raspberry Pi Computers

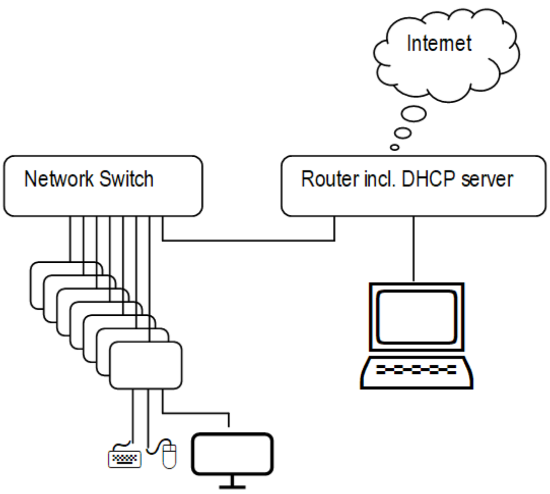


Figure 40 – A Depiction of the RPi-based Mesh Simulator

9 IMPLEMENTATION

9.1 Printed Circuit Board Design

The Printed Circuit Board (PCB) implementing the circuits described in *Sections* 3 *to* 6 has overall dimensions slightly smaller than the limits set out in *Section 2.1*. It is 230mm long and 80mm wide (the limits were 240mm × 90mm. A rendering of the PCB is shown in *Figure 41*, and a drawing with dimensions is included as *Appendix C* (*Section 14*).

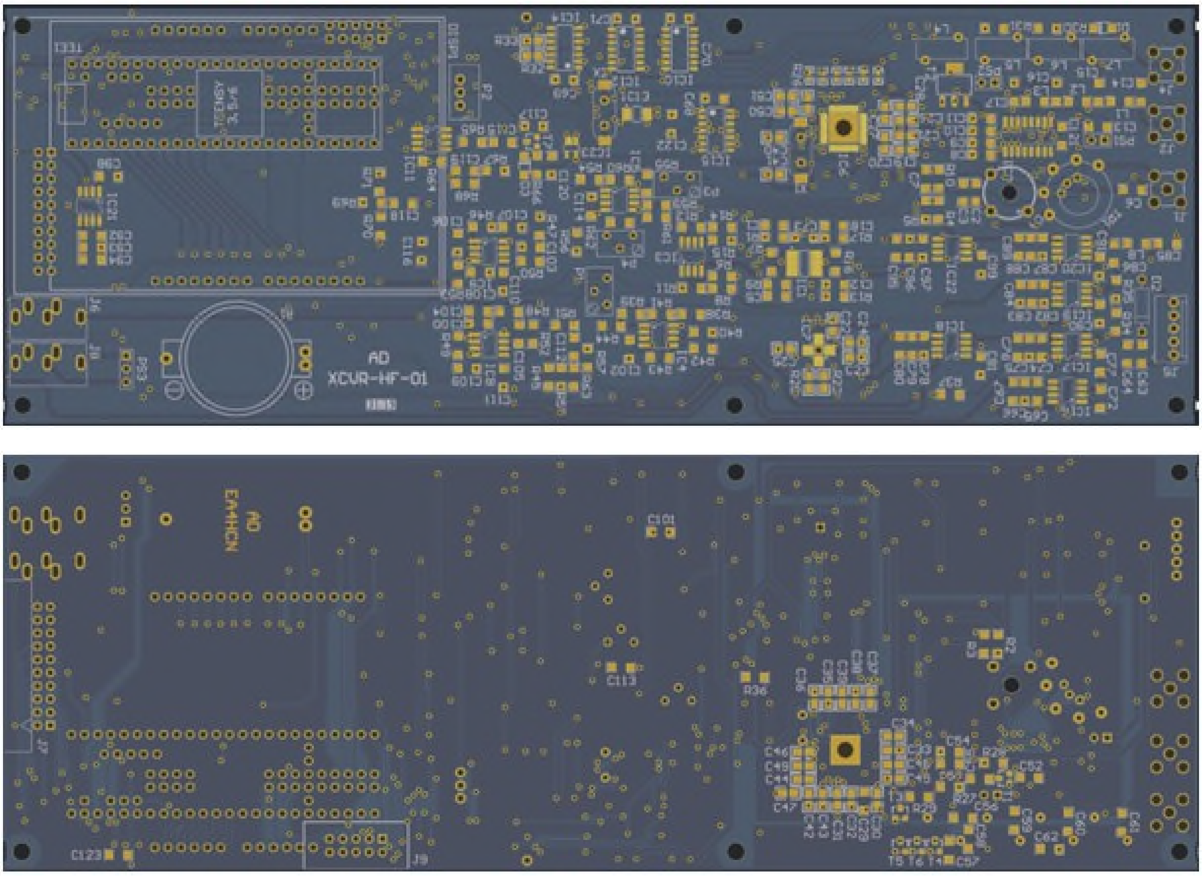


Figure 41 – XCVR-HF-01A PCB Rendering. Top: Upper Layer, Bottom: Bottom Layer

Although the design is pretty complex, a two-layer circuit was sufficient. The only significant detail is that analogue and RF grounds were drawn independently, and united at a single point to minimize ground loops.

9.2 Prototype Implementation and Testing

Figure 42 shows the prototype of the XCVR-HF-01A wireless mesh node with a full complement of accessories connected, in the configuration planned for use in the implementation of the fixed resilient sensor. This includes a colour display, a keyboard, a rotary encoder, antenna cables and gas (atmospheric composition) sensors.

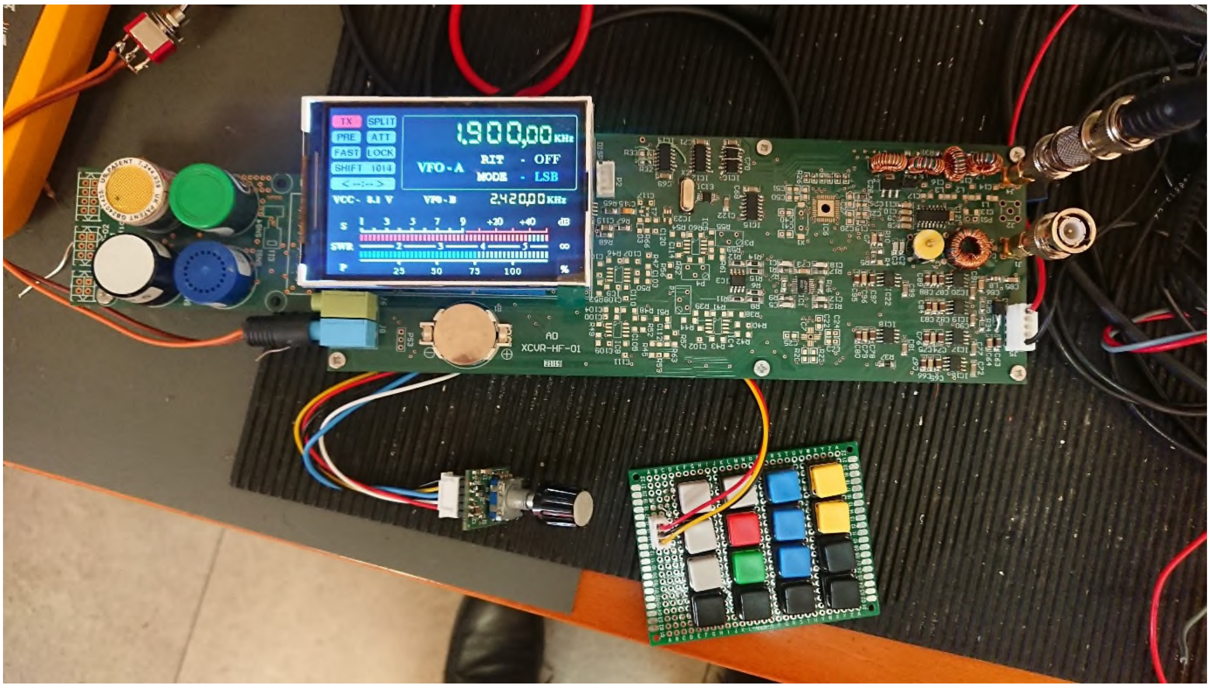


Figure 42 – XCVR-HF-01APrototype with LCD Screen and Sensor Interface Board

Extensive functional testing was performed while building the prototype. Of the tests performed, the most interesting are loopback tests, in which the RF signals generated by the transmitter are fed to the receiver. In the *Figure 43*, oscilloscope traces of the signals are shown.

