**SATT4 Comms Design Documentation. PSAT-1U** (rev y)21 Apr 2020

Bob Bruninga, WB4APR

Rev y 21 Aug 2020 Clarified all the Voltage Options

Rev x 5 Aug 2019, updated to PSAT1U version [and reversed BANK0 and BANK1 logic to true]

Rev u 29 Nov 2018 added photo of bottom of PCB board and needed 0.01 uF TXD filter capacitor.

Rev s 8 Nov 2018 added H1-16 data select driver of UHF PA to the schematic

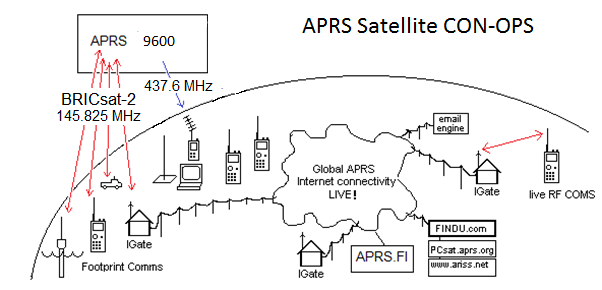
Rev p 21 July 2017 Added HIPWR command to docs

Rev o 3 May 2017 Updates LEDS and power budget. May need much more from USNAP1

Rev n 7 Mar 2016 brings docs current with latest REV 3 board order

The design goal of the SATT4 board is to be a low cost AX.25 and APRS compatible Cubesat comms board for telemetry, command, and control and to encourage Amateur Radio missions on student cubesats compatible with the worldwide network of volunteer ground stations. The comm board usually works on 145.825 MHz half duplex at 1200 baud to provide this compatibility with the APRS global network while also providing for command uplinks and telemetry downlinks. [It can also be tested on the 144.39 MHz terrestrial network with an outside antenna. Packets there will be captured into the APRS.fi web page.] This board also adds an optional CANSAT 9600 baud UHF downlink for higher data rate downlinks.

**The board fits the cubesat-104 Bus standard.**



**The APRS mission** extends the ongoing AX.25 packet radio space relay network pioneered by the US Naval Academy (USNA) since PCSAT-1 in 2001[1] and subsequent spacecraft [2] [3] [4] [5]. These satellites are vital in the amateur satellite service to provide a remote date relay for student and amateur experimenters around the world. This mission also encourages young people to participate in STEM related education with hands-on access to satellites that others can use and fit within the rules of the Amateur Satellite Service.The live data is visible on the links summarized on the USNA satellite downlink page: <http://aprs.org/sats.html>

**Power Options:** There have been so many different uses of the SATT4 and battery boards that extreme caution should be taken on each individual card power arrangement. Originally the system was designed for a 6-7 volt 1.5U cubesat with three 2.4v solar cells per side matched with five NiCd’s for a simple passive charging system requiring no active components. The battery board provided 6-7 volts and the individual boards had 5v regulators. Then the system was re-designed for a 1U system with two solar cells and 3 NiCd’s for a 3.6-4.2 battery system which required a boost-up 5v regulator on the battery board. Then the SATT4 was wired to run directly on the 5v including the watchdog timer. KNOW your board and do not Mix and Match.

**PSAT 1.5U:** For future projects we have gone back to a 1.5U cubesat and redesigned all the boards to work on only that 6-7 V voltage and to remove all the confusing traces and pads that are not applicable. This also involved total redesign of ADCS with new current sensors since the MAX chip is no longer available.

**SATT4 Specifications:** The SATT4 board’s VHF capability is a 2 Watt APRS transceiver capable of operating in the 2 meter amateur radio band. The board’s capabilities include the following:

* Design Operation from a 7 volt bus up to 12 volts max. [5v for the PSAT1U]
* An APRS Digipeater function to relay user packets from uplink to downlink
* Five analog inputs for simple telemetry in a 1-minute beacon
* 8 digital output bits that can be toggled by ground control for spacecraft functions.
* A serial port for uplink and downlink between other boards and the ground
* A CANSAT 9600 baud high speed UHF 2W downlink for data
* A Heartbeat watchdog timer that power cycles the board if Telemetry every fails
* An on/off Output bit and driver for an aux 200 mA load functions\*
* A password protected uplink command capability\*
* A post-launch TX holdoff delay\*

