



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



T-238 Operations Manual

This document was written July 2000. © 2000 Tucson Amateur Packet Radio Corp.

Reproduction or translation of any part of this work beyond that permitted by sections 107 or 108 of the 1976 United States Copyright Act (or its legal successor) without the express written permission of Tucson Amateur Packet Radio Corporation is unlawful except as noted below. Requests for permission to copy or for further information should be addressed to Tucson Amateur Packet Radio Corporation. Except as noted above, permission is hereby granted to any non-profit group or individual to reproduce any portion of this document provided

that: the reproduction is not sold for profit, the intent of the reproduction is to further disseminate information on Amateur Packet Radio, the reproduction is not used for advertising or otherwise promoting any specific commercial product, full credit is given to Tucson Amateur Packet Radio Corporation (including address) as the original source of information, and Tucson Amateur Packet Radio Corporation is notified in writing of the reproduction.

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.

Introduction	1
Section 1, Operating the Station.....	2
Section 2, Data Formats.....	5
Section 3, Updating Firmware Using MON08	7
Section 4, Using the Motorola Development Tools	8
Section 5, Connectors and Jumpers.....	9
Section 6, Applications.....	11

Introduction

The Operations Manual describes the operation and programming of the TAPR T238. Use this manual once you have assembled and verified the basic operations of the T238 following the instructions in the Assembly Manual. While the assembly manual can be for any T238 application, this operations manual is specifically for the weather application.

The Motorola MC68HC908GP32 microcontroller that comes with the T238 kit has been pre-programmed with the Dallas Semiconductor 1-Wire[®] APRS[®] Weather Station Application. If you've just finished the last step in the T238 Assembly Manual, you can proceed to the Section 1, Operating the Station in this manual.

Updates and improvements for the Weather Station Application will be posted on the TAPR FTP site at <ftp://ftp.tapr.org/wxsig/software>. You can update the program in your T238 using the DOS utility MON08. Section 3, Updating Firmware Using MON08 explains how.

The T238 is a general purpose microcontroller based platform that interfaces with Dallas Semiconductor's 1-Wire[®] products. The Weather Station Application is but the first of many applications that can be programmed into the T238. Any application can be programmed into the T238. Development tools for the Motorola MC68HC908GP32 microcontroller can be downloaded for free from the Internet. Section 4, Using the Motorola Development Tools tells you how and where.

The T238 interfaces to the outside world via serial connectors and headers. Jumper settings configure the T238 for normal operation or debugging (programming). Section 5, Connectors and Jumpers covers the various connectors, headers, and jumper settings on the T238.

The Section 6, Applications tells you where you can find more T-238 applications.

This manual is written with respect to the Revision 1.12.3 firmware release. See Section 6 for websites and discussion groups for newer firmware and manuals.

DALLAS SEMICONDUCTOR 1-WIRE® APRS® WEATHER STATION APPLICATION

Section 1, Operating the Station

The micro you received from TAPR is pre-programmed with this weather application.

The first step after powering up the T238 for the first time is to setup the Dallas Semiconductor 1-Wire® Weather Station sensor.

After powering up the T238 for the first time, you will get a message asking you to run the setup routines and you will be given the main menus. This will continue to happen until the setup routines have been run. Once you have run the setup routines, the weather station will immediately go into the data-gathering mode when powered up.

The first menu presented is always the “DISPLAY CURRENT DATA” menu. By pressing the Menu button, you can cycle through all the menu options. While displaying a menu option, pressing the Select button performs the action associated with that menu. If this is your first time powering up, you need to go through all the setup screens before selecting the “display current data” option. If you are coming back after setting up your weather station earlier, you can skip the items that don’t apply. The options are:

MENU TEXT	FUNCTION	WHEN TO SELECT
DISPLAY CURRENT DATA (SW Rev x.xx.xR)	Displays the current weather data.	Select last
SET ZULU/LOCAL TIME	Sets the internal clock.	Select on every restart
UART/DISPLAY OPTIONS	Sets various display and function parameters.	Select on first power-up and when you want to change a parameter
SENSOR SETUP	Sets up the weather station wind/temperature sensors	Select on first power-up and when you change sensor units
POSITION STRING SETUP	(Optional) Allows you to specify a position string for APRS packets.	Select on first power-up if you want a position string.

