TAPR™

Tucson Amateur Packet Radio
8987-309 E. Tanque Verde Rd #337
Tucson, Arizona • 85749-9399
Office: (940) 383-0000 • Fax: (940) 566-2544
Internet: TAPR@TAPR.ORG • www.tapr.org
Non-Profit Research and Development Corporation

APRS MIC-E

TAPR APRS™ MicEncoder™ Manual

The information contained in this document has been carefully checked and is believed to be entirely reliable. However, no responsibility is assumed for inaccuracies. Tucson Amateur Packet Radio Corporation (TAPR) reserves the right to make changes in any products to improve reliability, function or design without obligation to purchasers of previous equipment. TAPR does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey license under its patent rights or the rights of others.

Contents

APRS and Mic-Encoder are trademarks of Bob Bruninga, WB4APR. Parties interested in commercial developments and commercial sales of the MIC-E both within and outside of the amateur community should contact Clement Engineering.
http://www.agrelo.com/clement.html

Quick Start

If you want to get a quick start, here are the sections that you MUST read and complete

- Assembly Notes
- Check all on-board jumpers
- Build Radio Interface Header and set PTT jumper
- Connect your GPS
- Assembly of Unit
- Program/Setup (DOS or Mac software)

INTRODUCTION

As mobile GPS/APRS continues to grow, we can eliminate the need for every mobile to have a TNC, digital radio, and second antenna by simply integrating the position report into a very brief tone burst at the end of a voice transmission over any two-way radio. The APRS™ MIC-E™ is this solution. With the MIC-E™, no additional hardware is required in the vehicle, other than a GPS unit. The system not only reports position and vehicle type, but also one of 7 canned messages and 4 analog telemetry values.

Your APRS™ MIC-E™ Encoder Unit will provide you a quick, easy means of tracking your mobile operations when properly interfaced with your existing voice radio. Before operating your MIC-E on the air, you should inquire about APRS operations in your immediate area. Check 145.79 MHz (U.S.), 144.39 Mhz (Canada), and some areas may also use 145.01 MHz. By contacting APRS users in your local area you may find particular voice repeaters that already support MIC-E operations with a cross-band link to the local APRS network.

In order to make the APRS LOCATOR SYSTEM practical, the MIC-ENCODER has been designed with several design concepts:

- Interfaces to UNMODIFIED radios via the MIC connector
- Uses standard AX.25 for compatibility with existing TNC's
- Compresses position report into about 0.3 seconds
- Uses low enough in power to be powered from the MIC jack
- Accepts the readily available NMEA output from GPS receivers
- Optionally provides 4 analog channels for telemetry

The result is a 1200 baud position report compressed to 32 bytes including beginning and ending FLAGS. This equates to about 320ms, including CALLSIGN, DIGIPEATER PATHS, and a minimum message capability. Plus, the packets are still receivable on ANY AX.25 TNC.

The MIC-E™ (Mic Encoder) installs between your radio mic and radio and allows your GPS unit to transmit APRS AX.25 frames at designated intervals without needing a TNC! Very useful in many settings. You can listen to a discussion of the MIC-E by Bob Bruninga, WB4APR, while Bob uses one or read over the text about the APRS MicEncoder (http://www.tapr.org/tapr/html/virtual.html).

The heart of the MIC-E™ is a PIC chip processor that is now sold commercially by Clements Eng (http://home.navisoft.com/agrelo/clement.htm). In 1996, Gwyn Reedy, W1BEL, President of PacComm (http://www.paccomm.com) approached TAPR about handling the amateur version in a semi-kit solution. A semi-kit means that the board has already been built, but interface wires, box, and other items will still need to be configured and assembled for proper use by the owner.

The TAPR APRS MIC-E is the result of the collaborative process begun in October 1996 and finished in May of 1997.

ACKNOWLEDGMENTS

The Tucson Amateur Packet Radio (TAPR) APRS™ MIC-E™ Kit was made possible by the pioneering efforts of Bob Bruninga, WB4APR, Will Clements, N3XLM, Gwyn Reedy, W1BEL, Greg Jones, WD5IVD (TAPR Project Manager) and the active participation of a number of Amateur Radio operators and organizations. TAPR wishes to acknowledge their contribution to this effort.

Testers

Alan K. Unangst, WC7R, Prescott, AZ
Mike Parker, KT7D, Tucson, AZ
Don Hilman, VE7FBI, Victoria, B.C.
Steven Boyle, KD6WXD , Mountain View, CA
Cliff Buttschardt, K7RR, Morro Bay, CA
Stan Horzepa, WA1LOU, Wolcott, CT
William Covey, W1GTT, East Lyme, CT
Sam Guccione, K3BY, Camden, DE
Neil Lauritsen, KA3DBK, Clearwater, FL
Rich Garcia, N2CZF, Jupiter, FL
Steve Dimse, K4HG, Summerland Key, FL
Ralph Fowler, N4NEQ, Kennesaw, GA
David Chesser, KA9NHL, Dekalb, IL
Henry Van Bogaert, N9WMM, Midlothian, IL
Larry Keeran , K9ORP, Hudson, IL
Don Pfister, KA0JLF, Overland Park, KS
Tom Kinahan , N1CPE, Westboro, MA
Kevin Kelly, N6QAB , Lusby, MD
Will Clement, N3XLM, Arnold, MD
Bob Bruninga, WB4APR, Glen Burnie, MD
Jonathan R Brockmeier , KB8SWB, Holland, MI
Tony Drumm, AA0SM, Rochester, MN
Rich Josephson, WA0UNS, St. Cloud, MN
Jim Duncan, KU0G, Kansas City, MO
Dale Lam, WA0NKE, Kansas City, MO
Wilbur Goll, W0DEL, Kansas City, MO
Bob Derderian, N2IPH, Bridgeton, NJ
Mark Sproul, KB2ICI, North Brunswick, NJ
Bill Healy , N8KHN, Incline Village, NV
Alan Crosswell, N2YGK, Briarcliff Manor, NY
George V. Chapek, N2AIG, Scotia, NY
Michael E. Young, WB8CXO, Munroe Falls, Ohio
Steve Caesar, AH7E, West Carrollton, Ohio
Mark Humphrey, KE3XY, Chester Springs , PA
Gould Smith, WA4SXM, Knoxville, TN
Mark Endicott, WB0NOO, Nashville, TN
Ron Parsons, W5RKN, Austin, TX
Bob Winingham, KC5EJK, Dallas, TX
Harry Burford , KA0TTY, Coppell, TX
Mike Heskett, WB5QLD, Hurst, TX
Frank McJunkins, K7RSD, Seattle, WA

PARTS LIST

Please verify that all parts are present, checking the space ( ) provided as you locate and verify it against this list. You may wish to take this opportunity to sort the parts into a compartmented container, such as an egg carton, muffin tin, or other container as you inventory them. This will aid you in building.

( )	1	MicEncoder Circuit Board
( )	1	Assembly/Operations Manual
( )	1	MIC-E Software Disk (1.44M)

Mic Personality Module

to configure the microphone jack for your particular radio includes:

( )	2	16 Pin Machine Screw Dip Sockets
( )	1	1 ft. section of Olive Green 28 AWG Wire

Microphone Wiring

to wire a panel mount 8 Pin Microphone Connector to JP10 on the circuit board:

( )	1	8 Pin Panel Mount Microphone Connector
( )	1	3 in. section of RED 24 GS Stranded Wire
( )	1	3 in. section of GREEN 24 GS Stranded Wire
( )	1	3 in. section of YELLOW 24 GS Stranded Wire
( )	1	3 in. section of ORANGE 24 GS Stranded Wire
( )	1	3 in. section of BROWN 24 GS Stranded Wire
( )	1	3 in. section of WHITE 24 GS Stranded Wire
( )	1	3 in. section of BLUE 24 GS Stranded Wire
( )	1	3 in. section of BLACK 24 GS Stranded Wire
( )	1	3 in. section of 1/16 in. Heatshrink
( )	1	Plastic plug for front mic connector hole

Cabinet Enclosure

( )	1	Black Anodized Aluminum Cabinet
( )	1	MicEncoder Front Panel
( )	1	MicEncoder Rear Panel
( )	4	Black Self-tapping Screw
( )	4	Black Rubber Feet

Connectors

( )	1	3 ft. 8 Pin RJ-45 Modular Connector Cable for MicE to radio mic jack connection
( )	1	8 Pin Machine Screw Dip Socket prototyping header for APRS Telemetry System
( )	2	3.5mm Stereo Plug
( )	1	8 Pin Male Cable Mount Microphone Connector to use with RJ-45 modular connector cable
( )	1	2.1mm Power Plug to apply power to MicE, center positive

GPS Connectors

( )	4	Nylon Spacer
( )	4	4-40 Machine Nut
( )	4	4-40 Nylon Machine Screw

Final Assembly Items

( )	1	Plastic Plug for Rear Panel BNC Opening
( )	2	10mm Hexadecimal Switch Knob for MSG and Path Switches
( )	2	10mm Hexadecimal Switch Knob Cap
( )	1	Arrow Trimpot Knob for Period Adjust Trimpot
( )	1	Arrow Trimpot Knob Cap