(\*) These items will require future firmware engineering from Byonics.com

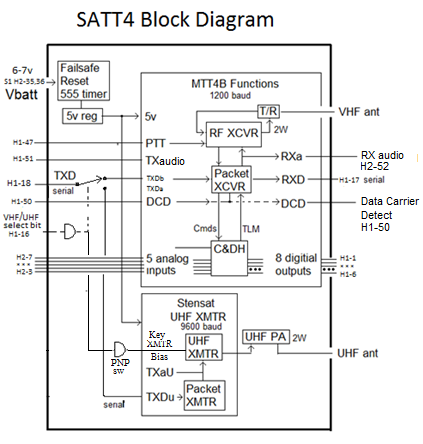


Figure 3: The SATT4 Inputs and Outputs

**The SATT4 Cubesat System:**  The board uses the processor and VHF XCVR of the Byonics.com MTT4B transceiver on 145.825 MHz. Data can be transmitted and received to and from the other cubesat boards via the internal TXD and RXD serial lines for the VHF link and can be transmitted on UHF via a TXD line to the CANSAT exciter for the UHF downlink. The CANSAT module is available from http://www.stensat.org/.

**SATT4 APRS Comms Board:** This comms board evolved from the MTT4B communications board available from Byonics.com. The earlier MT-TT4 was a precursor to the MTT4B and was flown by USNA on the PSAT-1 in May 2015 (and still operational). The MTT4B is the same basic system but with a new XCVR module and a 10W power amp for terrestrial use. In our design, we do not use the 10W PA but set the XCVR to high power (1.5W) with the new HIPWR=1 command. The default is 0.5W in low power. The current MTT4B has all of the functional capabilities that are needed for a Cubesat, however the dimensions of the MTT4B will not fit on the desired 10cm by 10cm satellite. Because of this, a new design was needed where the components of the MTT4B were rearranged to fit on a 10 cm x to cm board, and is shown below in figure 4. In addition, the 9600 baud UHF CANSAT module was included as well. The schematic is shown in Appendix A.

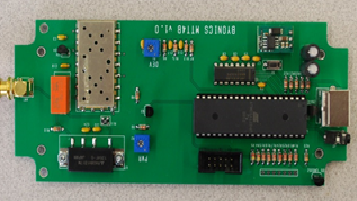




Figure 4. Layout challenge to reduce MTT4B Comm system to the SATT4 Cubesat-104 board

**SATT4 Dual Bank Switching:** The SATT4 can be programmed with two separate operating configurations or banks as detailed in Appendix B. The main use of this feature (on our cubesats) is to enable and disable user digipeating and to change uplink frequencies. The configurations in Bank0 and Bank1 are switched on the condition of the JP1 bit (the 5th Telemetry channel input). The default is Bank-0 if nothing is connected. A pull-down to ground enables Bank-1.

**Using the Bank Switch input also for Temperature:**  But we can still use this A/D input channel for Temperature by using the thermistor as a default pull-down (enabling Bank1 now as the default) and by paralleling the 30k thermistor with a 27.4k resistor to make sure even at low temperatures (high-z) there is still sufficient pull-down to hold BANK1. We need the pull-down because there is a phantom 30k or so internal pull-up on the input to the A/D and so with this combination of fixed pull-up and thermistor pull-down, then we can sense temperature on this channel in this state. This gives us telemetry counts between about 100 to 160 that vary with temperature as shown here.

Ch5 Temperature when $20 bit is OFF

When the $20 command bit goes high, there is a series 8.25k resistor to assure the bit is high and BANK0 is enabled but not so high that we would not get any variation with temperature. In this case, the R1 is 8.25k and R2 is the thermistor and parallel 27.4k voltage divider giving an A/D telemetry count from about 500 to 800 which is also proportional to temperature as shown here.

Temps for Ch3, Ch4 and Ch5y when $20 bit is ON (Ch 5 telemetry above 500). The +/-Z temps on PSAT2 were biased high with 15k for R1 for an obscure reason where 30k would be better.