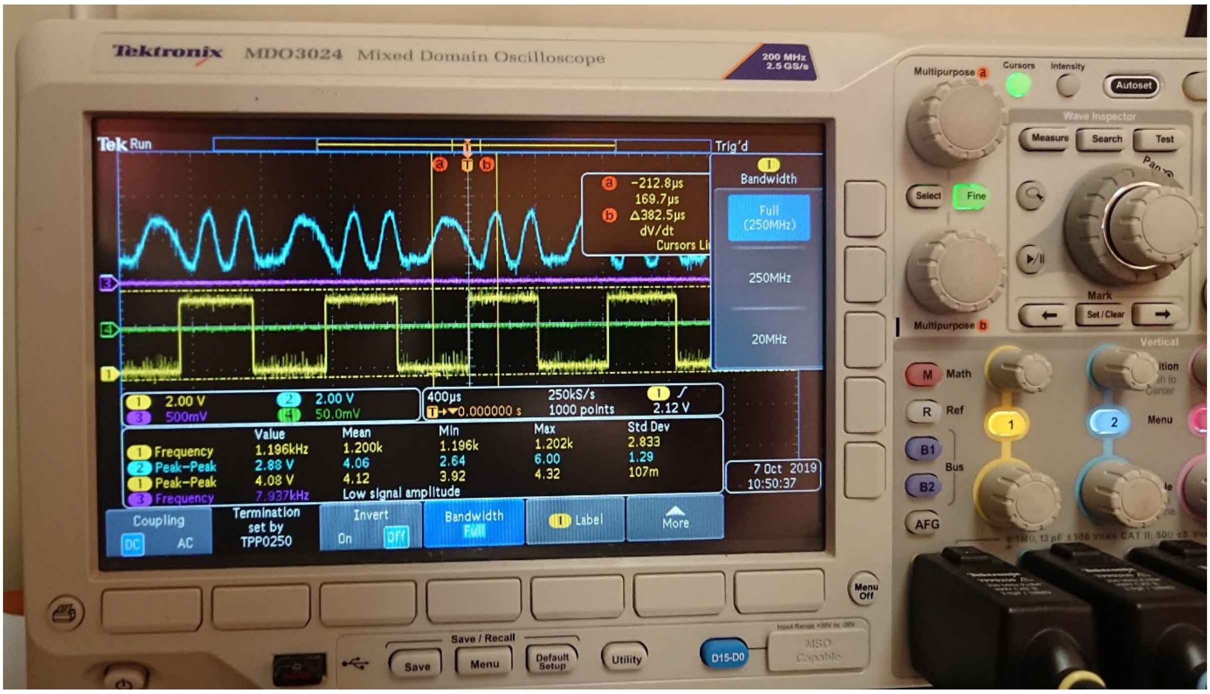


Figure 43 – XCVR-HF-01APrototype: Loopback Testing Waveforms (Unsaturated)

The yellow trace is the data signal, fed at 2400 baud (1,200Hz). The blue trace represents the analogue detected signal, which, as expected, has a 2,400Hz frequency when transmitting a “0” and 4,800Hz when transmitting a “1”. The synchronous transition (at 0V) between frequencies can be seen in the received waveform.

In *Figure 43*, the received signal is unsaturated, approximately sinusoidal. In *Figure 44*, it can be seen what happens when the receiver saturates and clips the signal. It must be noted that, in spite of clipping, the receiver signal is clean and easily processable by the microcomputer core.

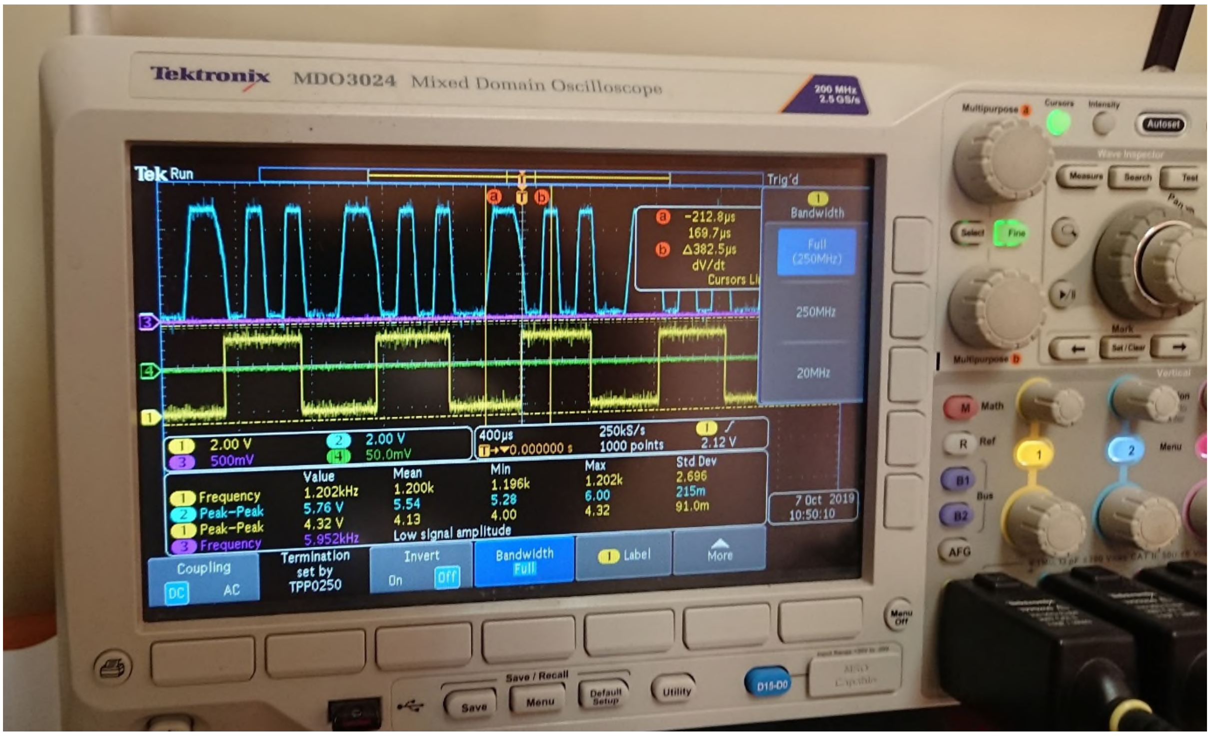


Figure 44 – XCVR-HF-01APrototype: Loopback Testing Waveforms (Saturated)

This page is intentionally left blank.

10 POSSIBLE IMPROVEMENTS & ATEX CERTIFICATION PATH

10.1 Possible Improvements

The modulation scheme presented in *Section 5* is non optimal in terms of bandwidth use. Instead of switching carriers only at 0°, they could be switched at both 0° and 180°, thereby allowing a doubling of the bit rate. Implementing this method would require some additional logic, including the selection between 0° and 180° (inverted) carriers.