ASSEMBLY

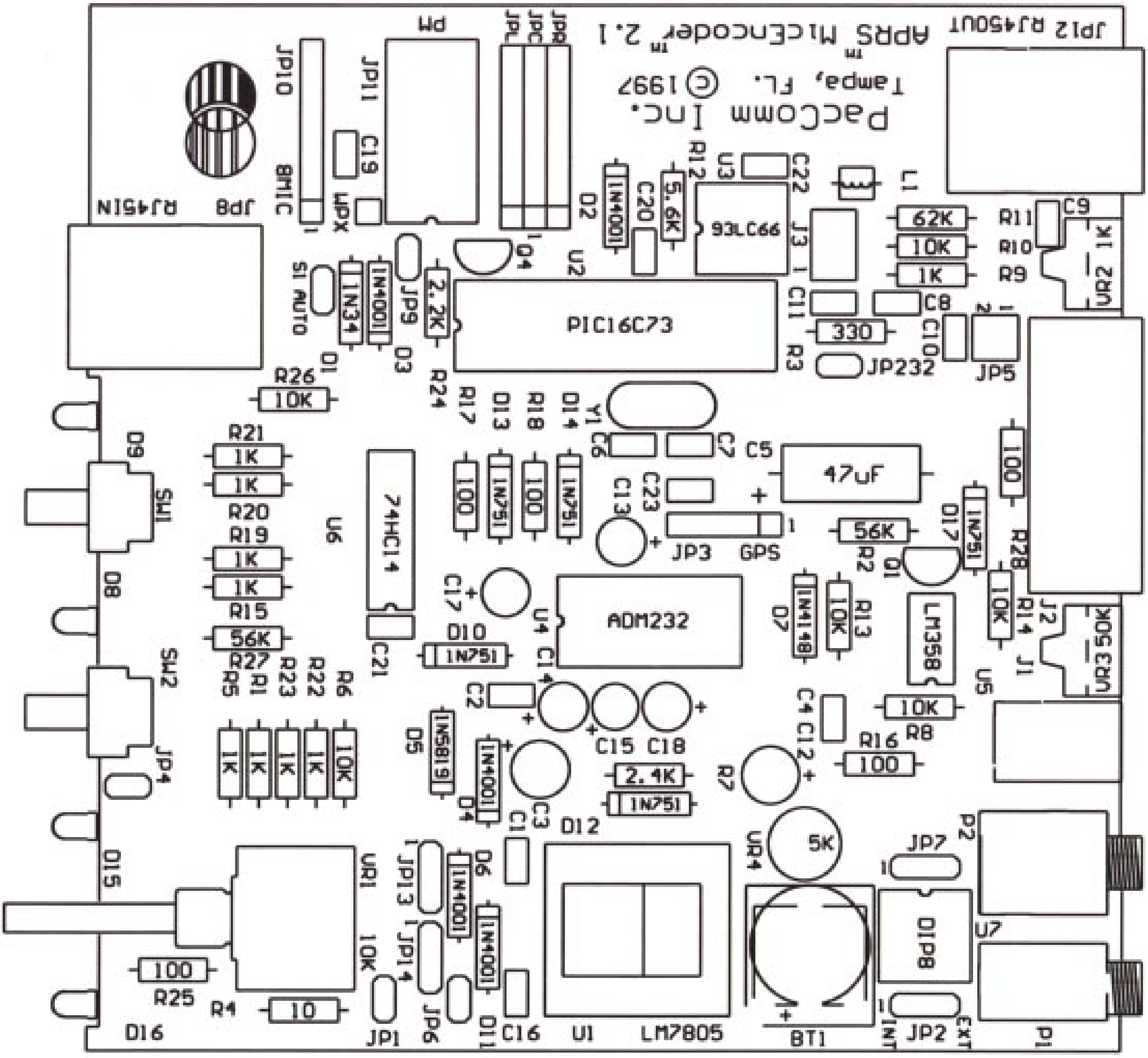
You are now ready to begin assembly, jumper configuration, and interfacing of the MIC-E. It is recommended that you use a very fine pointed soldering tip on an iron of relatively low wattage (25 watts maximum, 15 watts is ideal). In addition to the soldering iron and solder, you will need small flush or semi-flush cutting pliers and small-tipped long nosed pliers.

Pay careful attention to the directions that follow. Keep the tip of your soldering iron bright and clean, wiping it frequently on a wet rag or sponge. Make solder joints carefully, but swiftly. Two to three seconds should be enough time to apply heat to any joint.

The accompanying component placement drawing shows the placement of all parts on the circuit board.

Read this entire document prior to starting assembly of your kit. There are several final configurations for the MIC-E (depending on which type of radio you have).

( ) Carefully unpack and inspect the circuit board.


APRS™ MicEncoder™ 2.1
PacComm Inc.
Tampa, FL. © 1997
JP12 RJ45OUT
JP8 RJ45IN
JP10
8MIC
JP11
PM
WPX
C19
JPL
JPC
JPR
D2
U3
R12
5.6K
C20
1N4001
93LC66
C22
L1
J3
62K
R11
C9
10K
R10
1K
R9
VR2 1K
C11
C8
C10
330
R3
JP232
JP5
PIC16C73
U2
Q4
2.2K
JP9
1N4001
1N34
S1 AUTO
D1
D3
R24
R26
10K
R17
D13
R18
D14
Y1
C6
C7
C5
47uF
100
R28
D17
1N751
56K
R2
Q1
C23
C13
JP3
GPS
R21
1K
R20
1K
R19
1K
R15
1K
56K
R27
D9
SW1
D8
U6
74HC14
100
1N751
100
1N751
C17
U4
ADM232
D7
1N4148
R13
10K
LM358
R14
10K
J2
VR3 50K
J1
U5
C21
D10
1N751
C14
C2
C15
C18
SW2
R5
R1
R23
R22
R6
1K
1K
1K
1K
10K
JP4
D5
1N5819
1N4001
D4
C3
C4
C12
R16
100
R8
10K
R7
2.4K
1N751
D12
C1
VR4
5K
JP7
P2
D15
VR1
10K
JP13
D6
1N4001
JP14
1N4001
U7
DIP8
R25
100
R4
10
D16
JP1
JP6
D11
C16
U1
LM7805
BT1
INT
JP2
EXT
P1

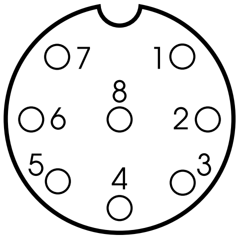
Assemble and Attach Panel Mount Mic Connector (JP10)

( ) Locate the 8 Pin Panel Mount Microphone Connector -- which is a plug (male) connector. If you are using an 8-pin mic, you will assemble the connector and soldered into JP10.

Included in your kit is a plastic plug which can be inserted into the hole if you chose not to install the 8-pin mic connector. The plug is not a perfect matc h for the hole, so care should be taken when installing. It can be done ;-)

( ) Locate the eight (8) 3" colored wires.

( ) Strip the wire back on one end by 1/4" Solder this end into the following PINs of the 8-pin connector.



Viewed from Rear

( ) 3 in. section of BLACK 24 GS Stranded Wire into PIN 8 of the 8-pin connector

( ) Install 3 in. section of RED 24 GS Stranded Wire into PIN 1 of the 8-pin connector

( ) 3 in. section of GREEN 24 GS Stranded Wire into PIN 2 of the 8-pin connector

( ) 3 in. section of YELLOW 24 GS Stranded Wire into PIN 3 of the 8-pin connector

( ) 3 in. section of ORANGE 24 GS Stranded Wire into PIN 4 of the 8-pin connector

( ) 3 in. section of BROWN 24 GS Stranded Wire into PIN 5 of the 8-pin connector

( ) 3 in. section of WHITE 24 GS Stranded Wire into PIN 6 of the 8-pin connector

( ) 3 in. section of BLUE 24 GS Stranded Wire into PIN 7 of the 8-pin connector

( ) Check all 8 connections. There should be no shorts or bad solder joints

( ) Now take the 3inch section of 1/16 inch Heatshrink and cut a 1/3inch section off. Slip the small section over each the wire and up over the end of the pin connector. Heat slightly to conform to wire and pin.

The purpose of the heatshrink is to keep the wires you just soldered to the connector stable and keep them from flexing and breaking off when you are removing the front face plate.

( ) You are now ready to attach the 8 wires to the MIC-E board.

( ) 3 inches of wire might be more then you want inside your enclosure.

If you are an experienced kit builder, you will have a feel for what length you will want to cut the wires down to.

If you are not an experienced kit builder, we suggest that you keep the 3 inches of wire and simply curl them together with a tie wrap after you finish soldering them to the board.

( ) Strip the wire back on one end by 1/4" of each wire. Solder this end into the following holes of the MIC-E board at JP-10

( ) Install 3 in. section of RED 24 GS Stranded Wire into PIN 1 of JP10

( ) 3 in. section of GREEN 24 GS Stranded Wire into PIN 2 of JP10

( ) 3 in. section of YELLOW 24 GS Stranded Wire into PIN 3 of JP10

( ) 3 in. section of ORANGE 24 GS Stranded Wire into PIN 4 of JP10

( ) 3 in. section of BROWN 24 GS Stranded Wire into PIN 5 of JP10

( ) 3 in. section of WHITE 24 GS Stranded Wire into PIN 6 of JP10

( ) 3 in. section of BLUE 24 GS Stranded Wire into PIN 7 of JP10

( ) 3 in. section of BLACK 24 GS Stranded Wire into PIN 8 of JP10

( ) Check all 8 connections. There should be no shorts or bad solder joints.

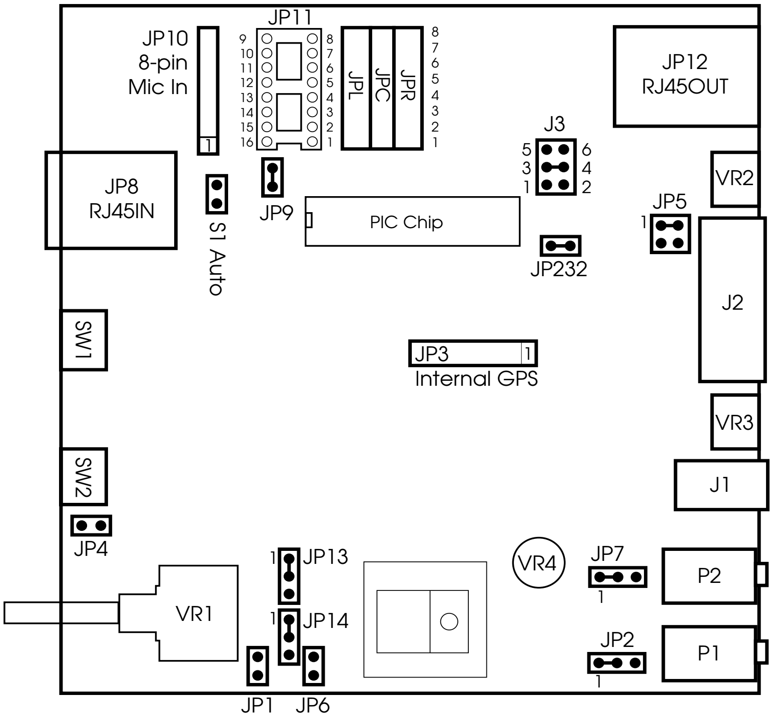
( ) Clip any excess wire on bottom (solder side) of the board.

NOTE -- BE VERY CAREFUL of these connections. Flexing the wires and other things can break them requiring them to be soldered again!

Configuration of MIC-E on-board Jumpers and Connectors

Contributed by Ron Parsons, W5RKN

Refer to figure of board that shows location of all jumpers (below).



JP1 REGULATOR BYPASS
DEFAULT is OFF. See JP6.

DO NOT CHANGE THIS UNLESS YOU KNOW WHAT YOU ARE DOING. YOU CAN ZAP YOUR UNIT IF YOU SET THE POWER INCORRECTLY.

With this jumper on, you can bypass the regulator chip when using microphone connector power (if your radio supplies it) via the personality jumper JP11 pin 16 or other power wired to wire point WPX (near JP10) and that power is less than 7.2 volts.

Do NOT use this option for supply voltages greater than 7.5 volts or you may cook everything. ***If the supply voltage is greater than 7.5 volts, use the Power Jack (J1).***

JP2 AD0 INPUT
DEFAULT is 1 and 2.

Warning: Do not put more than 5.0 volts on J2 pin 1. With jumper on pins 1 and 2, AD0 will read the supply voltage at the Power Jack (J1) in tenths of a volt. This means that a value of AD0 of 126 means 12.6 volts. With the jumper on 2 and 3, AD0 reads the value of the voltage on pin 1 of J2 in tenths of a volt.