**Bank 0 - TOCALL=APOFF: Bootup mode with minimum power. Ch5 is valid.**

**Bank 1 - TOCALL=APDIGI: normal, Digipeater ON, TLM Ch5 is meaningless**

To determine the correct thermistor bias resistors on Ch5 to allow the Ch5 input to be used for both commanding and temperature some tests were coinducted. Testing showed that the switchover point is about 460 out of the 999 possible telemetry range (almost half way) and that a 10k pull-up with a 7k pull down would not switch, but a 6k pull down would. We had also measured an internal 30k or so phantom pullup on this bit. To keep the default value low (BANK 1) in all conditions a 27.1k in parallel to the thermistor kept channel lower than the JP1 threshold even when the thermistor was cold (high z Similarly, using an 8.25k pullup when the $20 bit went high would guarantee a count above 500 for all temperature conditions. When no command has been sent, the command bit is floating and so any intermediate channel readings should be ignored.

**Telemetry Temperature Equations**: The conversion equations for the other 30k Thermister channels are shown above for PSAT2 when used with a 15k pullup resistor chosen for a unique condition on PSAT2. This gives less resolution at lower temperatures. It is better to use 30.1k resistors to center the temperature scale on 25 C to get more dynamic range.

Note, these trendline equations only show the first term to a single digit of precision and will lead to very big errors unless the original Excel can be used to reveal at least another one or 2 digits of precision on those higher order terms. A tiny truncation error when raised to the 4th power is very significant.

**SATT4 Telemetry:** The SATT4 board includes an inherent 5 channel analog telemetry capability. These 5 channels report voltages from 0 to 5 volts as a count from 0 to 999. The typical use of these channels are shown here. For Vin on Channel 1, we use a 190k/10k voltage divider to get volts in tenths up to 99.9 volts. Or 10.0k/10.0k for 0-10v range. For thermistors, we use 30k units and 30.1K precision pullups (3012).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Channel | Function | Pin # | Cubesat Bus | Notes |
| Ch 1 | System Voltage | PA7 / 33 | H2-7 R1/R2 On Comm board |  |
| Ch 2 | System Current | PA6 / 34 | H2-6 External |  |
| Ch 3 | Temp-SATT4 CPU | PA5 / 35 | H2-5 thermister on comm brd |  |
| Ch 4 | Temp-EPS | PA4 / 36 | H2-4 External |  |
| Ch 5 | Tbatt and Bnk-Sw | PA3 / 37 | H2-3 External but also bankSW | Shared w Bank switch |

**SATT4 Telemetry Formats:**

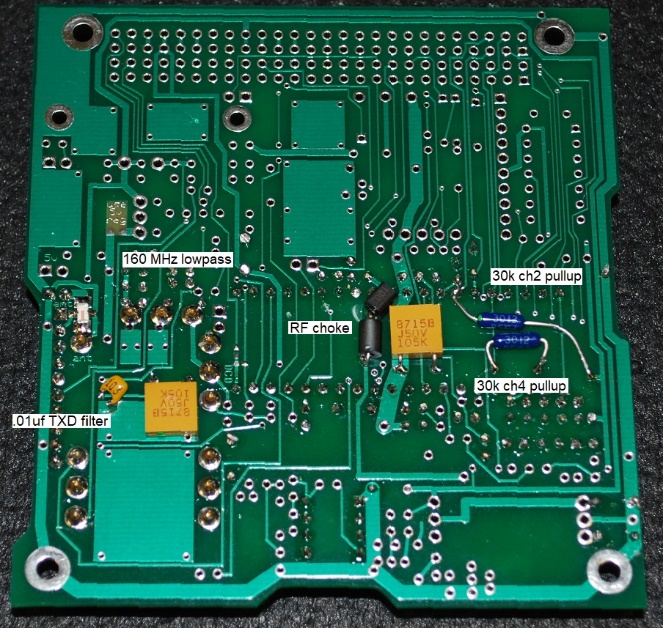
The Telemetry downlink originated from the SATT4 can contain three internal SATT4 generated packets; These are the fixed BText, the fixed LTEXT and the 5 channel Telemetry. In addition, the downlink can contain ANY type of any other packet format sent to it from any other processor on the spacecraft on the TXD line. In addition the downlink will also contain any USER digipeated packets if the digipeater settings in the selected BANK are enabled.