The screens are explained below in the order you should execute them when setting up the weather station for the first time.

“Sensor Setup”

This is the menu that is used to have the T238 (indoor unit) find the serial numbers of the various sensor units (outdoor units). This set-up process must be performed before any weather readings can be made.

When you first press the select button you will be reminded how to run the calibration process for each sensor (by pressing select when it is shown) or skipping it and leaving whatever settings were previously set (by

pressing menu). After that you will be asked if you would like to run the calibration process for each sensor.

Temperature: If only one sensor is present, then the ID is read and stored without any further questions or prompting. If the firmware detects two temperature sensors, it will need direction as to which sensor is the correct one. The most likely scenario for this is if you add an external temperature sensor to the 1-Wire® bus without removing the one that is inside the wind/temperature unit. This is a perfectly OK thing to do, but will require this extra step: You will get a message saying there are two temperature sensors and the IDs of both will be listed. Press the up/down buttons to put the asterisk next to the desired ID. Since in almost all cases you will be replacing an old DS1820 with a newer DS18S20, the ID with an eight in the fifth-to-last digit will be your new sensor. With the asterisk next to the correct sensor, press select.

Wind Speed: For wind speed a potential conflict exists with the rain gauge sensor. If the rain gauge is plugged in, you will get asked to either remove the rain gauge from the bus or select which sensor ID is the correct one. If you can, remove the rain gauge. If you can’t, your best bet is to try one, see if you get reasonable readings, and if not, try the other. Press the up or down buttons to move the asterisk next to the sensor you want and press select to choose it. Once you determine the correct one, it is advisable to write them down somewhere.

Wind Direction: When you push the select button, you will get a warning to make sure you really want to do this. Unless you are really good at reprogramming micros, you must have physical access to the sensor unit to perform this calibration. For initial testing it is fine to perform this calibration routine on a bench, but eventually you will need to do this with the sensor outdoors or know its final outdoor orientation with it sitting on your bench.

Make sure the wind direction sensor is stationary before saying it is OK to run the calibration. Press SELECT to start. If you have an original Dallas wind direction sensor or the early AAG sensors you will first need to “discover” each wind direction ID. If you have one of the later “Rev 3” AAG sensors you will go straight to the north calibration. For discovering each ID, you will be asked to rotate the weather vane clockwise slowly (looking from above). Almost any speed is acceptable as long as you don’t go backwards. Remember that clockwise assumes the sensor is right side up (weather vane down). All eight sensors must be acquired in the correct sequence three times in a row for the calibration to be considered successful. The sensor sequence numbers are displayed as they are acquired.

Errors can and do occasionally happen during this process and this is OK. When an error occurs, the counting will start over again. Several retries is probably OK, but more than that may suggest a problem. After the sequence is done, you will be asked to point the vane north (true north, not magnetic) and press select. This completes the wind/temperature sensor calibration.

After discovering the IDs (or if you didn't need to), you will be asked to tell the T238 where north is. Point the vane north (fin pointing south) and press SELECT.

Rain Gauge: You may have been asked to remove the rain gauge earlier in the process, you will now be asked to install it. The anemometer and rain gauge have the same sensor type (a counter). By removing the rain gauge for the anemometer and getting its ID, this routine goes out, finds the "other" ID, and assumes it is the rain gauge.

It is OK to re-run this calibration routine without a rain gauge installed. If this happens, you will be asked if you would like to remove the rain gauge ID from the sensor list.

In addition to finding out the rain gauge sensor ID, this routine reads the current rain count and stores it as a reference for the "user" rain setting. Whenever you want to reset this user setting to zero, simply re-run this calibration routine.

Humidity Sensor: This sensor has no conflicting addresses, so once confirming you want to run the calibration, it will go look for the sensor ID and if it finds it will store it for future reference. If no sensor is found you will be asked if you want to erase any existing ID in Flash had it existed.

You will also be asked if you want the temperature sensor in the humidity gauge to be the official temperature sensor. Alas for 1.12.3 the question is there, but the feature is not. This feature will be added in the next release.