JP3 Used to connect an internal GPS to the MIC-E.
This section requires that you solder the cable to your internal GPS (Garmin GPS-20, Trimble, etc) to these holes. (See section on installing a GPS)

Pin 1 - Connect to the TTL level (0V-5V) output of the GPS.
Pin 2 - Connect to the RS-232 level output of the GPS.
Pin 3 - Connect to the VBatt input of the GPS (not installed).
Pin 4 - Connect to the 5V input of the GPS.
Pin 5 - Connect to the ground connection of the GPS.

NOTE: Use either Pin 1 or 2, not both. If you use Pin 2, place a jumper on JP232. A 90-degree header may be placed in JP3 and used for the connection to the GPS if desired.

JP4 POSIT NOW
DEFAULT is OFF

Can be used to connect to an external momentary switch, which when closed, causes the MIC-E to send a Posit packet immediately. POSIT NOW can also be involked by connecting J2 pin 9 to ground momentarily.

JP5 Audio Rolloff Impedance Matching
DEFAULT on pins 1 and 2.

Leave this jumper on default if you don't understand what this does! If the jumper is on pins 3 and 4, the TX level pot is grounded through a .1 uF capacitor. On position 3-4 the capacitor in the circuit will add an extra pole of audio rolloff above 1600 Hz more or less depending on circuit load. This will unbalance the tones by 6dB. Users will find with the vagaries of mobile flutter, etc, that paying particular attention to their tone skew or balance will greatlly enhance there Mic-E performance. ***Trying to operate the Mic-E just outta-the-box on normal FIXED station packet is not going to give good performance under weak signal mobile operations though.***

JP6 5 VOLT POWER
DEFAULT is OFF.

This jumper must be ON if using 5.0 VDC external power via WPX (near JP10) or MICP (JP11 pin 16). This jumper must be OFF if using external power via WPX or MICP for voltages above 5.0 VDC and less than 7.2 VDC. This jumper bypasses the 10 ohm series resistor in the Zener regulator circuit and relies on the source impedance of the MIC circuit power for current limiting.

WARNING: Never use this jumper when supply voltages are above 5.8 volts.

JP7 RCV AUDIO AMP
DEFAULT on pins 1 and 2

With the jumper on 1 and 2, the receive audio (RXA) is not amplified prior to being passed to the MIC-E. With the jumper on 2 and 3, the receive audio (RXA) is amplified with the gain determined by VR3. Some radios, for example, the Kenwood 742, have a low-level audio output and need amplification. ***The green RX LED should flash or should be solid on with voice on the channel. If it does not, use the jumper on 2 and 3 and adjust VR3.***

JP8 The RJ-45 mic input connector.
This is a straight through connection to JP12 (except for PTT after configuration).

JP9 PTT Input Type
DEFAULT is ON

This jumper must be used if the radio has a separate PTT input. Some handi-talkies share the mic and PTT input. In this case, removing this jumper isolates the PTT from the mic with a 2.2K resistor. ***This may NOT work for your HT and should be tested carefully.***

JP10 The 8-pin (round) mic input connector.
This is a straight through connection to JP12 (except for PTT after configuration). You must solder the 8-pin round mic connector into these holes if your radio uses an 8-pin round mic connector.

JP11 Microphone Personality Module.
See section on constructing the radio interface header for further details on wiring this connector.

JP12 The RJ-45 mic output connector.
This is a straight through connection to JP8 and JP10 (except for PTT after configuration). The included 3 foot cable will be used to connect from this connector to your radio. If your radio uses a 8-pin connector, you will cut one end off the cable and attach the included 8-pin in-line connector.

JP13 POWER (*See JP14*)
DEFAULT on pins 1 and 2.
Use jumper 1 and 2 if using the Power Jack (J1).
Use jumper 2 and 3 if using external power (WPX or MICP).

JP14 POWER (*See JP13*)
DEFAULT is 1 and 2.
Use jumper 1 and 2 if using the Power Jack (J1).
Use jumper 2 and 3 if using external power (WPX or MICP).

JP232 NMEA RS-232 OUTPUT
DEFAULT is ON

This jumper must be used if NMEA RS-232 level output from the internal GPS is used.

J1 The Power Jack
Use this if the supply voltage is greater than 7.5 volts but less than 13.8 VDC. The center connector is positive.

J2 The DE-9 connection the rear panel.

Pin 1 - AD0 analog input if JP2, pins 2 and 3, are used.
Pin 2 - RS-232 output from MIC-E.
Used when programming the MIC-E.
Pin 3 - RS-232 input to the MIC-E.
This can be external GPS data or programming data for the MIC-E. Insert a stereo plug in P1 with no connections on tip and ring.
Pin 4 - May be used to power the MIC-E by putting a jumper on SP1. The voltage at this pin should be greater than 7.5 volts but less than 13.6 VDC.
Pin 5 - Ground
Pin 6 - AD4 analog input.
Pin 7 - no connection.
Pin 8 - AD2 analog input.
Pin 9 - AD3 analog input

J3 Mic-E LOADING
DEFAULT is on pins 3 and 4 for 10K.
Pins 1 and 2 provide 1K
Pins 5 and 6 provide 62K.

This jumper allows you to minimize the impedance loading of the Mic-E on your existing Mic Circuit. Use the highest value resistance that still gives suitable packet audio level without loading down the voice audio.

SP1 POWER Selection
This jumper must be used if the MIC-E is powered via J2, pin 4. See above. The solder pads are located on the bottom side of the board near J2.

P1 GPS data/MIC-E programming data jack
The tip is RS-232 data output. The ring is RS-232 data input. ***NOTE: If the MIC-E is programmed via J2, insert a stereo plug in P1 with no connections on tip and ring.***

Construction of Radio Interface Header Interfacing with your RADIO

Contributed by Bob Bruninga, WB4APR

The Mic-E is easily interfaced to any radio with a round 8-pin Mic jack (JP10) or the newer RJ-45 jack (JP8). All of the 8 pins from these front panel connectors (except for PTT after configuration) pass straight through the Mic-E, out the back (JP12) and to the radio unchanged, so there are no restrictions on any microphone/radio combination that use these two common connectors.

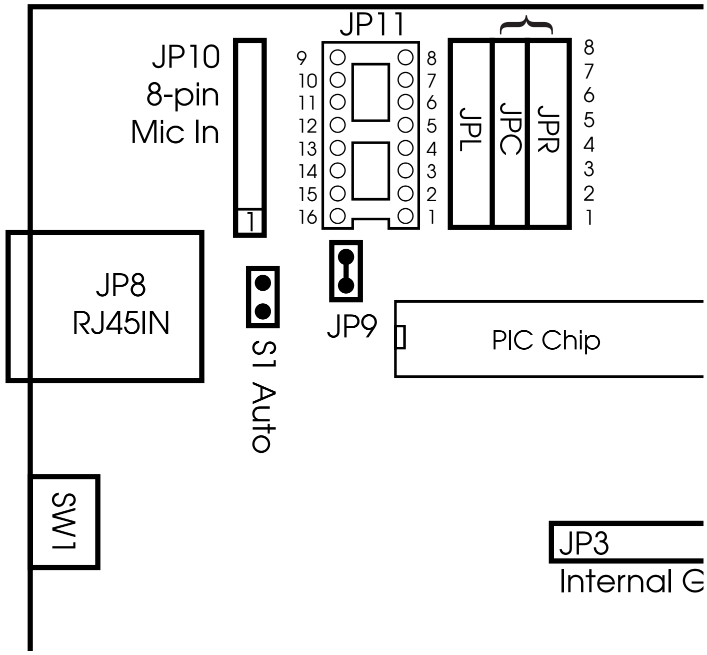
To configure the Mic-E to the particular arrangement for your radio, there are two steps:

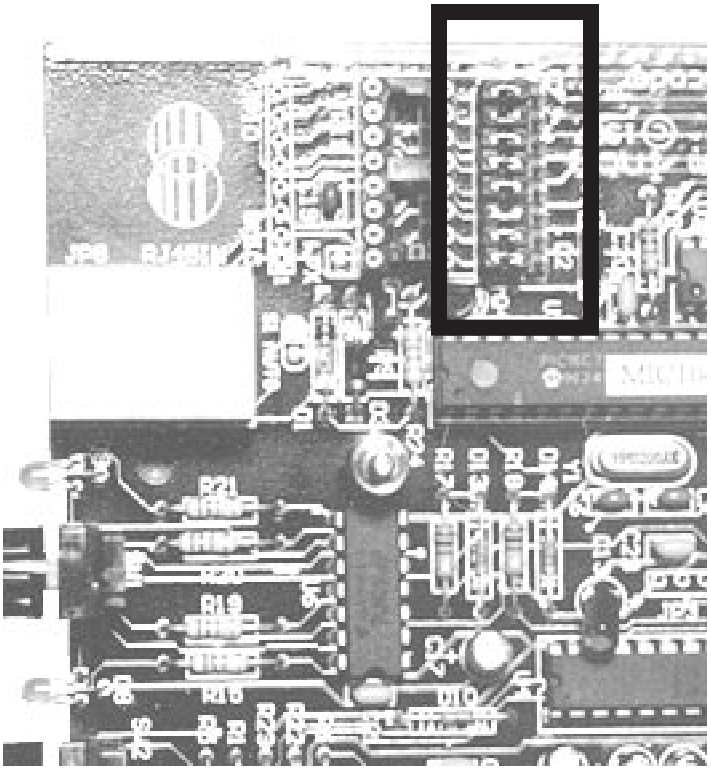
1) ***Isolate the PTT signal using JPR,JPC,JPL***

2) ***Configure the personality dip header (JP11) that connects the Mic-E to all the other signals it needs.***

PTT CONNECTION

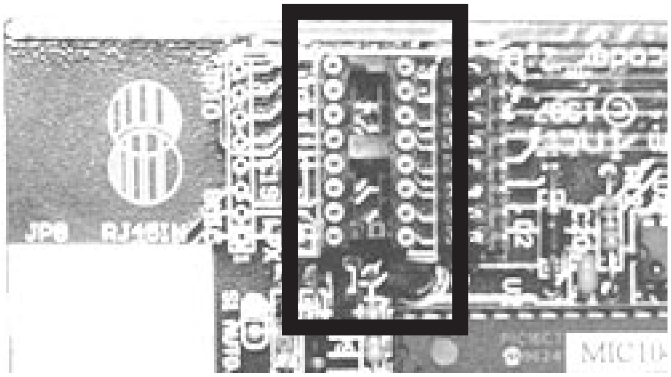
All 8 pins pass straight through the Mic-E via a row of 8 jumpers labeled JPR, JPC and JPL. All of these jumpers should be on the JPC to JPL position except for one. ***Place the one jumper that represents your PTT pin on the JPR-to-JPC position.***

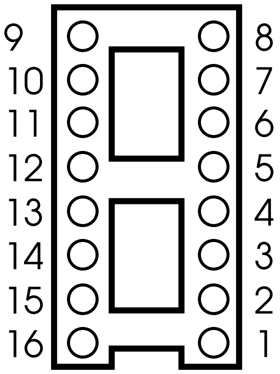


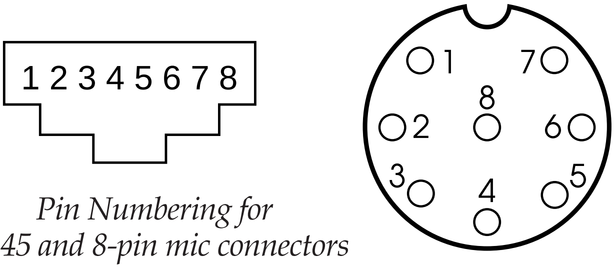


PERSONALITY HEADER

The Mic-E is shipped with an empty 16 pin DIP header for use on JP11. All you need to do is place 6 jumper wires to connect the 6 remaining Mic-E connections to the proper microphone pin for your radio.







