

**Table of Contents**

<b>Programming.....</b>	<b>1</b>
<b>Menu Selection &amp; Programming Controls.....</b>	<b>1</b>
<b>Menu Alphabet.....</b>	<b>2</b>
<b>Entering Program Mode.....</b>	<b>2</b>
<b>Menu Map.....</b>	<b>2</b>
<b>Options Menu.....</b>	<b>6</b>
<b>Calibrate Menu.....</b>	<b>7</b>
<b>Alarm Menu.....</b>	<b>9</b>
<b>Miscellaneous Menu.....</b>	<b>10</b>
<b>Serial Port Menu.....</b>	<b>12</b>
<b>Service Menu.....</b>	<b>12</b>
<b>Formula Menu.....</b>	<b>12</b>
<b>Log Menu.....</b>	<b>13</b>
<b>APPENDIX A.....</b>	<b>15</b>
<b>GPS SERIAL OUTPUT FORMATS.....</b>	<b>15</b>
<b>APPENDIX B.....</b>	<b>19</b>
<b>LOG MEMORY DETAILS.....</b>	<b>19</b>
<b>APPENDIX C.....</b>	<b>20</b>
<b>PROGRAMMING CHECKLIST.....</b>	<b>20</b>
<b>APPENDIX E.....</b>	<b>22</b>
<b>SOFTWARE VERSION DETAILS.....</b>	<b>22</b>

**NOTE TO THE USER**

The μMONITOR manuals are open to revision based on your needs. If you have suggestions for improvement or clarification, please write or call.

Rocky Mountain Instrument  
 PO Box 683 or 493 Hwy. 20 N  
 Thermopolis, WY 82443  
 (307) 864-9300  
 techsupport@rkymtn.com



## Programming

Rather than fix functions to certain design limits, the  $\mu$ MONITOR allows access to most of the many variables that mold the instrument to fit the performance envelope of different aircraft, as well as personal preferences of the pilot.

Programming, as used here, is more akin to choosing COOK or DEFROST on a microwave oven rather than the art/science of computer programming, so don't face this section with apprehension.

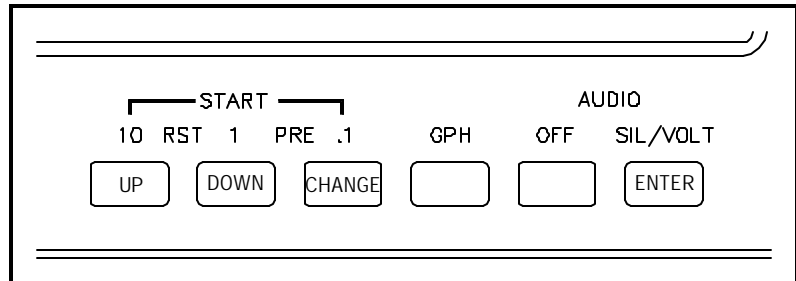
The majority of this section will only be used once—after installation. There will be portions that will be used to fine-tune the  $\mu$ MONITOR as experience dictates.

It is recommended that you first read the operations section of this manual that provides descriptions and/or definitions necessary for understanding the options and limits that can be selected during programming. It also helps to fill in Appendix C to use as a programming aid.

## Menu Selection & Programming Controls

Once the programming mode is entered, menu selection makes it easy to find the desired option or limit to be changed or operation to be performed. In the  $\mu$ MONITOR, selection from the first menu leads to another menu where another choice is selected. After two or three selections from the menus, the actual operation will be performed.

Once in the programming mode, only four controls are used: **[UP]**, **[DOWN]**, **[CHANGE]**, and **[ENTER]**. Four of the  $\mu$ MONITOR's buttons are renamed as shown.



**[CHANGE]** performs two functions; 1) rotating through the menu selections and; 2) changing digit values where data entry is necessary. Think of it as MENU SELECT and CHANGE DIGIT VALUE.

**[DOWN]** is a menu selector. Pressing it at a particular menu item leads down to the next menu level. Think of it as DOWN into menus.

**[UP]** is a conclude signal and acts opposite to **[DOWN]**. Pressing this button backs up through the menus so another choice can be made, and eventually returns the  $\mu$ MONITOR to normal operation. Think of it as BACK out of menus. When pressed in the middle of data entry, it also acts like an escape key on a computer, leaving that partial operation without changing anything.

**[ENTER]** is used to start the procedure of entering a number and stepping to the next digit. Think of it as START and NEXT.

## ***Menu Alphabet***

The menu selections during programming are shown as one to five letter words, abbreviations, or acronyms that will give the user as much information as possible in the small amount of space available. Some letters are difficult or impossible to represent on a display that is designed for numbers. Use your imagination when interpreting messages. Some letters like L work well while others like T are converted to lower case t. The lower case letter t still is not perfect—but close. Letters like K that are impossible are not used. The alphabet and numbers used on the display looks like this:

***A b C d E F G H I J \_ L N n O P \_ r S t U u \_ \_ y \_ 0 1 2 3 4 5 6 7 8 9***

In particular, note the difference between the letter B (shown on display as lower case b) and the number 6. Although the M and N shape are the same, the M is the larger. The letters U and V follow the same rule.

Throughout this manual letters that depict what is appearing on the display will look as close as possible to what the screen will show. For example, ALr uses the upper case “AL” and lower case “r” just like in the alphabet shown above.

## ***Entering Program Mode***

The program mode can be entered at any time when the μMONITOR is operating in its normal mode. There are two steps from normal operation to programming mode. The first step is to enter the backlight setting mode as detailed in the operations manual, by pressing the three grey/white 10, 1 and .1 buttons at the same time. Once in the backlight setting mode, and with the light intensity being shown on the display, pressing all three grey/white buttons again completes the second step. The second pressing of the three buttons must be completed without first pressing any of the backlight adjust buttons, and before the three second time-out lapses. The display then changes to show **OPt** in the RPM portion of the display. The **OPt** is one of the 1st level menu choices. The control combination was chosen to be awkward so there is no possibility of entering program mode by mistake. NOTE: It is recommended that the function switch always be set on LMT before trying to enter program mode. That way, an inadvertent, sloppy press of other than all three buttons won't result in an accidental change of a function that is hard to correct.

## ***Menu Map***

When the only item on the display is a three letter abbreviation in the lower left, the μMONITOR is in the 1st level menu. Pressing **[CHANGE]** will rotate through different 1st level menu selections. When the upper menu item is reached, the list will repeat over. The three letter 1st level menu selections as they appear on the display and their meanings are:

<b>OPt</b>	<b>CAL</b>	<b>ALr</b>	<b>MSC</b>	<b>SEr</b>	<b>Srv</b>	<b>FOr</b>	<b>LOG</b>
Options	Calibrate	Alarm	Misc.	Serial	Service	Formula	Log

At this point, pressing **[CHANGE]** will change 1st level menu selections, pressing **[DOWN]** will enter the 2nd level menu group that belongs to the 1st level selection displayed and pressing **[UP]** will return to normal operation.

