

NASA/TM-20205002770



Antenna and Flight Control Software for Conformal Phased Array Antenna Design (CPAAD)

Daniel R. Oldham
Glenn Research Center, Cleveland, Ohio

NASA STI Program . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program plays a key part in helping NASA maintain this important role.

The NASA STI Program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI Program provides access to the NASA Technical Report Server—Registered (NTRS Reg) and NASA Technical Report Server—Public (NTRS) thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counter-part of peer-reviewed formal professional papers, but has less stringent limitations on manuscript length and extent of graphic presentations.
- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., “quick-release” reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.
- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.
- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.
- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.
- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at <http://www.sti.nasa.gov>
- E-mail your question to help@sti.nasa.gov
- Fax your question to the NASA STI Information Desk at 757-864-6500
- Telephone the NASA STI Information Desk at 757-864-9658
- Write to:
NASA STI Program
Mail Stop 148
NASA Langley Research Center
Hampton, VA 23681-2199

NASA/TM-20205002770



Antenna and Flight Control Software for Conformal Phased Array Antenna Design (CPAAD)

Daniel R. Oldham
Glenn Research Center, Cleveland, Ohio

National Aeronautics and
Space Administration

Glenn Research Center
Cleveland, Ohio 44135

August 2020

Acknowledgments

Thank you goes to Chief Gene Fujikawa of the Cognitive Signal Processing Branch for support in completing this document.

This report is a formal draft or working paper, intended to solicit comments and ideas from a technical peer group.

This work was sponsored by the Transformative Aeronautics Concepts Program.

Trade names and trademarks are used in this report for identification only. Their usage does not constitute an official endorsement, either expressed or implied, by the National Aeronautics and Space Administration.

Level of Review: This material has been technically reviewed by technical management.

Available from

NASA STI Program
Mail Stop 148
NASA Langley Research Center
Hampton, VA 23681-2199

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161
703-605-6000

This report is available in electronic form at <http://www.sti.nasa.gov/> and <http://ntrs.nasa.gov/>

Antenna and Flight Control Software for Conformal Phased Array Antenna Design (CPAAD)

Daniel R. Oldham
National Aeronautics and Space Administration
Glenn Research Center
Cleveland, Ohio 44135

Abstract

The National Aeronautics and Space Administration (NASA) is developing lightweight conformal phased array antenna designs (CPAAD) for beyond line of sight communications (Ref. 1). The CPAAD system is developed to address reliable and affordable autonomous next generation aviation systems. The technology takes advantage of newly assigned provisional Ku-bands for UAVs. The antenna design is unique to avoid interference with the ground and the unconventional substrate reduces weight. The surface mounted conformal design allows for reduced drag during flight. The software developed for the CPAAD system operates the antenna hardware and provides a graphic user interface (GUI) for ground testing and flight operations.

Introduction

The Conformal Phased Array Antenna Design (CPAAD) is an 8 by 8 (64 elements) phased array antenna using an aerogel backbone. The antenna uses beam forming software for antenna control and interference mitigation. Characterization of the antenna is performed with planar nearfield and flight testing. Flight testing includes the flight control system on the aircraft and the GRC developed measurement ground system (Ref. 2).

The flight control system block diagram is shown in Figure 1. The flight control system for the aircraft includes power conditioning, the antenna, thermocouples, control computer, GPS receiver, frequency synthesizer, frequency reference and router. For the flight configuration, the router function was implemented within Ubuntu on the Jetson. The antenna, beam steering and flight control software are covered within this paper and the GRC developed measurement ground station and nearfield testing equipment (Ref. 3) are outside the scope of this document.

Conformal Phased Array Antenna Design and Electronics

The antenna and flight control software manage the antenna hardware and provides a graphical user interface for ground testing and flight operations. The control software operates the antenna RF devices, sets the frequency synthesizer, configures and reads the thermocouples, reads the GPS information and the frequency reference values. The software is written in C/lex/yacc and uses GTK 3.0 tool kit for the GUI applications. The software development includes two python programs to allow for import and export of CPAAD waveforms and antenna configurations. The antenna waveforms developed for nearfield testing are imported into the CPAAD software for archiving and flight testing. The software also allows for exporting CPAAD antenna configuration information.

Block Diagram for CPAAD

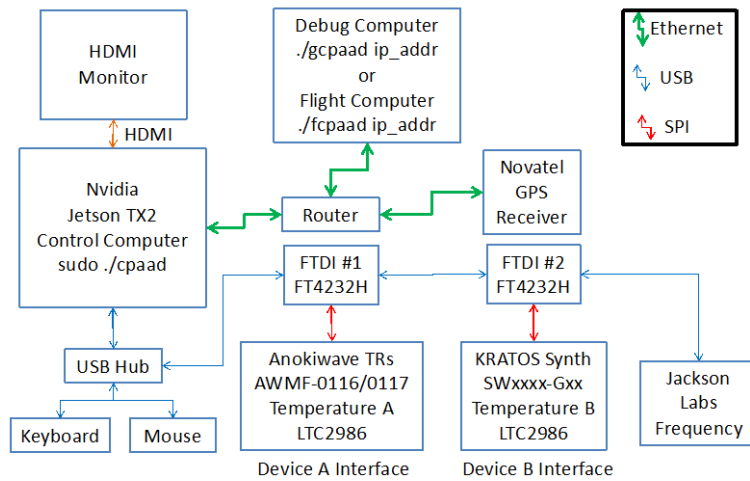


Figure 1.—Block Diagram for CPAAD.



Figure 2.—Rosie Embedded System with NVIDIA® Jetson TX2.

The control computer is a hardened NVIDIA Jetson TX2 with Jetpack 2 running Ubuntu 16.04 and is shown in Figure 2. The computer is designed to MIL STD 810g specifications and DO 160G for shock and vibration with IP68 ingress protection rating. The computer uses two FT4232H USB to SPI devices with LibMPSSE to control the antenna and thermocouples. The antenna SPI interface controls the 64 Anokiwave AWMF-0117 transmit receive (T/R) modules and Kratos frequency synthesizer. The thermocouples SPI controls two LTC2986 thermocouples each with four channel temperature monitoring. The HDMI monitor, keyboard and mouse are used in the laboratory for debugging and calibration operations. The antenna, flight control and beam steering software are designed to run on the Jetson TX2 as an integrated software system.

Conformal Phased Array Antenna Design Software

The antenna software is divided into four parts; antenna control software, beam steering software, flight control software and import-export software. The antenna control software is required to operate the antenna hardware and is the only software required to manage the system. The beam steering software is a GUI for testing and debugging the antenna system. The flight control software is a GUI used for flight operations. The antenna control and beam steering software targets a NVIDIA Jetson TX2 and the flight control software targets a laptop with Ubuntu 16.04.

The antenna is built out of ultra-lightweight low dielectric polymer aerogels with 64 Anokiwave AWMF-0117 transmit receive (T/R) modules on a waveguide backplane with heat sink. The antenna RF devices allow individual gain and phase control for beam steering and pattern synthesis. There is one T/R module chip per element using a 2.5×2.5 mm integrated circuit in BGA package. The initial antenna design allows for transmit and receive operations. The shift registers for RF transmit/receive functions are the 74LVC594A device but removed from the prototype design due to unexpected SPI operations. The removal of the shift registers locked the T/R module chip into transmit only operations. The software uses the LTC2986 Digital Temperature Measurement System with four thermocouples and thermistor per board. The Kratos Frequency Synthesizer is used to generate a reference 6 to 18 GHz signal. A Jackson Labs GPIO device is used to determine if the antenna waveform RF signal is locked. The Jackson Labs is a 100 MHz double oven crystal oscillator that is locked to GPS. It is used as the reference for the Kratos synthesizer to generate the Ku-band signal. The software uses a Novatel GPS receiver for current position information. The GPS position information is used to calculate the current pointing angle and direction. The information is applied back to the beam steering software.