Pin Numbering for
RJ-45 and 8-pin mic connectors
Looking into panel jack

Note, it is important to use the separate Mic-Ground in order to minimize ground loop problems, noise, alternator whine etc. The receiver audio must be squelched. If squelched audio is not available on the mic jack, use speaker audio via the rear panel connector.

RECOMMENDED PROCEDURE

1) First, with no personality jumper plugged in and all of the PTT jumpers in the JPL position, plug in your mic and plug in the Mic-E to your radio. Verify normal operation of the Microphone and radio.

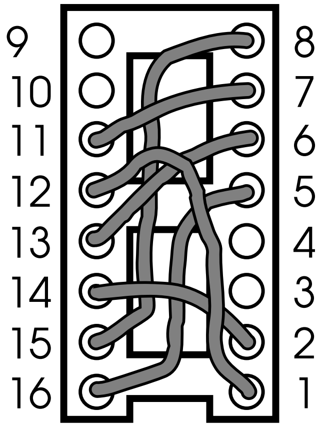
2) Move the PTT jumper and place wires in the DIP header for your radio and microphone personality. Test Mic-E operation.

   It is suggested that you cut small segments from the 1 ft. section of Olive Green 28 AWG Wire provided with the kit. Simply insert the small jumper segments into the holes of the dip header to make the necessary jumpers between pins.

3) Once proper operation is verified including clean audio with no noise, alternator whine, or other troubles, then solder your jumper wires on the dip header and stick on a label for your radio type.

   If you have a radio not in the chart below, please let us know what the configuration is so that it can be included on the web page and added to later manuals

Example wiring for the Kenwood 8-pin mic:



Mic-E Function	MicE Dip Header		Kenwood 8-pin	Kenwood RJ-45	Icom 8-pin	Icom RJ-45	Yaesu 8-pin	Azden 8-pin
Mic power	16	to	5	7	2	1	2	?
Radio/PTT gnd	15	to	8	6	6	7	7	6
Microphone PTT	14	to	2	5	5	4	6	7
Receiver Audio	13	to	6	2	8	3	4	?
Microphone audio	12	to	1	3	1	6	8	1
Microphone gnd	11	to	7	4	7	5	6	?

Mic-E Function	MicE Dip Header		Alinco 8-pin	Radio Shack RJ-45	Standard Rado 8-pin	Standard Rado RJ-45	Your Mic	Your Mic
Mic power	16	to	5	7	?	1		
Radio/PTT gnd	15	to	8	2	8	5		
Microphone PTT	14	to	2	6	6	4		
Receiver Audio	13	to	6	?	3	3		
Microphone audio	12	to	1	5	1	6		
Microphone gnd	11	to	7	2	?	5		

Interfacing with GPSs

You MUST use a GPS receiver in order for the MIC-E to operate properly. Your position will not be sent without a GPS connected to the unit. You may either install an internal GPS receiver board inside the MIC-E case or connect an external GPS to the unit.

If your transmitted packets show P's and 0's in the destination field, e.g., "PP0P0P", then your GPS is either not receiving data or is not connected to the MIC-E.

Types Supported

Any GPS unit that provides NMEA output can be used with the MIC-E.

Attaching Internally

The following GPS units are supported for internal installation: Garmin GPS-20, Motorola ONCORE VP, Trimble SveeSixCM3, and Rockwell Jupiter. You use JP3 to connect the internal GPS to the MIC-E. The following table shows the connections required for each GPS. Pinout for JP3 is shown earlier in manual.

NOTE: Use either Pin 1 or 2, not both. If you use Pin 2, place a jumper on JP232.

MicEncoder		Trimble Svee Six-CM3	Garmin GPS-20	Rockwell Jupiter
NMEA-TTL	Pin 1	3	N/A	11
MNEA-232	Pin 2	N/C	4	N/A
BATT	Pin 3	4	N/A	3
VCC	Pin 4	2	10	2
GND	Pin 5	8	8	10

Attaching Externally

You must use RS-232 level output from the GPS. An external GPS provides NMEA data to the MIC-E via the Ext GPS/PGM stereo jack (P1, ring) or the Serial Port (J2, pin 3). Be sure to attach the ground connection as well. If you use J2, pin 3 for GPS input, insert a blank stereo plug into P1.

Interfacing with your Computer

There are two ways to interface to the PIC for programming: via the DB-9 (J2) or the External GPS/ PGM stereo plug (P1).

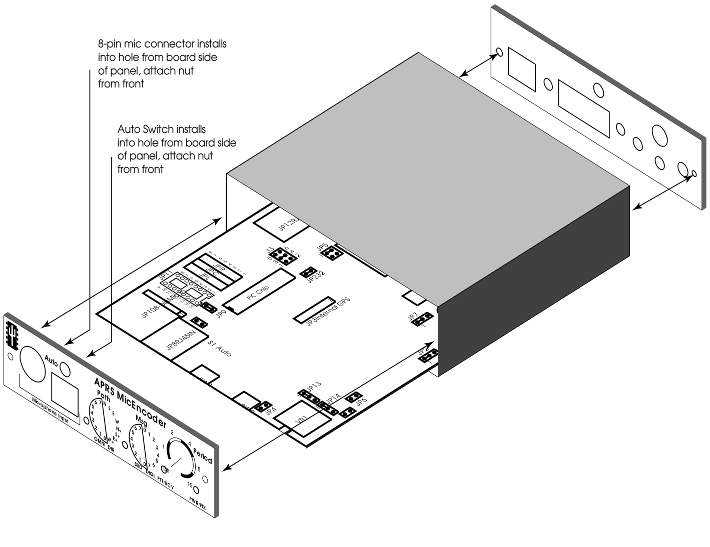
J2 - You should connect only pins 2, 3, and 5 if you are using the DB-9 interface J2. A standard DB-9 serial cable MAY work, but be very carefully in it use since other signals are present on the connector. If you use J2 for programming input, you must use RS-232 level output from the computer and insert a blank stereo plug into P1.

P1 - You must use RS-232 level output from the computer and connect the Mic-E input (TxD) to the ring of P1 and the Mic-E output (RxD) to the tip of P1. Ground is the shank. Do not connect anything to J2.

Turn the MIC-E on before executing MIC107.EXE or the Macintosh equivalent.

Assembly of the Unit

Assembly can take place at any time. It is suggested that you test and configure your unit with the case off on your workbench before final assembly. Assembling the unit can stress certain parts of the kit, so repeated opening and closing can cause damage and you should be aware of this and be careful each time you take the unit out of the case.



Assemble the unit by inserting the circuit board (with or without an attached internal GPS) in the case in the lowest possible position. Place the rear cover of the unit in place and use the two mounting screws to secure the plate to the case.

Carefully position and attach the 8-pin mic connector (JP8) and the Auto switch (S1) to the front panel.

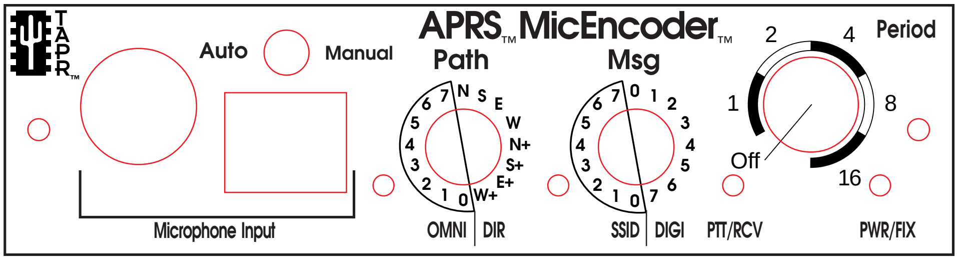
Then attach the front panel to the enclosure securing it to the case with the two mounting screws. Pay particular attention to wires attaching JP8 -- ***DO NOT STRESS OR BREAK THEM.***

After you have completed assembly of the unit you will want to test the unit.

Operational Overview

Contributed by Ron Parsons, W5RKN

Microphone In Connectors:
The MIC-E supports both RJ-45 and 8-pin microphone connectors. Both microphone connectors in the front are wired together and either can be used. This allows the MIC-E to be reconfigured by changing the personality module on the inside of the unit if you change radios.



Auto/Manual Switch:
When set to AUTO, the MIC-E will transmit a packet automatically whenever the AUTO interval you configured with MIC-E SETUP program has gone by. The MIC-E will only automatically transmit if the radio is quiet for the number of seconds configured in the QUIET interval. You should also make sure your TXDelay is set sufficiently long to allow any necessary keyup delay before data is transmitted. For example, if you are working through a repeater system with CTCSS and remote receivers, the delay may need to be set as high as 500 or 1000 milliseconds to avoid clipping the beginning of the packet.

When set to MANUAL, the MIC-E will only transmit packets after you release the PTT switch. It will never transmit automatically. When set to MANUAL, the RCV LED is always illuminated to indicate this hold-off condition.

Normally the MIC-E will only send a POSIT if the POSIT timer has elapsed AND the user has been talking AND releases his PTT. In the AUTO mode, however, after a specified AUTO time period, AND after the repeater has been silent for the QUIET period, then a position packet will be initiated and transmitted. **Note**: for the QUIET timer to work properly, the receiver's audio must be set high enough to occasionally flicker or fully light the RCV LED.

TXDelay Sets the key up delay for AUTO packets
AUTO N Sets AUTO period as N * POS period

AUTO has two uses. On a voice repeater you might set QUIET to long enough to be sure the repeater is really not BUSY before you kerchunk it. But if you use AUTO on the APRS packet channel, then you want QUIET to be ZERO so that it acts only for collision avoidance. If QUIET is non zero and you have a busy APRS channel it may never transmit!. You can override this if you need to, by simply turning the volume down, but then you don't have collision avoidance. So this is a trade-off. *Just be sure to never turn your radio volume down on a REPEATER if you have AUTO on, or you will BRAAAP other people...*

PTT/RCV LED:
This LED is connected to both a simple audio rectifier of the receiver audio and the PTT circuit.

RED - In Manual or Auto mode, this LED lights to indicate that the MIC-E is preparing to transmit or is transmitting its data packet at the end of your current voice transmission. In addition, in Auto mode, this LED lights to indicate that the MIC-E is transmitting its data packet at the scheduled interval.

