MICRO LAMBDA WIRELESS, INC.

MLAT Attenuator User Manual

MICRO LAMBDA WIRELESS, INC.

MLAT Attenuator User Manual

© Micro Lambda Wireless, Inc.
46515 Landing Pkwy. Fremont, CA 94538
Phone 510.770.9221 • Fax 510.770.9213

Table of Contents

1.0 Introduction

This manual describes the setup, operation and remote communication for the MLAT Attenuator. The model and serial numbers are located on the label located on the top cover of the unit. Each unit has a separate, custom specification sheet for the particular model defining the Attenuator's frequency range, RF characteristics and options.

General operating/programming instructions are located herein.
The USB Flash Drive supplied with the package contains a **SetupMLAT.msi** file, when executed, it will create a folder named "MLAT Support Files" on the desktop containing short cuts to the manual, documentation and programs for interfacing the product with a personal computer. This **SetupMLAT.msi** file is compatible with Windows 7, 8, and 10. The most current versions of these files, new offerings and standard Attenuator specifications can be downloaded at our web site: http://www.microlambdawireless.com

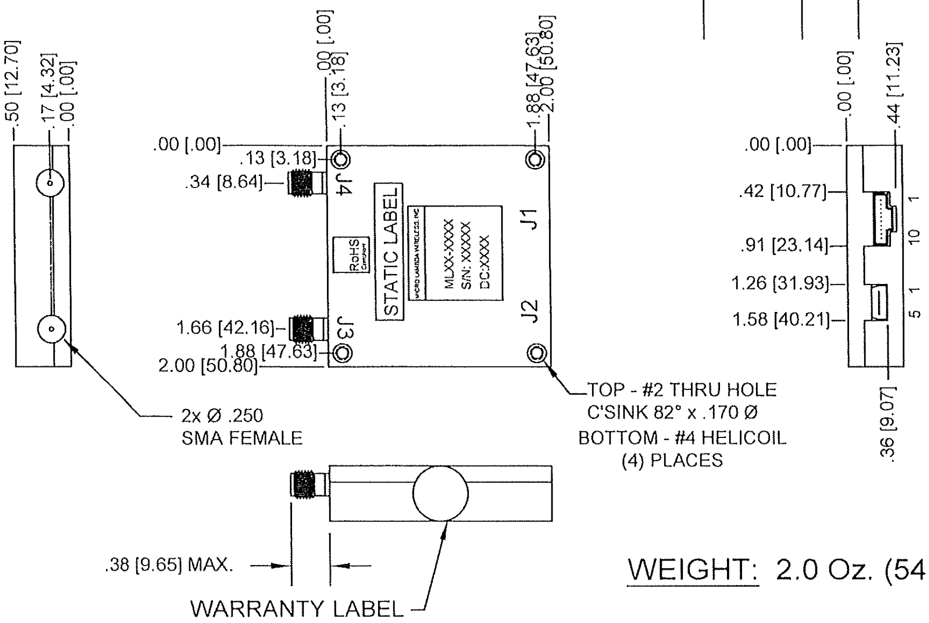
2.0 Package Contents

Item	**Quantity**
MLAT Series Attenuator	Purchased Qty.
DC Power mating connector and crimp pins	Purchased Qty.
USB A male to USB Micro-B cable	Purchased Qty.
USB Flash Drive(Contains manual, and PC software)	1 each
MLAT Quick Start Guide (Printed)	1 each

3.0 General Overview of Product Capabilities

The MLAT series of broadband frequency Attenuators are general purpose single channel programmable attenuators suitable for a wide range of applications. They can be supplied in 31.5 dB or 63.0 dB models and have the options of an integrated amplifier, and or a level flatness calibration attenuator. These products can be used in test and measurement equipment, production test, be integrated in a system where signal source attenuation is required, and many other applications. The units are 2.0" x 2.0" x 0.50" high and will fit nicely in a PXI chassis slot. The attenuator is housed in a compact package with SMA Female RF Connectors. A 10 pin Molex connector, and a USB Micro B are provided for power and control ports. The standard frequency range is 0.01 to 21.0 GHz. Two types of interface come standard: USB for trouble free connection to a personal computer and a 4-wire serial interface for use in the customers system. Outline drawing 99-0221-027 shown on the next page defines the mechanical configuration. The drawing is displayed mainly for the mechanical and the connection information. See your model specification sheet or contact Micro Lambda Wireless, Inc. for details. Operating temperature ranges outside the 0 to 50 Deg. C range are also available.