At the 2nd level, menu selection is continued using the **[CHANGE]** button. All 2nd level menu items will

appear on the display in the clock position. Pressing **[DOWN]** at this point will enter the 3rd level where the selections or entries are made. Pressing **[UP]** will return to the 1st level.

At the 3rd level, menu selection and entries are made using the **[CHANGE]** and **[ENTER]** buttons. Where possible, the 3rd level menu items are shown in a third portion of the display. However, because of the limited space, some of the 3rd level menu items replace the 2nd level items in the clock portion of the display so you can only see the 1st level and 3rd level menus. See the examples at the end of this section explaining the two methods of changing/entering new options or values.

The menu map below contains a description of the function that will be changed by each selection. 2nd/3rd level menu selections that need additional explanation are described more fully later in this section of the manual.

1st level menu	2nd level menu	3rd level menu/action	Description
OPt (Options)	PERCt	Calibrate Pwr%	In-flight calibration of Pwr%
	tEMP	C or F	Temperature readings in degrees C or F
	FULty	G or L	Fuel in Gallons or Liters
	TPUL	1, 2, 3, 4 & 8	Number of tachometer pulses per revolution
CAL (Calibrate)	CHt	Calibrate/disable	Cylinder head temperature
	Ot	Calibrate/disable	Oil temperature
	MAP	Calibrate/disable	Manifold pressure
	AMP	Calibrate/disable	Ammeter
	OP	Calibrate/disable	Oil pressure
	FP	Calibrate/disable	Fuel pressure
	Ct	Calibrate/disable	Carburetor temperature
	vOLtS	Calibrate	Voltage
	OAt	Calibrate/disable	Outside air temperature
	EGt	Calibrate/disable	Exhaust gas temperature
ALr (Alarm)	CHt	view/enter high & low cylinder head temperature alarms	
	Ot	view/enter high & low oil temperature alarms	
	MAP	view/enter high & low manifold pressure alarms	
	AMP	view/enter high & low ammeter alarms	
	OP	view/enter high & low oil pressure alarms	
	FP	view/enter high & low fuel pressure alarms	
	Ct	view/enter high & low carburetor temperature alarms	
	rEdLn	view/enter high RPM redline alarm	
LOFUL	view/enter low fuel quantity low alarm		
MSC (Misc.)	FLPrE	view/enter full fuel quantity preload	
	FFFAC	view/enter fuel flow sensor K-factor (pulses per 1/10 gallon or liter)	
	TTIME	view/enter beginning tachtime	
	TTrPM	view/enter tachtime cruise RPM	
	TMPRE	view/enter countdown timer preload value	
	HObbS	view/enter beginning Hobbs hours	

	SrLnO	view/enter serial number	
	CLndr	view/enter year, month and day of the internal real-time clock	
SrEr (RS232)	OUtPt	OFF, Gar, SHF, SHAd	select serial port transmission mode
	InPUt	OFF, RMI	select serial port receive mode
	bAUd	1200 to 19200	select serial port baud rate
Srv (Service)	vEr	view software version number	
	SHO1	display static in-flight display	
	dSPly	display all segments and characters of the display	
	bEEP	activate the headset and cockpit audio alarms	
	UPLOd	field upgrade/upload of μMONITOR software	
FOR (Formula)	CHt	change cylinder head temperature formula for calculations	
	Ot	change oil temperature formula for calculations	
	MAP	change manifold pressure formula for calculations	
	AMP	change ammeter formula for calculations	
	OP	change oil pressure formula for calculations	
	FP	change fuel pressure formula for calculations	
	Ct	change carburetor temperature formula for calculations	
	Oat	change outside air temperature formula for calculations	
	EGt	change exhaust gas temperature formula for calculations	
LOG (Log)	LiME	1, 2, 3, 4, 5, 6, 10	select minutes between logging data to memory
	dUMP	UP	dump logged data to RS232 serial port
	CLr	UP	clear log memory

**TO CHANGE MENU LEVELS** - Using the previous menu map as a reference, pressing **[DOWN]** moves to the right and pressing **[UP]** moves to the left. Pressing **[UP]** when at the 1st level returns the μMONITOR to normal operation.

**TO CHANGE SELECTIONS AT EACH LEVEL** - Press **[CHANGE]** and observe the menu selection change on the display. The order of menu selections is as they are shown in the menu map. A selection can be reached only by stepping through the menu list until that selection shows on the display. If you go past a desired selection, you will have to continue until reaching the last item when the list will repeat.

The 1st level menu appears in the first three RPM digits in the upper part of the display. When at the 1st level the rest of the display is blank and **[CHANGE]** will change 1st level selections.

The 2nd level menu appears in the clock digits in the lower center of the display. When at the 2nd level only the 1st and 2nd level menus are on the display and **[CHANGE]** will change 2nd level selections.

Any other display format will indicate that the 3rd level is active. Sometimes the 2nd level menu display in the clock position will be replaced by the function being shown or entered.

The actual program changes are made at the 3rd menu level. The 1st and 2nd menu levels are just a means to get to the proper function to change.

Following are programming examples that represent the two ways of changing program data. The examples assume that the program mode has just been entered and that the 1st level selection **OPT** is already on the display.

**EXAMPLE 1** – How to change the fuel flow sensor K-factor from 2800 to 2850.

Press **[CHANGE]** three times. The 1st level menu changes to **MSC**. Press **[DOWN]** to select miscellaneous as the 1st level menu to use. **FLPrE** appears in the clock position. Press **[CHANGE]** once and **FFFAC** appears as the 2nd level menu selection. Press **[DOWN]** to select fuel flow sensor K-factor as the 2nd level menu to use. **0 2 8 0 0** appears on the display in place of the 2nd level menu selection. If you just wanted to see what the current K-factor is, you would return to the 2nd level menu at this point by pressing **[UP]**. The following will finish the entry:

action	display	comment
press <b>[ENTER]</b>	<b>0 - - -</b>	begin the entry procedure (1st digit already at zero)
press <b>[ENTER]</b>	<b>0 0 - -</b>	leave 1st digit at zero and start 2nd digit
press <b>[CHANGE]</b>	<b>0 2 - -</b>	two presses changes 2nd digit to 2
press <b>[ENTER]</b>	<b>0 2 0 - -</b>	leave 2nd digit at 2 and start 3rd digit
press <b>[CHANGE]</b>	<b>0 2 8 - -</b>	eight presses changes 3rd digit to 8
press <b>[ENTER]</b>	<b>0 2 8 0 -</b>	leave 3rd digit at 8 and start 4th digit
press <b>[CHANGE]</b>	<b>0 2 8 5 -</b>	five presses changes 4th digit to 5
press <b>[ENTER]</b>	<b>0 2 8 5 0</b>	leave 4th digit at 5 and start 5th digit
press <b>[ENTER]</b>	<b>F F F A C</b>	leave 5th digit at 0 and go back to 2nd level menu

At this point press **[UP]** to return to the 1st level menu or make additional 2nd level menu selections. At any time before the final **[ENTER]** on the last digit, pressing **[UP]** would return to the 2nd level menu and leave the fuel flow sensor K-factor equal to 2800.