GREEN - There is audio being received by the radio and being detected by the MIC-E. This will cause the MIC-E to hold off transmitting its packet. The packet will be transmitted after the QUIET interval has elapsed after the detected audio ceases.

If the LED has not been driven by the receiver for the period of the QUIET timer, then an AUTO packet is possible. If AUTO is set to Manual, then the switch permanently lights the LED and prevents all AUTO-packets. Users must set the receiver audio high enough to tickle or fully light the RCV LED frequently in order for holdoff to work.

WARNING!: If you turn the audio down because the spouse wants quiet, then your Mic-E will not have a holdoff signal and the AUTO timer may transmit on the repeater over other users if AUTO is enabled!

PWR/FIX LED:
The PWR/FIX LED is RED for about a second when power is first applied. When the GPS is absent or no fix is available, the LED is YELLOW (GREEN + RED which looks YELLOW). While a valid GPS fix is available, the LED will momentarialy flicker RED about once a second.

Path Switch (SW1)

The light just to the lower-left of the Path Switch (D9) will have the following colors:

Position	Path	Color
0	0	off
1	1	off
2	2	off
3	3	off
4	4	green
5	5	green
6	6	green
7	7	green
8	0	red
9	1	red
A	2	red
B	3	red
C	4	yellow
D	5	yellow
E	6	yellow
F	7	yellow

In the following example, assume the MIC-ENCODER has been loaded with the VIA string of RELAY,WIDE,WIDE,DIG4,DIG5,DIG6,DIG7

D/O	PATH	SSID	DIGI MODE	SSID MODE AS DIGIPEATED BY THE NODE
0	0	0	none	none
0	1	1	RELAY	WIDE-1
0	2	2	RELAY,WIDE	WIDE-2
0	3	3	RELAY,WIDE,WIDE	WIDE-3
0	4	4	DIG4	WIDE-4
0	5	5	DIG4,DIG5	WIDE-5
0	6	6	DIG4,DIG5,DIG6	WIDE-6
0	7	7	DIG4,DIG5,DIG6,DIG7	WIDE-7
1	0	8	none	NORTH UNPROTO path
1	1	9	RELAY	SOUTH UNPROTO path
1	2	10	RELAY,WIDE	EAST UNPROTO path
1	3	11	RELAY,WIDE,WIDE	WEST UNPROTO path
1	4	12	DIG4	NORTH UNPROTO path + WIDE
1	5	13	DIG4,DIG5	SOUTH UNPROTO path + WIDE
1	6	14	DIG4,DIG5,DIG6	EAST UNPROTO path + WIDE
1	7	15	DIG4,DIG5,DIG6,DIG7	WEST UNPROTO path + WIDE

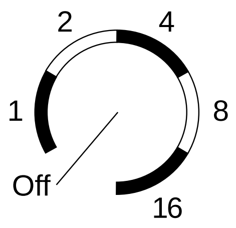
Message Switch (SW2):

The light just to the lower-left of the Message Switch (D8) will have the following colors:

Position	Msg	LED Color	Mode
0	0	off	SSID Mode
1	1	off	
2	2	off	
3	3	off	
4	4	green	
5	5	green	
6	6	green	
7	7	green	
8	0	red	DIGI Mode
9	1	red	
A	2	red	
B	3	red	
C	4	yellow	
D	5	yellow	
E	6	yellow	
F	7	yellow	

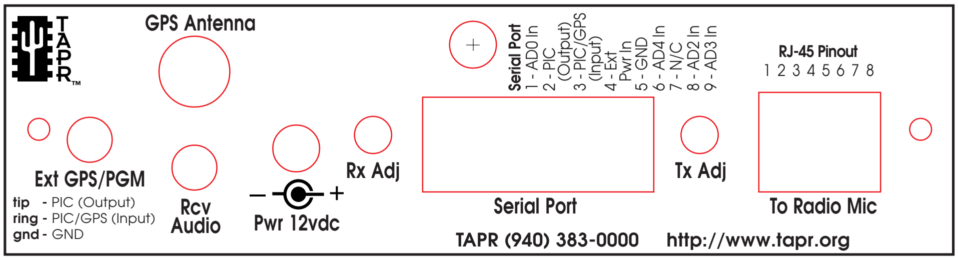
MSG	Icon Color	DEFAULT DEFINITION	
0	normal	Off duty	
1	normal	Enroute	
2	normal	In Service	
3	normal	Returning	
4	dim yellow	Committed	
5	Brt yellow	Special	
6	dim red	PRIORITY	Trips alarms & centers all maps to unit
7	Brt red	EMERGENCY!	Trips alarms & centers all maps to unit

Period:



The value of the period knob is a multiplier for the PERIOD that was configured into the MIC-E using the SETUP program. For example, if you configured the PERIOD to once per 60 seconds and then set the period knob to the 4 position, the PERIOD will be once every 4 minutes.

Setting the knob to the off position turns off the MIC-E.



Ext GPS/PGM:
Enables connection of an external GPS's NMEA-0183 serial data using a mini-plug. This is the primary method of feeding external GPS data into the MIC-E.

GPS Antenna BNC hole:
Use this hole to mount a female BNC connector attached to your internal GPS engine. If you are not installing an internal GPS, place the plug in the hole.

Receive Audio In:
If receive audio is not available on your microphone connector, then use this jack to provide a hold-off signal which prevents the unit from transmitting in Auto mode when the channel is busy. Connect to a speaker jack on your radio, for example, and adjust the audio level and the RXA trim pot until the RCV LED at least flickers or is full on at all levels of normal use.

Receive Audio Level:
Use this adjustment in conjunction with your speaker level to cause the RCV LED to at least flicker whenever the radio is receiving (unsquelched) but be off when there is no traffic. ***Overdriving the level is not a problem as long as it goes out when the channel is clear.***

Transmit Audio Level:
Adjust the transmit level if the MIC-E does not provide enough audio for your radio. Not all radios require this to be adjusted. Your deviation should be about 3KHz (or less if you are willing to give up a little performance for the sake of softer tones and less objection from voice users during the mic-e introduction period).

Power 12volts:
Center pin is positive.

Serial Port:

Pin 1 - AD0 analog input if JP2, pins 2 and 3, are used.
Pin 2 - RS-232 output from MIC-E.
Used when programming the MIC-E.
Pin 3 - RS-232 input to the MIC-E.
This can be external GPS data or programming data for the MIC-E. Insert a stereo plug in P1 with no connections on tip and ring.
Pin 4 - May be used to power the MIC-E by putting a jumper on SP1. The voltage at this pin should be greater than 7.5 volts but less than 13.6 VDC.
Pin 5 - Ground
Pin 6 - AD4 analog input.
Pin 7 - no connection.
Pin 8 - AD2 analog input.
Pin 9 - AD3 analog input

To Radio Mic:
Connect from here to the MIC jack of your radio. A RJ-45 to RJ-45 patch cord is provided in the kit. If your radio uses a 8-pin connector, cut one end off and use the supplied 8-pin mic connector. Match wire 1-8 to pins 1-8 respectively.

DOS Users Guide (quick start)

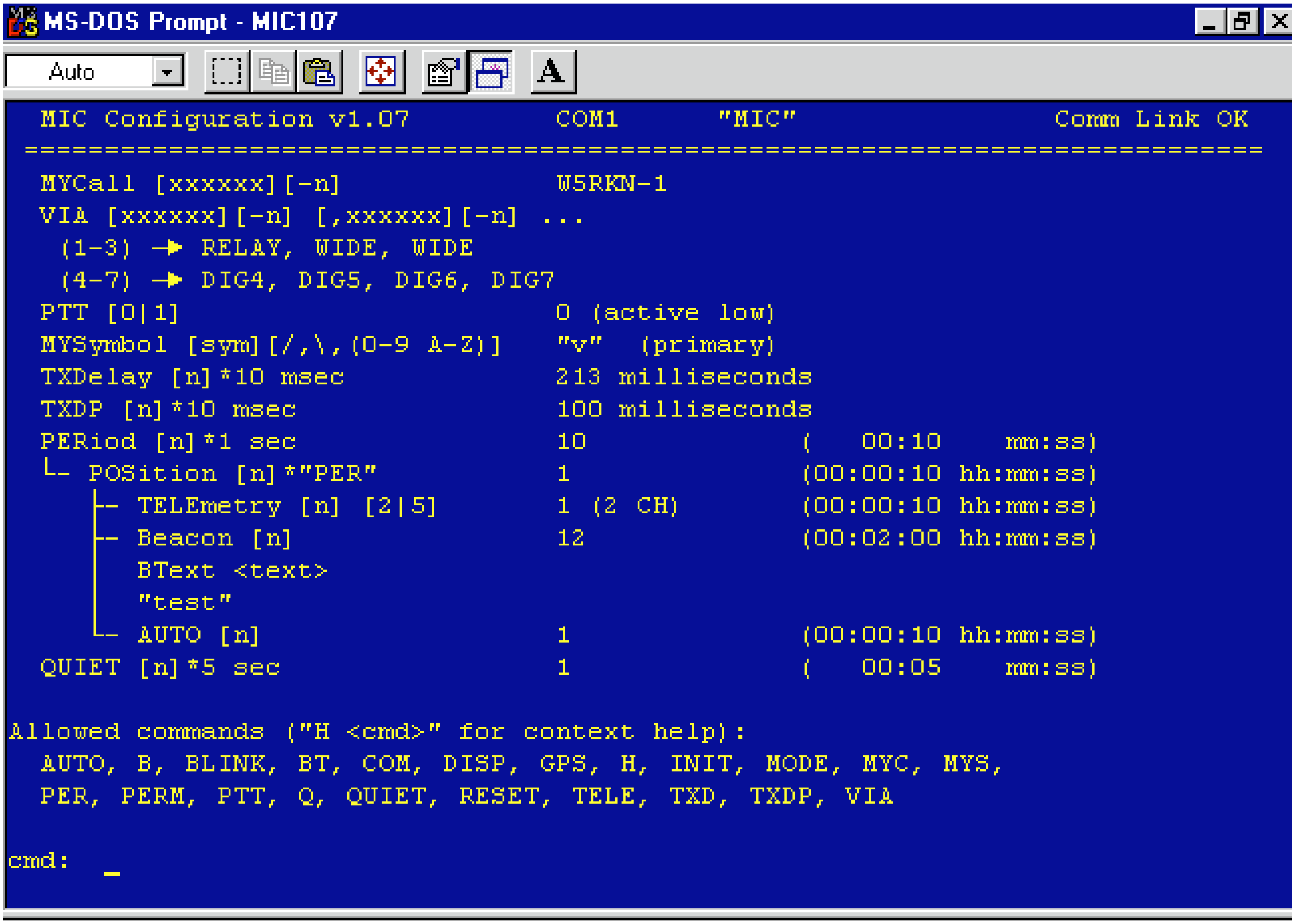
Contributed by James Duncan, KU0G

Initial Programming and Testing
Load the DOS setup program into a convenient computer and connect a serial cable to COM1 or COM2. The MIC-E end of the cable should ONLY use pins 2, 3 and 5!!! Do NOT attempt to use other pins as the port on the MIC-E is NOT necessarily compatible with a normal serial cable and damage to your computer or the MIC-E may result!