REV	DESCRIPTIONS	DATE	APP'D	NOTES
-	PRELIMINARY DRAWING #	11/21/19	DS	
A	INITIAL RELEASE	8/18/2020	DS	



WEIGHT: 2.0 Oz. (54.0 g.)

J1
POWER / SPI INTERFACE

PIN	FUNCTION
1	MOSI - DATA IN
2	MISO - DATA OUT
3	CLOCK - SERIAL CLOCK
4	SS - SLAVE SELECT (*)
5	GROUND - LOGIC
6	N/C
7	N/C
8	N/C
9	+ 5V (IF NO USB CONN.)
10	GROUND

J2
USB INTERFACE (MICRO-B)

PIN	FUNCTION
1	+ V (+ 5V)
2	D -
3	D +
4	GROUND
5	GROUND

J3: RF INPUT, SMA FEMALE
J4: RF OUTPUT, SMA FEMALE

NOTES:

1. J1 MALE: MOLEX# 5015681007
2. J1 MATES WITH FEMALE: MOLEX# 5013301000
3. CRIMP PIN: MOLEX#5013340000
4. J1 POWER SUPPLY INPUTS NOT REQUIRED FOR USB OPERATION.
5. SMA ENGAGE/DISENGAGE TORQUE: 2 INCH-POUNDS MAX.
6. SMA MATING TORQUE (ROTATIONAL): 8 INCH POUNDS MAX.
7. SMA CONTACT RETENTION: 6 LBS. MIN. AXIAL FORCE ON MATING END. 4 IN-OZ MIN. RADIAL TORQUE.
8. (*) ACTIVE LOW

UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES
TOLERANCE ARE :
FRACTIONS ± ± | DECIMALS .xx ± .02 [.52] .xxx ± .010 [.254] | ANGLES

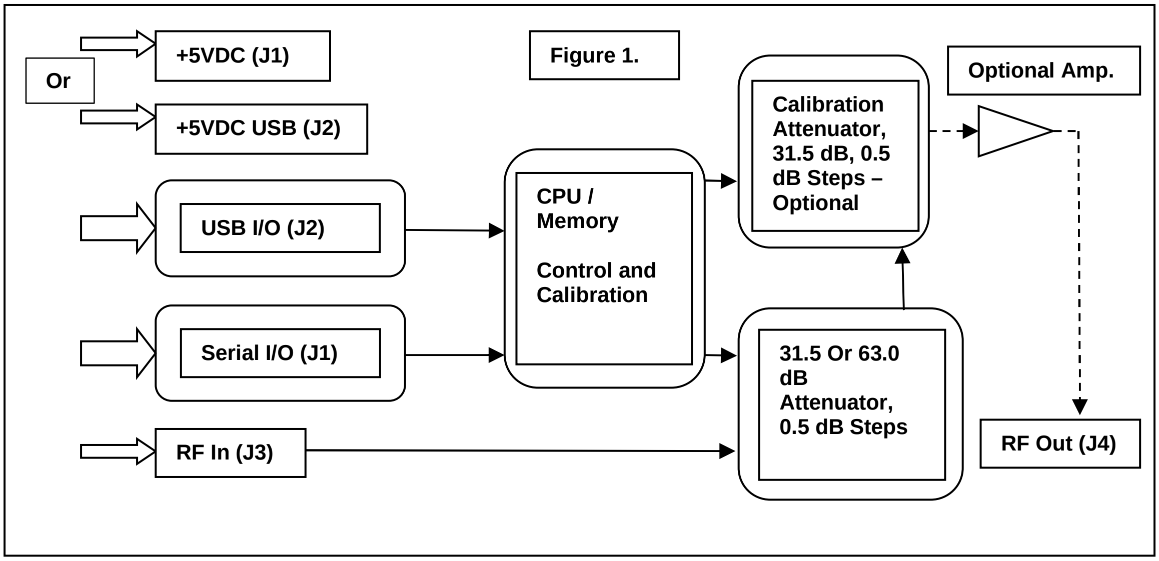
CONTRACT NO.		
	APPROVALS	DATE
MATERIAL ALUMINUM	DRAWN N.NGUYEN	8/18/2020
	CHECKED DS	8/18/20
FINISH NICKEL	ISSUED	
DO NOT SCALE DRAWING		

MICRO LAMBDA WIRELESS, INC.