"UART/Display Options"

Here is where you configure the weather station for various modes and options. When you first enter this mode you are reminded how to navigate these screens. Press Up/Down to adjust the displayed setting and Select to save/continue. The options are:

UART Settings:	
5-min avg (TNC-dir)	Send averaged data every 5 minutes in APRS format. Puts TNC in conversation mode. Expected data rate is 4800 baud.
5-min avg (TNC-LTP)	Send averaged data every 5 minutes in APRS format (For Kantronics KPC3+ LTP command mode) Expected data rate is 2400 baud
5-second live (Computer)	Send raw samples every 5 seconds in APRS format. Expected data rate is 9600 baud.
5-second live (Peet)	Send raw samples every 5 seconds in Peet format. Expected data rate is 2400 baud.

The UART settings are intended to select between the anticipated modes for the weather station. TNC modes average the data and send out only the averaged data every five minutes. Computer and Peet modes just send off each 5-second sample over the UART and let the computer do any data processing. For a full description on the data logging functions of the weather station see the "DATA FORMATS" section.

When you change the data format, the baud rate is automatically set to the expected data rate listed in the table above. If you wish to over-ride this, you can do so on the baud rate menu (next menu).

UART Baud Rate:	
2400, 4800, 9600	Overrides default for a specific baud rate

In general you should use automatically selected baud rate. For most of the data formats the baud rate is part of the defined format. If you do need to select a specific baud rate you can do so with this menu.

Wind Direction Text:	
Numeric Degrees	LCD shows numeric degrees for wind direction
Compass (E, N, etc)	LCD shows compass points for wind direction

The wind direction text is for the LCD. You can choose between the direction being shown as numeric degrees or compass points. This only affects the display, all the transmissions out the UART still use numeric degrees.

Select Units:	
English	LCD shows weather using English units (even though the English use metric now!). Transmitted data still follows units of published specification.
Metric	LCD shows weather data using metric units. Transmitted data still follows units of published specification.

You have the option of displaying your weather data in English or Metric units. This selection affects the LCD only, not the data transmitted by the UART. The data sent by the UARTS continues to follow the units specified by the selected protocol.

"Set Zulu Date/Time"

This menu item is to set up the local time. When you press the select button you will be presented with a date and time field to set up. The date is in the form MM/DD/YY and the time is in the form HH:MM. This is important for several reasons. The main one is that the current Zulu (UTC) time is sent as part of every weather packet sent out, and weather data with the wrong time tag is in error and may be ignored by APRS display programs. For each field, use the up and down arrows to select the correct value (holding the button down will auto-repeat) and press the select button to go to the next field. After finishing the minutes field, you will be asked the current local hour. After that, you are done and get returned to the main menu.

"Position String Setup"

This menu is used to enter a position string. The position string is only used in the TNC formats and is optional. You are first asked if you want a string, and if so, you are asked to enter the string. The format is "ddmm.mmN/ddmm.mmW" (or S/E depending on where you are in the world). The cursor starts at the first digit. Press the up and down buttons to select the digit/letter you want, then press select to go on to the next character. Note that the format assumes decimal notation for the minutes, so 45 degrees, 30 minutes, 15 seconds would be entered as 4530.25.

"Display Current Data"

This menu item gathers weather data, displays it on the LCD, and sends it over the serial port. How it does all this depends on how options are selected before. While showing data, pressing the select button cycles through various display modes. These modes are:

Currently	
Temp 62°F L 46 H 76	Current Temperature, low/high since midnight
Wnd WSW 2M G 5 x 10	Wind direction, Wind speed, Max speed last 5 mins. and max speed since midnight.
Rain 0.00" 65%RH	Total rain since midnight, relative humidity.

Yesterday	
Temp L 49°F H 61°F	Yesterday low and high temperature
Wind Gust 14 MPH	Yesterday peak gust
Rain 0.00"	Yesterday total rain

Rain	
Today 0.00 Yest 0.00	Rain since midnight, total rain yesterday
Month 0.00 User 0.00	Rain since start of month, Rain since last calibration
Hour 0.00	Rain in last 60 minutes

Status	
06/04/00 01:59 Zulu	Current Month, day, year, hour, minute UTC
18:00 Lcl Bus:5.00v	Current local hour, Bus voltage
Sample: 10/60	Intervals processed / intervals in 5 minute period