**EXAMPLE 2** – Change the number of tachometer pulses per engine revolution from 2 to 1.

Press **[DOWN]** to select the options menu as the 1st level menu to use. **tEMP** appears in the clock position. Press **[CHANGE]** two times until **tPUL** appears as the 2nd level menu selection. Press **[DOWN]** to select tachometer pulses per revolution as the 2nd level menu to use. The number **2** will appear just to the right of **tPUL**. Press **[CHANGE]** four times to rotate through the choices until the digit reads **1**. Press **[UP]** to return to the 2nd level menu. Note that this is not an entry procedure, and unlike example number one, where pressing **[UP]** would leave with no change, whatever is last displayed on the display is the final result. Press **[UP]** to return to the 1st level menu or select other option menu items to program.

## **Options Menu**

**PErCt** – This entry is to allow the μMONITOR to generate a percent-power factor to match your aircraft and engine combination. This is done in-flight, after leveling off. Using the pilots operating handbook, the engine must be set to a 65% power setting as accurately as possible at an altitude below 10,000 feet. Usually the engine should be leaned to 25 to 50 degrees Fahrenheit rich of peak and everything should be stabilized. Once the power setting is stable, enter programming mode, and then enter this option. Its location in the menu map makes it the easiest menu item to find, as **Opt** is the first menu item in the first level and **PErCt** is the first menu item in the second level. Once the **[DOWN]** button is pressed, the calculation is made and either **Good** or **Err** will appear to let you know the outcome. The only option at this point is to press **[UP]** and the unit will immediately return to normal operation.

**IMPORTANT!** The percent power algorithm is based on fuel flow to the engine. If you change, for any reason, the fuel flow sensor k-factor using the **MSC/FFFAC** menu item, you must re-generate the percent-power factor using this procedure.

Prior to this calculation, the μMONITOR uses a default factor value that may or may not be close to indicating a correct percent-power indication, so it is probable that the percent-power indication may indicate **EE** (out of range) during your first take-off. Note that after returning to normal operation from any programming function (including this one), the **EE** may appear for up to ten seconds as the calculations that were delayed during programming catch up.

**tEMP** – This option selects whether the unit displays temperatures in degrees Fahrenheit or Celsius. Note that all internal calculations are done in degrees Fahrenheit because of better resolution. Those results are only converted to degrees Celsius before display if the **C** option is selected. Even with degrees **F** selected for display, all serial airdata/fuel output is in degrees Celsius, per the output format standards.

Note, however, that the log memory data dump temperature values will be in the units selected. The resulting data file, however, does not contain information as to what temperature units are being used. If you decide to change from one unit to the other, you may wish to clear the log memory so that data dumps will not contain both temperature units.

**FULty** – This option selects whether fuel results are displayed in gallons or liters. In both cases, the display designator is **FUL**. Results in gallons will always have a decimal point on the display for both fuel quantity remaining and fuel flow, while liter displays will not have a decimal for fuel quantity, but will have a decimal for fuel flow. Be sure that the fuel flow sensor K-factor **MSC/FFFAC** value matches with the **FULty** selection. See the **MSC/FFFAC** details.

Note that the log memory data dump fuel values will be in the units selected. The resulting data file, however, does not contain information as to what fuel units are being used. If you decide to change from one unit to the other, you may wish to clear the log memory so that data dumps will not contain both fuel units.

**tPUL** - The tachometer pulses per engine revolution option must be set to match the number of pulses per revolution that the source of the tach signal is putting out, or the calculation of the RPM will be in error. Typically, magneto “P” leads and electronic ignition systems for 4 cylinder engines have tach outputs of two



pulses per revolution. This is the default setting of the μMONITOR. Magneto “P” leads and electronic ignition systems for 6 cylinder engines have tach outputs of three pulses per revolution.

Connection to a RMI supplied Hall Effect magnetic pickup would result in one pulse per revolution. This must not be confused with electronic ignition systems whose timing depends on a similar Hall Effect sensor.

In that case, the tach output of the system governs. Other choices are provided for non-standard tach sources

### **Calibrate Menu**

Calibration is necessary because each sensor, even of the same type for the same function, is slightly different. Calibration of each function requires that the sensor be connected and operational. With the exception of the ammeter sensor, calibration is a matter of correction of just a few units of difference. So if there is a 20 digit difference between what the function is displaying and what it SHOULD read, there is a problem that calibration won't correct, and troubleshooting is what is necessary. Refer to the troubleshooting appendix in the installation manual.

Performing calibration for each function is the same. Once the function to be calibrated is selected, the calculated value is then shown on the display, updated a few times per second. When the displayed value is stable, and the true value of that function is known, the correct value is entered, and the computer calculates and stores into nonvolatile memory a correction value. For instance, if the oil pressure is displaying a value of minus 2 psi when the engine is not running, a new, correct value of zero should be entered.

**DISABLE FUNCTION** – The calibration procedure also allows you to disable any function where you may not have a sensor attached. If you have a fixed pitch prop, you probably do not have a manifold pressure sensor connected. Instead of having a zero, or some other value displayed, programming the unit to disable the manifold pressure function will result in just a blank spot on the display at that function's position. Normally, once you have the function displayed that you want to calibrate, you would expect that pressing the **[DOWN]** button would go directly to the calibration entry procedure. There is one additional step added during calibration. When the **[DOWN]** button is pressed an **I** or **U** will appear to the right of the function's description in the clock position of the display. At this point pressing the **[CHANGE]** button will toggle the letter back and forth. An **I** indicates “INSTALLED” and a **U** indicates “UNINSTALLED”.

Of course, the default setting for all functions is “installed” when the unit is initialized. If you just wish to mark this function as “uninstalled”, change this letter to **U** and press **[UP]** to leave. Pressing **[DOWN]** with the indicator at **U** will be ignored, as we can't calibrate a sensor that is not installed.

Each function is displayed in its own position on the display during the calibration procedure and the entry procedure is the same as outlined in example number one at the end of the previous Menu Map section.

**CHT & EGT** – The easiest and most accurate way to calibrate these thermocouples (TCs) is to put the sensor into boiling water and calibrate to 212 degrees Fahrenheit. This calibration is adequate throughout the function's range. Although it is tempting to calibrate the CHT temperature using an infrared temperature gun, these measurement tools can have larger errors between actual and their readings because of sensor location and other factors. Any other temperature measuring instrument with its sensor in contact with the sensor under calibration is also acceptable.