MLAT ATTENUATOR 2" X 2" X 0.5"

SIZE	CAGE No	DWG NO	REV.
	0RN63	99 - 0071 - 027	A

A simple block diagram is shown in Figure 1. The port configuration is shown in drawing 99-0071-027, the Attenuator provides an RF input, J3. An RF output, J4. A DC power/serial interface port J1, and a USB Micro-B interface connector J2. The Attenuator value is adjusted via the J1 serial connection or the USB port J2. A DC voltage of +5.0V with adequate current to operate the product must be applied to J1, or Via J2, USB in order to communicate with the product.



4.0 Option Information

This product has four options as follows:

1. Option A is an attenuation range of 31.5 dB, in 0.5 dB steps.
2. Option B is an attenuation range of 63.0 dB, in 0.5 dB steps.
3. Option D is User initiated level flatness correction.
4. Option E is internal gain stage in the RF path, frequency range; 10MHz to 22.0 GHz. See MLAT specification sheet for details.

4.1 Option D Description and Operation

This option adds a user initiated level flatness correction that is stored in nonvolatile memory. The frequency range, and step size can be defined, in MHz, and stored in nonvolatile memory. FA = start frequency, FB = end frequency, and FC = correction frequency increment (i.e. FA2000, FB8000, FC50). The source Cal level point SCL command, "SCL0.0" stores, in memory the RF output level that you would like to calibrate at. (i.e. 0.0 dBm). Note: The RF output of the attenuator, with all attenuation and correction set to minimum, must be greater than the SCL level setting, across the FA to FB range. Set your signal source to FA, then measure the RF output power of the attenuator, send (decrement) the LC command (LC-0.5) until the measured RF output reaches the desired "SCL" level, +/- 0.5 dB. Send the SC0 command to store the first calibration point (i.e. last LC setting sent LC-2.5 = 2000 MHz @ 0.0 dBm) into Cal memory location 0 (which would now read back as -2.5). Now increment the signal source frequency by FC (50 MHz), repeat the LC routine until SCL is reached, send

SC1 command to store 2050 MHz @ 0.0 dBm. Repeat until FB is reached. The correction mode can be enabled / disabled via the C1 / C0 commands. With calibration complete, enable correction "C1". Send the signal source to 2000 MHz. Send the attenuator the frequency command F2000. The attenuator will pull the 2000 MHz correction data from nonvolatile memory, and set the RF output to be 0.0 dBm, +/-0.5 dB. You can now set the attenuator, attenuation level to any setting in the attenuators range (i.e. L-45.5). The RF level accuracy will be relative to the 0.0 dB Cal point insertion loss (-45.5 dBm, +/-0.5 dB). If you set the signal source frequency to a new frequency, set the attenuator frequency to the same frequency as the source, and the attenuator will output the frequency / level corrected attenuation level setting. You can read the calibration table starting at memory location 500 (FA) to 2000 (FB). Read memory command = R500, 0 data = no correction, valid correction = a negative number between 0 an –31.5, in 0.5 increments. Please note: There are only 1500 memory locations to store data, for a full frequency range calibration, keep FC >= 25.0 MHz. If your signal source has abrupt RF power frequency band changes, select an FC increment that will match the frequency band changes. If the signal source is substituted, a new calibration run will be required.

Note: Before calibration, all attenuation and calibration settings should be set to 0.0 dB.

4.2 Option E Description and Operation

This option adds an internal 10.0 to 22.0 GHz GaAs MMIC RF amplifier to the attenuators RF output port. This will eliminate the insertion loss associated with the attenuators RF path. This will lead to some gain from the attenuator, at the expense of frequency range, noise figure, harmonics, and increased power consumption.

Notes:

1. Insertion loss of the optioned attenuator must be subtracted for correct gain numbers.
2. This option is available with options A and D.

5.0 Setup and Operation

This product is designed for a 0 to 50 Degree C environment and should not be subjected to humidity >95%. Use proper ESD handling procedures. Verify that all external RF/microwave cables and components connected to the unit are in good working condition.

5.1 Connections

Connect to the preferred interface port, Serial or USB. USB Micro-B to USB Male - A cable for connection to a host PC USB port, the USB interface is USB 1.1 and 2.0 compatible. The serial port is a 0-5V, CMOS/TTL compatible port; it is very similar to an SPI communication port. Clock, Data In, Data Out, and Select/Enable operate in the typical serial communication format, except that the maximum clock speed is 5 MHz.