After assembling the unit, you will need to program the MIC-E! Apply power to the MIC-E and turn the unit on. Run the programming utility from the DOS prompt. The program will start and the unit will search for your MIC-E. Don't panic if you get the message Comm Link FAILURE. This MAY be the fault of the serial port on your computer and does not necessarily mean that your MIC-E is not receiving input data. Issue the RESET command. This will most likely result in your now being able to communicate fully with the unit.

Next, be sure that the mode of the unit is MIC rather than MIM. The MODE of the unit is indicated on the status line at the top of your screen. It should show "MIC". If it shows "MIM", issue the command "MODE MIC". Once you have done this you are ready to continue programming.

For additional information, refer to the documentation for the DOS setup program.



```
  MIC Configuration v1.07         COM1      "MIC"                   Comm Link OK
 ===========================================================================
  MYCall [xxxxxx][-n]             W5RKN-1
  VIA [xxxxxx][-n] [,xxxxxx][-n] ...
   (1-3) → RELAY, WIDE, WIDE
   (4-7) → DIG4, DIG5, DIG6, DIG7
  PTT [0|1]                       0 (active low)
  MYSymbol [sym][/,\,(0-9 A-Z)]   "v"  (primary)
  TXDelay [n]*10 msec             213 milliseconds
  TXDP [n]*10 msec                100 milliseconds
  PERiod [n]*1 sec                10             (   00:10    mm:ss)
  └─ POSition [n]*"PER"           1              (00:00:10 hh:mm:ss)
     ├─ TELEmetry [n] [2|5]       1 (2 CH)       (00:00:10 hh:mm:ss)
     ├─ Beacon [n]                12             (00:02:00 hh:mm:ss)
     │  BText <text>
     │  "test"
     └─ AUTO [n]                  1              (00:00:10 hh:mm:ss)
  QUIET [n]*5 sec                 1              (   00:05    mm:ss)

Allowed commands ("H <cmd>" for context help):
  AUTO, B, BLINK, BT, COM, DISP, GPS, H, INIT, MODE, MYC, MYS,
  PER, PERM, PTT, Q, QUIET, RESET, TELE, TXD, TXDP, VIA

cmd: _
```

Programming

The MIC-E/MIM chip functions just as a TNC does and you will notice that the command set follows TNC commands that you are already familiar with. Here are some recommendations for your initial setup:

MYCALL	Use your call and one of the standard APRS SSIDS (i.e. -9 for car).
VIA	Although 7 digipeaters can be entered, the Mic-E only uses them in two groups. For path switch positions 1-3 the first three digipeaters are used. For path switch positions 4-7, only the digis beginning wiht the 4th entry are used.
BT	Enters a Beacon Text which is sent in plain-text ASCII at the Beacon rate you set.
MYS	This sets the icon code that APRS will use to display your MIC-E. Refer to SYMBOLS.TXT contained in the APRS documentation for the character codes.
PERM	Use this command once you have all data set as you prefer. This will download the data to your MIC-E.
RESET	After perming, you must either cycle power on the MIC-E or issue the RESET command if you want to begin using the new configuration.

TESTING

Have a scanner or second radio close at hand so you can monitor your testing. Program in your local APRS digital frequency and let’s start testing!

• Double check all of your programming, wiring and connect the unit to your radio.
• Turn on the unit, set the PATH switch to OMNI 2
• Switch the unit into AUTO mode and rotate the Period Pot until the red PTT LED lights.

• Your radio should transmit and you should immediately hear a digipeat of your packet. If you do NOT hear a data burst corresponding to your transmitter keying up, check your microphone wiring installation. If you DO hear data but no digipeat, don’t worry you’re operating.

• Leave a convenient computer on line running APRS and do several Period Pot tests in the AUTO mode.

Note: It is helpful to have a laptop computer at your testing point so you can see what you’re sending.

• If your icon pops up on the APRS screen your are fully operational.

• Go ahead and experiment with the various message/status functions by rotating the MSG switch from one position to the next and watch the transmitted results!

• Try position #7. Your APRS computer should re-center the map on your icon and paint you BRIGHT RED while triggering an alarm tone. ***Please do this on a simplex frequency or without a digipeater so th at you dont bother all APRS stations in yoru network with a false alarm.***

You should now start testing the unit in Manual mode by changing the PATH and MSG switches and keying your microphone. Not every transmission will trigger a MIC-E packet; however, the RED PTT LED will light when you have the key down and the MIC-E has an packet to transmit. When you release your microphone you will note a slight delay in the transmitter unkeying while the data packet is sent!

CONGRATULATIONS!

Your MIC-E is now operational and you’re ready to use this unit in the wonderful world of APRS!

MAC Users Guide (quick start)

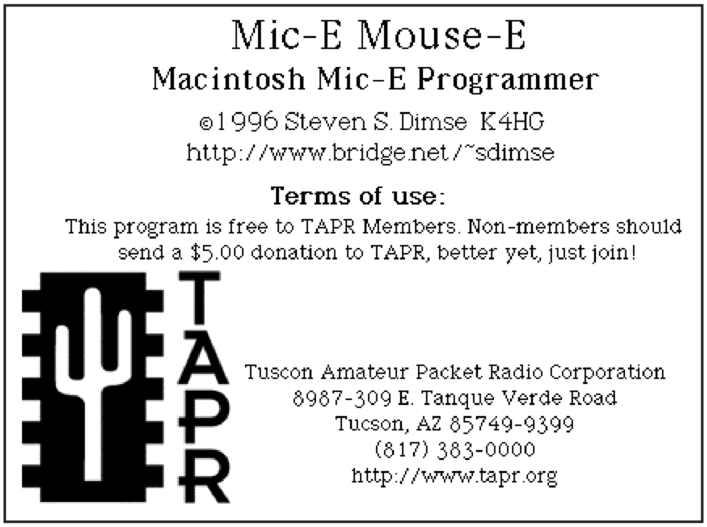
Contributed by Steve Dimse

Initial Programming and Testing

The first step to using the Mic-E is obtaining a cable for programming the unit. Cables may either be bought or made. The easy way is to use a standard modem cable, and buy a DB-25 to DB-9 adapter. I've never seen a mac->DB-9 cable for sale. If you want to make your own, use a mini-din 8 cable, cut one end off, and figure out the color scheme by testing each line, there is no standard (see below).

Before assembling the unit, you will need to program the MIC-E! You will need to obtain the Mic-E Mouse-E program. Latest version is available on ftp://ftp.tapr.org/tapr/mice/software. When the program is run, an initialized window opens. There are two pages of info available, selected by the tabs at the top of the window as shown by Figures 1 and 2 (next page). See the documentation for descriptions of these parameters.

Attach your serial cable, plug a non-attached stereo plug into P1 (Ext GPS/PGM) and apply power to the MIC-E and turn the unit on. The first time you use the transmit or receive command the program will let you pick the port to which the Mic-E is connected. The other serial parameters are set correctly (4800-8-N-1) and should not be changed. You can use the receive command to verify the data has been sent correctly. The Mic-E data can be saved with the Save As command so it does not need to be reentered. Once you have typed in the desired values for both pages of data, it is sent to the Mic-E by using the transmit command in the Mic-E menu.



Mic-E Mouse-E

Macintosh Mic-E Programmer

©1996 Steven S. Dimse K4HG
http://www.bridge.net/~sdimse

Terms of use:

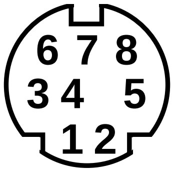
This program is free to TAPR Members. Non-members should send a $5.00 donation to TAPR, better yet, just join!

Tuscon Amateur Packet Radio Corporation
8987-309 E. Tanque Verde Road
Tucson, AZ 85749-9399
(817) 383-0000
http://www.tapr.org

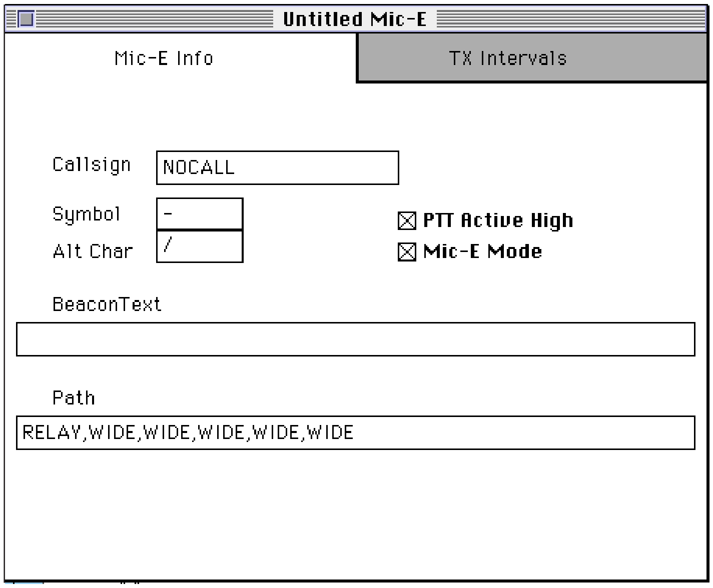
Mini DIN-8 Macintosh Serial Connector Pinout

This is the pinout of the Macintosh connector as if you were looking at the (male) connector on the cable; the (female) Macintosh connector will be a mirror image.