CPAAD Software Configuration and Setup Operations

The CPAAD software operates on a PC or Virtual Machine using Ubuntu 16.04 with several software packages installed. Start with Ubuntu 16.04 and add flex, bison, swig, octave and python-dev software. Remove libftdi1 and add libftdi1-2 and add libftdi1-dev.

The MPSSE Library is required for the FTDI FT4232H devices for USB to SPI and GPIO signal processing. Use the following command to get the libmpsse library source code. You will need a github user account and password to download the files.

```
git clone https://github.com/devttys0/libmpsse.git
```

Note, there is an error in the source code for libmpsse library. Edit the libmpsse/src/fast.c file and change line 178 to call the function fast_build_block_buffer instead of build_block_buffer. See the information below for the corrected source line of code. Make the change to the source code and save the file.

```
if(fast_build_block_buffer(mpsse, mpsse->txrx, (unsigned char *) (wdata + n),  
rxsize, &data_size) == MPSSE_OK)
```

To configure the libmpsse library, cd into the libmpsse source directory and execute the following commands. The extra parameters define CFLAGS to find the ftdi.h file located in the /usr/include/libftdi1/ directory and the LDFLAGS sets the libmpsse library in the /usr/local/lib directory. Run the makefile and then use sudo privilege to install the software.

```
cd libmpsse/src  
./configure CFLAGS=-I/usr/include/libftdi1 LDFLAGS=-L/usr/local/lib  
make  
sudo make install
```

The commands build the mpsse software and install the libmpsse library into the /usr/local/lib directory.

CPAAD Software Source Code

The CPAAD software is now ready to be installed. Use the git command below to obtain the CPAAD software source code. The software comes as a tar file (cpaad288.tar 1341680 bytes).

```
git clone https://gitlab.grc.nasa.gov/daniel.r.oldham/cpaad
```

Execute the following command to extract the files. The files are defined in Table 1 with the CPAAD software directory location and the size in bytes of the files.

```
tar -xvf cpaad288.tar
```

The agile CPAAD software development method allows the software to track the antenna research development cycle and provides antenna applications for debug and flight testing. The first antenna prototype is a discrete two element antenna with frequency synthesizer and the original software is in directory cpaad/. The second antenna prototype is the 64-element antenna and the software is updated to version cpaad2/.

TABLE 1.—CPAAD SOFTWARE SOURCE FILES

| <i>directory/filename</i> | <i>bytes</i> | <i>directory/filename</i> | <i>bytes</i> |
|------------------------------------|--------------|--------------------------------------|--------------|
| cpaad2/Linux/cpaad.py | 18824 | cpaad2/fcpaad/Linux/freqERROR.jpg | 11740 |
| cpaad2/Linux/Makefile | 14730 | cpaad2/fcpaad/Linux/freqOFF.jpg | 11646 |
| cpaad2/Linux/NovatelControl2.py | 40719 | cpaad2/fcpaad/Linux/freqON.jpg | 11466 |
| cpaad2/Linux/pcpaad.py | 8942 | cpaad2/fcpaad/Linux/freqWARNING.jpg | 12718 |
| cpaad2/source/cpaad.c | 132664 | cpaad2/fcpaad/Linux/gpsERROR.jpg | 11777 |
| cpaad2/source/dencode.c | 25656 | cpaad2/fcpaad/Linux/gpsOFF.jpg | 11126 |
| cpaad2/source/lex.l | 5677 | cpaad2/fcpaad/Linux/gpsON.jpg | 11120 |
| cpaad2/source/LTC2986.c | 16634 | cpaad2/fcpaad/Linux/gpsWARNING.jpg | 12840 |
| cpaad2/source/yacc.y | 16539 | cpaad2/fcpaad/Linux/jlabERROR.jpg | 11759 |
| cpaad2/include/cpaad.h | 18195 | cpaad2/fcpaad/Linux/jlabLOCKED.jpg | 11959 |
| cpaad2/include/dencode.h | 3019 | cpaad2/fcpaad/Linux/jlabOFF.jpg | 11186 |
| cpaad2/include/LTC2986.h | 22430 | cpaad2/fcpaad/Linux/jlabUNLOCKED.jpg | 13265 |
| cpaad2/gcpaad/source/gcpaad.c | 470352 | cpaad2/fcpaad/Linux/Makefile | 1613 |
| cpaad2/gcpaad/include/gcpaad.h | 7261 | cpaad2/fcpaad/Linux/tempAERROR.jpg | 11903 |
| cpaad2/gcpaad/Linux/Makefile | 1565 | cpaad2/fcpaad/Linux/tempAOFF.jpg | 11496 |
| cpaad2/fcpaad/source/fcpaad.c | 41267 | cpaad2/fcpaad/Linux/tempAON.jpg | 11290 |
| cpaad2/fcpaad/include/fcpaad.h | 2230 | cpaad2/fcpaad/Linux/tempAWARNING.jpg | 13311 |
| cpaad2/fcpaad/Linux/antERROR.jpg | 11806 | cpaad2/fcpaad/Linux/tempBERROR.jpg | 12120 |
| cpaad2/fcpaad/Linux/antOFF.jpg | 11019 | cpaad2/fcpaad/Linux/tempBOFF.jpg | 11978 |
| cpaad2/fcpaad/Linux/antON.jpg | 11166 | cpaad2/fcpaad/Linux/tempBON.jpg | 11830 |
| cpaad2/fcpaad/Linux/antWARNING.jpg | 13082 | cpaad2/fcpaad/Linux/tempBWARNING.jpg | 13140 |
| cpaad2/fcpaad/Linux/config1.jpg | 10606 | cpaad2/beam_steering/beam_steering.m | 12620 |
| cpaad2/fcpaad/Linux/config2.jpg | 11119 | cpaad2/beam_steering/cpaad_codegen.m | 9954 |
| cpaad2/fcpaad/Linux/config3.jpg | 10761 | cpaad2/beam_steering/iseven.m | 141 |
| cpaad2/fcpaad/Linux/config4.jpg | 10332 | cpaad2/beam_steering/pat2uvplot.m | 1553 |
| cpaad2/fcpaad/Linux/config5.jpg | 11344 | cpaad2/beam_steering/taylor_win.m | 585 |

The software follows a similar directory layout between the three software applications; cpaad, gcpaad and fcpaad. The source directory contains the C/lex/yacc source files, the include directory the include files, the Linux directory the Makefile, icons, python and resulting executables. The cpaad software is shared so some functions are used with the two GUI software applications. The two GUI applications are in a directory below the antenna software cpaad/gcaad and cpaad/fcaad. The beam steering software is a Matlab/Octave program and located in the cpaad/beam_steering directory. The antenna pointing direction and beam steering parameters are used to generate the CPAAD antenna waveform (filename.cpa). The software makes a call into the computer system to execute the beam steering software and downloads the generated beam steering output file to the antenna.