Error Counters	
Temp 000 Rain 000	Temperature CRC errors, Rain CRC errors
Wspd 000 Wdir 000	Wind speed CRC errors, Wind direction CRC errors
Hum 000 016400910601	Humidity CRC errors, Humidity sensor raw data

For any screen, the data is updated every five seconds. If there are communications errors, the data that got corrupted will be replaced with dashes and that sensors error count will increment. Ideally, this should be rare, (maximum of about 10 per day). If it does start to happen often, you should try to see what you can do to correct the problem. The counters roll back to 0 after a count of 63

While sensor errors are possible (and to a small extent probable). Hopefully you should NEVER see any erroneous data transmitted, all data is checked with CRCs to ensure accuracy. Note that your very first wind speed reading will be dashes as two consecutive wind speed counter readings are needed before a wind speed can be determined. This is not an error and does not increase the wind speed error counter.

Bus voltage is a measurement of the 1-wire bus voltage as it leaves the T238. There are several failure modes where this voltage can be low. If your displayed voltage is above 4.00V you should be OK. A voltage between 4V and 3.5V is marginal, and anything less than 3.5V will likely result in many errors reading data. If you have a low bus voltage your best hope is to selectively remove 1-wire devices from the bus to isolate which device is causing the problem and then repair or replace that device.

As a temporary debug effort the raw humidity sensor data is displayed on the debug screen. This will go away in future releases.

DS1994 Real Time Clock Support

The DS1994 is an I-button device that contains both a real-time-clock and some non-volatile RAM. If a DS1994 is on the 1-wire bus during power up and the clock is programmed to be running, the T238 firmware assumes the device has the correct time and the local clock is set to match the DS1994 time. A short message is displayed during power up noting that the time was obtained from the DS1994. This is particularly handy if you experience regular power outages and do not want to reset the clock every time. The RAM is currently not used, but some data logging functions are envisioned.

When you program the DS1994 using your computer, set the time on the DS1994 to UTC. After powering up for the first time using the DS1994, re-run the set-time routine once more to set the correct local time. This saves the UTC to local time offset in permanent memory and then you should never need to set the clock manually again.

Missing Rain Gauge Support

If the rain gauge sensor is missing during power-up a short message is displayed noting the missing sensor and rain data is ignored (and error counts do not increment) until power is cycled again. This feature allows the rain gauge to be removed for the winter in the colder climates without having to recalibrate sensors to remove then re-add the sensor during the spring.

Section 2, Data Formats

The weather station currently supports sending data in two basic formats using four specific modes. Selecting which format you want is done using the "UART/DISPLAY OPTIONS" menu and selecting which format you want. They are listed below:

Selection	Recipient	Data Format
5-min averaged (TNC-dir)	TNC	APRS Format
5-min averaged (TNC-LTP)	TNC	APRS Format
5-second live (Computer)	Computer	APRS Format
5-second live (Peet)	Computer	Peet format

Note that whatever mode is selected, the data displayed on the LCD is unaffected.

5-Minute Averaged (TNC-Dir) Mode

The TNC mode is intended for just that, hooking up to a TNC. If your intentions are to hook up to a computer, you can safely ignore this section, it is pretty much for ham radio folk only. In this mode, when you start gathering data, commands are sent out to the TNC to initialize it as a weather broadcaster. The TNC is programmed for the un-attached conversation mode so that whenever a string of data is sent to the TNC, it immediately transmits it.

Except for the newer Kantronics KPC3+'s, the GPS logging mode found in most newer TNCs is not used as it is not ideal for this application. (See next section for the KPC3+) The logging modes on TNCs assume that data is constantly being fed to the TNC and at the appropriate interval, the latest complete set of data received is transmitted. For GPS operations this is fine, but for weather operations averaged data is much more valuable. If the weather station transmitted averaged data every five minutes and the TNC was set up to transmit data every five minutes, when they got out of sync, the data transmitted would be nearly 10 minutes old.

Forgive me a small soapbox: AVERAGED data is important! This is especially true for wind direction. An instantaneous wind sample at a particular moment in time is useless. If you look at wind vane on any weather station, you can see how much it varies over a period of even a few seconds, never mind five minutes. Taking a snapshot once every five minutes (like you get with all the off-the-shelf amateur weather stations) is horribly inaccurate. For this weather station we go to great lengths to figure out a good averaged wind speed that is much more representative of what the real wind direction is. We now return to the regular dispassionate manual.