1	HSKo	Output Handshake (Zilog 8530 DTR pin)
2	HSKi/CLK	Input Handshake *OR* External Clock
3	TxD-	Transmit data (-)
4	Ground	Signal ground
5	RxD-	Receive data (-)
6	TxD+	Transmit data (+)
7	GPi	General Purpose input
8	RxD+	Receive data (+)



Macintosh DIN-8 Pin	 Signal		Mic-E DB-9 Signal	 Pin
1	HSKo	(nc)		
2	HSKi	(nc)		
3	TxD-		TxD	3
4	GND		GND	7
8	RxD+			
5	RxD-		RxD	2
6	TxD+	(nc)		
7	GPi	(nc)		
shield			shield	



Callsign NOCALL

Symbol -

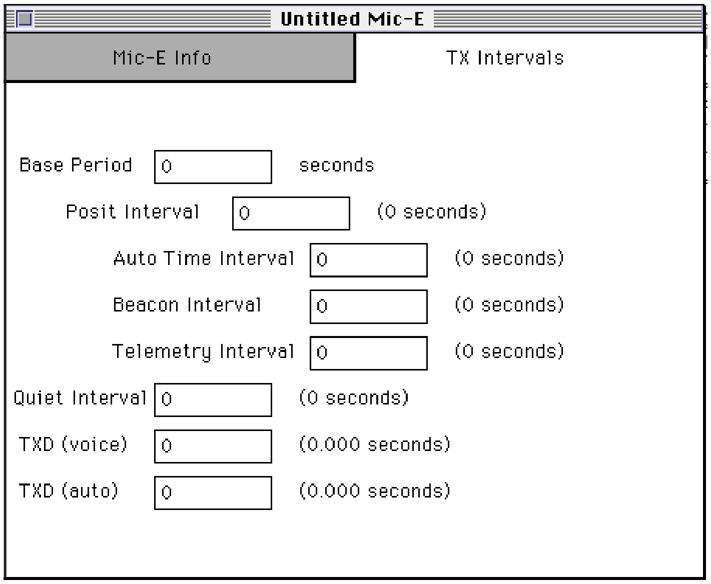
Alt Char /

☒ PTT Active High

☒ Mic-E Mode

BeaconText

Path RELAY,WIDE,WIDE,WIDE,WIDE,WIDE



Base Period	0	seconds
Posit Interval	0	(0 seconds)
Auto Time Interval	0	(0 seconds)
Beacon Interval	0	(0 seconds)
Telemetry Interval	0	(0 seconds)
Quiet Interval	0	(0 seconds)
TXD (voice)	0	(0.000 seconds)
TXD (auto)	0	(0.000 seconds)

TESTING

Have a scanner or second radio close at hand so you can monitor your testing. Program in your local APRS digital frequency and let's start testing!

- Double check all of your programming, wiring and connect the unit to your radio.

- Turn on the unit, set the PATH switch to OMNI 2

- Switch the unit into AUTO mode and rotate the Period Pot until the red PTT LED lights.

- Your radio should transmit and you should immediately hear a digipeat of your packet. If you do NOT hear a data burst corresponding to your transmitter keying up, check your microphone wiring installation. If you DO hear data but no digipeat, don't worry you're operating.

- Leave a convenient computer on line running APRS and do several Period Pot tests in the AUTO mode.

Note: It is helpful to have a laptop computer at your testing point so you can see what you're sending.

- If your icon pops up on the APRS screen your are fully operational.

- Go ahead and experiment with the various message/status functions by rotating the MSG switch from one position to the next and watch the transmitted results!

- Try position #7. Your APRS computer should re-center the map on your icon and paint you BRIGHT RED while triggering an alarm tone.

You should now start testing the unit in Manual mode by changing the PATH and MSG switches and keying your microphone. Not every transmission will trigger a MIC-E packet. However the RED PTT LED will light when you have the key down and the MIC-E has an packet to transmit. When you release your microphone you will note a slight delay in the transmitter unkeying while the data packet is sent!

CONGRATULATIONS!

Your MIC-E is now operational and you're ready to use this unit in the wonderful world of APRS!

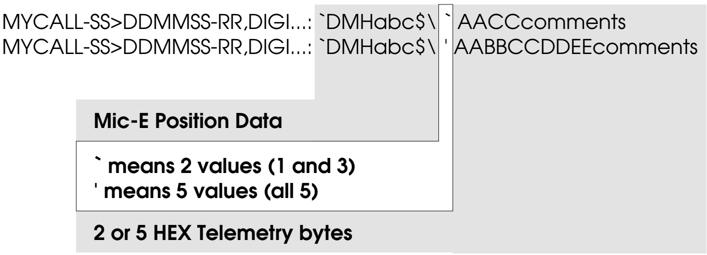
Mic-Encoder APRS TELEMETRY SYSTEM

Contributed by Bob Bruninga, WB4APR

The Mic-E can include up to 5 analog values in its position packets. There are two basic telemetry formats in the Mic-E. The original APRS format is transmitted in the MIM mode and includes a serial number, the five analog values as decimals between 000 and 255, and the eight bits as 0's or 1's as follows:

T#sss,111,222,333,444,555,xxxxxxxx

In Mic-Encoder mode, there are no digital bits, since they are used internally for the message and path bits. Further, the 5 analog values are transmitted as dual byte hexadecimal values. To keep the packets as short as possible for the Mic-E, there is also a short option which only sends the AD0 and AD2 values. Telemetry is optional and if present is appended after the Position report and before any optional Beacon Text (see below).



```
MYCALL-SS>DDMMSS-RR,DIGI...: `DMHabc$\ ` AACCcomments
MYCALL-SS>DDMMSS-RR,DIGI...: `DMHabc$\ ' AABBCCDDEEcomments
```

Although all 5 analog values are available in the Mic-E, some of the channels have dual usage. The following details the configuration of each of these analog inputs:

AD0 - JP2 selects whether this pin is internally connected or routed to the DE-9 connector pin 1. When internal is selected, a simple 10k and 2.4k resistor divider gives battery voltage in tenths. ie, 138 would mean 13.8 volts. 25.5 volts is the highest possible reading. When routed externally, this is a 0-5 volt input only. When external signals are used, the pin is protected for momentary surges up to +/-30 volts.

AD1 - Connected to the MSB of the MESSAGE switch so that it is used to select between SSID and DIGI mode. When the messages 8-F are selected, AD1 is grounded and the message LED glows RED/YELLOW. This selects DIGI mode. Note, when SSID mode is selected, the AD1 pin may still be used as an analog input as long as its value is kept above 008 or about .16 volts. AD1 may be connected to the DE-9 pin 7 by installing a 100 ohm resistor at pads RX. When this is in place, this pin IS NOT PROTECTED and voltages beyond 0 to 5 volts MUST NOT BE EXCEEDED. Analog signals for AD1 must have a low source impedance to swamp out the effect of the 56k pullup resistor. Alternatively, the 56k resistor may be replaced with a 470k value to increase the impedance of this pin, and the Mic-E will continue to operate normally, but it will cause the MSG LED to show RED all the time.

AD2 - Connected to the DE-9 connector pin 8. This pin is protected to +/- 30 volts and is also connected to an internal 8 pin dip socket where the user can add a thermistor for reading temperature. See section below on temperature.

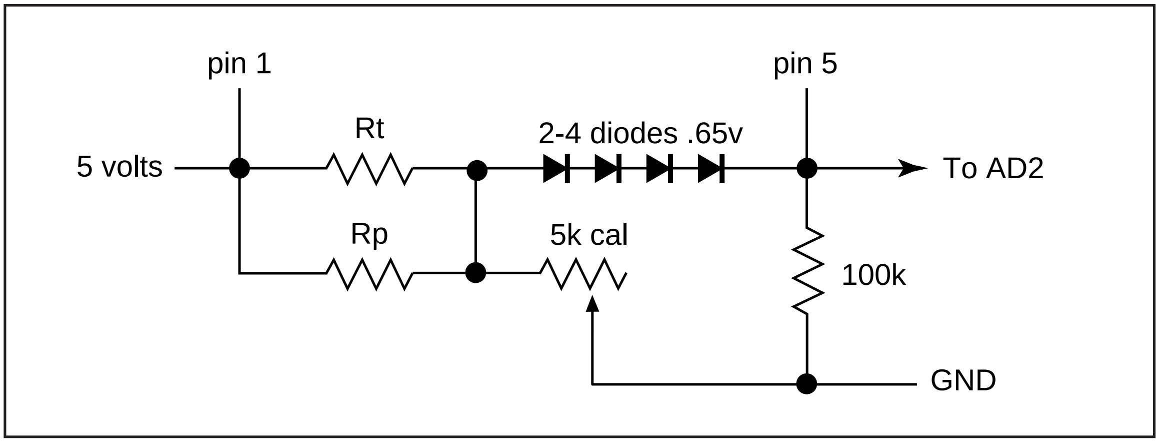
AD3 - Connected to the Period Pot. Useful for reading the setting of this pot or for using the pot as a human entry analog input.

AD4 - Connected to the DE-9 connector pin 6. This pin is protected to +/- 30 volts

PROTOTYPING HEADER

To facilitate analog telemetry experimentation, the AD0, AD1, AD2, and AD4 values are made available on an 8 pin DIP header. Also a 5k pot is available for calibration adjustments. The following temperature circuit will result in analog values equal to the temperature in degrees Fahrenheit (see next page).

Pin numbers are shown for the 8 pin dip header. The 5k calibration pot and the 100k input load are fixed. But by choosing appropriate values for Rt, Rp and the number of diode voltage drops, this circuit can be linearized for many different temperature ranges. A program called MIC-TEMP.BAS which is included in all APRS distributions can be used to select the values of Rp and the number of .65 diode voltage drops to use.



Three useful combinations are:

Rt = 3k ohm Digi-Key part # KC004E-ND $2.09 47-130 deg +/- 2 deg
Rb = 5.6k and 2 diode drops 62-122 deg +/- .5 deg

Rt = 1k ohm Digi-Key part # KC001E-ND $2.17   0-87 deg +/- 2 deg
Rb = 6.2k and 4 diode drops

Rt = 1k ohm Digi-Key part # KC001E-ND $2.17   5-77 deg +/- 2 deg
Rb = 5.1k and 4 diode drops   10-70 deg +/- 1 deg

APRS TELEMETRY RECEIVING SYSTEM

Although the APRS Telemetry page displays default definitions for these five analog values, it can be re-configured over-the-air to show real engineering values and units. This means that APRS does not need to be programmed for each different application. These parameters may be uploaded to all APRS stations live via four one-line BULLETINS. The first one defines the telemetry labels, the second defines the units, the third defines the telemetry equations, and the forth defines the project name and digital bit definitions. For details refer to the TELEMTRY.TXT file included with APRS.

Communicating with the Mic-E (Specification)

This chapter documents the protocol used to communicate between the Mic-E (and the MIM) and the computer during configuration. Most users have no need to read or understand this chapter, as they will use one of the configuration programs that run under DOS or MacOS. It is included for those who wish to develop their own software.

Configuration Software Protocol for the MIC-E:

Command strings are sent to the MIM Module/MIC-Encoder via the serial port using 4800 baud N81 protocol. Each command has the following format (the pipe symbol (|) separates individual portions). The number of bytes in each portion is shown in ():

| 0xFF(1) | command number(1) | data (many) | CRC(1) | **(2) |

Because of this command structure, no data element may be 0xFF (255). Also, an asterisk (*) may be part of the message, but a double asterisk will end the command.

The CRC is calculated using all characters preceding it except the 0xFF. The CRC is formed by adding all preceding bytes, discarding any carries, and negating the result. In other words, the MIC-E will discard the 0xFF and sum all bytes up to (but excluding) the '**'. The result should be 0.

Command Number	Function
0	Software reset
1	Store configuration data (in EEPROM)
2	Store beacon text
3	Store Morse ID (MIM module only)
4	Read configuration data
5	Read beacon text
6	Read Morse ID
7	Initialize EEPROM to default

Data for each command.