Another possible modification would be to change the circuit in the clock generator to allowing working in the higher part of HF band. In its current version, the maximum carrier frequency is around 20MHz, while there is no lower limit in practical terms. With some simple modifications, the upper range can be extended to 30 – 40MHz. The drawbacks are reduced lower range (minimum frequency would be around 4MHz) and that it would not be possible to implement the synchronous digital modulation scheme presented in *Section 5.2*. In practical terms, this implies that the so modified device could be used efficiently only for voice transmission.

A further possibility is adding some kind of Automatic Gain Control, to keep the level of the recovered audio fairly constant. This has importance only for analogue audio reception using amplitude or single sideband modulation,

A last issue is power control. It seems that a better RF power control scheme, which would need minimal modifications, could use T2 as a PWM controlled voltage regulator. As output power is directly related with voltage applied to the PA, this method would allow a finer and wider power regulation.

10.2 Atex Certification Path

As the prototype was intended as a proof of concept, the limiting components needed to implement a fully compliant intrinsically safe (Ex ia, M1) design were not implemented in it.

However, the path to implement intrinsic safety is quite straightforward. No special issues are expected. The power requirements of the current version of the design are not too high (less than 250mA at 12V), and therefore can be supplied from an intrinsically safe power supply or battery.

Moreover, the power supply design must be modified, adding fuses and limiting resistors for limiting currents and temperatures. Zener diodes will be needed to guarantee a limited voltage in the output of the regulators, in order to allow the amount of capacitance (hundreds of μF) needed for the proper operation of the electronic circuits.

These modifications could be made after the end of the project, as part of exploitation strategy.

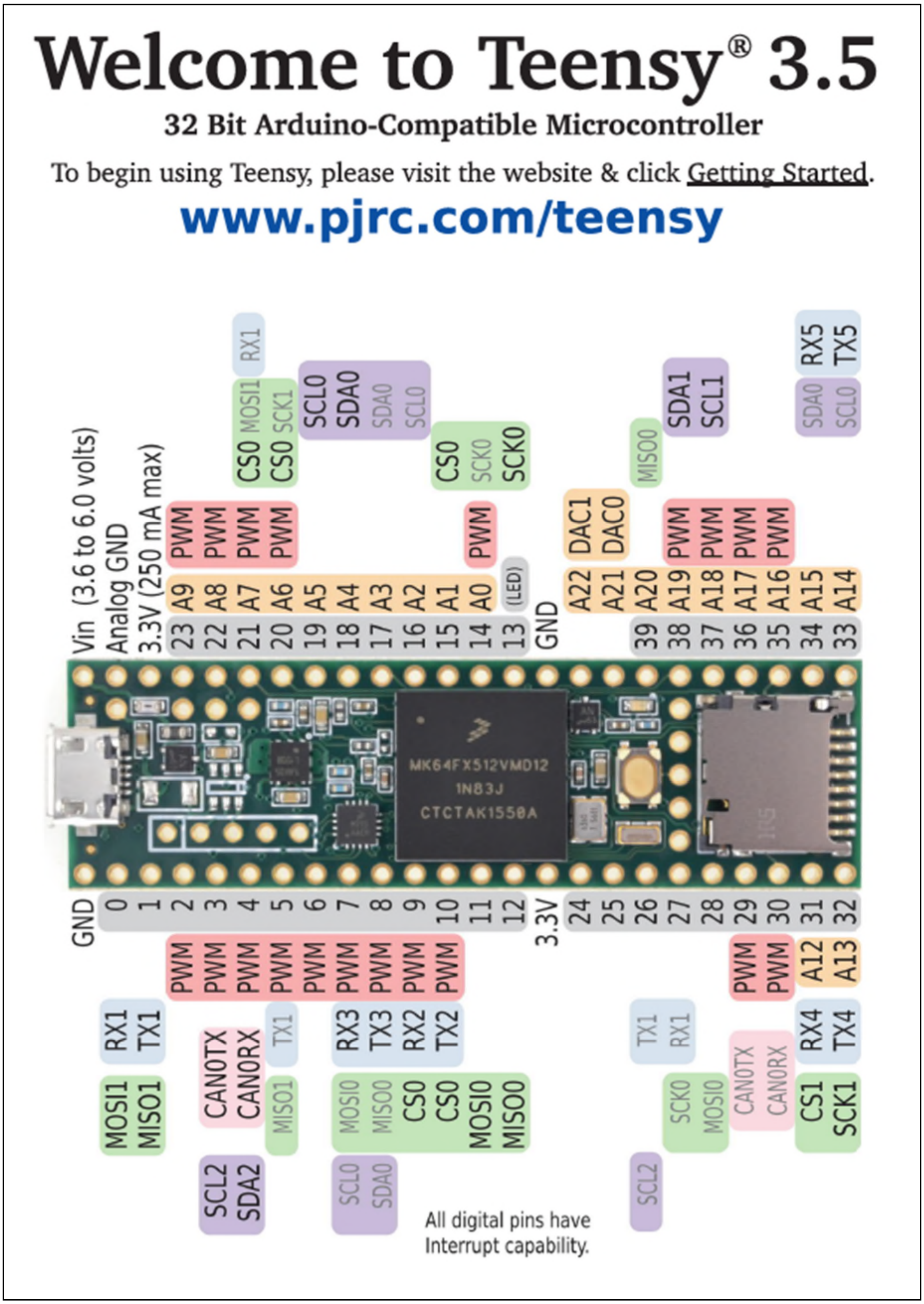
11 FURTHER INFORMATION

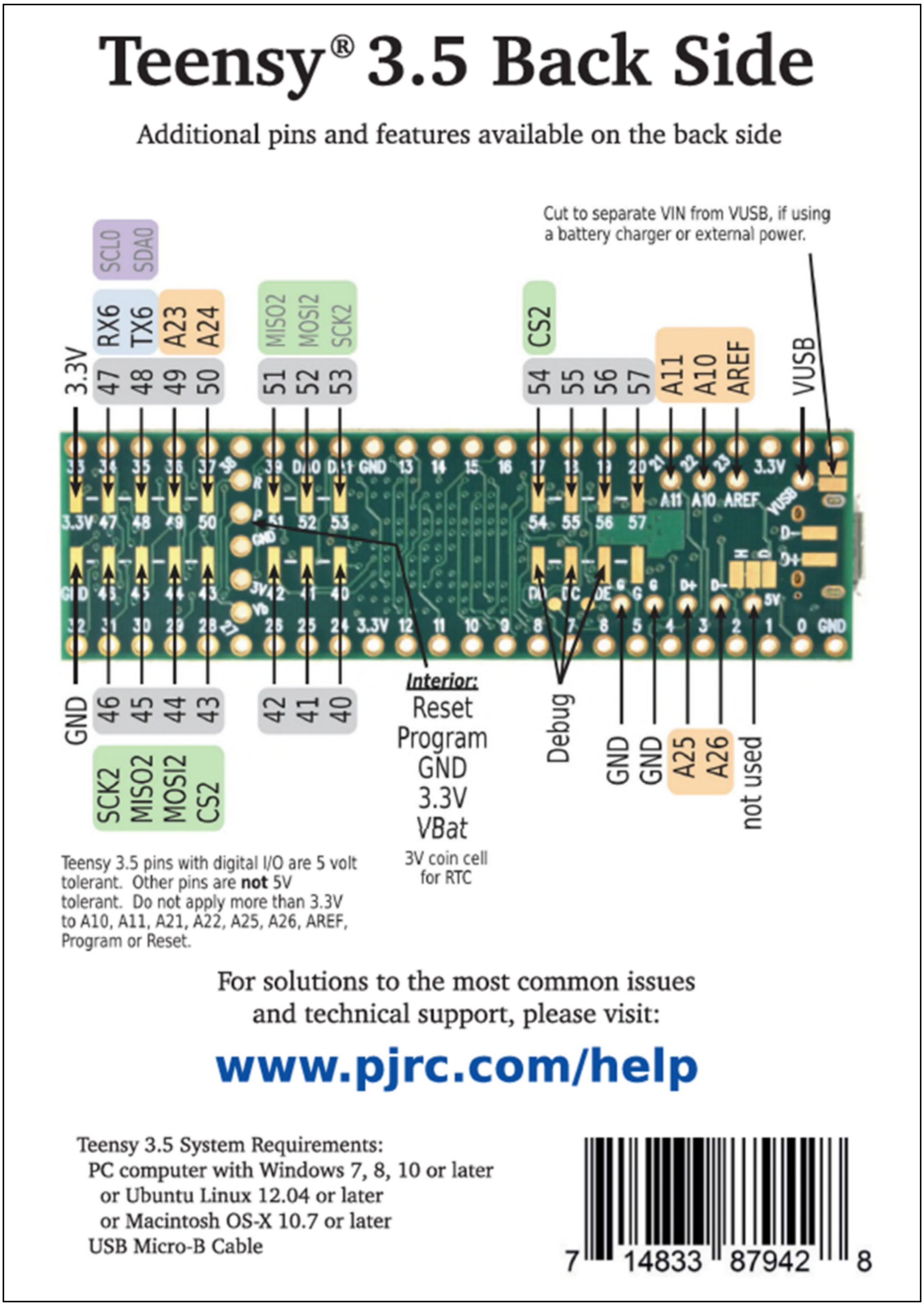
Further information about the TTE mesh node design is available from the Robotics Lab Research Group at Universidad Carlos III de Madrid, Avda. de la Universidad 30, Leganes 28911, Madrid, Spain, http://roboticslab.uc3m.es.