The Select line going low tells the attenuator to listen for clocked data. The Data-In line is used to write data commands to the Attenuator. The Clock line is used to clock each data bit in or out of the unit as required. The Data-Out line is used to read data from the Attenuator if a command requested it. To read data from the unit, the data is clocked into the controller by clocking out data (Writing) all 0's to the MLAT and the controller reads during this time, this is done in a second Select sequence. See section 6.0/6.1 serial communication for details.
For Serial communications, +5.0 VDC on J1 pin 9, and a common ground for both the supply, and Logic ground on pin 10 are required. Note the power supply inputs are protected from damage up to +6 VDC, and are also protected from negative voltages.
If USB communications is used, the USB port of the PC controlling the MLAT will power the unit.

5.2 MLAT operation

Apply the power supply voltage or USB connection to the unit. The unit should be operating at the last Level, and Frequency (Optional) it was set to before power down; this would typically be; Attenuation Level Min., and Frequency Min. when shipped from the factory.

6.0 Controlling the MLAT using a personal computer

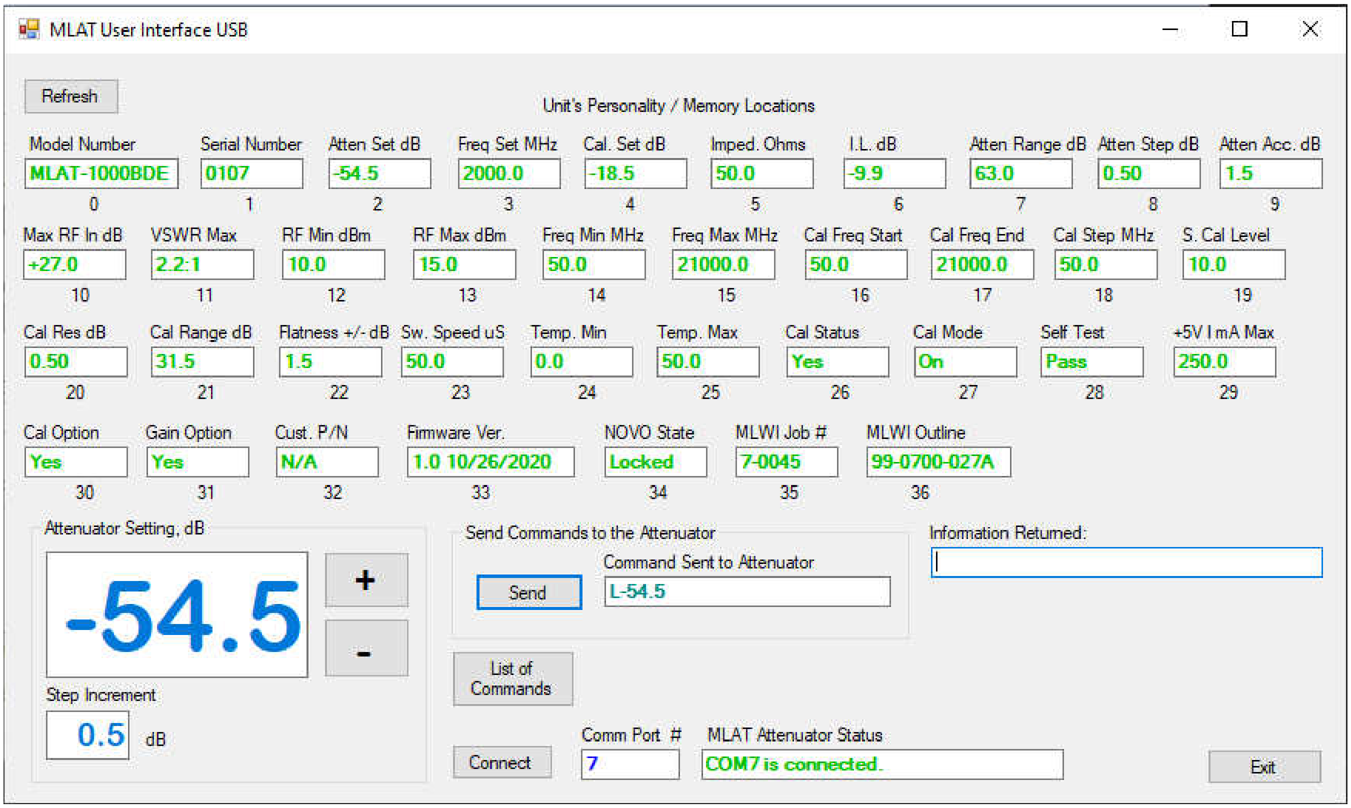
The MLAT Attenuator can be controlled by a personal computer for Demo purposes. The requirements for this are as follows: A USB Ver. 1.1 thru 3.0 port on the PC, Windows 7, 8, and 10 (32 or 64 bit versions), the programs included on the MLAT Support USB Flash Drive and a power supply, or USB port capable of supplying the DC Voltage and Current required to operate the MLAT Attenuator if the Serial interface is used.