Command Number	Data
0	NONE (any text will be disregarded)
1	configuration string (see below)
2	beacon text (up to 80 characters)
3	morse ID (up to 12 characters)
4	NONE (text will be sent from the MIC-E to the PC/MAC)
5	NONE (ditto)
6	NONE (ditto)
7	NONE

Configuration string — number of bytes for each portion shown in ().

```
| AP(2) | auto(1) | quiet(1) | txsync(1) | period(1) | flag bits(1) |
...
| beacon(1) | morse(1) | tele(1) | posit(1) | symbol(1) | txdp(1) | ...
| NMEA string like "GPRMC" (5) | reserved(2) | ...
| callsign(6) | ssid(1) | more?(1) | ...
| via1(6) | ssid1(1) | more?(1) | ...
| via2(6) | ssid2(1) | more?(1) | ...
:
:
| via7(6) | ssid7(1) | 0(1) |
```

Notes on the configuration string:

(1) “period” is base time period in seconds (1-254)

(2) “posit” is the position reporting interval in multiples of “period”. (1-254)

(3) “auto” is the auto time, in multiples of the position reporting interval (which itself is “posit * period”).

(4) “quiet” is the quiet time, in multiples of 5 seconds (0-254).

(5) “txsync” is the number of sync characters to send for automatic position reports. One sync character lasts 8/1200 second, so you can figure out how many syncs to send to give a particular “txdelay”.

(6) Flag bits should be: 000001p0 where p=0/1 is PTT active state. The “1” means MIC-E operation vice MIM Module operation.

(7) “beacon” is the beacon text interval, in multiples of the position reporting interval.

(8) “morse” is the Morse ID reporting interval. Does not apply to MIC-E.

(9) “tele” is the telemetry reporting interval, in multiples of the position reporting interval.

(10) “symbol” is the APRS symbol to display — any ASCII character (0-127)

(11) “txdp” is the number of sync characters to send for position reports which follow voice operation. Typically this is fewer than “txsync” since the repeater is already keyed up.

(12) NMEA string only applies to the MIM module, which can parse any NMEA string. However, you must have these 5 bytes here as place holders.

(13) The “callsign” is 6 characters, padded with spaces (0x20) as necessary.

(14) SSID is one byte, 0-15.

(15) “more?” is a single byte: 0=no via path follows, 1=path follows.

(16) All hops on the via path are coded the same way as the callsign. Each should have it’s “more?” equal to 1 except the last one which is 0, to indicate that no further hops follow.

Note: The callsign and ssid are NOT bit-shifted to the left as AX.25 requires. The MIC-E does this.

Note: The Beacon Text is just ASCII text. The Morse ID does not apply to the MIC-Encoder.

Note: Times like “auto” and “quiet” are NOT ASCII, simply binary.

Example strings (a ‘\’ character preceeds each byte — but don’t include it in the actual string. ‘\x’ is for HEX):

Software reset: ‘\xFF\0\0\**’

Configuration string:

Position reports every 60 seconds (\x3C), telemetry every 2nd posit, beacon text every 10th posit, auto reports every 4 posits, quiet time 15 seconds (i.e., 3 x 5 seconds), 32 sync characters (\x20) for automatic transmissions (txsync), 16 syncs (\x10) following voice transmissions, PTT active low (0), APRS symbol ‘$’, callsign N3XLM-7, via: relay,wide,wb4apr-11.

‘\xFF\x01\A\P\x04\x03\x20\x3C\x04\x0A\x00\x02\x01\$\x10 ...
\G\P\R\M\C\x00\x00\N\3\X\L\M\x07\x01\R\E\L\A\Y\’ ‘\x00\x01 ...
\W\I\D\E\’ ‘\’ ‘\x00\x01\W\B\4\A\P\R\x0B\x00\<CRC>**’

Note that the ‘4’ in WB4APR is an ASCII ‘4’ whereas \x04 is HEX. I did not compute the CRC for this string, but it is a single byte.

Read configuration: ‘\xFF\x04\xFC\**’

(Then be ready to parse the configuration string and display the current hardware configuration.)

APRS Mic Encoder Message Formats

Contributed by Alan Crosswell, N2YGK

This chapter describes the on-air protocol used by the APRS Mic Encoder. You don't need to read it unless you are writing software to decode Mic Encoder packets, for example. Reference is made to the AX.25 standard, *AX.25 Amateur Packet-Radio Link-Layer Protocol Version 2.0*, October 1984 which is available at **http://www.tapr.org/tapr/html/ax25.html**.

APRS and Mic-Encoder are trademarks of Bob Bruninga, WB4APR. Parties interested in commercial developments and commercial sales of the MIC-E both within and outside of the amateur community should contact Clement Engineering. http://www.agrelo.com/clement.html.
It is published here to document the TAPR product and to allow amateur experimentation. This ownership does not intend to restrict individual amateur designs and construction, but it does require that all commercial developments and commercial sales both within and outside of the amateur community must license the trademarks from Bob Bruninga and Clement Engineering who welcome such applications.

The APRS Mic Encoder compresses the APRS position report and message bits into the destination address and information fields of a standard AX.25 UI (unproto) frame. Although the destination address appears to be quite unconventional, it is a valid AX.25 address, consisting only of printable (shifted) 7-bit ASCII values. The result of this compression is a frame 232 bits long which contains:

- Transmitting station's callsign and secondary station ID (SSID);
- Latitude & Longitude in degrees, minutes, and hundredths of minutes;
- Bearing in degrees;
- Velocity in MPH;
- Three binary message bits (8 values)
- A display symbol code (e.g. car, truck, RV, etc.); and
- A specially encoded digipeater path (only understood by "APRS SSID digipeaters").

To this basic message, the following additions are sometimes made:

- [+40..88 bits] Two or five 8-bit analog telemetry values (analog-to-digital values ranging from 0..255).
- [+N bits] Arbitrary beacon text.
- [+56..336 bits] Conventional (non-SSID) digipeater list ("VIA" path).

Figure 1 shows the basic AX.25 UI frame. Figures 2 and 3 describe the special encoding.

Flag	Address	Control	PID	Info	FCS	Flag
01111110	112/560 Bits	00000011	11110000	72... N Bits	16 Bits	01111110
	(see Fig. 2)	UI	no level 3	(see Fig. 3)		

Fig. 1 — Mic-E UI frame

Callsign: Latitude & msg, etc. bits						SSID
Ar1DDDD0	Br1DDDD0	Cr1MMMM0	Nr1MMMM0	Lr1HHHH0	Wr1HHHH0	CrrSSID0

Fig. 2 — Destination Address Encoding

Destination Field

Since the MIC-E is transmitting unconnected (broadcast) AX.25 frames, the destination address does not have to be a callsign-SSID as is required for connected-mode (two-way station-to-station) communications. Conventional APRS transmissions set the destination address to a generic broadcast callsign such as **APRS**. The MIC-E compressed transmission instead makes good use of these seven bytes by embedding the latitude and various flag and message bits as follows:

- Six binary decimal latitude 4-bit nibbles giving degrees, minutes, & hundredths of minutes (DDDD DDDD MMMM MMMM HHHH HHHH).
- [ABC] Three message bits, complemented (e.g. message 5 is 010).
- [N] Northern/Southern latitude bit (N = 1, S = 0).
- [W] Eastern/Western longitude bit (W = 1, E = 0).
- [L] 100's of longitude degrees (L = 1 means add 100 degrees to longitude in the Info field.)
- [C] Command/Response flag (see AX.25 specification).
- [r] reserved for future use.
- [SSID] AX.25 SSID, interpreted as:

0	Use conventional digipeater path, if any
1	WIDE-1 omnidirectional flooding
2	WIDE-2
3	WIDE-3
4	WIDE-4
5	WIDE-5
6	WIDE-6
7	WIDE-7

8	North path
9	South path
10	East path
11	West path
12	North path + WIDE
13	South path + WIDE
14	East path + WIDE
15	West path + WIDE

Information Field

The Information field (text) is used to complete the position report that was begun in the destination address. The encoding used is different from the destination address since the content is not constrained to be printable, shifted 7-bit ASCII, as it is in the address. However, full 8-bit binary is not used: all values are offset by 28 and further operations described below) are performed on some of the values to make the data close to if not completely printable ASCII. This enables one to use a TNC in MONitor mode to display the data without confusion over whether a carriage-return is part of the data or the end of line. Also, software that suffers from the limitations of only accepting 7-bit printable ASCII is accommodated.

NOTE: If the first character of the information field is not one of the four valid MIC-E flag characters (1Ch, 1Dh, 60h or 27h), then the frame is not a MIC-E compressed report and must be interpreted as a conventional AX.25 packet.

Fig. 3a — Information Field Encoding: Mandatory fields

Flag	**Longitude**			**Speed & Course SPD & CSE**			**Symbol**	**Table**
F	D+28	M+28	H+28	SP+28	DC+28	SE+28	$	T

The information field is encoded as follows:

- [F] Mic-E flag, one of:

60h	GPS data is valid.
27h	GPS data is old.
1Ch	GPS data is valid (Rev. 0 beta units only)
1Dh	GPS data is old (Rev. 0 beta units only)

- [D + 28] Binary degrees of longitude. To decode:
  1.subtract 28
  2.subtract 80 if 180 <= D <= 189
  3.or, subtract 190 if 190 <= D <= 199
  4.if the [L] bit is set, add 100 to get final value of longitude.
- [M + 28] Binary minutes of longitude. To decode:
  1.subtract 28
  2.subtract 60 if M > = 60
- [H + 28] Binary hundredths of minutes of longitude. Note that 00..03 hundredths will be unprintable.
- [SP + 28] First part of speed in MPH. Subtract 28 and multiply by 10.
- [DC + 28] Second part of speed and first part of course. Subtract 28 and divide by ten. Quotient is units of speed. Remainder is hundreds of degrees.
- [SE + 28] Second part of course in degrees. Subtract 28, the add to remainder, above.

Finally, make these course and speed adjustments:

- If speed >= 800 MPH, subtract 800.
- If course >= 400, subtract 400.

The remaining fields are:

- [$]Symbol to use (see APRSdos README\SYMBOLS.TXT)
- [T] Primary/alternate symbol table flag (Rev. 1 and higher only), one of:
  2Fh Use primary APRS symbol table.
  5Ch Use alternate APRS symbol table.

The information field is optionally extended either with telemetry values or comments/btext. If the next byte is one of the flag characters (60h, 27h, 1Dh), then it contains telemetry.

.Fig. 3b — Information Field Encoding: Optional telemetry (follows after mandatory fields)

Flag	**Telemetry**				
F	11	22	33	44	55

The flag is one of:
60h 2 printable hex telemetry values follow (channels 1 and 3).
27h 5 printable hex telemetry values follow.
1Dh 5 binary telemetry values follow (Rev. 0 beta units only)

Each telemetry value is a 2 digit printable hex representation of a binary value ranging from 0..255. For example, 254 is represented as ‘FE’. (Except Rev. 0 beta units.)

For comments/btext, the remainder of the information field simply contains textual data as configured by the MIC-E user. You can put in a standard APRS-formatted comment (see APRSdos README\PROTOCOL.TXT) which will cause the APRS display software to override any position data the Mic-E has encoded. This is most useful if you are using a Mic-E without a GPS.