In the un-attached mode, the TNC is in conversation mode all the time but not attached to any other TNC. As such, it sends the data to the UNPROTO destination you programmed into your TNC. This should be something like "APRSW via {your favorite path}". When five minutes of data have been averaged, the weather station sends

the data to the TNC and it is transmitted as soon as the TNC finds an opening in the channel.

When the weather station is in TNC mode, it will try to initialize the TNC when you tell it to start displaying data. The sequence consists of flushing any pending data out of the TNC and sending four commands and a version string to the TNC. The sequence is:

^C	to make sure we are in command mode
Monitor off	to keep data from coming back into the TNC
Echo off	to also keep data from coming back into the TNC
Convers	to get into conversation mode
Wx start (x.xx.x)	broadcast the current firmware version.

If you also use the TNC for packet messages and the like, you are going to need some kind of data switch for the TNC to select between the computer and weather station. When you change over to the computer, you will need to issue two commands to get the TNC back into a computer-friendly mode. The first is a "AC" to get back to command mode and finally "echo on" to be able to see what you are typing. Setting a function key to send this script is particularly handy.

While in display mode the weather station sends out data to the TNC. The data is sent out every five minutes or 60 samples. The data consists of the averaged wind direction, speed, and temperature. The string transmitted follows the APRS weather format and looks like the following:

_MMDDHHMMcxxxxsxxxgxxxTxxxPxxxxrxxxelw where

_	is the character to flag weather data
MMDD	is the month and day (Zulu)
HHMM	is the hour and minute (Zulu)
cxxx	is wind direction (degrees)
sxxx	is wind speed (MPH)
gxxx	is wind gust (MPH)
Txxx	is temperature (Farenheight)
Pxxx	Is rain since midnight (1/100ths of an inch)
rxxx	is rain past hour (1/100ths of an inch)
hxx	is the humidity
Elw	is the project identifier (Embedded, 1-wire)

The only part of the string that may not be 100% APRS-compatible is the project identifier. The current spec does not include an option for a project such as this (choices are PIC-E, Dos, Win, Mac, and Linux). The APRS folk acknowledge this small hole in the spec and will update the spec soon. Until then I will leave the existing format.

The baud rate chosen for TNC mode is 4800 baud. This was chosen primarily because that is what most of the

GPS receivers use and we wanted to be able to use the same baud rate everywhere.

5-Minute Averaged (Tnc-Ltp) Mode

This mode is very similar to the 5-Minute Averaged (TNC-dir) mode. This mode is specifically for the Kantronics KPC3+ TNCs that have a firmware revision greater than 8.2. These TNCs essentially have a second serial port that can accept input-only data while leaving the main serial port for other uses. This port was intended for a GPS input, but works just fine for the weather input as well. The data sent in this mode is identical to the direct mode in all ways except that it does not send the initialization strings, only the weather data.

5-Second Live (Computer) Mode:

In computer mode only raw (un-averaged) data is transmitted. The data is sent out every five seconds at 9600 baud. This mode is much simpler than the TNC mode as there are no special initialization commands. All the intelligence is assumed to be in the computer. The data transmitted is very similar to the APRS format used in TNC mode except the gust data is omitted. The format is as follows:

`_MMDDHHMMcxxxxsxxxtxxxPxxxxrxxxe1w` where

<code>_</code>	is the character to flag weather data
<code>MMDD</code>	is the month and day (Zulu)
<code>HHMM</code>	is the hour and minute (Zulu)
<code>cxxx</code>	is wind direction (degrees)
<code>sxxx</code>	is wind speed (MPH)
<code>txxx</code>	is temperature (Farenheight)
<code>Pxxx</code>	Is rain since midnight (100ths of an inch)
<code>rxxx</code>	is rain past hour (100ths of an inch)
<code>e1w</code>	Is my project identifier (Embedded, 1-wire)

DEBUG STRINGS

For both the TNC and computer modes, debug strings are added on to the end of the APRS string for debug purposes. The Peet format does not support additional data and does not support this feature. The debug string consists of a multi-byte hexadecimal string after the "e1w" identifier. The first byte is the message identifier and the subsequent bytes are the data bytes. Currently only two message types exist.