SATT4 telemetry: PSAT-2>APRSAT,ARISS:T#SSS,111,222,333,444,555,10101010

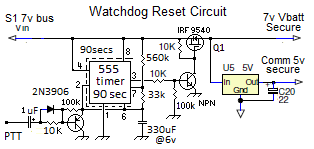
SATT4 BTEXT: PSAT-2>APRSAT,ARISS:any fixed BTEXT you want to go here

SATT4 LTEXT: PSAT-2>APRSAT,ARISS:any fixed LTEXT you want to go here

Other CPU data:: PSAT-2>APRSAT,ARISS:Any other CPU data goes here



**Bottom PCB Modifications:** On the bottom of the board are the RF choke between the CPU power leads, two large 1 uF power bus bypass capacitors and one 0.1 uF filter capacitor necessary to make TTL transmit Data properly received by the CPU. Also there are two 30k thermistor pullup resistors. The lower U shaped one is required for Ch4 XCVR temperature and is simpler to install on bottom of board than on top-under-the-chip. The other one is optional if ch-2 is not used for I-current but is used for an external Thermister on that Icur line.

****

**Watchdog Timer Reset Circuit:** A special circuit is added to the comms board to recover the comms board from SEU events and latch-up and to reboot the comms board via a 2 second power-cycle whenever the 1 minute telemetry is lost. This circuit consists of a 555 timer chip and transistor switch to the input of a DC/DC regulator and can be used up to 12v which is the max rating of the 555 chip. For the PSAT1U design the 5V regulator is on the Battery board.

**5V Regulators:** The SATT4 contains a 555 Watchdog timer circuit and FET switch that will cycle power to the Vbatt bus if the SATT4 one-minute telemetry is lost. All circuit cards in the spacecraft with chips and processors can run from this protected Vbatt power and can be recycled in case of latchup or single-event-upsets in space. A second low-dropout regulator powers the separate higher current requirement of the UHF power amp directly from the battery lines since these passive componnents are keyed on/off and hopefully do not latchup. The exact wiring when using the optional UHF circuit on the PSAT1U has not been confirmed or tested.

**The SATT4’s Power Budget** is shown in Table 1. The SAFE mode currents represent the always-on loads on the spacecraft. The additional columns show the total load current in the additional modes. Currently the budget shows that not all modes can operate full time.

SubFunction Current Duty Safe mode Operate

----------------- ---------- ------- ------------ ----------

SATT4 RX 120 mA 100% 120 mA 120 mA

SATT4 TLM 750 mA 2% 15 mA 15 mA

UHF TX stby 20 mA\*100% 0 20 mA

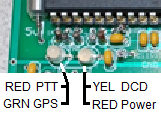
UHF PA stby 400 mA\* 2% 0 8 mA

UHF XMIT 750 mA 2% 0 15 mA

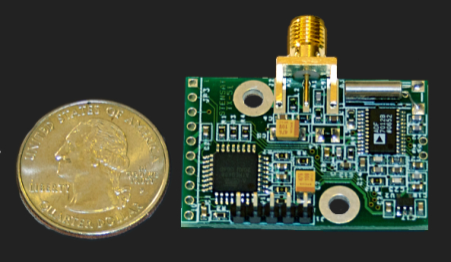
------------------------------------------- ---------- ----------

Totals 135 mA 200 mA

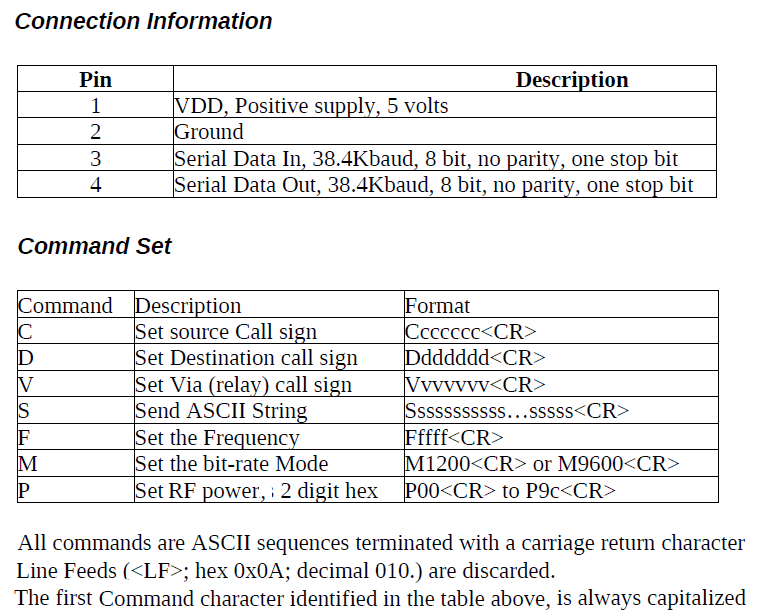
\* only drawn when UHF Select bit is LOW