**OIL & OUTSIDE AIR TEMPERATURE** – The easiest and most accurate way to calibrate these sensors is

in boiling water as explained in the CHT & EGT section above. Be careful that these sensors are either completely potted into their waterproof housings or care taken that the bare wire leads of the sensors do not contact the water. The current carried through the water between the sensor leads will degrade the accuracy. Any other temperature measuring instrument with its sensor in contact with the sensor under calibration is also acceptable.

**CARBURETOR TEMPERATURE** – The sensor immersed in an equal mixture of ice and water (let stabilize for about five minutes) will result in a temperature of 32 degrees Fahrenheit. Any other temperature measuring instrument with its sensor in contact with the sensor under calibration is also acceptable.

**OIL & FUEL PRESSURE & AMPS** – All of these sensors can be calibrated to zero when the engine is not running.

**VOLTS** – This function will be more than accurate without calibration unless there is an isolating diode installed between the aircraft bus and the μMONITOR. The voltage displayed is measured at the input to the μMONITOR. If a diode is installed, use a voltmeter to measure the voltage at the aircraft battery for a true calibration value.

**MANIFOLD PRESSURE** - To calibrate manifold pressure, the local barometric pressure is needed. Note that the altimeter setting from the tower or ATIS is NOT the same as what the manifold pressure should read. If your elevation is 1000 feet or less and an adjustable altimeter is handy, just crank the adjusting knob until the altimeter indicates sea level (zero feet) and the setting in the window will equal the correct manifold pressure.

If that won't work, use the following table and instructions.

STANDARD DAY TABLE

ALTITUDE (FEET)	PRESSURE (InHg)	PRESSURE CHANGE/FOOT
-1000	31.0185	?
-900	30.9073	.001112
0	29.9213	.001096
500	29.3846	.001073
1000	28.8557	.001058
1500	28.3345	.001042
2000	27.8210	.001027
3000	26.8167	.001004
4000	25.8418	.000975
6000	23.9782	.000932
8000	22.2250	.000877
10000	20.5770	.000824

Call the closest airport or service for their current altimeter setting and their elevation. Ask if they can convert the altimeter setting to actual pressure in InHg (actual barometric pressure not corrected for elevation). If they can, it will save doing the following calculation.

1. Station elevation..... (example)(3570) \_\_\_\_\_

- 2. Station altimeter setting..... (30.15) \_\_\_\_\_
- 3. Next higher altitude from table..... (4000) \_\_\_\_\_
- 4. Standard pressure from table for step 3.....(25.8418) \_\_\_\_\_
- 5. Subtract step 1 from step 3.....(4000 - 3570 = 430) \_\_\_\_\_
- 6. Pressure change per foot for step 3 .....(.000975) \_\_\_\_\_
- 7. Step 5 times step 6.....(430 x .000975 = .4193) \_\_\_\_\_
- 8. Add step 4 and step 7..... (25.8418 + .4193 = 26.2611) \_\_\_\_\_
- 9. Subtract step 8 from 29.9213..... (29.9213 - 26.2611 = 3.660) \_\_\_\_\_

Step 9 represents the difference in pressure (in InHg) between the station and sea level.

- 10. Subtract step 9 from step 2.....(30.15 - 3.66 = 26.49) \_\_\_\_\_

If your elevation is different than the elevation of the station where you got the altimeter setting, adjust the result of step 10 by an amount equal to step 6 times the difference between your elevation and the station's elevation. If you are higher than the station, subtract the correction, otherwise add it.

- 11. Final local pressure in InHg ..... \_\_\_\_\_

**Alarm Menu**

Most of the μMONITOR's functions have two alarm settings; high and low. The high alarm will activate whenever the value being displayed is in excess of the high alarm setting. The low alarm will activate whenever the value being displayed is less than the low alarm setting. An alarm setting left at zero cancels only that high or low alarm for that function, not both. For most functions, you will want either a high setting or a low setting, but not both, so one will be set at a value and the other will be left at zero.

Normally, once you have the function displayed that you want to set its alarm value, you would expect that pressing the [DOWN] button would go directly to the alarm setting entry procedure. There is one additional step added during alarm setting. When the [DOWN] button is pressed an **H** or **L** will appear to the right of the function's description in the clock position of the display. At this point pressing the [CHANGE] button will toggle the letter back and forth. An **H** indicates a high alarm setting will be done and an **L** indicates a low alarm setting will be done. When the [DOWN] button is pressed and the alarm setting is displayed, the **H** or **L** will stay as a reminder.

**rEdLn & LOFUL** – Neither of these function's alarm settings have both a high and a low setting, so the **H/L** designator is not used. When the RPM redline is displayed, it will replace the 1st level menu until the entry is finished.

### **Miscellaneous Menu**

Some of the values in this category are displayed/entered in a portion of the display already occupied by a menu level description. The description will be replaced when the entry is completed.

**FLPrE** – This entry is the measured full fuel capacity of your aircraft, and is used to quickly enter the fuel aboard when you top off the tanks, using the **[PRELOAD]** entry buttons.

**FFFAC** – This entry is the number of pulses that the fuel flow sensor outputs for one-tenth of one unit that the μMONITOR is using for fuel... either gallons, Imperial gallons or liters. Each FloScan sensor is individually calibrated and comes with this number (pulses for 1/10 of a US gallon) printed on the sensor's attached tag.

**NOTE:** To convert the pulses per gallon to Imperial gallons, multiply the pulses per 1/10<sup>th</sup> gallon from the sensor by 1.20. To convert to liters multiply by 2.64.

**Tweaking the FFFAC** - With some careful record keeping, the μMONITOR can be fine tuned for fuel quantity remaining. Even though the unit shows fuel remaining, it really is measuring the FUEL USED and subtracting it from the initial amount aboard. If it is found that the FUEL REMAINING is consistently off, the **FFFAC** can be changed to correct.

The following **example** explains how to determine the amount and direction of the correction:

μMONITOR current fuel calibration number = 2942

1st flight:

- before takeoff, tanks topped to known 50.0 gallons aboard
  - after landing, before fueling, μMONITOR fuel remaining reads 22.5 gallons  
therefore: μMONITOR says  $(50.0 - 22.5) = 27.5$  gallons used
- tanks are then topped off and actual fueling amount is found to have been 27.0 gallons

2nd flight:

- at intermediate stop 15.0 gallons added to aircraft (tanks not topped)
- 15.0 gallons added to μMONITOR before takeoff from intermediate stop
- at destination, before fueling, μMONITOR fuel remaining reads 21.4 gal  
therefore: μMONITOR says  $(50.0 - 21.4) + 15.0 = 43.6$  gallons used
- tanks are then topped off and actual fueling amount is for to have been 27.6 gallons  
therefore: actual fuel used is  $15.0 + 27.6 = 42.6$  gallons

RECAP:

- 1st flight	μMONITOR = 27.5	actual = 27.0
- 2nd flight	μMONITOR = 43.6	actual = 42.6
total	71.1	69.6

The new calibration number is:

$(\mu\text{MONITOR fuel used/actual fuel used}) \times \text{current calibration number}$

$$(71.1/69.6) \times 2942 = 3005$$

If the formula is used, the direction of the correction in addition to the amount of the correction is automatically determined.