Please contact Alberto Jardón at ajardon@ing.uc3m.es.

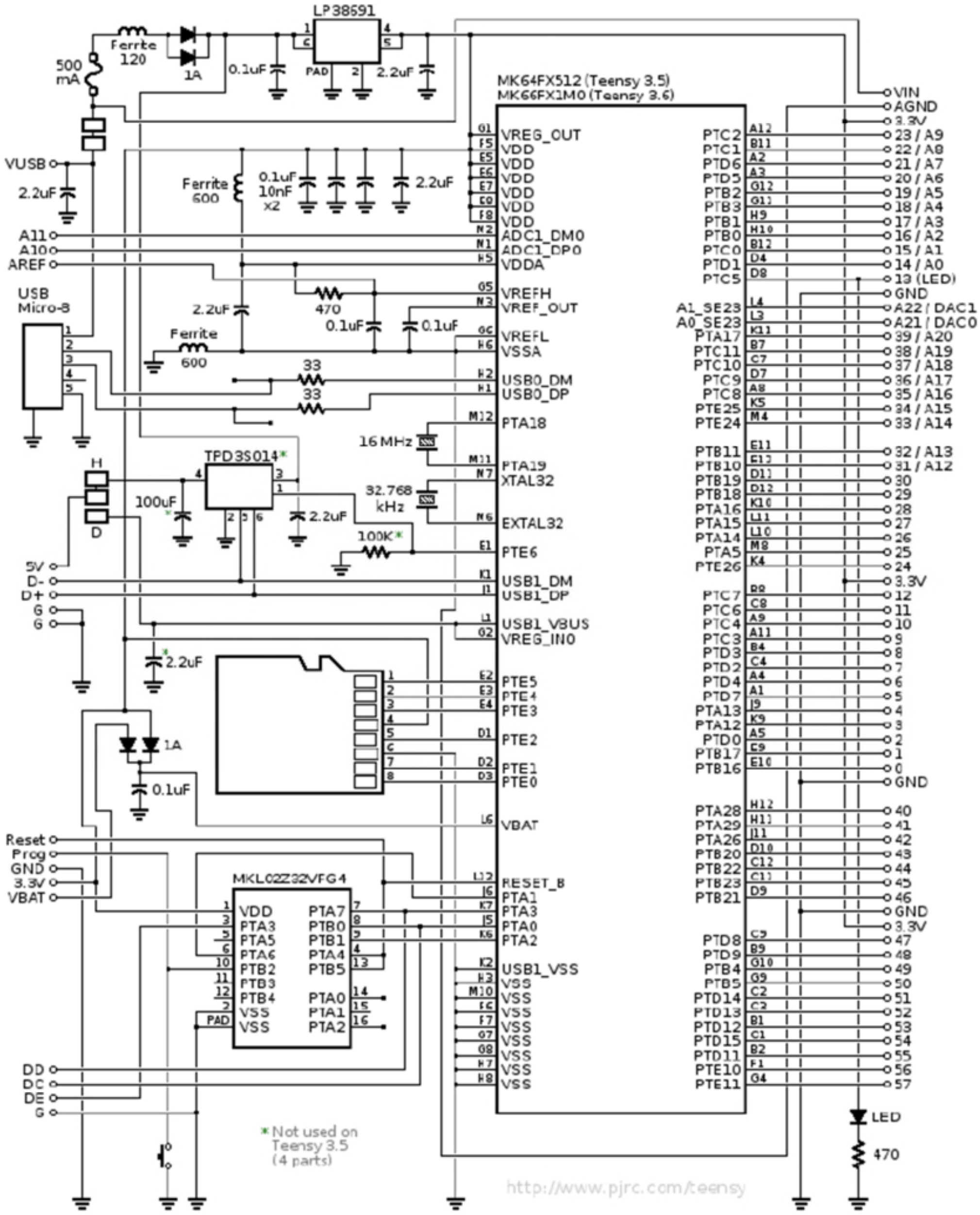
This page is intentionally left blank.