**LED Indicators:** The SATT4 has the same LEDS as the Byonics MTT4B. The function colors are as indicated here. To save about 5 mA of power, be sure to change the LEDS paramater to OFF in the final flight code at least in the default BANK0 (SAFE mode). Leaving them on in BANK1, which is usually enabled when there is plenty of power, allows you to send ground command and see the LED come on during final integration testing.

**Sun Bit:** To save power, there is a Sun-bit defined on the cubesat104 bus to indicate when the spacecraft is in sun. This can be used as a power-saving technique where applicable. The main power saving techniques in the SATT4, are to 1) bank switch to a lower duty cycle operating scenario, 2) operate with squelch set, and/or 3) put the receiver into to power-saving mode. This mode puts the receiver to sleep for 900ms of every second if nothing is heard for significant savings, but it also guarantees the first packet will not be decoded unless PTT is keyed a second before first transmission. In PSAT2 the bank switching meant turning on and off user digipeating. This bit could be used for example to power down about 10 minutes (TBD) after entering eclipse which should be when the spacecraft will be near local midnight depending on season. But the power saved by using squelch and power-saving are so great that this auto-mode switchign based on the sun bit is not needed. These systems then power back up about ten minutes later (TBD) to be ready at sunrise for user load. This could saves about 10% of user load power (10 out of every 90 minutes) but pales compared to the 70% savings of the power-save mode on the receiver.

****

**Optional High Speed UHF Downlink:** The SATT4 board also includes a UHF 9600 baud downlink CANSAT transmitter made by Stensat with an added UHF power amplifier. The H1-18 TXD data line from the Cubesat is switched from the default 1200 baud VHF transmitter to the UHF 9600 baud transmitter when the H1-16 Select bit is pulled low. The TXD line is TTL level and is always operated at 9600 baud. CANSAT UHF transmitter



**UHF Downlink Format: Note,** The factory assembled CANSATs operate with a TTL input of 38.4 kBps and a channel rate of 1200 baud. We special ordered CANSAT modules set for 9600 baud TTL inputs though the channel rate still defaults to 1200 baud.. The command to change channel rate is M9600. The serial port rate cannot be changed. Any data to be transmitted on the UHF dowlink must have the first character be an “S” (for send) and be terminated with a <CR>. Any other character may be interpreted by the CANSAT as setup commands that can possibly scramble the CANSET settings. The CANSAT module commands are as shown above.

**To Transmit on VHF:** 1) Float or HIGH on H1-16

2) Send any data on the H1-18 serial TTL TXD line

3) End each line with a <CR>

**To Transmit on UHF:** 1) Ground or LOW H1-16

2) Send “M9600” once on H1-18 serial line

3) Send all subsequent

**IMPORTANT:** The H1-16 bit going-low not only selects UHF, but also biases the final Power Amp in preparation for transmitting. Just turning on the bias draws 400 mA or almost 2 watts of wasted power. So be sure to always keep H1-16 selected HIGH when not transmitting. And for safety, in your main code periodically send an H1-16 HIGH just to make sure it stays that way except when actually transmitting. Your software needs to bring this pin low a few ms before sending the transmit data, and needs to remain low several ms after the data transfer is complete, because there is a processing delay in the CANSAT module when preparing the data into an AX.25 packet and then the TXD delay and actual length of transmission. So your timing should estimate the length of this delay based on the length of the transmitted data.