**TTIME** – This entry is to set the beginning tachtime on the aircraft engine in tenths of hours. The μMONITOR comes with this value initialized to zero assuming a new engine is being installed. If the aircraft engine already has tach hours, this entry should be set accordingly.

**TTrPM** – This value is a constant that the computer uses to keep track of tach time on the engine. All tach hour recording tachometers have this constant, sometimes called “cruise RPM”. If this constant is set at 2400 RPM, then the tach hours will increase one hour for every wristwatch hour when the engine is turning at 2400 RPM. If you are setting on the deck at 1200 RPM, you would have to set for two wristwatch hours for tach hours to increase by one hour. The whole idea is that tach hours represent engine wear, and there is more wear at higher RPMs. You can enter a higher cruise RPM value so that the engine will accumulate less hours... but you’re just fooling yourself.

Due to internal calculations, the μMONITOR does not allow tachtime cruise RPMs that exactly match what is being entered. As an example, if you enter 2400, and then reenter the function to see what it displays, it will show 2389 RPM, which is what the computer will be using to calculate tachtime. If you really do want a higher cruise RPM, add 50 to what you previously entered and try again. In this case entering 2450 RPM will display 2432 when you reenter the procedure. If you want the result that is closest to 2400, then you would want to go back to the original entry.

**TMPRE** – This value is used to quickly enter a starting time into the count-down timer using the **[PRELOAD]** entry buttons. The default value is 30.0, which is handy to preload the timer which is then started before takeoff. The timer will then give an alarm at 30 minutes and every hour thereafter, which is helpful for switching fuel tanks to keep the wings balanced. You can reprogram the value to any other time that you would use regularly.

**Hobbs** - This entry is to set the beginning Hobbs on the aircraft in tenths of hours. The μMONITOR comes with this value initialized to zero assuming a new aircraft. If the aircraft already has Hobbs hours, this entry should be set accordingly. Hobbs time accumulates equal to clock time whenever the RPM is over 200 RPM. The μMONITOR uses RPM to turn on the Hobbs clock because oil pressure indicates maximum scale when the oil pressure sensor is disconnected or defective. Hobbs time can only be viewed here. After viewing, pressing **[UP]** will leave the total unaffected.

**SrLnO** – This entry is for storage of the μMONITOR’s serial number.

**CLndr** – This procedure is to enter the year, month and date of the internal clock. The μMONITOR’s clock power input must be connected per the installation manual for this data to remain valid. The clock memory is different from the nonvolatile memory that stores other values, and will lose all its time and calendar data when the unit is shut off unless connected properly. Calendar data are viewable only in this procedure and is not needed to keep GMT and LMT correct. However, calendar data is needed to time stamp engine data that is periodically stored to log memory.

A **[DOWN]** button press starts the procedure, where three values are displayed, along with a label, in year,

month and day order. Each value has two digits. If the value is correct, pressing **[UP]** will step to the next value. Pressing **[ENTER]** will start the normal data entry procedure until a final **[ENTER]** on completion of the new data will also step to the next value, or return to the 2nd level menu if that value was the day.

### **Serial Port Menu**

**OUtPt** - If no serial output is necessary from the μMONITOR, turn the serial port transmission off using the **OFF** menu selection (default setting).

The serial formats for **GAR** (Garmin), **ShAdF** (Shadin Fuel) and **ShAd** (Shadin FADC Z format) are described in Appendix A of this section.

**INPUt** - If no serial input is necessary to the μMONITOR, turn the serial port reception off using the **OFF** menu selection (default setting).

The **rMI** selection allows air data from the RMI μENCODER to be input into the μMONITOR. For those output formats using airdata values, this input data will be passed on through the serial port. If the output format includes compass heading, magnetic heading will also be passed on if the μENCODER also has a compass module attached.

**bAUd** - Baud rate selections are 1200, 2400, 4800, 9600 and 19200 (default 9600). Note that all baud rate settings for all the instruments in the serial port path should be the same... μENCODER, μMONITOR and GPS.

### **Service Menu**

**vEr** – This displays the version number of the installed software. No entry mode.

**SHO1** – This displays a fixed screen display for photo purposes.

**dSPLy** – This displays all digit segments and descriptors to test the LCD.

**bEEP** – This allows testing of the audio alarms. Press **[CHANGE]** to alternate between on/off.

**UPLOd** – This entry allows the customer to perform software upgrades for the μMONITOR in the aircraft using a laptop computer. The upgrade software, up loader software and instructions can be downloaded from the [www.rkymtn.com](http://www.rkymtn.com) website as new versions are developed. The baud rate used by both the μMONITOR and the upload software are fixed by both the μMONITOR software and the uploader software, so the current baud rate setting is immaterial.

### **Formula Menu**

This section describes entry of values used by the internal computer to calculate and smooth the results shown for each function on the display. Different calculation values are sometimes needed when a sensor with different characteristics than the standard type are used. The supplied software has default values that are loaded into the nonvolatile memory when the unit is first turned on. These default values match the sensors supplied with the μMONITOR, so this section of programming is seldom used.

The **FIL** value used in the calculation is the called the filter factor, and determines how fast the display will show a change in the function's value. The range allowed for the filter factor is 1 to 15, and the smaller the number the faster the display reacts. The μMONITOR calculates and displays each function twice per second. With a filter factor of 1, an instantaneous change of 10 calculated degrees, psi or whatever will show up immediately. With a filter factor of 4, it will take 4 calculations for the full change of 10 to reach its new value, with each change moving one fourth of the final result. The circuitry of the μMONITOR also contains filtering, so it is a combination of that and the software filtering that determines the overall smoothing. The default filter factor values should work in most installations, giving a good compromise between displaying the result as soon as possible and eliminating the annoying "jitter" of digits jumping around. In the entry process below, you can enter in any three number between 0 and 255 for the **FIL** value. **However**, the value is checked before storage in nonvolatile memory and zero is changed to the minimum value of 1, and anything 16 or over is changed to the maximum of 15.

A **[DOWN]** button press starts the procedure, where four values are displayed, along with a label, in filter factor (smoothing), offset, multiply, divide and correction order. Each value has three digits. If the value is correct, pressing **[UP]** will step to the next value. Pressing **[ENTER]** will start the normal data entry procedure until a final **[ENTER]** on completion of the new data will also step to the next value, or return to the 2nd level menu if that value was the correction.