6.1 Installing the documentation and control software

The MLAT Support USB Flash Drive supplied with the MLAT contains the file named **SetupMLAT.msi**. Execute this file to install the manual, documentation and control programs for PC interface. The setup file, when run, will create a folder named "MLAT Support Files" on the computer desktop with short-cuts to the documentation and interface program.
(Note: This setup file must be executed for the USB PC interface program to operate correctly, dll files will be installed to the system directory.)

6.2 USB Interface

The MLAT product, when connected using the USB interface, appears as a USB Serial Device to the Windows operating system. The USB driver is supplied with the windows operating system, and is installed automatically when the unit is connected to the PC's USB port. The unit can be identified in the Windows "Device Manager" as a ???????, under the "Ports" folder. The MLAT may be controlled remotely via a USB connection using the supplied “MLAT User Interface USB.exe” program. A screen capture of this program is shown in Figure 2. A list of all commands that the unit will respond to, can be accessed through the button called “List of Commands”. On the program screen you will see some limited information about the unit. Commands may be sent to and information received from the unit, as required. The unit’s Attenuation Level can also be adjusted, by using the “+” and “-” buttons, the Attenuation will increment and decrement based on the dB number shown in the Step Increment box. This number can be changed to any valid step size within the attenuation range limits of the unit. The current frequency setting is also shown. The program can be used to connect to multiple units; choose the Com port of interest, and click the “Connect” button. Press the “Refresh” button to update the unit’s information.



Unit's Personality / Memory Locations

Model Number	Serial Number	Atten Set dB	Freq Set MHz	Cal. Set dB	Imped. Ohms	I.L. dB	Atten Range dB	Atten Step dB	Atten Acc. dB
MLAT-1000BDE	0107	-54.5	2000.0	-18.5	50.0	-9.9	63.0	0.50	1.5
0	1	2	3	4	5	6	7	8	9
Max RF In dB	VSWR Max	RF Min dBm	RF Max dBm	Freq Min MHz	Freq Max MHz	Cal Freq Start	Cal Freq End	Cal Step MHz	S. Cal Level
+27.0	2.2:1	10.0	15.0	50.0	21000.0	50.0	21000.0	50.0	10.0
10	11	12	13	14	15	16	17	18	19
Cal Res dB	Cal Range dB	Flatness +/- dB	Sw. Speed uS	Temp. Min	Temp. Max	Cal Status	Cal Mode	Self Test	+5V I mA Max
0.50	31.5	1.5	50.0	0.0	50.0	Yes	On	Pass	250.0
20	21	22	23	24	25	26	27	28	29
Cal Option	Gain Option	Cust. P/N	Firmware Ver.	NOVO State	MLWI Job #	MLWI Outline			
Yes	Yes	N/A	1.0 10/26/2020	Locked	7-0045	99-0700-027A			
30	31	32	33	34	35	36			

Attenuator Setting, dB
-54.5

Step Increment
0.5 dB

Send Commands to the Attenuator
Command Sent to Attenuator: L-54.5

Infomation Retumed:

Comm Port #: 7
MLAT Attenuator Status: COM7 is connected.