Header	Data
00	aa bb cc Firmware revision aa is huge rev (01 for 1.11.0) bb is major rev (0b for 1.11.0) cc is minor rev (00 for 1.11.0)
01	aa bb cc dd ee Error counters aa = Temperature error counter bb = Wind Speed error counter cc = Wind Direction error counter dd = Rain Guage error counter ee = Bus static voltage (ff=5.00V)
02	aaaa bbbb cccc Humidity Raw Data aaaa = Sensor VCC Value bbbb = Sensor Output Value cccc = Sensor Temperature

5-Minute Live (Peet) Mode:

This mode is intended to emulate a Peet weather sensor so other computer programs (and standalone displays) that expect the Peet format can accept data from this weather station. The format I intend to support is called the "data logging mode". This mode sends data in the following format:

`!!aaaaabbbbccccdddeeeffffgggghhhhiijjjkkkk` where

<code>!!</code>	is the header
<code>aaaa</code>	is wind speed (.1KPH)
<code>bbbb</code>	is wind direction (00-ff)
<code>cccc</code>	is Temperature in (.1 degrees F)
<code>dddd</code>	is Long term rain (---- for now)
<code>eeee</code>	is Barometric pressure (---- for now)
<code>ffff</code>	is indoor temperature (---- for now)
<code>gggg</code>	is outside humidity
<code>hhhh</code>	is inside humidity (---- for now)
<code>iiii</code>	Date (day of year)
<code>jjjj</code>	Time (minute of day)
<code>kkkk</code>	Rain since Midnight (100ths of an inch)

All data fields are 4-digit ASCII hex values. For full details on this format, consult <http://www.peetbros.com> and select "Serial data specs." On that sheet, look for the "Data Logging mode" section and all will be revealed.

Section 3, Updating Firmware

Using MON08

NOTE: The micro in the TAPR T238 Kit is pre-programmed. This section is needed if you are building a T238 from scratch or you have a firmware update.

There are actually two programs inside the micro. The first is a downloader that allows updating the weather station code without having to use all the Motorola tools. The second is the weather station program itself that gets loaded by the downloader. You can think of the downloader as a miniature operating system and the weather station code as an application. If you got a kit from TAPR, the micro already has both programs loaded and is ready to run. If you have a working weather station and just want to upgrade to the latest version of weather station code, you only need to download new code. If you are building this weather station from scratch or have corrupted the downloader, then you are going to need to completely reprogram the micro using the (free) Motorola development tools.

Downloading New Code:

To just update the weather station code, you can use this simpler download process. You will need two files from the website. The first file is the latest object code, a file called WX08.S19. You will also need the DOS program MON08.EXE. After downloading these two files, reboot your computer in DOS mode (not a DOS window!) and be in a directory that has both these files.

- 1) Power to the weather station should be off.

- 2) Make sure the jumpers for OJ1 and OJ2 to the 1-2 position (this is the normal position).
- 3) Attach a serial cable from your computer to port J4 on the weather station.
- 4) Press and hold down the up and down buttons.
- 5) With those buttons pressed, apply power to the controller
- 6) You should get an LCD message stating the monitor version number.
- 7) On your computer, enter MON08 WX08.S19 followed by 1 or 2 for your com port number.
- 8) The MON08 program should now program the new code into your micro.
- 9) When complete the weather station should automatically start to run.

The two most likely places for problems are in steps 6 and 8. If you do not get a monitor message on power-up with the up and down buttons pressed, make sure all the jumpers and connectors are in the right place. If all checks out and you still do not get the message, then the monitor may have been corrupted for some reason. If this is the case, you have no choice but to use the Motorola tools to reprogram the entire micro. The second place for problems is during the programming process. It is essential to be in DOS mode, not a DOS window for the downloader to run. If you run into problems, just try a few times. If that still doesn't work, then your only recourse is to use the Motorola tools.

The DOS downloader isn't the greatest way to download new code, but for now it is all I have. A windows-based program would be nice, but is still only a concept. I'm not a windows programmer. Any volunteers?