The following table shows values for some different sensors (default sensors are marked \*):

Function	Sensor	FIL	OFF	MUL	dIv	COr
CHT *	Type J Thermocouple	4	0	9	10	32
Oil Temp *	AD590	4	546	179	199	32
MAP *	RMI/GM (10.0-59.0 InHg)	4	130	29	50	100
Amp *	Amloc AMP-50	4	632	27	158	0
Amp	MicroSwitch CSLA1CE	4	633	1	4	0
Oil psi *	VDO 0-10 bar #360-430	4	100	1	4	0
Fuel psi *	VDO 0-2 bar #360-043	4	95	1	10	0
Fuel psi	VDO 0-10 bar #360-430	4	100	47	105	0
Carb Temp *	RMI #52-0001	4	56	33	148	0
Carb Temp	Richter B-5 Probe (SN < 590)	4	57	21	94	0
OAT *	AD590	4	546	179	199	32
EGT *	Type K Thermocouple	4	0	254	141	32

### Log Menu

At periodic intervals, when the engine is running (RPM over 200) the μMONITOR "logs" all its data to a block of nonvolatile memory. This data is time stamped with year, month, day and GMT time of day. Using another log menu item, this data can be dumped out the serial port for capture and analysis to a PC or laptop.

Currently, RMI does not have software available for this task, but the dump format is detailed in Appendix B for those who have the expertise. Also, EGView™ software from EGTrends, Inc. at [www.EGTrends.com](http://www.EGTrends.com) is capable of accepting the μMONITOR format in its list of engine monitors.

The nonvolatile log memory has a capacity of 992 blocks of data. When the log memory is full, it starts

writing over the oldest data. At one minute intervals, the memory can hold 16.5 hours of data. At the maximum log interval of every 10 minutes, the memory can hold 165 hours of data before starting to overwrite older blocks.

**LtIME** – This procedure allows selection of the number of minutes between storage of blocks. On entry of the procedure, the current interval setting is displayed to the right of the “LOG” showing in the clock position of the display. Pressing **[CHANGE]** rotates through the interval choices. Pressing the **[UP]** will store the interval choice being displayed and return to the 2nd level menu.

**dUMP** – This procedure dumps all stored data blocks out the serial port starting with the oldest and finishing with the last logged entry. On entry to the procedure, the data is immediately dumped. There is no indication on the display that data is being dumped. However, when all the data has been sent, **UP** will appear as a signal that the procedure has finished and that actually pressing **[UP]** will return to the 2nd level menu. This can be repeated over if necessary.

The dump procedure uses the  $\mu$ MONITOR’s current baud rate as set by the **Ser/bAUD** menu item.

**CLr** – This procedure clears all blocks in the logged memory as if it had never been used. You would probably only need to perform this if a different engine were being installed. There is no way to recover the data after entering this procedure. On entry to the procedure, the memory is cleared and **UP** will appear as a signal that the procedure has finished and that actually pressing **[UP]** will return to the 2nd level menu.



## \* APPENDIX A \*

**GPS Serial Output Formats**

The serial port output formats are provided for those of you that have the capability and desire to interface to the μMONITOR with your own computer or equipment or to supply air data information to any of the GPS systems on the market.

**STANDARD RMI INPUT FORMAT** - The standard Rocky Mountain Instrument serial port format contains all of the μENCODER functions including those not shown on its display. The string of data is output from the μENCODER about once per second with a baud rate set using the μENCODER **SEr/BAUd/xxxxx** programming menus. Protocol is 1 start bit, 8 data bits and 1 stop bit. Each byte of data is an ASCII character (with exceptions noted).

Each field is input with the most significant digit first which may contain leading blanks or a minus sign. The sign byte is not in a fixed position, i.e. -1234 and (blank)-567 are two valid inputs for the same function. Positive numerical fields are assumed unless a minus is present. If a field is out of limits for the μENCODER the field will be filled with Es (for error). If the temperature probe of the μENCODER is inoperative or disconnected, temperature dependent functions such as true airspeed, density altitude etc. will have their fields filled with Es.

The standard format is:

Off-set	Len.	Name	Description
0	1	ID#1	hex ident byte - always 0xFF
1	1	ID#2	hex ident byte - always 0x81
2	1	Mode	hex output format code - always 0x01 for standard format
3	5	IALT <sup>1</sup>	indicated altitude in feet
8	5	PALT <sup>1</sup>	pressure altitude in feet
13	5	DALT <sup>1</sup>	density altitude in feet
18	5	TALT <sup>1</sup>	true altitude in feet (see warning in the μENCODER manual)
23	5	AlertALT	alert altitude currently set (use indicator flags to see if alert is active)
28	3	IAS	indicated airspeed in knots or MPH
31	3	TAS	true airspeed in knots or MPH
34	4	Mach	indicated airspeed in Mach (assume decimal, (blank)453 = .453 Mach)
38	4	VSI <sup>1</sup>	VSI in 10's of FPM - multiply output times 10
42	3	OAT <sup>1</sup>	indicated outside air temperature (°C only)
45	3	TAT <sup>1</sup>	true air temperature (static temperature) (°C only)
48	4	InHg	altimeter setting in inches of mercury OR millibars (decimal assumed between 2nd and 3rd digits for inches of mercury)
52	4	Inds	BINARY bytes containing bit indicator flags (gear, flap etc.) - bit=1 means indicator is being shown on display - see below for breakdown
56	4	Hdg	compass heading from external heading sensor (assume decimal ddd.d)
60	1	CkByte	BINARY check byte – adding all bytes starting with mode and including the CkByte (ignoring carries) will equal zero if a valid string received.

<sup>1</sup> Can contain minus sign.

Indicator byte	Bit#	Desc	Indicator byte	Bit#	Desc
<b>Ind1</b>	0	FT/ALT	<b>Ind2</b>	0	CONV
	1	°F		1	TAS
	2	HOLD		2	KT
	3	Vno		3	IAS
	4	Vne		4	UPATI
	5	APP		5	DOWNATI
	6	FLAP		6	M
	7	GEAR	7	Mach decimal point	
Indicator byte	Bit#	Desc	Indicator byte	Bit#	Desc
<b>Ind3</b>	0	FPM	<b>Ind4</b>	0	ignore
	1	TAT		1	ignore
	2	°C		2	ignore
	3	LGT		3	ignore
	4	HDG		4	ignore
	5	InHg		5	ignore
	6	Mb		6	ignore
	7	baro setting decimal	7	ignore	

**CAUTION:** Do not use true altitude for terrain clearance. See μENCODER operations manual.

**NOTE:** You can use the TAS, KT, IAS and M indicators in Ind2 byte to resolve whether the unit is using knots or miles per hour for airspeed. If both M and IAS are set to 1 at the same time OR both M and TAS are set to 1 at the same time, then MPH is being used. By the same token if either both KT and IAS are set to 1 OR both KT and TAS are set to 1 at the same time, then knots are being used for airspeed.