12 APPENDIX A – TEENSY 3.5 MANUFACTUER'S DATA



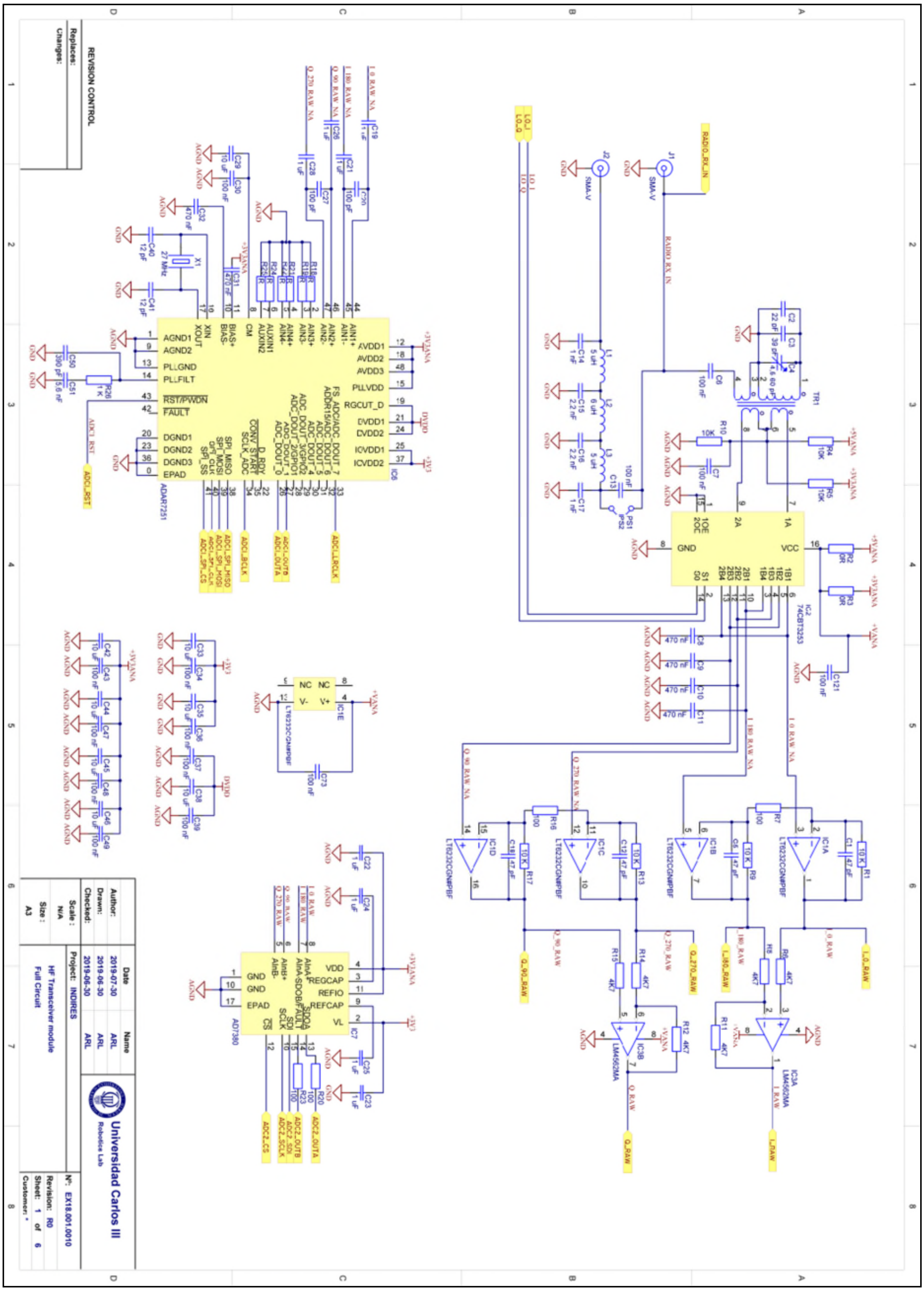

Teensy® 3.5 Back Side
Additional pins and features available on the back side
Cut to separate VIN from VUSB, if using a battery charger or external power.
SCL0 SDA0
RX6 TX6 A23 A24
MISO2 MOSI2 SCK2
CS2
3.3V 47 48 49 50 51 52 53 54 55 56 57 A11 A10 AREF VUSB
GND 46 45 44 43 42 41 40
SCK2 MISO2 MOSI2 CS2
Interior:
Reset
Program
GND
3.3V
VBat
3V coin cell for RTC
Debug
GND GND A25 A26 not used
Teensy 3.5 pins with digital I/O are 5 volt tolerant. Other pins are not 5V tolerant. Do not apply more than 3.3V to A10, A11, A21, A22, A25, A26, AREF, Program or Reset.
For solutions to the most common issues and technical support, please visit:
www.pjrc.com/help
Teensy 3.5 System Requirements:
PC computer with Windows 7, 8, 10 or later
or Ubuntu Linux 12.04 or later
or Macintosh OS-X 10.7 or later
USB Micro-B Cable
7 14833 87942 8