Section 4, Using The Motorola Development Tools

In addition to being a ready-made kit with the weather application pre-programmed into the microcontroller, all the necessary hardware is provided for the more adventurous people to develop their own applications. A list of available development tools for the MC68HC908 can be found at http://mcu.motps.com/dev_tools/index.html. Free development environment is available from P&E Microsystems at <http://www.pemicro.com/ics08/>. Download the file "ICS08GP Software for 68HC908GP20/GP32" which contains the following programs:

WINIDE08 – Integrated Development Environment w/
Assembler
PROG08SZ – MON08 Flash/EEPROM Programmer
ICS08Z – In-Circuit Simulator
ICD08SZ – Real Time In-Circuit Debugger

If you are starting with a blank micro, wanting to do some of your own code development, or the monitor program has been corrupted somehow, you will need to use the Motorola development tools. The toolset can be found by following the 6808 link on the links page. It is a nice toolset with plenty of decent documentation. Describing the toolset is way beyond the scope of this manual,

however. Here is the basic data you need: The debug port is J3 and is fully RS-232 compatible. OJ1 selects the voltage to the IRQ pin. For debug mode you want VPP, so short pins 2 and 3. For normal operation, short pins 1 and 2. OJ2 selects the clock frequency for the micro. For debug operations you need 9.8304MHz, so short pins 2 and 3. For normal operation you want 32MHz, so short pins 1 and 2.

If you want to start playing with the source code, you can still run and debug the code. At the beginning of the WX08.ASM file is an equate called DEBUG. Setting that equate re-times all the 1-Wire and UART communications to use the 9.8304MHz clock versus the 32MHz clock. This mode also leaves the reset vector blank so you can always come up in debug mode without having to resort to VPP all the time.

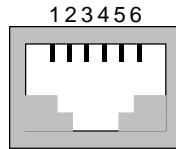
Note that the weather station stores the serial numbers for all your sensors in flash memory. When you use the debug tools and erase the flash, you are also going to erase all your sensor calibration values too. One option you have (besides doing the calibration every time) is to use the tools to save your data and then re-program it back in later. The source-code file wireids.asm is where these IDs should be located. In that source file are instructions on how to put your IDs into that file to save you having to re-learn the IDs every time you reprogram the Flash.

Section 5, Connectors And Jumpers

This section describes all the interfaces and options for the T238 board.

J2: 1-Wire Interface

Jack J2 is the 1-Wire interface. It is a 6-6 RJ-11. The pin-out of J2 is:



Looking into RJ-11 jack

Pin Number	Function
1	No-connect or +5V
2	Ground
3	Data
4	Ground
5	+5V
6	No-connect

J3: Debug Interface

J3 is the debug interface. It is intended to be hooked up to your computers serial port when you are writing and debugging new software. It is also your last resort backup if the downloader code in the microcontroller code gets corrupted. If you are not writing or debugging programs, this port should never be needed. J3 is configured as a Data Communication Equipment (DCE) serial port (i.e. modem port). Any Data Terminal Equipment (DTE) such as a computer can connect directly pin-for-pin to the port. The pin numbers are labeled on the connector. The pin functions are as follows:

Pin Number	Pin Function
1	Signal detect (N/C)
2	Tx Data out
3	Rx Data in
4	DTR (N/C)
5	GND
6	DSR (always high)
7	RTS (N/C)
8	CTS (always high)
9	Ring Indicator (N/C)

J4: Computer Data Connector

J4 is the data interface to the computer. If you want the data from the weather station to go to a computer, use this connector. You should be able to plug in a normal serial cable between this connector and your computer. Use either this header or J5, do not use both at the same time. J4 is configured as a Data Communication Equipment (DCE) serial port (i.e. modem port). Any Data Terminal Equipment (DTE) such as a computer can connect directly

pin-for-pin to the port. The pin numbers are labeled on the connector. The pin functions are as follows:

Pin Number	Pin Function
1	Signal detect (N/C)
2	Tx Data out
3	Rx Data in
4	DTR (N/C)
5	GND
6	DSR (always high)
7	RTS (N/C)
8	CTS (always high)
9	Ring Indicator (N/C)