**SATT4B Uplink Command BITS:** The SATT4B has a command BYTE consisting of 8 uplink command configurable output bits ($XX). The $02 (internal) and $20 bank-select bits are already used internally and should not be toggled. The rest can be defined for payload use. NOTE: The output bits are initially tri-state, meaning they are floating at a high impedance. It is best to design all control circuits to default to HIGH to match this condition and to be pulled low when commanded. As soon as any !OUT $XX command is sent, however, they are forced high and low as defined by the bits in the $XX command. So careful thinking is required. IE, a $00 command would turn on ALL FUNCTIONS at once. Conversly, a $FF command is benign because it simply switches the output bits from floating-to their default HIGH state. Also, always keep the $02 bit high since it is used internally. And be careful of the $20 bit because it causes a bank switch and that might change the callsign for subsequent commands. We hope the future flight SATT4 firmware allows these commands to be protected by a password system\*.

**Proper I/O Interface Design:** It is important to think through what these bits are connected to, whether you want them active high or low, and what happens when the SATT4 power is temporarily removed and these bits go low. It is important to isolate them with a series resistance so that the SATT4 5v bus can go low during a reset (an external input held at 5v will hold the CPU still running!) Or what the external circuit on another card does to these as inputs when that board is powered down and drags them low. We finally settled on using these bits as active HIGH with pull down resistors to solve most of these issues.

SATT4 Command !OUT $XX bit designations **–**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Function | Bit | MTT4-B  I/O Pins | SATTT4B  Functioin | Wire  Use | Fig LLL. I/O Pin diagram | BS2  Pin |
| Tristate, floats before first cmd |  |  |  |  |  |  |
| $00 all functions off |  |  |  |  |  |  |
| $01 TBD | 1 | 22 / PC0 |  |  | 1 | 8 |
| $02 SATT4 intl (xcvr setup) | 2 | 23 / PC1 | Sets RF Freq |  | 2 | 9 |
| $04 TBD | 3 | 24 / PC2 |  |  | 3 | 10 |
| $08 TBD | 4 | 25 / PC3 |  |  | 4 | 11 |
| $10 AUX board PwrON | 5 | 26 / PC4 | Voice Pwr-on |  | 5 | 12 |
| $20 Bank switch (only) | 6 | 27 / PC5 | Voice Audio |  | 6 | 13 |
| $40 TBD | 7 | 28 / PC6 |  |  | 7 | 14 |
| $80 TBD | 8 | 29 / PC7 |  |  | 8 | 15 |

**SPECIAL FIRMWARE REQUIREMENTS FOR SATT4**: These ideas will hopefully be negotiated with Byonics to make the SATT4 not only more useful for space but also for general users of the MTT4B product.

1. A launch TX delay? (Can fix w TLM and DIGI commanding) – FCC REQUIRED
2. Possible on/off BIT control of the GPS switch - nice
3. Possible usage of the PD7 bit? And the PB4, 5 and 6 bits? - nice
4. Possible XCVR settings for PL, squelch or power saving? - DONE
5. Password “abc” string protection on the !OUT $XXabc cmd - GOOD FOR ALL
6. De-conflict use of PA3 analog input for Bank Switching - NEEDED
7. Extension of !OUT $XX commands to include:

$K0/$K1 for setting bank 0 or 1

$T0/$T1 for setting Telemetry On/Off, and timing

$U0/$U1 for turning on and off User digipeating

$S0/$S1 for turning On OFF the Status Beacon

$G0/$G1 turning On/Off the GPS power control bit

**Configuring the SATT4 Settings:**

**Serial Port**: A big difference between the SATT4 and the stock MTT4B is the absence of a MAX232 TTL to RS-232 conversion chip. This chip is noisy and reduces RF receive performance by as much as 8 dB, so it is not used on the SATT4. As a result, the serial TXD line on the SATT4 is TTL sense-and-levels whereas on the MTT4B they are RS232 sense and levels (inverted from each other).

**GSE and Configuration Connector:** We use an RS232 to TTL adapter to connect to the GSE (Ground Service Equipment) pins on the SATT4 for working with a PC. This connector can also jumper the TXD and RXD to/from the 104 bus on other pins so they can be separately used for troubleshooting. For flight, two jumper wires are used on the flight GSE plug to connect the SATT4 serial TXD and RXD to the other board’s TXD’s and RXD’s. Notice that all boards can monitor the SATT4’s RXD. But if they want to transmit, they should be hard pulldowns and light pullups so these multiple drivers are not fighting each other.