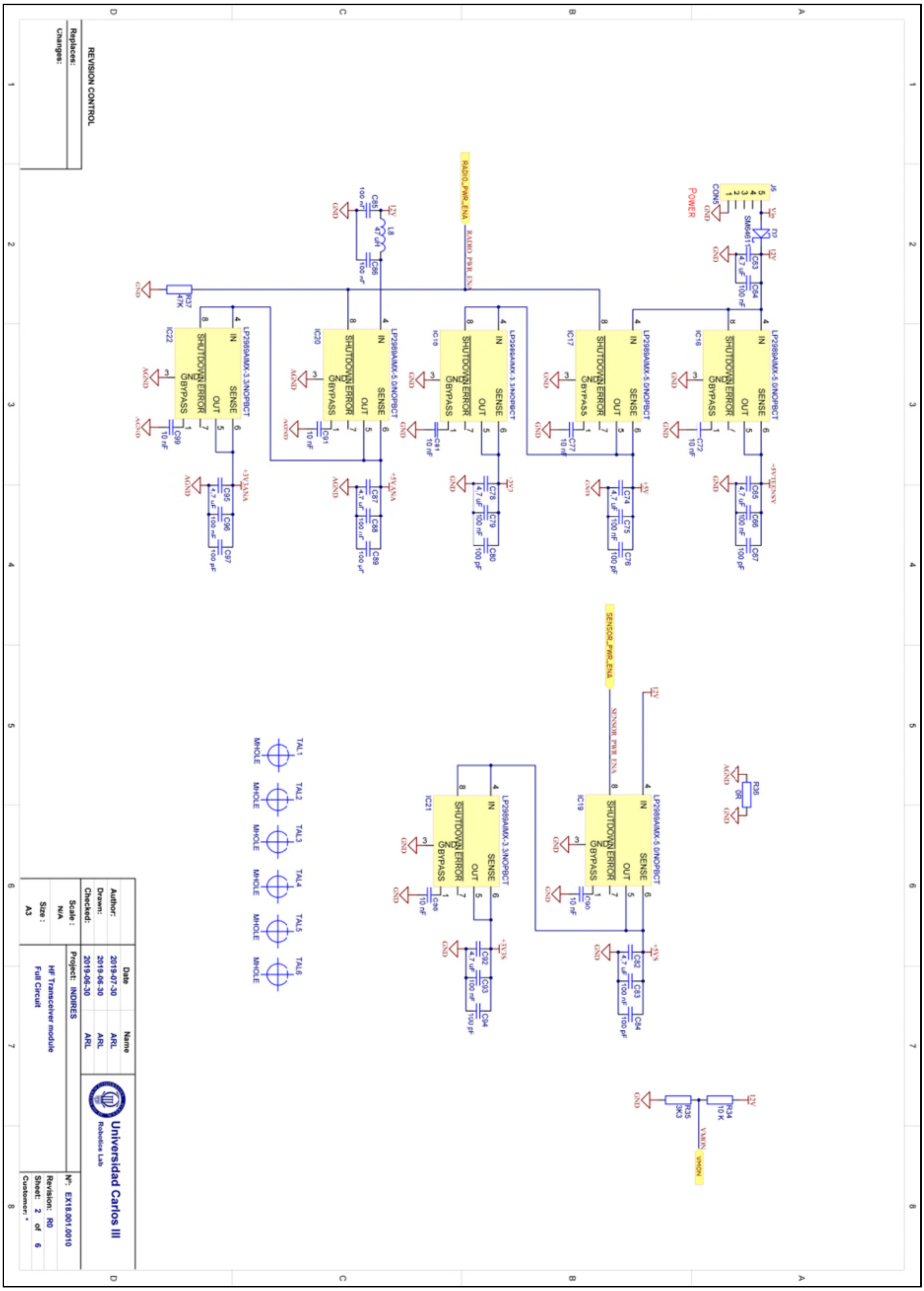
Teensy 3.5 & 3.6 Schematic

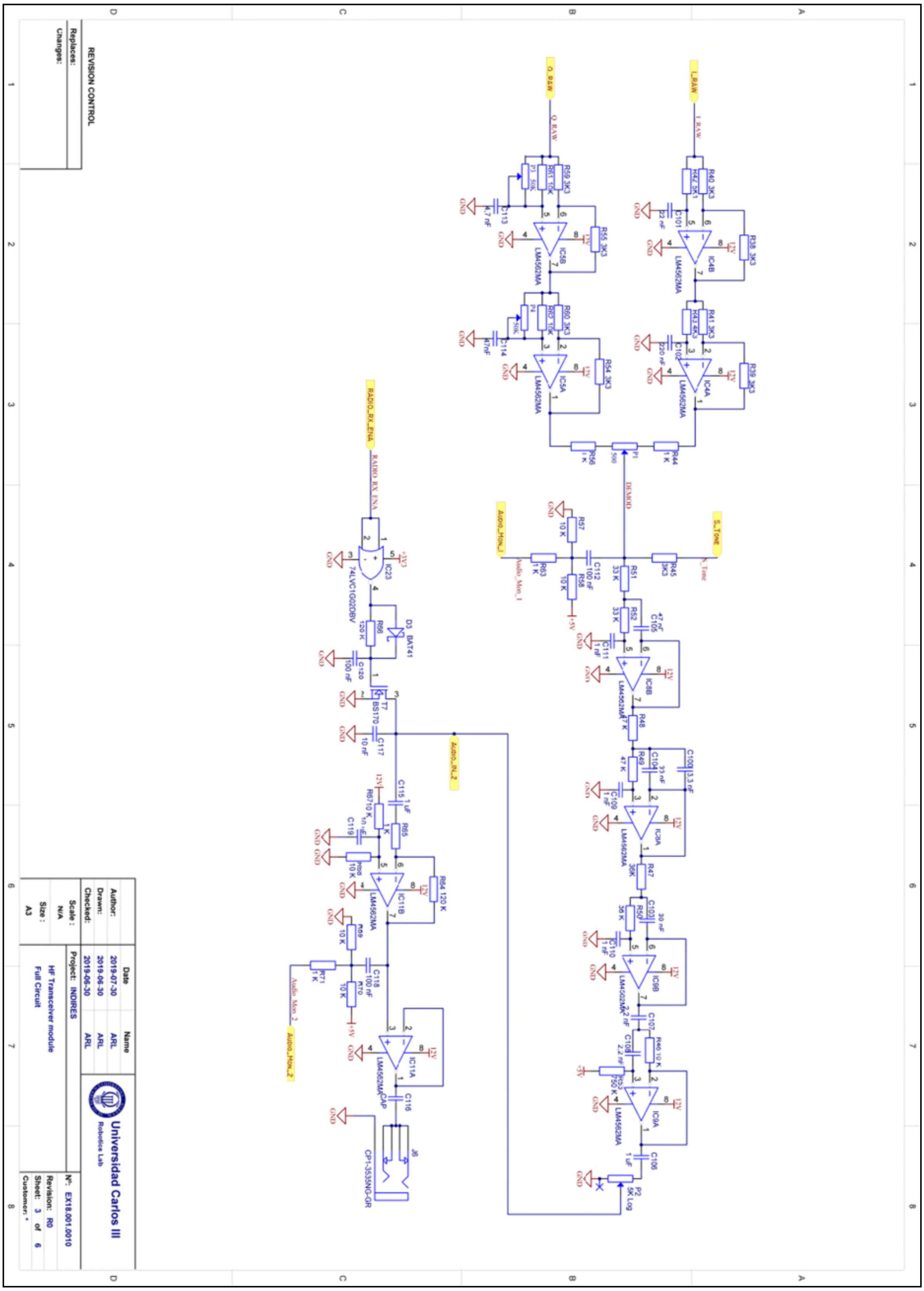


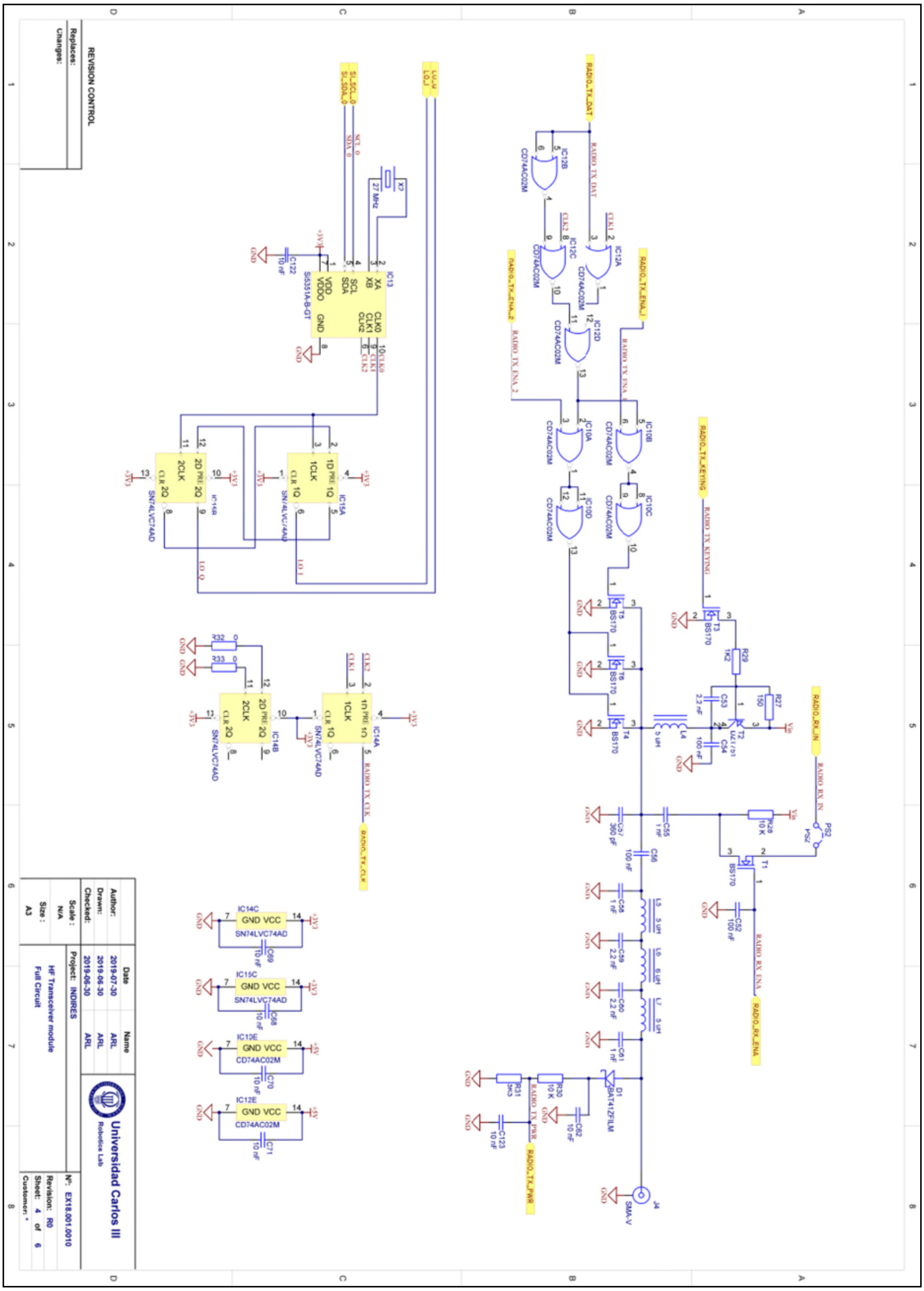
This page is intentionally left blank.