J5: TNC Data Connector

J5 is the data interface to the TNC. If you want the data from the weather station to go to a TNC, use this connector. You should be able to plug in a normal serial cable between this connector and your TNC. Use either this header or J4, do not use both at the same time. J5 is configured as a Data Terminal Equipment (DTE) serial port (i.e. computer port). Any Data Communications Equipment (DCE) device can connect pin-for-pin to the port. The pin numbers are labeled on the connector. The pin functions are as follows:

Pin Number	Pin Function
1	Signal detect (N/C)
2	Rx Data in
3	Tx Data out
4	DTR (N/C)
5	GND
6	DSR (N/C)
7	RTS (N/C)
8	CTS (N/C)
9	Ring Indicator (N/C)

Headers

J6: Expansion Header

J6 is a header with all the unused (or slightly used) pins from the microcontroller along with some other signals we thought might be handy. If you want to add extra hardware to the main board, we hope we included all the signals you need. Here are the signals available:

Pin	Function	Pin	Function
1	Port B, bit 7	2	Port C bit 0
3	Port C bit 1	4	Port C bit 2
5	Port C bit 3	6	Port C bit 5
7	Port D bit 0	8	Port D bit 1
9	Port D bit 2	10	Port D bit 3
11	Port D bit 5	12	N/C
13	32MHz Clock	14	1-Wire data
15	+5V	16	GND

J7: LCD Header

J7 is the header for the Liquid Crystal Display. A standard cable should be able to plug between this header and an LCD with a 2x7 header on it. If the LCD you have is a 1x14 header, then you will need an adapter. To save microcontroller pins, the interface to the LCD uses the 4-bit mode instead of the 8-bit mode, so the lower four data pins are not connected. The signals are as follows:

Pin	Function	Pin	Function
1	GND	2	+5V
3	VLC (contrast adjust)	4	Register Select
5	Read/Write	6	Enable
7	N/C	8	N/C
9	N/C	10	N/C
11	Data 4	12	Data 5
13	Data 6	14	Data 7

J8: LCD Backlight

This two-pin header is for the LCD backlight. Pin 1 is the positive voltage, Pin 2 is the negative voltage. R13 is the current limit resistor if you are using an LED backlit display. Pick R13 per the specifications for the display if it is different from the one supplied by TAPR.

J9: Front Panel

Use this connector if you want to have the four switches and LED on a separate panel that is not part of the main board. This is intended for a board in a case where you want to mount the switches and LED to a faceplate instead

of using the locations on the main board. The pin functions are as follows:

Pin	Function	Pin	Function
1	Up switch	2	Down Switch
3	Select switch	4	Menu Switch
5	Heartbeat LED	6	N/C
7	N/C	8	N/C
9	+5V	10	GND

Jumpers

The following tables summarize the jumper selections on the T-238.

OJ1, OJ2 – Programming Selection:

OJ1 and OJ2 are for selecting between the “normal” and debug modes of the microcontroller.

Option Jumper	1-2 position	2-3 position
OJ1 – Clock Select	32MHz (Normal mode)	9.8304MHz (Debug mode)
OJ2 – IRQ Voltage	+5V (Normal mode)	+9V (Debug mode)

MODE	OJ1	OJ2
Normal Operation	1-2	1-2
Debugging	2-3	2-3

Section 6, Applications

This manual covers the first application of the T-238 board. There is nothing about the hardware that makes it a weather station nor anything that prevents you from creating your own applications. The full source code including all the drivers for the 1-Wire[®] bus, LCD, and UART are available to help you create your own applications. As you or anyone else develops these programs, we encourage you to let TAPR know so we can add them to the list of available applications for this board.

As T-238 programs are developed they will be displayed or linked from the TAPR Weather web site located at <http://www.tapr.org/taprf/html/Fwx.html>. If you've created

a project and would like to display or link to it, contact TAPR at tapr@tapr.org.

Updated documentation can be found at <http://www.tapr.org/taprf/html/Fwx.html> or <ftp://ftp.tapr.org/wxsig/docs/>.

Contributed files can be found at <ftp://ftp.tapr.org/wxsig/software/>. Files can be anonymously uploaded to <ftp://ftp.tapr.org/wxsig/upload/>.

T-238 is discussed on the WXSIG Special Interest Group. You can join WX SIG via the web at <http://www.tapr.org/cgi-bin/lyris.pl?join=wxsig>.