CPAAD Software Build Process

The CPAAD software is built using the three Makefiles in the different Linux directories. First build the antenna control software then the beam steering software then the flight control software. Use the following commands to build the three software applications. The files will compile and link without any warnings or errors.

```
cd cpaad2/Linux
make
cd ../gcpaad/Linux
make
cd ../../fcpaad/Linux
make
```

To rebuild the software, use the following command to remove the application software from the directory first.

```
make clean
```

CPAAD Software Startup

The antenna control software is the only software required to manage and operate the antenna system. The software uses a common command interpreter so the GUI functions can be executed from the antenna control command line. The two GUI programs provide a user interface to allow for easy access to the antenna operations. The GUI programs use TCP/IP client server functions (port 6000) for command processing and UDP/IP multicast functions (239.0.0.2 port 3241) for telemetry data. The software is started using root privilege with the following command.

```
cd cpaad2/Linux
sudo ./cpaad [command line parameters]
```

The command line parameters change the default variables of the software before the software starts. The following parameters are accepted by the command line parser and are shown with the default values in Table 2.

Once the software starts the cpaad> prompt appears on the console window unless there is an error that prevents the software from running. There must be at least one FT4332 device connected to the USB port for the software to start. The following parameters, shown in Table 3 are accepted by the run time parser along with the parameters defined above. If there is an error, then use the +debug and +yydebug flags on the command line to turn verbose messaging ON to get a detailed error message.

TABLE 2.—CPAAD COMMAND LINE PARAMETERS

| Parameter | Comments | Default Value |
|-----------------------|--|-----------------------------|
| ip address | IP Address of Novatel Device | 192.168.1.3 |
| frequency | Kratos Frequency Synthesizer in GHZ | 14.25 |
| -B[baud] | Jackson labs Frequency Counter Baud Rate | -B115200 |
| -D[device] | Jackson labs Frequency Counter Device Name | -D/dev/ttyUSB8 |
| [+ -]server | Start or Stop the TCP/IP client/server | +server |
| [+ -]shift | Include or Exclude the TR shift registers | -shift |
| [+ -]frequency | Include or Exclude frequency synthesizer | +frequency |
| [+ -]jllabs | Include or Exclude the Jackson Labs Counter | +jllabs |
| [+ -]novatel | Include or Exclude the NovAtel GPS | +novatel |
| [+ -]ltc2869 | Include or Exclude the Temperature Devices | +ltc2869 |
| [+ -]telemetry | Include or Exclude the Telemetry Packets | +telemetry |
| [+ -][yy]debug | Turn debug or yydebug messaging ON/OFF | -debug -yydebug |
| [+ -]packet | Turn packet messaging (verbose) ON/OFF | -packet |
| run stop | Turn temperature and frequency monitoring ON/OFF | run |
| load | Load Antenna Waveform or Configuration file | novatel.cfg |
| file.[cpa cfg] | Load Antenna Waveform or Configuration file | filename.cpa |
| save | Save the current Antenna Waveform to file | savefile.cpa |
| abort | Abort the file load operation | |
| AA AB BA BB | Set parameter for SPI Device and Interface | AA |
| SPIO SPI1 ... | Set the SPI device mode | SPIO |
| [baud] | Set the SPI device baud | TEN_KHZ FOUR_HUNDRED_KHZ |
| [endian] | Set the SPI device endian | LSB_MSB |

TABLE 3.—CPAAD RUNTIME COMMANDS

| Runtime Parameters | Comments |
|---|---|
| TR[1-64] [tx rx off] | Set TR device to TX RX OFF |
| [+ -]dev [a b]_int[a b c]_1[0-7] | Set or Reset GPIO device |
| set | Set the frequency device in GHZ (14.25) |
| cmd | Set the antenna waveform |
| temp | Read the temperature values |
| setup | Initialize the temperature devices |
| reset | Reset the temperature devices |
| latch | Cycle the antenna GPIO latch |
| sw1 | Set GPIO for SW1 configuration |
| sw2 | Set GPIO for SW2 configuration |
| status | Print the CPAAD status to the screen |
| tr | Print the TR devices to the screen |
| swap | Swap FT4323H devices A and B |
| context | Print the context of FT4323H devices |
| pins | Print the GPIO pins of the devices |
| novatel | Configure the NovAtel Device |
| cfg | Export the NovAtel Configuration |
| hex | Load the data/hexfile.hex Waveform |
| filename.hex | Load the filename.hex Waveform |
| quit exit | Quit or Exit the CPAAD software |

CPAAD GUI Beamforming Software

The CPAAD beamforming software provides a GUI for the user to control the antenna system and create antenna waveform files. The software is started using root privilege with the following command. The optional default IP address is the local 127.0.0.1 address or enter the IP address of the antenna control computer for a remote IP connection.

```
cd cpaad2/gcpaad/Linux
sudo ./gcpaad [ip address]
```

The CPAAD antenna software starts and the main window is shown in Figure 3. The window contains the beam steering algorithm, the software control window and the device control panels. The antenna array editor windows shown in Figure 4 and Figure 5 allows editing phased array antenna values. The antenna array editor window allows editing of each element or all 64 elements at a time. The design monitor in Figure 6 and Figure 7 displays the current values of the 64 devices and is color coded to show transmit/receive mode, phase angle, amplitude level and signal polarity.

The CPAAD antenna devices have polarity based on the physical package orientation. Due to the placement of the columns the devices in every other column is turned 90°. To make the appearance consistent to the user the software automatically adjusts the waveform polarity when writing the waveform data to the antenna system.