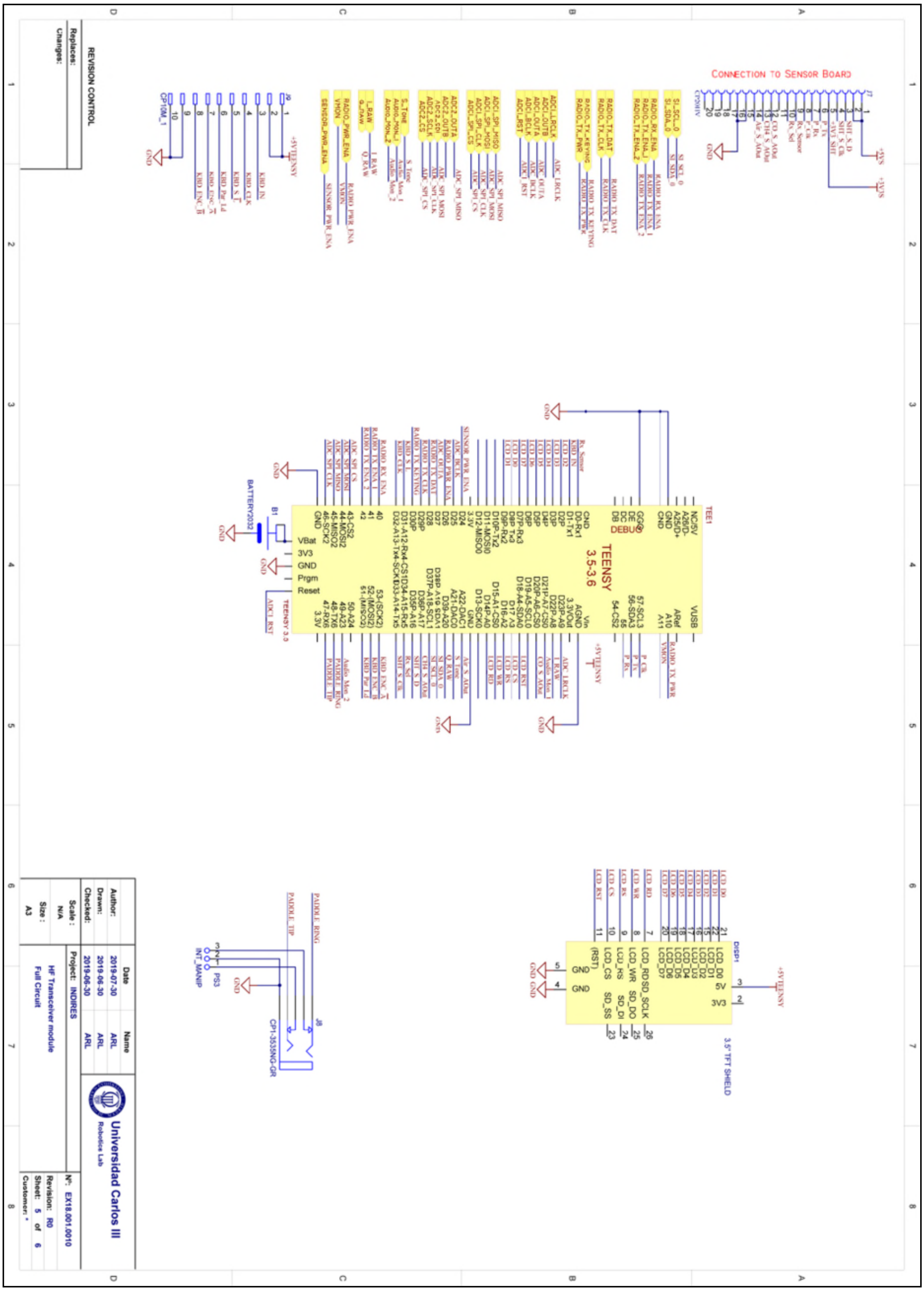
13 APPENDIX B – XCVR-HF-01A WMN CIRCUIT DIAGRAMS




REVISION CONTROL
Replaces:
Changes:
Universidad Carlos III
Robotics Lab
Project: INDIRES
HF Transceiver module
Full Circuit
Author: 2019-07-30 ARL
Drawn: 2019-06-30 ARL
Checked: 2019-06-30 ARL
Scale: N/A
Size: A3
Nº: EX18.001.0010
Revision: R0
Sheet: 2 of 6
Customer: *
RADIO_PWR_ENA
SENSOR_PWR_ENA
POWER
SHUTDOWN ERROR
GND
BYPASS
IN
SENSE
OUT
LP2989AIMX-5.0NOPBCT
LP2989AIMX-3.3NOPBCT
MHOLE

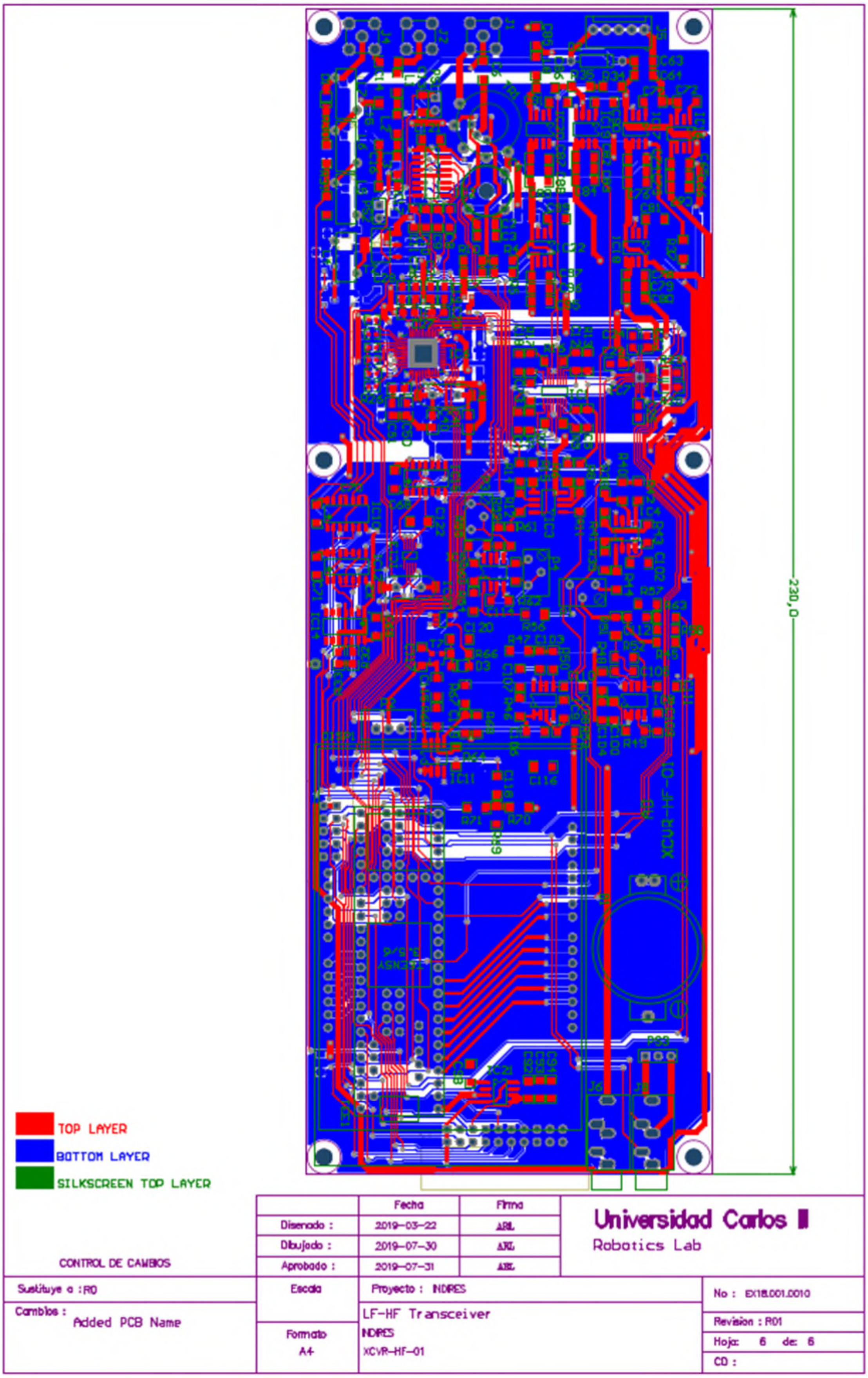

REVISION CONTROL
Replaces:
Changes:
Date
Name
Author: 2019-07-30 ARL
Drawn: 2019-06-30 ARL
Checked: 2019-06-30 ARL
Project: INDIRES
HF Transceiver module
Full Circuit
Scale: N/A
Size: A3
Universidad Carlos III
Robotics Lab
Nº: EX18.001.0010
Revision: R0
Sheet: 3 of 6
Customer: -


REVISION CONTROL
Replaces:
Changes:
Universidad Carlos III
Robotics Lab
Project: INDIRES
HF Transceiver module
Full Circuit
Nº: EX18.001.0010
Revision: R0
Sheet: 4 of 6
Customer: *
Author: 2019-07-30 ARL
Drawn: 2019-06-30 ARL
Checked: 2019-06-30 ARL
Scale: N/A
Size: A3
RADIO_TX_DAT
RADIO_TX_ENA_I
RADIO_TX_ENA_2
RADIO_TX_KEYING
RADIO_RX_IN
RADIO_RX_ENA
RADIO_TX_PWR
RADIO_TX_CLK
SI_SCL_0
SI_SDA_0
LO_I
LO_Q
SMA-V


REVISION CONTROL
Replaces:
Changes:
Connection to Sensor Board
TEENSY 3.5-3.6
DEBUG
3.5" TFT SHIELD
Universidad Carlos III
Robotics Lab
HF Transceiver module
Full Circuit

This page is intentionally left blank.