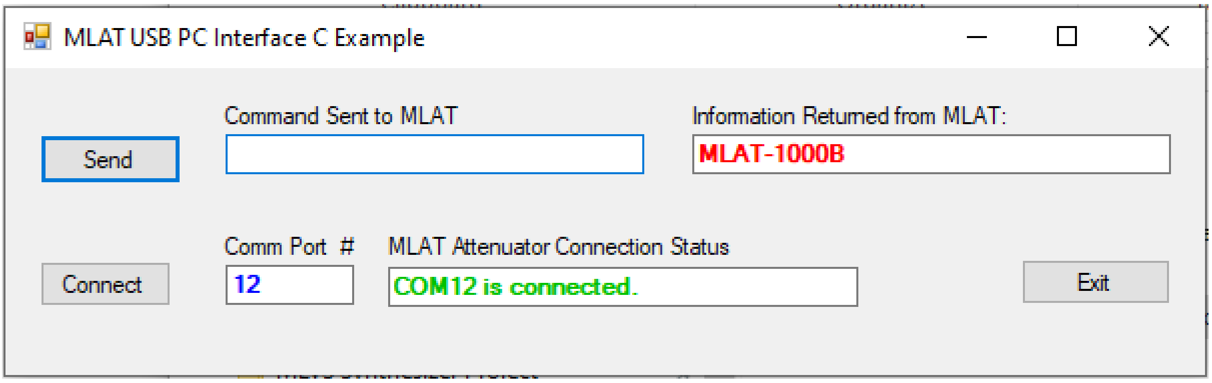
Figure 2

6.3 USB CDC Virtual Com Port PC Interface example in C++

Included in the installed support files folder is an example of a simple USB CDC Virtual Com Port interface program written in C. The project file and source code were written using Microsoft Visual Studio Community 2019, C#. Visual Studio Community 2019, can be downloaded for free at https://visualstudio.microsoft.com/. After you register this free version of the programming environment, you can create and edit C programs. This source code will allow you to get started integrating the MLAT USB communication into your own C programs. The example program is a simple interface in which you can send and receive ASCII characters. The interface tests for a connection to the MLAT

Attenuator on a USB port by looking for a return string identifying the MLAT unit after searching through all attached USB devices.

An executable version of the program is located on the USB Flash Drive, in the root directory. The file name is – MLAT USB PC Interface C Example.exe. A screen shot of the interface is shown in figure 3.



Command Sent to MLAT:

Information Returned from MLAT: MLAT-1000B

Comm Port # 12

MLAT Attenuator Connection Status: COM12 is connected.

Figure 3

7.0 Serial interface

The MLAT can be programmed using a 4 wire serial bus. The timing diagram is shown in Figure 4, and 5. The Four control lines are as follows:

SS – Slave Select (J1 – Pin 4)	Input	Active Low. Enables the shifting of data into the internal command buffer. Also serves as a command terminator when it goes HIGH. Note: All command + data, command strings must be sent using a single Select Line Cycle, same for data read.
CLOCK – S. Clock (J1 – Pin 3)	Input	Data is clocked into the unit on the falling edge of Clock (SPI Mode 1) and DATA OUT is valid at this time. The maximum clock rate of this line is 5 MHz. This line should be maintained in a LOW state at the application of Select to prevent confusion.
MOSI – Data In (J1 – Pin 1)	Input	Input data pin. Data is sent MSB first. Data must be stable 2 usec. before the CLOCK line goes high and 2 usec. after the clock goes low. (Setup / Hold time) All DATA in/out is in ASCII format.
MISO – Data Out (J1 – Pin 2)	Output	This line is used to pass internal information from the Attenuator to the user. Data is guaranteed to be valid on the falling edge of the clock signal. Data is sent out MSB first. All DATA in/out is in ASCII format.
Logic Ground (J1 – Pin 5)	Input	Logic Ground should be tied to power supply common, and to the controlling circuit.

Recommended send data sequence: The format is one select sequence per command send string, and one select sequence per data receive. The string length of each is 24 bytes of ASCII characters. To be safe, you can always send 24 characters to the unit, and receive 24 characters from the unit, every time you communicate with the unit. To shorten the communication time, on the send sequence, stop sending after a NULL is seen in the string, then set SS high. The same can be done for the receive sequence (See the Arduino serial interface example for reference). A decimal point is not required for frequency commands, 1 MHz resolution is all that is required. So the ASCII command to set the frequency to 8000 MHz - "F8000" would be sent as follows: Set select low, clock out 0x46 0x38 0x30 0x30 0x30 and a 0x00 for termination. When select goes high, this tells the unit that the user is done and to starts interpreting the command. The unit is now set to 8000 MHz, for frequency correction. If the command is not understood, the unit will do nothing. Recommended read data sequence: The returned data is variable in length; however it is recommended that the full 24 bytes of data be read to clear the buffer, or if a NULL is received, the unit is done sending, and the buffer is cleared. Set select low, send the desired read command and set select high. The unit interprets the command and places the requested data in its buffer. Then set select low and clock out 24 ASCII nulls while clocking in the data, then set select high. Example: To read the units current Level setting, set select low and send ASCII L? (0x4C 0x3F), set select high. Set select low and send 24 0x00 while reading the data line and clocking in bits. Set select high (or if NULL received, set select high). The information should be similar to -15.5, in ASCII. All of the memory locations in the unit can be read in this manner, using the R command.