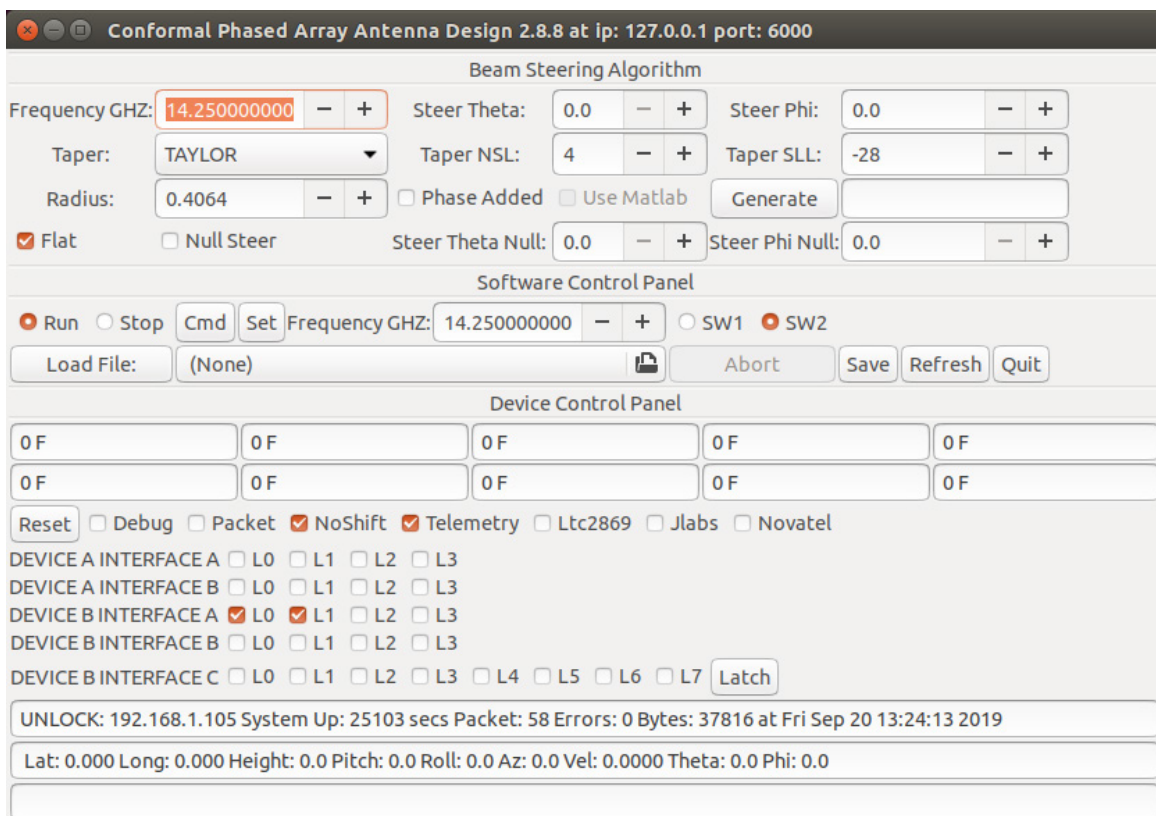


Figure 3.—CPAAD Main Control Window.

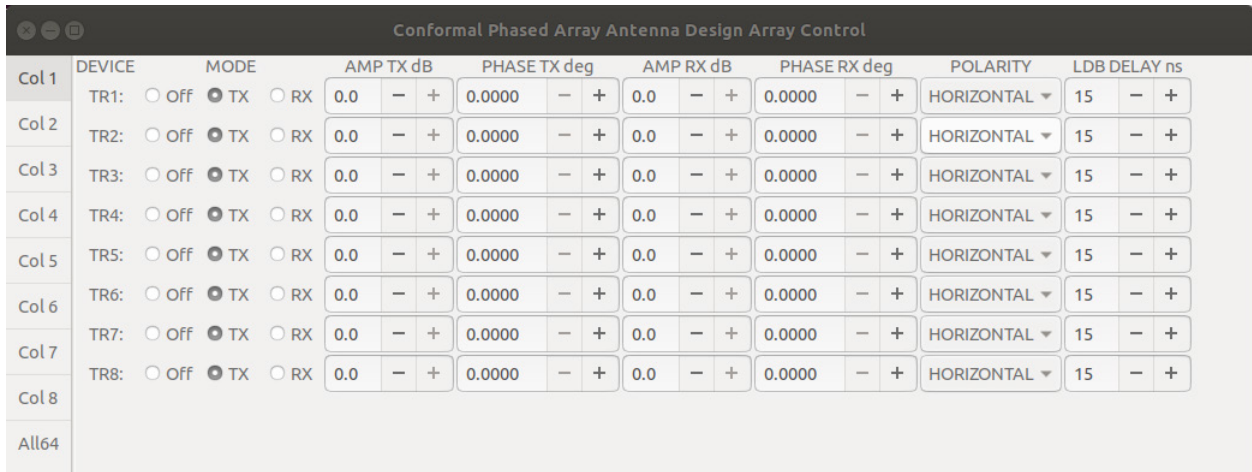


Figure 4.—CPAAD Antenna Array Editor Window.

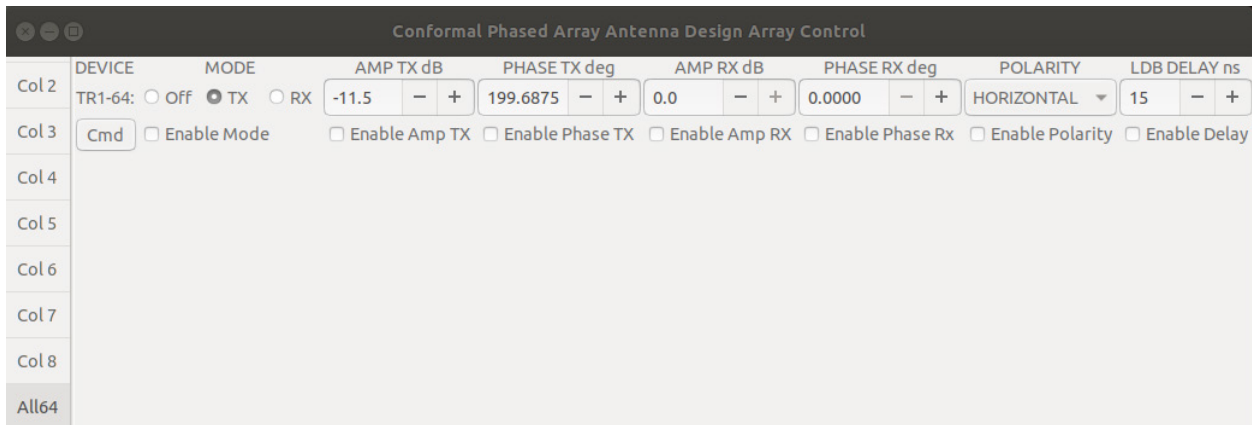


Figure 5.—CPAAD All 64 Array Elements Editor.

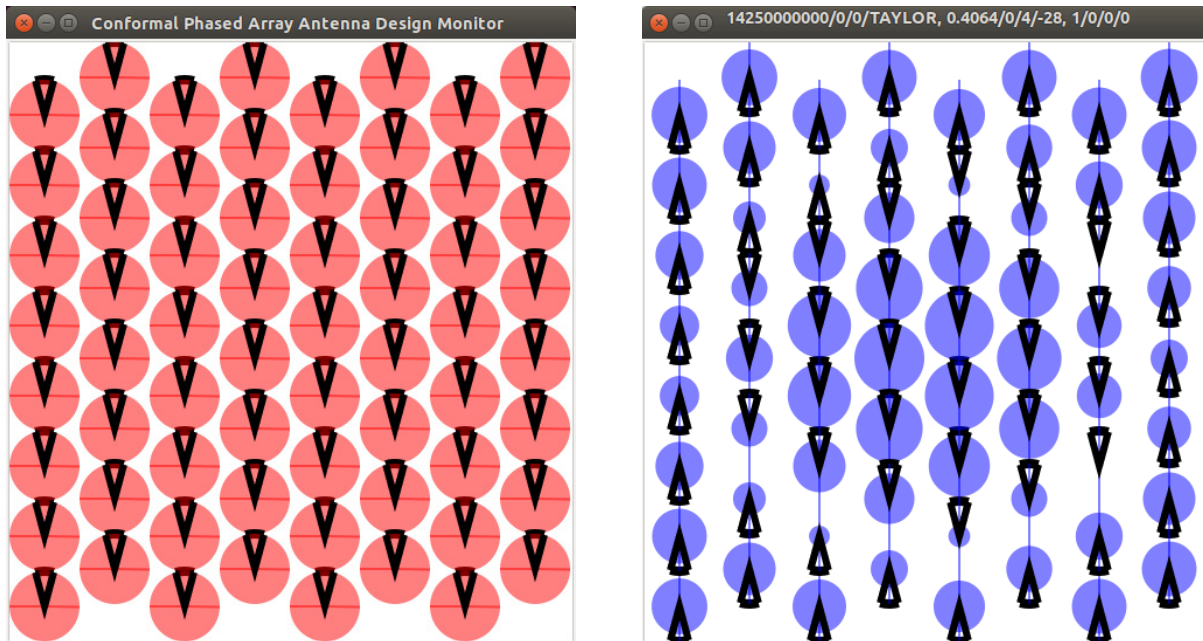


Figure 6.—CPAAD Array Monitor.