All GPS systems expect it airdata input to be in knots for true and indicated airspeeds. If the μMONITOR detects that the μENCODER is sending it airspeed in miles per hour, it will convert them to knots before they are used in any airdata serial outputs.

**SHADIN FUEL FORMAT** - This format is for fuel information only, and would not normally be used if airdata is available from the μMONITOR's input port. Protocol is 1 start bit, 8 data bits and 1 stop bit. Each byte of data is an ASCII character.

The μMONITOR's Shadin fuel format is 55 bytes long and is as follows:

```
<STX>G0047.3<sp>0009.8<sp>*****<sp>*****<sp>0018.8<sp>*****<sp>*****<sp>123<ETX>
```

Where:

<STX>	start-transmit character (0x02)
G	units designation (G=gallons, L=liters)
0047.3	total fuel remaining (i.e., ASCII-coded decimal format: 0x30,0x30,0x34,0x37,0x2e,0x33)
<sp>	space (0x20)

```

0009.8      fuel flow rate, total
*****
*****      fuel flow rate, engine one (asterisks for single engine)
*****
*****      fuel flow rate, engine two (asterisks for single engine)
0018.8      fuel used, total
*****
*****      fuel used, engine one (asterisks for single engine)
*****
*****      fuel used, engine two (asterisks for single engine)
123         checksum of bytes 2 through 51 (from G through character before this checksum)
<ETX>      end-transmit character (0x03)

```

The checksum is calculated by adding each byte in the message (2 through 51), such that carries are discarded to give a one byte result. The ASCII-coded decimal representation of the byte is given, ranging from 0 (0x30, 0x30, 0x30) to 255 (0x32, 0x35, 0x35).

**GARMIN 'G' FORMAT** – This format is preferable for GPS systems that accept it, as it contains the maximum amount of airdata along with fuel data. Of course, it is only practical to use it when the RMI airdata from the μENCODER is being used to feed the μMONITOR's input serial port.

The byte protocol is one start bit, 8 data bits and one stop bit.

In the table of data fields below, the message format column uses 'd' to represent a decimal digit (0-9) and uses 's' to represent a sign indicator ('+' or '-'). This table describes the data of each message string and the units used. The field width includes all characters in the message, including the terminating CR and LF. All messages start with a 'G', followed by a single alphanumeric character which is the item designator. Although the 'G' format can contain additional fields (identified by other item designators), the following table only shows the items output by the μMONITOR:

Item Desg.	Message Format	Message Mnemonic	Field Width	Message Description
(STX)				Start of transmission (0x02)
GA	ddd	IAS	7	Indicated Air Speed in knots
GB	ddd	TAS	7	True Air Speed in knots
GC	ddd	MACH	7	Mach Speed in thousandths
GD	sdddd	PALT	9	Pressure Altitude in tens of feet
GE	sdddd	DALT	9	Density Altitude in tens of feet
GF	sdd	OAT	7	Outside Air Temperature (Total) in degrees C
GG	sdd	TAT	7	True Air Temperature (Static) in degrees C
GK	sddd	VSPD	8	Vertical Speed in tens of feet per minute
GL	ddd	HEAD	7	Heading in degrees from magnetic north
GO	dddd	FF	8	Single Engine Fuel Flow in tenths of gallons/hour
GP	dddd	FU	9	Single Engine Fuel Used in tenths of gallons
GQ	ddd	ERR	7	Error Log (000=no errors)
GR	dddd.d	FR	10	Fuel Remaining in gallons
Ga	sdddd	IALT	9	Indicated Altitude (Baro corrected) in tens of feet
Gb	dd.dd	BARO	9	Current Barometric Pressure Setting in InHg
G*	ddd	CKSUM	7	Checksum of all characters preceding this record
<ETX>				End of transmission (0x03)

CKSUM is a one byte checksum (discarding carries), including all characters from the initial STX up through the line feed preceding the checksum message. It does not include itself.

There is no CR, LF following <STX> or <ETX>.

**SHADIN FADC Z FORMAT** - This format is the second best for GPS systems that accept it. It does not have quite as complete information as the Garmin format. Of course, it is only practical to use it when the RMI airdata from the μENCODER is being used to feed the μMONITOR's input serial port.

The byte protocol is one start bit, 8 data bits and one stop bit.

In the table of data fields below, the message format column uses 'd' to represent a decimal digit (0-9) and uses 's' to represent a sign indicator ('+' or '-'). This table describes the data of each message string and the units used. The field width includes all characters in the message, including the terminating CR and LF. All messages start with a 'Z', followed by a single alphanumeric character which is the item designator.

Although the 'Z' format can contain additional fields (identified by other item designators), the following table only shows the items output by the μMONITOR:

Item Desg.	Message Format	Message Mnemonic	Field Width	Message Description
(STX)				Start of transmission (0x02)
ZA	ddd	IAS	7	Indicated Air Speed in knots
ZB	ddd	TAS	7	True Air Speed in knots
ZC	ddd	MACH	7	Mach Speed in thousandths
ZD	sdddd	PALT	9	Pressure Altitude in tens of feet
ZE	sdddd	DALT	9	Density Altitude in tens of feet
ZF	sdd	OAT	7	Outside Air Temperature (Total) in degrees C
ZG	sdd	TAT	7	True Air Temperature (Static) in degrees C
ZK	sddd	VSPD	8	Vertical Speed in tens of feet per minute
ZL	ddd	HEAD	7	Heading in degrees from magnetic north
ZO	dddd	FF	8	Single Engine Fuel Flow in tenths of gallons/hour
ZP	dddd	FU	9	Single Engine Fuel Used in tenths of gallons
ZQ	ddd	ERR	7	Error Log (000=no errors)
ZR	ddd	CKSUM	7	Checksum of all characters preceding this record
<ETX>				End of transmission (0x03)

CKSUM is a one byte checksum (discarding carries), including all characters from the initial STX up through the line feed preceding the checksum message. It does not include itself.

There is no CR, LF following <STX> or <ETX>.

## \* APPENDIX B \*

**Log Memory Details**

The log dump is all in comma-delimited ASCII form with a two line header, and then records for each packet stored. The format is for easy importation into a spreadsheet. The number of packets dumped depends on if the log memory is full or not. The dump order is from oldest packet to newest packet. Log packets are stored in the log memory only if the engine is running (RPMs greater than 200).

The two line header is only output once at the beginning of the log dump. The first line is a title line indicating the RMI μMONITOR. The second line is a column header line with acronyms, in order, of the fields in each record.

