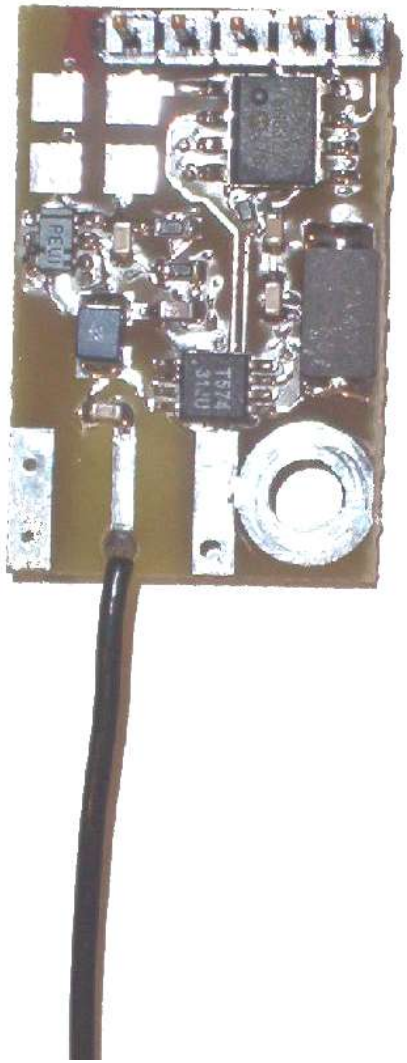
Section 5
Radio Communications

In this section, you will learn about sending telemetry data using a radio transmitter and a ground station.

You will learn how to write software to send telemetry.
You will learn how to set up a ground station and collect the telemetry.

RF Communications

- The transmitter module consists of a processor and a transmitter integrated circuit.
- The processor collects the data from the satellite computer and generates an AX.25 protocol data stream. The data stream is sent to transmitter integrated circuit as two tones, 1200 Hz and 2400 Hz, which represents a logic '1' and '0'. The RF transmitter IC uses the tones to FM modulate a 433 MHz carrier signal. The modulated carrier signal is transmitted on a wire antenna.
- The data is sent at a rate of 1200 bits per second. This is not a fast rate. The rate was chosen because the amount of data to be transmitted is small and the transmitter circuitry can be kept simple. Higher data rates require more complex circuitry and a more complicated ground station.



Communications Protocols

- Most communications use some type of protocol. A protocol is basically a method of formatting information and a communication method. A data structure is used that contains information for addressing a receiver, identifying the transmitter, identifying the type of data, and error checking and correction.
- A protocol can also include sequence of operations. For instance, there can be a protocol for requesting a connection, acknowledging receipt of data, request for retransmission, etc.
- A protocol can be a simple as a description of the transmission format.
- For example:

Header 8 bits	Data bytes (32) 32 bytes	Checksum 16 bits

- The protocol on the right is nothing more than a format. The first eight bits are the header which would be used by the receiver to detect the start of a message. The next thirty-two bytes are the data. The last sixteen bits are for the checksum which is used by the receiver to determine if all the data was received without error.

AX.25 Protocol

- The AX.25 protocol is a digital communications protocol for amateur radio. It has been modified from the X.25 protocol to support amateur radio call signs. The CanSat uses a a subset of the AX.25 protocol called the UI frame. AX.25 protocol is used for another type of communications called APRS, Automatic Position Reporting System. APRS uses the AX.25 protocol to transmit bursts of data. It is widely used in the amateur radio community for sending weather reports, position coordinates and other short sets of data. The CanSat uses the APRS format.