14 APPENDIX C – XCVR-HF-01A WMN PCB LAYOUT



This page is intentionally left blank.

15 APPENDIX D – SPECIFICATIONS OF COMMUNICATIONS PROTOCOLS

Outline specifications for the protocols have been created as follows.

15.1 Specification for SDP – Sensor Data Protocol

1. Sensors shall communicate with the ALCP handler by sending plain-text human-readable data. Binary data shall not be used.
2. If a handshake is required over a serial interface, this shall use the **X-ON**, **X-OFF** method
3. The responsibility for interpreting the data shall reside with the *receiving* node on the network. But the *sending* sensor shall provide sufficient information to allow this to happen.
4. *Example 1:* Suppose a sensor sends a temperature reading. It should send this as human-readable text, such as **123=24.5** [sensor #123, temperature 24.5 C] or more verbosely (and as might be produced from an**http_build_query()** function) as **sensor=123&temp=24.5C**. The information must be human-readable and sufficient in scope for the receiving node to be able to parse it.
5. *Example 2*: If a sensor sends audio data, it should send the binary data in a base64 encoding that is compatible with the ALCP.

15.2 Specification for ALCP – Application Layer Communications Protocol

1. The client and server communicate by sending plain-text messages. The client sends a *request* to the server and the server sends its *response* to the client.
2. This is a stateless protocol. Session control should be handled by the user.
3. *Except that…* the request headers can contain a timestamp, together with the timestamp of the previous session. Essentially this allows request bodies to be stitched together in sequence by the Application.
4. The INDIRES network does not split Application Layer data into packets for routing. Instead it treats each ALCP request as a single packet and routes it as one entity. It is the responsibility of the client to manage the size and sequence of the ALCP packets, using the TIME, LEN, PREV and SEQ headers.
5. The data is conveyed as 'printable text' using ASCII codes **0x20** to **0x7E**. Additionally,**0x0D**, **0x0A** shall be used to terminate a line and **0x09** shall be interpreted as a space character. No other characters are allowed.
6. An ALCP *request* comprises…
    - A request line
    - Request headers
    - An empty line
    - An optional message body

7. The request, status, and other header fields must each end with **\r\n**(i.e. CRLF, or **0x0D, 0x0A**).
8. The message body may contain any valid characters (see above), in whatever context required by the application. It is *suggested* that, for ease of debugging, lines are limited to 128 characters in length and are terminated by **\n**or **\r\n**. Each line, including its terminating CRLF, *mustnot* be more than 1024 characters long.
9. It is suggested that byte-stream data (such as audio) is base-64 encoded and printed at not more than 128 characters to the line. The base-64 decoding algorithm must ignore CRLF sequences within the data.

10. Header fields are colon-separated key-value pairs. Field names are case-insensitive. White space is allowed before and after the field value, and will be ignored; and it may be removed by the transport agent. Note that colon is a valid character in the value field (unlike in HTTP).
11. Header lines must *not* be wrapped (unlike in HTTP).
12. Headers may occur in any order. Headers outside the scope of this specification can be added as comment fields by preceding them with "X–".
13. The *ALCP request* is similar to an HTTP request, in that the client can specify a node (similar to an IP address), a port number, and a file to fetch. The client expects a *listener* to be present on the indicated port and to understand how to process the data.
14. An alternative scheme is provided for where, instead of using the port number to specify the data processing, this is specified in the APP field of the request header. If the APP field is present, the server should handle the data in accordance with the APP header or, if it cannot, it should issue an error message, rather than attempt any default processing.
15. An ALCP *response* comprises…
    - A status line
    - Response headers
    - An empty line
    - An optional message body
16. The same syntax applies to the response as to the request.
17. An ALCP response is *optional*. If there *is* a response then it comes from the destination, and can be used as a handshake; but handshaking may also be carried out at a lower layer in the communications operation. Obtaining a response is similar (superficially) to the operation of fetching a web page.
18. Status numbers shall be created as required during development. We require only a small sub-set of the full set of HTTP status codes, which shall include
    - ALCP 200 OK
    - ALCP 400 Bad Request
    - ALCP 403 Forbidden
    - ALCP 404 Not Found
    - ALCP 500 Internal Server Error
19. Values for ALCP Response Headers shall be created as required during development. Largely this is because we are not expecting a response body in most situations unless handshaking takes place at ALCP level. (See the HAND Request Header).

15.3 Specification for RLP – Routing Layer Protocol

The Application Layer packets are routed around the network and a routing table is required to achieve this.

20. The INDIRES network does not split Application Layer data into packets for routing. Instead it treats each ALCP request as a single packet and routes it as one entity. It is the responsibility of the client to manage the size and sequence of the ALCP packets, using the TIME, LEN, PREV and SEQ headers.

21. The RLP adds additional headers to the ALCP packet to indicate the previous and future hops of the data. See the table below.
22. The action of the RLP is as follows
    - Determine a route to the destination specified in the ALCP headers
    - Add the NEXT header (as described below)
    - Update the PATH header (as described below)
    - Send the ALCP packet to the next node along the route

Field Name	Description	Example of Field Value	Usage
DEST	Destination. A node, port and file on the network. The port defaults to zero. The file defaults to null.	0FA:23/ack.txt	Required
TIME	Unix timestamp	1549552874	Required
APP	"content type" indicator: use the named "App". This tells the destination what to do with the message body. For example, is it a text message? Is it audio speech? Is it a payload for onward transmission to an Internet URL? This field is optional. If it is present, the server must obey it or issue an error message. If it is absent, the server should use the default App specified for the port that is being accessed.	speech/LPC100 text/sms text/http	Optional
DATE	Human-readable date, for debugging purposes	Thu, 07 Feb 2019 15:57:24 +0000	Optional
ENC	A "content type" indicator: encoding method. Defaults to TEXT	base64	Optional
FROM	Human-readable origin (node address and port) for debugging purposes	005:23	Optional
HAND	Application Layer handshake. Specifies whether there should be an ALCP response to this data transmission. Defaults to 'Yes'	No	Optional
LEN	Length of message body; defaults to zero	354	Optional
PREV	Previous timestamp, allows packet re-assembly	1549552873:994	Optional
SEQ	Millisecond timestamp, used with TIME as a packet identifier	123	Optional
SUBJ	Optional Subject, for debugging purposes	voice	Optional

Table 6 – ALCP Request Headers

Field Name	Description	Example of Field Value	Usage
NEXT	The next node to which this packet is being forwarded.	0F2	Required
PATH	The complete path taken so far, read from left to right. The first value is the source of the packet; the last value is the current node that is processing the packet. This header is modified when the data is routed.	006 007 0F0	Required

Table 7 – RLP Headers that are added to each ALCP packet