**Using the SATT4 Bus serial pins when boards are stacked:** Instead of using the ten pin GSE connector on the edge of the SATT4 board that is inaccessible when stacked, the serial bus pins can be used with the right jumpers. First, make a 4 pin header with the TXD and RXD loop-back jumpers for pins 3,4,5,6 of the edge connector, and then ground the TXDB bus line at its 2-way-3pin select location to supply a ground, and then run a jumper from 5V on the 74HC00 chip to the top of the 3rd diode to the right to use the VHF/UHF select pin to supply 5v to the above TTL to RS232 adapter. This allows a 6 pin header to plug the TTL adapter into the 6 pins of the header. Notice this trick cannot be used if the optional 9600 baud transmitter is used.

**Dual Ports:** Also notice that the MTT4B and SATT4 actually have two serial ports. The A port is used for firmware loading and general serial communications and the B port was designed in the original MTT4B for connection of a 4800 baud GPS. GPS is not used on a cubesat, so this second port is configured to operate at 9600 baud as a second serial text data line using the BMODE = TEXT and BBAUD = 9600 commands (or 19200 if desired).

To configure the SATT4, connect a PC serial port via an RS232 to TTL converter to the TXD and RXD pins on the GSE connector 3rd and 4th pins up from the bottom (pin 2 is Gnd) or to the special 6 pin header noted above. This is Port A. The converter we used also needs 5v to power it so the 7th pin up provides that power. Prepare your terminal program to operate at 19.2 kbps, since the SATT4 always begins at 19200 for possible configuration-mode during the first 3 seconds. (Use PuTTY). After the initial 3 second optional period for configuration, and no action is taken, then the SATT4 will switch to whatever baud rate is configured in the ABAUD or BBAUD parameters (often 9600). But for simplicity and student labs, we now keep both at 19200 unless chosen otherwise.

Power up the SATT4 board and within the first 3 seconds you will see the message “Hit ESC three times to go to setup”. If you do that correctly, you will get the SATT4 prompt. At this point, you can type any of the commands (without their parameters) as shown in Appendix B to see the current settings. Be careful. Any change you make will be remembered.

Once you have finished checking or changing parameters, you can cycle power to return to normal operations or you can type the QUIT command. Remember, during normal operations the baud rate will be whatever is in the ABAUD and BBAUD commands. The 19.2 kBaud is only required for entering the settings mode in the first 3 seconds.

A complete list of all commands is in the byonics.com doucmentation for the MTT4B or the TinyTrack 4.

References:

[1] <http://aprs.org/pcsat.html> The original USNA APRS Satelliet PCSAT-1

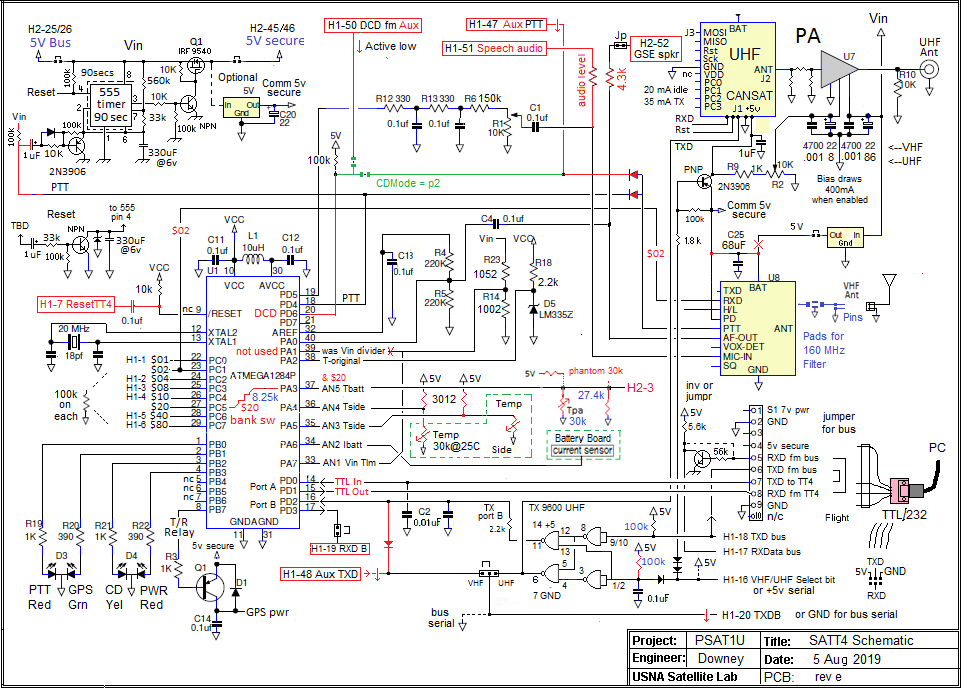
[2] <http://aprs.org/pcsat-2.html> PCSAT-2 flew attached to the ISS in 2006