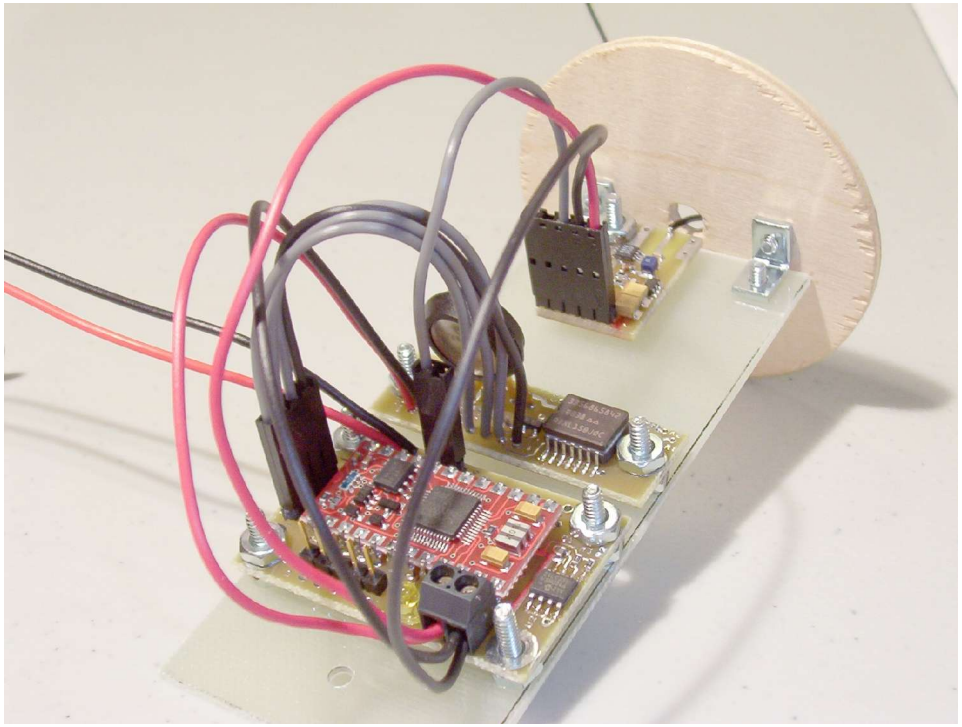
The AX.25 Frame

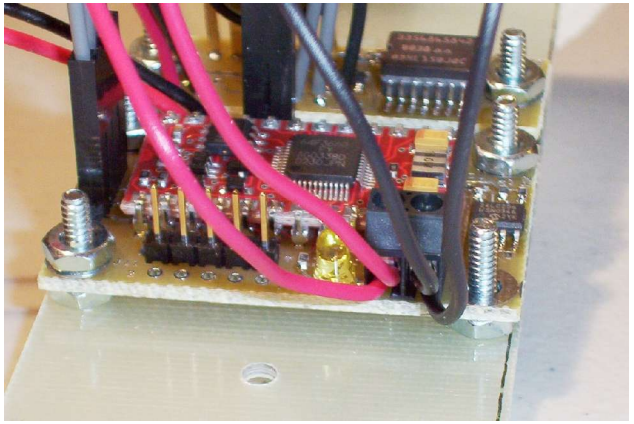
All APRS transmissions use AX.25 UI-frames, with 9 fields of data:

AX.25 UI-Frame Format								
Flag	Destination Address	Source Address	Digipeater Addresses (0-8)	Control Field (UI)	Protocol ID	Information Field	FCS	Flag
Bytes: 1	7	7	0-56	1	1	1-256	2	1

Installing the Transmitter

- Attach the wire harness to the transmitter as shown in the top picture.
- Connect the two power wires to the same terminal block on the processor module. The wire wire connects in the same place as the sensor board power wire. The black wire connects to the other terminal as shown in the bottom picture. Do not connect the wires in backwards. Damage can occur.
- Connect the two pin connector to the processor module as shown in the top picture.





Communication Signals

- Radio communications allows information to be conveyed using electromagnetic waves. The simplest method is Morse code. A radio signal is turned on and off at certain times to generate a pattern that conveys information. Morse code uses short and long radio pulses.
- More complicated communications use various modulation techniques. Most have heard of AM and FM radio. AM radio modulates a radio signal by adjusting the amplitude of the radio signal. FM radio modulates by shifting the frequency of the radio signal.
- Data communications can use a combination of AM and FM modulation. Using the combination of modulation helps increase the amount of information sent.
- The CanSat uses FM modulation. It modulates the frequency of the radio signal with two different audio tones similar to FM radio.

Communications System

- To send data to the transmitter module, the following command is used:

serout datapin,baudmode,[“S”,var1,var2,var3,…,varN,13]

- There are several parts to the above command:
  - Datapin specifies the pin for the data signal. P15 is used.
  - baudmode specifies the baud rate of the serial port. 'I1200' is used
  - Var_ is the list of variables to send to the transmitter module.
  - The “S” tells the transmitter to transmit the data. The 13 tells the transmitter that there is no more data.