A sample dump looks like this:

```
Rocky Mountain Instrument microMonitor<cr><lf>
YR,MO,DY,GMT,OT,CAT,OAT,OP,FP,MAP,PWR,RPM,FF,FQ,VLT,AMP,C1,E1<cr><lf>
05,02,25,0842,229,48,-2,72,4,22.3,65,2210,8.4,42.9,14.4,16,392,1355<cr><lf>
05,02,25,0844,229,48,-2,72,4,22.3,65,2210,8.4,42.9,14.4,16,392,1355<cr><lf>
.....
05,02,25,1122,229,48,-2,72,4,22.3,65,2210,8.4,42.9,14.4,16,392,1355<cr><lf>
```

YR	Year
MO	Month
DY	Day
GMT	Zulu Time, i.e. 8:42 = 8:42am, GMT and 13:24 = 1:24pm
OT	Oil Temp
CAT	Carburetor Air Temp
OAT	Outside Air Temp
OP	Oil Press.
FP	Fuel Press.
MAP	Manifold Pressure
PWR	Percent Power
RPM	RPM
FF	Fuel flow (hundredths if gallons, tenths if liters)
FQ	Fuel Quantity Remaining (tenths if gallons, units if liters)
VLT	Volts
AMP	Amps
C1	CHT 1 Temperature
E1	EGT 1 Temperature

Although the μMONITOR is programmable for gallons or liters for fuel units, and degrees Celsius or Fahrenheit for temperature, there is nothing in the data dump to differentiate which units are being stored. Obviously, switching back and forth between different units will make the data dump confusing.

The only practical way to differentiate between different flights is to parse the GMT. Since the max log interval is 10 minutes, any gap greater than that would indicate a separation between flights.

## \* APPENDIX C \*

<b>Programming Checklist</b>
------------------------------

This checklist serves to help select and record your programming option choices and engine performance values. The list is in the same order as it appears while the μMONITOR is in the programming mode. Refer to the *Menu Map* section and details following it for complete information on each entry. The asterisk (\*) and small print (2900) represent default selections and values.

Opt/PErCt	Not settable	In-flight calibration of Pwr% only
OPt/tEMP	* <input type="checkbox"/> F <input type="checkbox"/> C	temperature units in degrees Fahrenheit or Celsius
OPt/FULty	* <input type="checkbox"/> G <input type="checkbox"/> L	fuel units in gallons or liters
OPt/TPUL	<input type="checkbox"/> 1 * <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 8	number of tachometer pulses per revolution for RPM
CAL/CHt	<input type="checkbox"/> Completed, date _____	calibrate cylinder head temperature
CAL/Ot	<input type="checkbox"/> Completed, date _____	calibrate oil temperature
CAL/MAP	<input type="checkbox"/> Completed, date _____	calibrate manifold pressure
CAL/AMP	<input type="checkbox"/> Completed, date _____	calibrate ammeter
CAL/OP	<input type="checkbox"/> Completed, date _____	calibrate oil pressure
CAL/FP	<input type="checkbox"/> Completed, date _____	calibrate fuel pressure
CAL/Ct	<input type="checkbox"/> Completed, date _____	calibrate carburetor temperature
CAL/VOLtS	<input type="checkbox"/> Completed, date _____	calibrate volts
CAL/OAt	<input type="checkbox"/> Completed, date _____	calibrate outside air temperature
CAL/EGt	<input type="checkbox"/> Completed, date _____	calibrate exhaust gas temperature

Although both high and low alarm settings are allowed, generally one will be left at zero, meaning no alarm. See details in the main programming manual section on alarms.

ALr/CHt	H _____ 390F	L _____	cylinder head temperature high/low alarm
ALr/Ot	H _____ 230F	L _____	oil temperature high/low alarm
ALr/MAP	H _____ 58.0	L _____	manifold pressure high/low alarm
ALr/AMP	H _____	L _____ 2	ammeter high/low alarm
ALr/OP	H _____	L _____ 20	oil pressure high/low alarm
ALr/FP	H _____	L _____ 2	fuel pressure high/low alarm
ALr/Ct	H _____	L _____ 40F	carburetor temperature high/low alarm
ALr/rEdLn	H _____ 2700		tachometer RPM redline alarm
ALr/LOFUL		L _____ 5.0	low fuel quantity alarm
MSC/FLPrE	_____ 50.0		full fuel quantity preload
MSC/FFFAC	_____ 2900		fuel flow sensor K-factor
MSC/TTIME	_____ 0.0		tachtime
MSC/TTrPM	_____ 2432		tachtime cruise RPM
MSC/TMPrE	_____ 30.0		timer preload
MSC/HObbS	_____ 0.0		Hobbs time
MSC/SrLnO	_____ 0		serial number
MSC/CLndr	_____ ??		year, month & date calendar
SEr/OUtPt	* <input type="checkbox"/> OFF <input type="checkbox"/> GAr <input type="checkbox"/> SHF <input type="checkbox"/> SHAd		serial port fuel and/or airdata transmission format
SEr/InPUt	* <input type="checkbox"/> OFF <input type="checkbox"/> RMI		serial port receive mode

Srv/bAUd	<input type="checkbox"/> 1200 <input type="checkbox"/> 2400 <input type="checkbox"/> 4800 * <input type="checkbox"/> 9600 <input type="checkbox"/> 19200	serial port baud rate
Srv/vEr	Not settable	view software version
Srv/SHO1	Not settable	view static demo mode
Srv/dSPly	Not settable	view display test
Srv/bEEP	Maintenance routine	audio test
Formula section:		see detail section.
LOG/TPUL	<input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5 * <input type="checkbox"/> 6 <input type="checkbox"/> 10	log period in minutes
LOG/DUMP	Not settable	dump log memory
LOG/CLr	Not settable	clear log memory

**\* APPENDIX D \***

**Software Version Details**

The following list shows changes made to each version of software. Note new features that have been added since your previous version and refer to the current Operation and Programming manuals for details on the additions.

Version 06.06      7/1/07

1. Revamped code in Fuel in calculating FF. No customer observed changes.
2. Changed fuel flow smoothing factor from a stored NVRAM value to a constant to prevent possible inaccurate fuel flow calculations.
3. Corrected code so that volts is the default display instead of OAT when carburetor temp is not installed, as per the operations manual.

Version 06.05      1/24/07

1. Correct bug in storing calendar info into NVRAM. Year, Month or Day values were being corrupted between entry and storage.

Version 06.04      12/26/06

1. Added to RMI utility, no customer observed changes.

Version 06.03      10/3/2005

1. Correct bug in RS232 GAr and SHA serial output formats.

Version 06.02      5/17/2005 First Release

Per version 06.00 Operations and Programming Manuals

1. Correct NVRAM/IDATA pointer mismatch.

Version 06.01      4/18/2005      Beta Test

1. Correct bug in calibration module.

Version 06.00      2/16/2005      Beta Test

Version 6.00 4/1/05 new version for this hardware (plug in CPU/memory assembly)