Figure 7.—Example Antenna Waveforms.

The CPAAD antenna software uses electronic beam forming software to create antenna waveforms. The software algorithm redirects side lobe energy away from the ground and towards the target. The Beam Steering Algorithm is written in Matlab/Octave and the source code is located in the `cpaad2/beam_steering` directory. The user enters the parameters in the top panel and selects the “Generate” button. The software creates a `test.m` file with the GUI parameters and the Matlab program generates the antenna waveform into the `cpaad2/data` directory. The filename is created based on the parameters of the Beam Steering Algorithm panel. The software waveform is then loaded into the antenna system and the display windows are updated. The time period to create the antenna waveform is displayed in the window next to the generate button. If both Matlab and Octave are on the system, then select the “Use Matlab” button to switch between using the two software packages.

CPAAD Software Control Panel

The Software Control Panel operates the antenna system. There are buttons to change the monitoring thread state, set the frequency, set the antenna configuration and load or save antenna waveform files. There is a button to exit the software and a button to refresh the screen from the antenna control software. The original antenna development used only point to point messaging (Refresh) to update the screen. When the thermocouples were added to the design then multi-casting (Telemetry) was added to update the screen once a second.

CPAAD Device Control Panel and Messages

The Device Control Panel is the lowest level device functions in the Main Control Window. The temperatures from the thermocouples are shown with the software state flags, device GPIO, telemetry, refresh status and the message windows. Operating any of these buttons is not normally recommended and is for debugging purposes only. Below the Device Control Panel are the telemetry, GPS and message displays. The telemetry data includes the frequency lock indicator (LOCK/UNLOCK), antenna control

computer IP address, system status and current date and time. The GPS display includes the current position, height, pitch, roll, azimuth, velocity and antenna pointing direction. The antenna control software uses the current GPS position to calculate the antenna pointing direction (theta, phi).

CPAAD Software IP Address and DHCP

If the CPAAD software is run on the same computer then the software will use the local IP address (127.0.0.1) for commands and telemetry messaging. If the beam forming software or flight software is on a remote computer then plug the remote computer into the router to obtain a DHCP address. If the antenna control computer IP address is unknown, then start the CPAAD software without the IP address and the software will display the IP address in the telemetry message display. Then exit the software and restart using the IP address on the command line to connect to the antenna control computer.

CPAAD Antenna Array Control Editor

The CPAAD Array Control window allows for editing the individual array elements or all 64 elements at a time. To display the eight devices in a column, select the column of the device (1-8) on the left tab. Note, the device mode (OFF, TX, RX) is locked in TX mode only. Adjust the amplitude, phase or polarity using the window controls in any of the columns. The window data is not applied to the antenna until the CMD button is clicked on the Main Control Window or the CMD button on the All64 display window. If refresh is clicked or telemetry is ON, then the antenna GUI data will be overwritten with the current antenna data and all edits will be lost.

When using the All64 antenna array control, shown in Figure 5, select the enable button below each parameter to be adjusted to select those parameters for application to all the antenna elements. Change the amplitude, phase, polarity and select the CMD button to write the data to the antenna.

CPAAD Antenna Design Monitor

The CPAAD Design Monitor displays the 64 elements of the phase array and examples are shown in Figure 6 and Figure 7. The monitor display shows the mode, amplitude, phase and polarity. The mode is shown as red for transmit and blue for receive. The size of the circle indicates the amplitude value determined by the attenuation setting. The direction of the horn indicates the phase angle and the short line indicates the polarity. By default the antenna is configured as a transmitter, at zero degrees, zero attenuation and horizontal polarity. The window on the right shows the antenna waveform as a receiver with vertical polarity.

Two example waveforms are shown in Figure 7 using the Taylor and Uniform tapers at 14.25 GHZ.

CPAAD Flight Control Software

The CPAAD Flight Control Software allows the user to download five antenna waveforms, monitor temperatures, display device status and messaging. The software is started using root privilege with the following command.

```
cd cpaad2/fcpaad/Linux
sudo ./fcpaad [ip address]
```

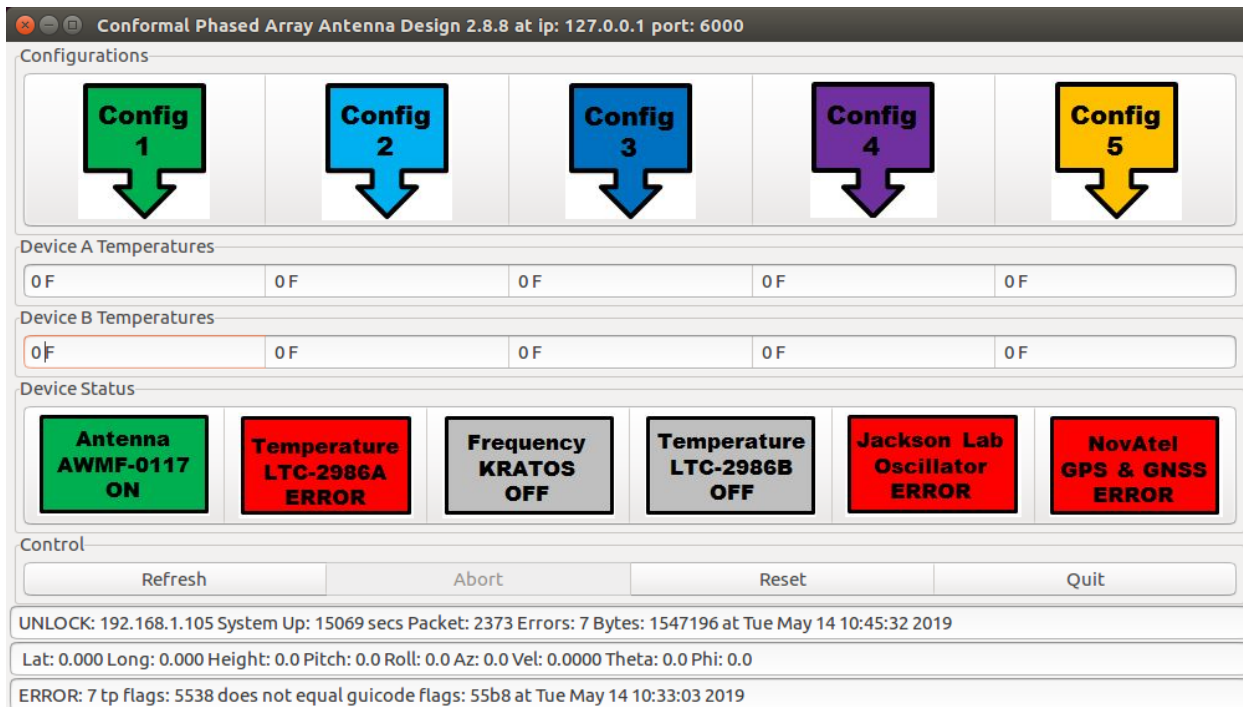


Figure 8.—Flight Control Software.

The CPAAD flight control software starts and the flight control software window will appear on the screen as shown in Figure 8.

The CPAAD flight control software has five buttons along the top to select an antenna waveform file to be downloaded to the antenna system. If the antenna waveform has delay instructions, then the “Abort” button will become active and allow the user to abort the load operation.