How to use the Transmitter

- One of the first things needed to be done with the transmitter is setting the call sign. All call sign is used to identify the transmitter. An example is “KN4JHF”.
- To load the trasnmitter module with a call sign, commands have to be sent to the transmitter. The command format is “Wxxnn” where xx is the character position and nn is the hexidecimal value of the character. This is how the transmitter module works.
- The following command statements set the call sign to “KN4JHF”.

```
serout P15, i1200,[“W002B”]

serout P15, i1200,[“W012E”]

serout P15, i1200,[“W0234”]

serout P15, i1200,[“W032A”]

serout P15, i1200,[“W0428”]

serout P15, i1200,[“W0526”]
```

- The following page lists the ASCII table. It shows the character, the decimal value, octal value and hexidecimal value.

ASCII Table

Char	Dec	Oct	Hex	Char	Dec	Oct	Hex	Char	Dec	Oct	Hex	Char	Dec	Oct	Hex
(nul)	0	0000	0x00	(sp)	32	0040	0x20	@	64	0100	0x40	`	96	0140	0x60
(soh)	1	0001	0x01	!	33	0041	0x21	A	65	0101	0x41	a	97	0141	0x61
(stx)	2	0002	0x02	"	34	0042	0x22	B	66	0102	0x42	b	98	0142	0x62
(etx)	3	0003	0x03	#	35	0043	0x23	C	67	0103	0x43	c	99	0143	0x63
(eot)	4	0004	0x04	$	36	0044	0x24	D	68	0104	0x44	d	100	0144	0x64
(enq)	5	0005	0x05	%	37	0045	0x25	E	69	0105	0x45	e	101	0145	0x65
(ack)	6	0006	0x06	&	38	0046	0x26	F	70	0106	0x46	f	102	0146	0x66
(bel)	7	0007	0x07	'	39	0047	0x27	G	71	0107	0x47	g	103	0147	0x67
(bs)	8	0010	0x08	(	40	0050	0x28	H	72	0110	0x48	h	104	0150	0x68
(ht)	9	0011	0x09	)	41	0051	0x29	I	73	0111	0x49	i	105	0151	0x69
(nl)	10	0012	0x0a	*	42	0052	0x2a	J	74	0112	0x4a	j	106	0152	0x6a
(vt)	11	0013	0x0b	+	43	0053	0x2b	K	75	0113	0x4b	k	107	0153	0x6b
(np)	12	0014	0x0c	,	44	0054	0x2c	L	76	0114	0x4c	l	108	0154	0x6c
(cr)	13	0015	0x0d	-	45	0055	0x2d	M	77	0115	0x4d	m	109	0155	0x6d
(so)	14	0016	0x0e	.	46	0056	0x2e	N	78	0116	0x4e	n	110	0156	0x6e
(si)	15	0017	0x0f	/	47	0057	0x2f	O	79	0117	0x4f	o	111	0157	0x6f
(dle)	16	0020	0x10	0	48	0060	0x30	P	80	0120	0x50	p	112	0160	0x70
(dc1)	17	0021	0x11	1	49	0061	0x31	Q	81	0121	0x51	q	113	0161	0x71
(dc2)	18	0022	0x12	2	50	0062	0x32	R	82	0122	0x52	r	114	0162	0x72
(dc3)	19	0023	0x13	3	51	0063	0x33	S	83	0123	0x53	s	115	0163	0x73
(dc4)	20	0024	0x14	4	52	0064	0x34	T	84	0124	0x54	t	116	0164	0x74
(nak)	21	0025	0x15	5	53	0065	0x35	U	85	0125	0x55	u	117	0165	0x75
(syn)	22	0026	0x16	6	54	0066	0x36	V	86	0126	0x56	v	118	0166	0x76
(etb)	23	0027	0x17	7	55	0067	0x37	W	87	0127	0x57	w	119	0167	0x77
(can)	24	0030	0x18	8	56	0070	0x38	X	88	0130	0x58	x	120	0170	0x78
(em)	25	0031	0x19	9	57	0071	0x39	Y	89	0131	0x59	y	121	0171	0x79
(sub)	26	0032	0x1a	:	58	0072	0x3a	Z	90	0132	0x5a	z	122	0172	0x7a
(esc)	27	0033	0x1b	;	59	0073	0x3b	[	91	0133	0x5b	{	123	0173	0x7b
(fs)	28	0034	0x1c	<	60	0074	0x3c	\	92	0134	0x5c	|	124	0174	0x7c
(gs)	29	0035	0x1d	=	61	0075	0x3d	]	93	0135	0x5d	}	125	0175	0x7d
(rs)	30	0036	0x1e	>	62	0076	0x3e	^	94	0136	0x5e	~	126	0176	0x7e
(us)	31	0037	0x1f	?	63	0077	0x3f	_	95	0137	0x5f	(del)	127	0177	0x7f