Please note: Some commands can take a very long time to complete, such as "ID" – Initialize Cal Data to 0 (3 seconds), and "ST" – Self Test (1.5 seconds), please allow a considerable delay between sending the command, and trying to read data back or sending another command. A +4.0 to +6.0 VDC power supply must be connected to J1 – 9 (+V), and J1 – 10 (Ground) for proper serial operation. The USB port of a PC can also be used to power the unit during serial Interface operation. The MLAT serial interface lines operate on internal 3.3V logic of a microcontroller; this should allow the unit to communicate in systems operating with 2.7V, 3.3V and 5.0V serial control lines.

7.1 Serial interface timing:

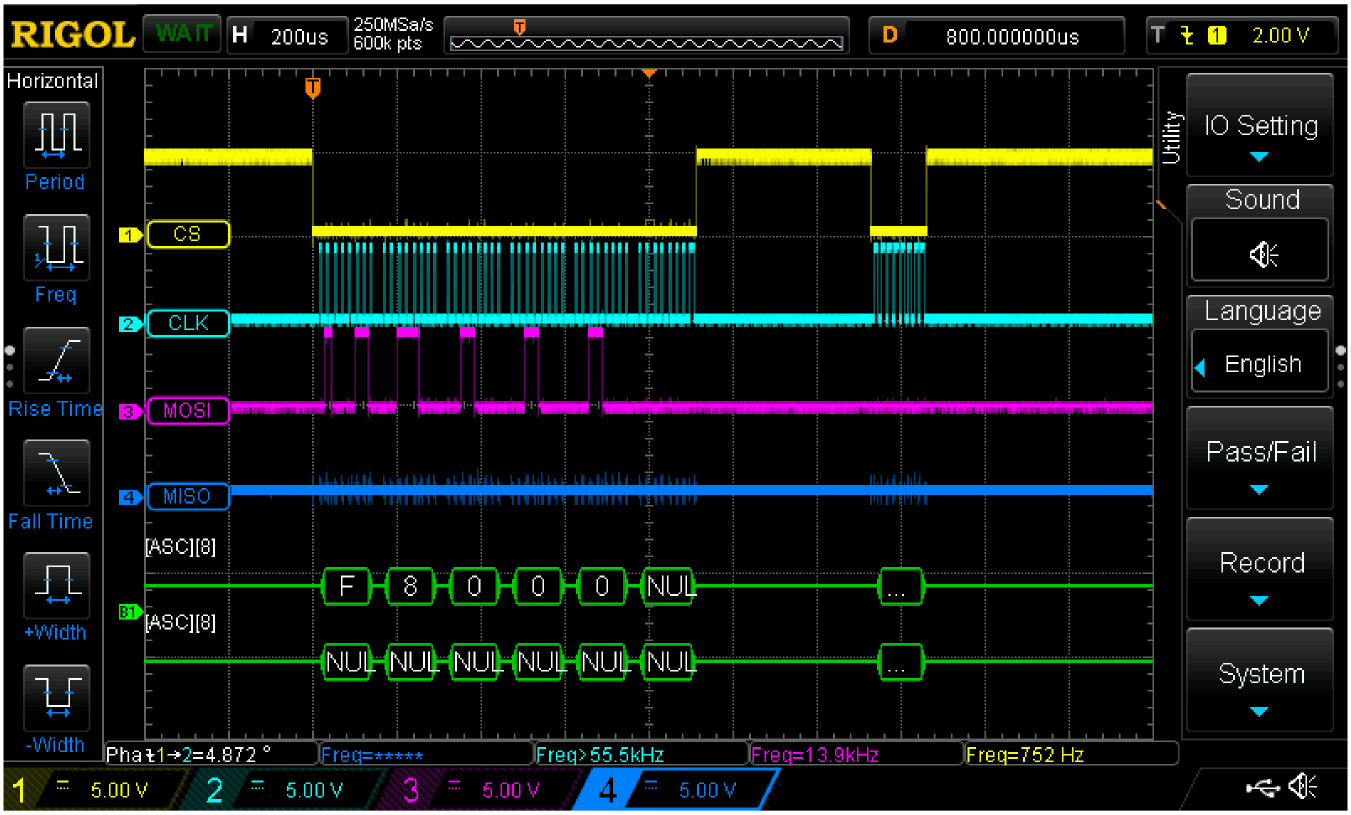


Figure 4 – Send Command Sequence, read back of a single NULL.