The five configuration buttons load a waveform file with the names; config1.cpa, config2.cpa, config3.cpa, config4.cpa and config5.cpa. The files are located in the cpaad2/data directory. Any CPAAD waveform file generated can be used and only needs to have the correct filename. Copy any antenna waveform file to the config filename to use that waveform file with the configuration download button.

The “Refresh” button is a command to update the display values. When telemetry is running the display values are automatically updated once per second. The “Reset” button is a command to reset the SPI devices (antenna, frequency, thermocouples). The “Quit” button exits the software. The temperature values are displayed in the device A and B temperature displays. The device status is shown with color icons to indicate the state of the devices.

The telemetry message shows the antenna status, antenna computer IP address, the system up time, the number of telemetry packets, number of errors detected, the number of packet bytes and the current date and time. The position message shows the GPS position (lat, long), height, pitch, roll, azimuth, velocity and antenna pointing direction. The bottom message display shows the last system or error message and records the message to the fcpaad.log file.

The Device Status shows the state of each device. There are four states per device and shown in Table 4.

TABLE 4.—DEVICE STATUS STATES

| | | | | | |
|--|--|---|--|--|---|
| Antenna AWMF-0117 ON | Temperature LTC-2986A ON | Frequency KRATOS ON | Temperature LTC-2986B ON | Jackson Lab Oscillator LOCKED | NovAtel GPS & GNSS ON |
| Antenna AWMF-0117 WARNING | Temperature LTC-2986A WARNING | Frequency KRATOS WARNING | Temperature LTC-2986B WARNING | Jackson Lab Oscillator UNLOCKED | NovAtel GPS & GNSS WARNING |
| Antenna AWMF-0117 ERROR | Temperature LTC-2986A ERROR | Frequency KRATOS ERROR | Temperature LTC-2986B ERROR | Jackson Lab Oscillator ERROR | NovAtel GPS & GNSS ERROR |
| Antenna AWMF-0117 OFF | Temperature LTC-2986A OFF | Frequency KRATOS OFF | Temperature LTC-2986B OFF | Jackson Lab Oscillator OFF | NovAtel GPS & GNSS OFF |

CPAAD Antenna Waveform and Configuration Format

The CPAAD antenna waveform and configuration format is defined below as a grammar. A grammar has non-terminal symbols, terminal symbols, productions and a start symbol. The waveform and configuration files can have antenna commands, waveforms, configuration data and comments. Comments are C style between “/*” and “*/” or // to end of the line. The SET command loads the frequency synthesizer with the default frequency or with the value of the parameter (14.25 GHZ). The CMD command loads the current waveform into the antenna system. The RUN and STOP commands start and stop the temperature and frequency monitoring threads. The SW1 and SW2 commands set GPIO pins. These commands are from the original software version and not used.

The # delay introduces a delay in the load file operation and the value is in microseconds. If the value is greater than 1,000,000 then the delay is in seconds. The IPADDRESS is the IP address of the NovAtel device. The variable ‘=’ number is for loading the configuration file parameters. The antenna device production allows for setting the RF mode (TX, RX, OFF) or with a single amplitude and phase value or with all the antenna device parameters.

CPAAD Antenna Waveform Grammar

```

file      : commands
          | file commands
          ;
commands : SET
          | SET number
          | CMD
          | RUN
          | STOP
          | SW1
          | SW2
          | '#' delay
    
```



```

        | device shift [ amp phase [ amp phase ] [ polarity ] [ delay ] ]
        | variable '=' number
        | IPADDRESS
        ;
device   : TR1
        | ...
        | TR64
        ;
shift    : OFF
        | TX
        | RX
        ;
amp      : number
        ;
phase    : number
        ;
polarity : HORIZONTAL
        | VERTICAL
        ;
delay    : number
        ;
variable : PL_ANT1_X
        | PL_ANT1_Y
        | PL_ANT1_Z
        | PL_ANT2_X
        | PL_ANT2_Y
        | PL_ANT2_Z
        | PL_IMU_X_ROT
        | PL_IMU_Y_ROT
        | PL_IMU_Z_ROT
        | ARRAY_INVEH_X
        | ARRAY_INVEH_Y
        | ARRAY_INVEH_Z
        | ARRAY_INVEH_ROT_ROLL
        | ARRAY_INVEH_ROT_PITCH
        | ARRAY_INVEH_ROT_AZ
        | MGS_LAT
        | MGS_LON
        | MGS_H
        ;
number   : LONG
        | DOUBLE
        ;

```

CPAAD Novatel Configuration File

The CPAAD Novatel GPS device configuration file is used in the antenna pointing algorithm and defines the orientation of antenna on the aircraft. Use the `cpaad2/Linux/NovatelControl2.py` program to generate the `novatel.cfg` file. An example of a Novatel configuration file is shown below.

```

192.168.1.3
pl_imu_z_rot = -90
array_inveh_x = -0.6096
array_inveh_y = 0.0889
array_inveh_z = 0
array_inveh_rot_roll = -90
array_inveh_rot_pitch = 0
array_inveh_rot_az = 0
mgs_lat = 41.4112906
mgs_lon = -81.8726012
mgs_h = 432

```

CPAAD Antenna Waveform File

The CPAAD antenna waveform file is used to configure the antenna RF devices. An example file is shown below.

```
/*
 * output filename: ../../data/th_0_0_taylor_4_-28_1_1_0_0.cpa
 * date-time: 17-Jun-2019 12:19:49
 * frequency: 14250000000.000000
 * steer_theta: 0
 * steer_phi: 0
 * taper_mode: 4 taylor window with taper_nsl and taper_sll config
 * radius: 0.4064
 * phase_added: 0
 * taper_nsl: 4
 * taper_sll: -28
 * flat: 1
 * null_steer: 1
 * steer_theta_null: 0
 * steer_phi_null: 0
 */
SET 14.25
TR1 TX -6.5 180 HORIZONTAL
TR2 TX -7 180 HORIZONTAL
TR3 TX -10 180 HORIZONTAL
TR4 TX -14 180 HORIZONTAL
TR5 TX -14 180 HORIZONTAL
TR6 TX -10 180 HORIZONTAL
TR7 TX -7 180 HORIZONTAL
TR8 TX -6.5 180 HORIZONTAL
TR9 TX -6.5 180 HORIZONTAL
TR10 TX -8 180 HORIZONTAL
TR11 TX -17 180 HORIZONTAL
TR12 TX -15.5 0 HORIZONTAL
TR13 TX -10.5 0 HORIZONTAL
TR14 TX -15.5 0 HORIZONTAL
TR15 TX -17 180 HORIZONTAL
TR16 TX -8 180 HORIZONTAL
TR17 TX -7.5 180 HORIZONTAL
TR18 TX -22.5 180 HORIZONTAL
TR19 TX -8 0 HORIZONTAL
TR20 TX -3 0 HORIZONTAL
TR21 TX -3 0 HORIZONTAL
TR22 TX -8 0 HORIZONTAL
TR23 TX -22.5 180 HORIZONTAL
TR24 TX -7.5 180 HORIZONTAL
TR25 TX -7 180 HORIZONTAL
TR26 TX -15 180 HORIZONTAL
TR27 TX -9 0 HORIZONTAL
TR28 TX -1.5 0 HORIZONTAL
TR29 TX 0 0 HORIZONTAL
TR30 TX -1.5 0 HORIZONTAL
TR31 TX -9 0 HORIZONTAL
TR32 TX -15 180 HORIZONTAL
TR33 TX -8.5 180 HORIZONTAL
TR34 TX -22 0 HORIZONTAL
TR35 TX -4 0 HORIZONTAL
TR36 TX -0.5 0 HORIZONTAL
TR37 TX -0.5 0 HORIZONTAL
TR38 TX -4 0 HORIZONTAL
TR39 TX -22 0 HORIZONTAL
TR40 TX -8.5 180 HORIZONTAL
TR41 TX -7 180 HORIZONTAL
TR42 TX -11 180 HORIZONTAL
```

```

TR43 TX -15.5 0 HORIZONTAL
TR44 TX -4.5 0 HORIZONTAL
TR45 TX -2.5 0 HORIZONTAL
TR46 TX -4.5 0 HORIZONTAL
TR47 TX -15.5 0 HORIZONTAL
TR48 TX -11 180 HORIZONTAL
TR49 TX -7 180 HORIZONTAL
TR50 TX -10.5 180 HORIZONTAL
TR51 TX -31.5 0 HORIZONTAL
TR52 TX -11.5 0 HORIZONTAL
TR53 TX -11.5 0 HORIZONTAL
TR54 TX -31.5 0 HORIZONTAL
TR55 TX -10.5 180 HORIZONTAL
TR56 TX -7 180 HORIZONTAL
TR57 TX -6 180 HORIZONTAL
TR58 TX -7 180 HORIZONTAL
TR59 TX -8 180 HORIZONTAL
TR60 TX -12 180 HORIZONTAL
TR61 TX -15 180 HORIZONTAL
TR62 TX -12 180 HORIZONTAL
TR63 TX -8 180 HORIZONTAL
TR64 TX -7 180 HORIZONTAL
CMD