How to use the Transmitter

- Here is an example on how to use the transmitter

```
serout P15,i1200,[“S”,“this is a test”,13]
```

- The above instructions send a sentence to the transmitter.
- The transmitter can only transmit ASCII values. These are values that represent the numbers and letters that can be displayed on a terminal. In order to do that, any variables sent must have a modifier to convert it to ASCII.

```
Temp var word

Temp = 254

serout P15,i1200,[“S”,dec temp,13]
```

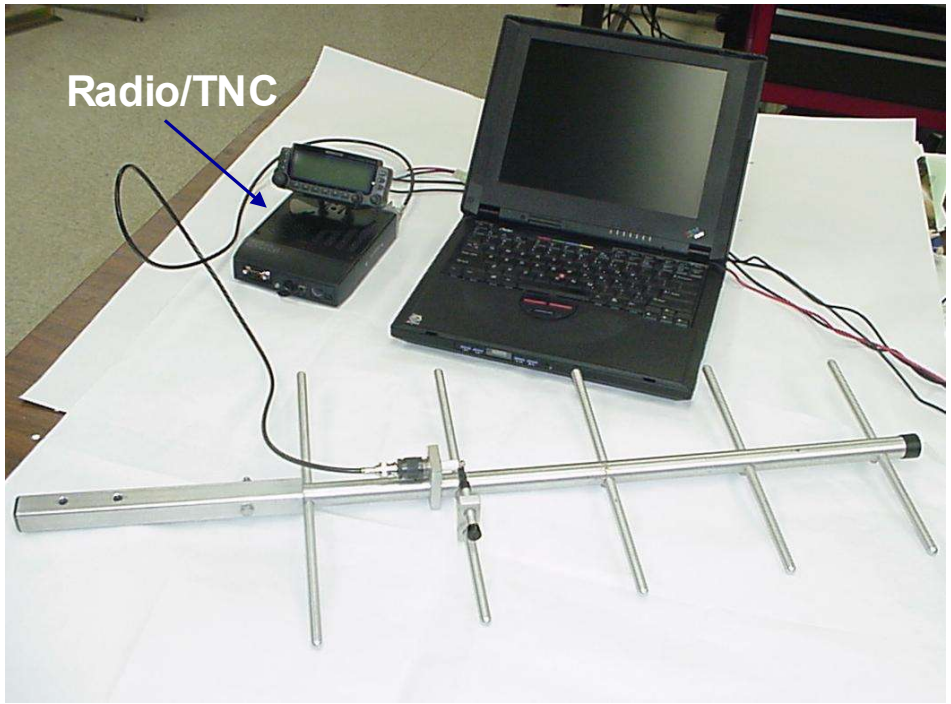
- The above sends out a the variable temp as a decimal number. Just like the debug command, you can add many variables in the bracket separated by commas.

Experimenting with the Transmitter

- Now it is time to experiment with the transmitter. Use the program to read the temperature sensor and insert the serout command to send the temperature data to the transmitter.
- Set up the ground station to receive and record the transmission. The instructions follow.

Ground Station

- The ground station is where the data is collected from the CanSat. The ground station consists of an antenna, a receiver, a TNC, and a computer.
- The antenna is a yagi type antenna. It is designed to focus the antenna in one direction. This focusing adds signal gain to the antenna. It's like an amplifier. The yagi antenna used for the ground station doubles the signal level received. When the CanSat is launched in the air, you need to point the antenna in the direction of the CanSat. If you point away, you may lose the signal.
- The receiver is a radio that detects and amplifies the signal. It also demodulates. The modulation used is FM or frequency modulation. The radio demodulates the signal and generates and audio frequency signal which is sent to the computer.
- The computer uses a program to extract the telemetry from the audio signal. The telemetry data can be stored for processing later.



The picture above shows the antenna and receiver.
The receiver has a built in TNC.
The radio connects to the laptop with a serial cable.

Communications Exercise

- Write a program to read the sensors, calculate the pressure and temperature, and transmit the results. Insert a delay so that transmission is not continuous.
- Since there is one receiver, only one group can test their software at a time. Turns will be taken.

Output Format

Format the output as follows:

T1 Pressure Temperature

T1 is for the team number. Each team will be a different number.
Pressure – put the pressure value here.
Temperature – put the temperature value here.

Example:

T1 102.123 24.123

Summary

At the end of this section, you should know what telemetry data is and how to send telemetry. You should know how to operate a ground station.