[3] <http://aprs.org/ande.html> ANDE Deployed form the ISS in 2006

[4] <http://aprs.org/raft.html> RAFT Deployed from the ISS in 2006

[5] <http://aprs.org/psat.html> PSAT-1 deployed in May 2015

Appendix A: This does NOT have the many changes for the final USNAP1 with UHF



**Appendix B**. Setup and Programming commands for SATT4 Processor v0.73. The SATT4 has two banks for two different flight configurations. Normally, the default is BANK0, but when sharing the bank-switch bit for another channel of Temperatuyre telemetry, the bit is biased low with the thermistor so that the BANK 1 is the default. To switch banks, the !OUT $20 bit must be commanded high.

Command Default BANK0 APRS Mode BANK1 (DEFAULT) same as Bank0 except…)

---------------- --------------- ------------------------------- ----------------------------------------------

MYCALL NOCALL SATX SATX-1 (\* means different)

AMODE Text text text

ABAUD 19200 19200 19200 (or can be 9600)

ALTNET APTT4 APDIGI APOFF

ALIAS1 TEMP APRSAT none

ALIAS2 % ARISS none

ALIAS3 % WIDE none

BANK 0/1 0 0 1 🡸 set during programming

BMODE GPS text text

BBAUD 4800 9600 9600

BNKMODE 0 1 1 🡸 uses JP1 for bank switching

[uses PA3, CPU pin 37, J3?]

BTEXT >TT4 alpha >DIGI is on >145.980MHz No Digi.

BPERIOD 0 153 every 2m 33s 240 once every 4 minutes

CDMODE Tones Pin2 Pin2 <uses J1 pin 2 [GND means busy] >

<to holdoff packet while voice is Xmitted>

DIGIID true TRUE TRUE <inserts MYCall in digipeated packets>

DIGIMY false TRUE FALSE

DUPETIME 30 10 10

HEADERLN false false false

LEDS true FALSE for flight false for flight

MSGCMD false true true <allows !OUT $xx commands >

NODISP false true true <Required for $xx PortC msg commanding>

PATH1 WIDE1-1 ARISS (via ISS) ARISS

PATH2,3 WIDE2-1 (none) none

PKTICOM true true true <Enables received packets to PC text port>

PKTOCOM False true true <Lets CPUs see our TX packets also>

PPERIOD 0 0 0 <posit Prd (In sec), not used

RXAMP 5 5 5 5 <RX gain. Use MONITOR mode to adjust>

RXFREQ 144.39 145.825 145.980 < for DTMF uplinks on PSAT2)

RXSQUELCH 0 2 2 (higher to assure silence and save 10 mA)

RXMODE ? 1 1 Auto Power Save (sleeps RX, saves 40 mA)

TELHIRES true TRUE TRUE <enables 0-999 values>

TELVOLT true False false

TELTEMP true False false

TELREAD true False false <Turns off human readable telemetry>

TPERIOD 0 60 60 Telemetry required for 555 watchdog timer

TXD 40 25 25 <20 = about 150ms> CHECK? Seems too fast

TXLEVEL 128 128? 128 <TBD for flight>

TXTWIST 50 50? Is now zero? 50 <TBD for flight>

TXFREQ 144.39 145.825 145.825

RXSQUELCH 1 2 2

HIPWR 0 1 1 sets power to 1W not 0.2W

Appendix C

The definitions of the pins of the Cubesat bus used by the SATT4 are indicated in the notes across the top. The command bits are on the left. Then the analog bus pins, some serial data pins then the battery and separation switch bus. Finally, on the right are four User defined pins and the six special user pins used for our particular PSAT-2 which have to do with sharing data and TX audio with the DTMF/Voice system. This does not have the final arrangements used for UHF on USNAP1.