```

CPAAD LTC-2986A Code Generation

The CPAAD thermocouples use the LTC-2986A devices for temperature monitoring. The devices are programmed with SPI signals to setup the registers for the thermocouple design and then queried to obtain the temperature data. To effectively program the device, a starter kit is provided with software to generate the thermocouple C program. The starter kit includes a Linduino and device daughter board for thermocouple testing. The thermocouple circuit is laid out with the QuickEval test bench and the thermocouple program is generated. The generated program runs on the Linduino to control the device daughter board. The generated program is then ported to use the libmpsse library on the Jetson TX2. The QuickEval design windows are shown in Figure 9 based on the CPAAD schematics.

CPAAD Antenna Python Programs

The CPAAD antenna software includes two python programs. The `cpaad2/Linux/NovatelControl2.py` program is used to generate the GPS antenna configuration file (`novatel.cfg`). The `cpaad2/Linux/cpaad.py` program is used to import and export CPAAD waveform files (`filename.cpa`). The python programs allow antenna configurations and antenna waveforms to be imported and exported for the laboratory and near field testing facilities. A python waveform example is provided (`cpaad2/Linux/pcpaad.py`) along with the source code files. To use the import/export python program, include the program as part of the python program and execute functions to set, get, encode, decode, read, write and print antenna waveforms. The following example shows some of the import and export operations.

```

import cpaad
default1 = [
    0, 0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 100, 0, 0, 0,
    0, 0, 100, 0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 100,
    0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 68, 0, 0, 0, 0, 0, 68, 0, 0, 0, 0,
    0, 68, 0, 0, 0, 0, 0, 68, 0, 0, 0, 0, 0, 68, 0, 0, 0, 0, 0, 68, 0, 0,
    0, 0, 0, 68, 0, 0, 0, 0, 0, 68, 0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0,
    100, 0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 100, 0, 0,

```

```

0, 0, 0, 100, 0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0,
68, 0, 0, 0, 0, 0, 68, 0, 0, 0, 0, 0, 68, 0, 0, 0, 0, 0, 68, 0, 0, 0,
0, 0, 68, 0, 0, 0, 0, 0, 68, 0, 0, 0, 0, 0, 68, 0, 0, 0, 0, 0, 68, 0,
0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 100, 0, 0, 0, 0,
0, 100, 0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 100, 0,
0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 68, 0, 0, 0, 0, 0, 68, 0, 0, 0, 0, 0,
68, 0, 0, 0, 0, 0, 68, 0, 0, 0, 0, 0, 68, 0, 0, 0, 0, 0, 68, 0, 0, 0,
0, 0, 68, 0, 0, 0, 0, 0, 68, 0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 100,
0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 100, 0, 0, 0,
0, 0, 100, 0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 100, 0, 0, 0, 0, 0, 68,
0, 0, 0, 0, 68, 0, 0, 0, 0, 0, 68, 0, 0, 0, 0, 0, 68, 0, 0, 0, 0,
0, 68, 0, 0, 0, 0, 0, 68, 0, 0, 0, 0, 0, 68, 0, 0, 0, 0, 0, 68 ]

```

```

cpaad.decode( default1)
cpaad.set_shift( cpaad.TR1, "TX") # (off tx rx)
cpaad.set_tx_amp( cpaad.TR1, -31.5) # (-31.5 to 0)
cpaad.set_tx_phase( cpaad.TR1, 360) # (0 to 360)
cpaad.set_polarity( cpaad.TR1, "HORIZONTAL")
cpaad.set_delay( cpaad.TR1, 15) # (0 to 105)
cpaad.write( "null1.cpa")
cpaad.status()
cpaad.read( "null1.cpa")
null1 = cpaad.encode()

```

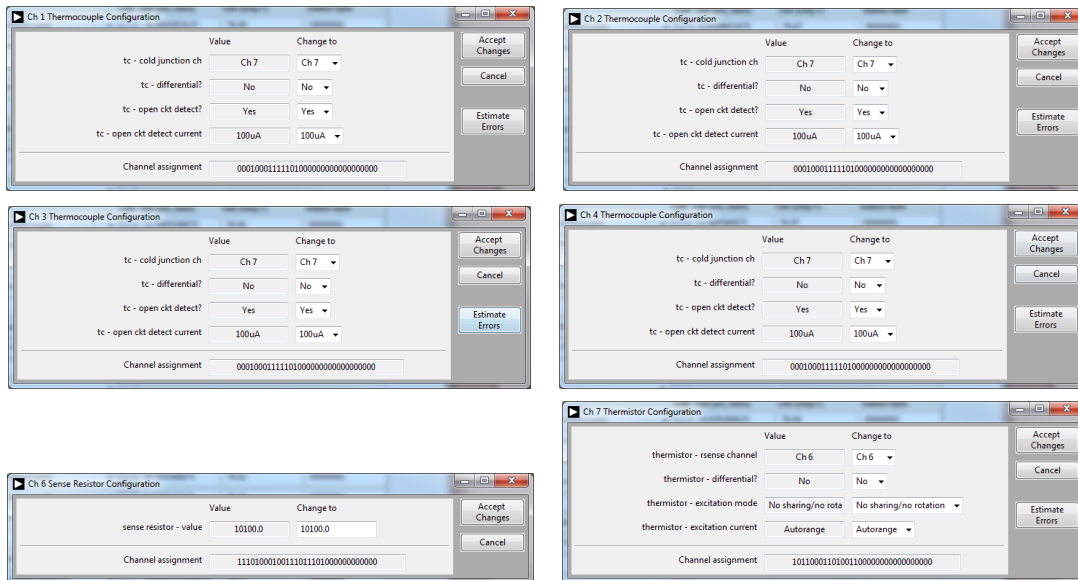
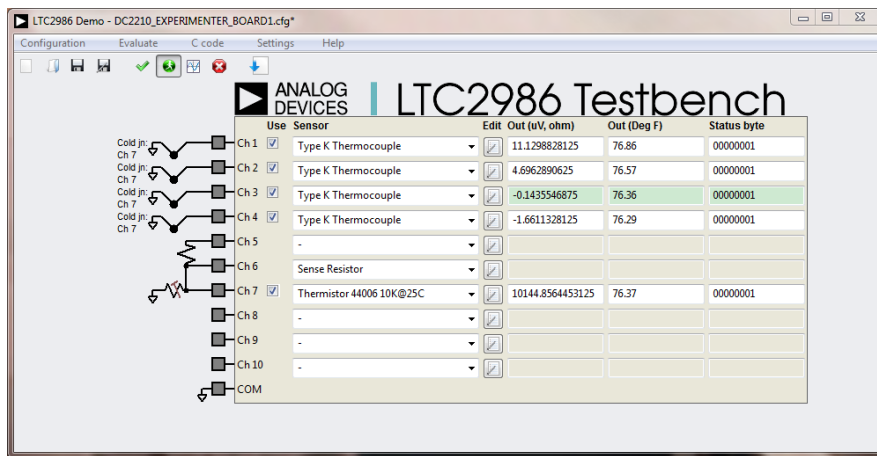


Figure 9.—LTC2986 Testbench Configuration.

CPAAD Antenna Hex Import Format

The CPAAD antenna software includes a hex waveform import utility. The hex format allows antenna waveforms of any length and values to be created for debugging operations. The file is loaded with the load command or entering the filename on the command line. When the filename has the .hex file extension then the file is imported in hexadecimal format. Edit the file to use two-byte hexadecimal format values. An example program is shown below.

```
55 00 00 00 00 00 64 00 00 00 00 00 64 00 00 00
00 00 64 00 00 00 00 00 64 00 00 00 00 00 64 00
00 00 00 00 64 00 00 00 00 00 64 00 00 00 00 00
64 55 55 00 00 00 00 00 44 00 00 00 00 00 44 00
00 00 00 00 44 00 00 00 00 00 44 00 00 00 00 00
44 00 00 00 00 00 44 00 00 00 00 00 44 00 00 00
00 00 44 55 55 00 00 00 00 00 64 00 00 00 00 00
64 00 00 00 00 00 64 00 00 00 00 00 64 00 00 00
00 00 64 00 00 00 00 00 64 00 00 00 00 00 64 00
00 00 00 00 64 55 55 00 00 00 00 00 44 00 00 00
00 00 44 00 00 00 00 00 44 00 00 00 00 00 44 00
00 00 00 00 44 00 00 00 00 00 44 00 00 00 00 00
44 00 00 00 00 00 44 55 55 00 00 00 00 00 64 00
00 00 00 00 64 00 00 00 00 00 64 00 00 00 00 00
64 00 00 00 00 00 64 00 00 00 00 00 64 00 00 00
00 00 64 00 00 00 00 00 64 55 55 00 00 00 00 00
44 00 00 00 00 00 44 00 00 00 00 00 44 00 00 00
00 00 44 00 00 00 00 00 44 00 00 00 00 00 44 00
00 00 00 00 44 00 00 00 00 00 44 55 55 00 00 00
00 00 64 00 00 00 00 00 64 00 00 00 00 00 64 00
00 00 00 00 64 00 00 00 00 00 64 00 00 00 00 00
64 00 00 00 00 00 64 00 00 00 00 00 64 55 55 00
00 00 00 00 44 00 00 00 00 00 44 00 00 00 00 00
44 00 00 00 00 00 44 00 00 00 00 00 44 00 00 00
00 00 44 00 00 00 00 00 44 00 00 00 00 00 44 55
```

CPAAD Novatel GPS Message Format

The CPAAD antenna control software queries the Novatel GPS messages at default IP address 192.168.1.3 port 3005. The software issues the following command and the reply follows. The reply is decoded and the antenna pointing direction is calculated and the software is updated.

```
LOG INSPVAXA ONTIME 2
#INSPVAXA,ICOM5,
0,78.0,FINESTEER
ING,2049,488028.
000,02004000,471
d,14970;WAITING_
AZIMUTH,SINGLE,4
1.41301388275,-8
1.86560429616,24
2.0968,-34.9000,
0.0143,-0.0064,-
0.0050,66.909168
255,-66.17585136
3,-0.000000000,1
.4037,1.1239,2.6
311,0.1678,0.134
3,0.3145,3.0000,
3.0000,180.0000,
01000000,0*ffb91
a3e.
```

CPAAD Jackson Laboratory GPIB Message Format

The CPAAD antenna control software queries the Jackson Laboratory test equipment over the GPIB interface to determine frequency lock status. The software uses /dev/ttyUSB8 for the default device. The sync? command provides the lock status. If the value is 0 then the status is UNLOCKED. If the value is 1 then the status is LOCKED. Below is an example of the communication between the software and the Jackson Laboratory frequency reference.

```
syst:comm:ser:pro off
*IDN?
Jackson Labs, RCM, Firmware Rev 1.2
sync?
1PPS SOURCE MODE : AUTO
1PPS SOURCE STATE : 1PPS
1PPS LOCK STATUS : 0
1PPS OUTPUT ON RESET : OFF
HOLDOVER STATE: ON
LAST HOLDOVER DURATION : 18,1
FREQ ERROR ESTIMATE : 1.00E-08
TIME INTERVAL DIFFERENCE : 0.000E+00
TIME INTERVAL THRESHOLD : 300
HEALTH STATUS : 0x10228
10MHz PLL BW : NARROW
100MHz PLL BW : NARROW
GPIB interface: /dev/ttyUSB8
GPIB baud: 115200
```

CPAAD System Conclusions

The CPAAD system is designed to address affordable autonomous next generation aviation communications. The CPAAD antenna is built out of ultra-lightweight low dielectric polymer aerogels. The conformal design is used to reduce drag and increase simplicity. The CPAAD system uses a hardened NVIDIA Jetson TX2 as the antenna control computer. The CPAAD software integrates sixty-four RF transmit/receive modules, two thermocouples, a frequency synthesizer, a GPS receiver and a frequency reference. The CPAAD software operates the antenna system and provides GUI applications for creating beam forming waveforms and flight testing. The CPAAD software determines the antenna pointing direction and generates the corresponding beam steering antenna waveforms. The CPAAD waveforms files are text formatted and can be imported and exported from the CPAAD system. The CPAAD software and device drivers can be reused on future phased array antenna control systems.

References

1. Design and Development of Aerogel-Based Antennas for Aerospace Applications: A Final Report to the NARI Seedling. August 01, 2014. <https://ntrs.nasa.gov/search.jsp?R=20140017756>.
2. Downey, J.M., B.L. Schoenholz, M.T. Piasecki, and R.J. Kerczewski, "Phased Array Antenna for the Mitigation of UAS Interference," 2018 ICNS Conference, Washington DC, USA, April 2018.
3. P.A. Slater, J.M. Downey, M.T. Piasecki, and B.L. Schoenholz, "Portable Laser Guided Robotic Metrology System," 2019 Antenna Measurement Techniques Association Symposium (AMTA), San Diego, CA, USA, 2019.