Continued:

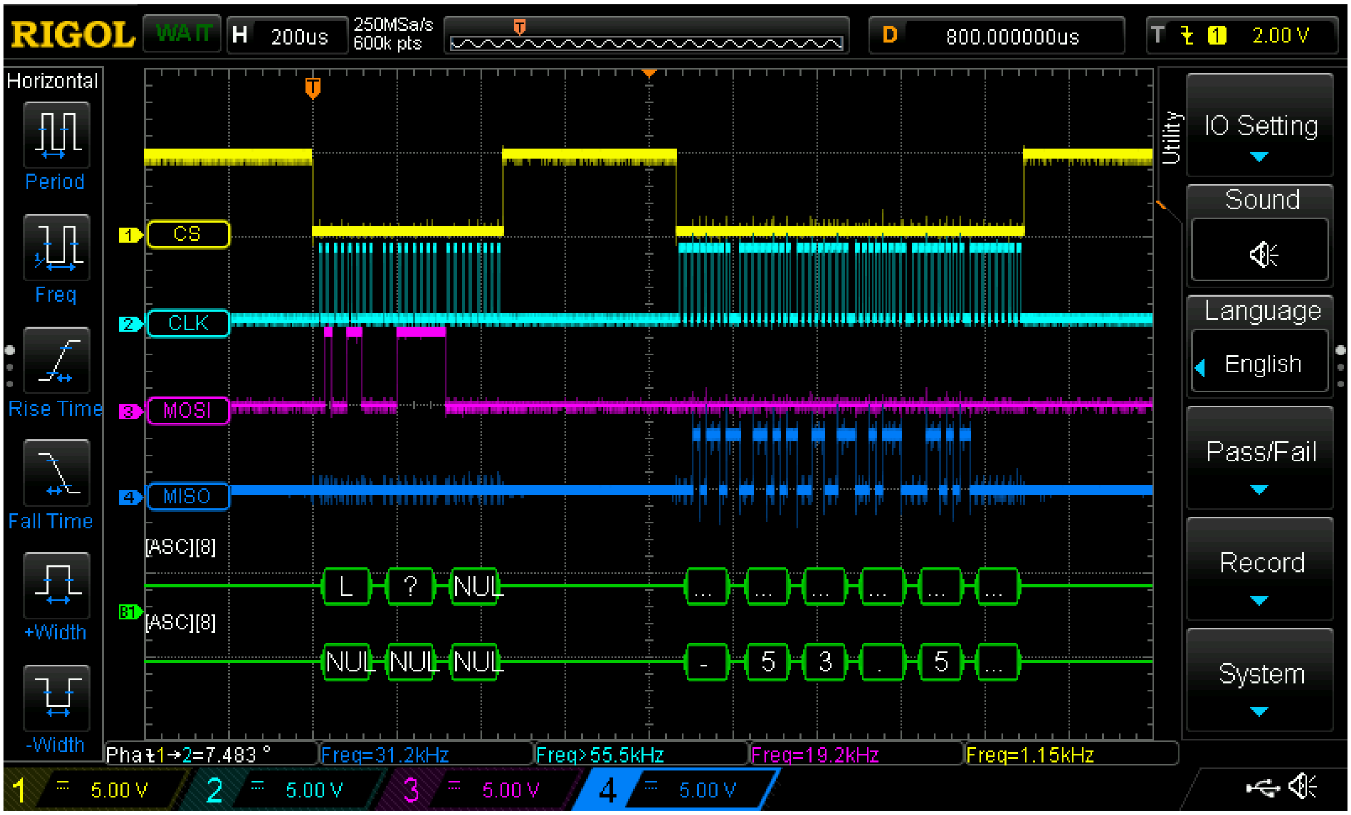


Figure 5 - Read Data Sequence.

8.0 Communication syntax

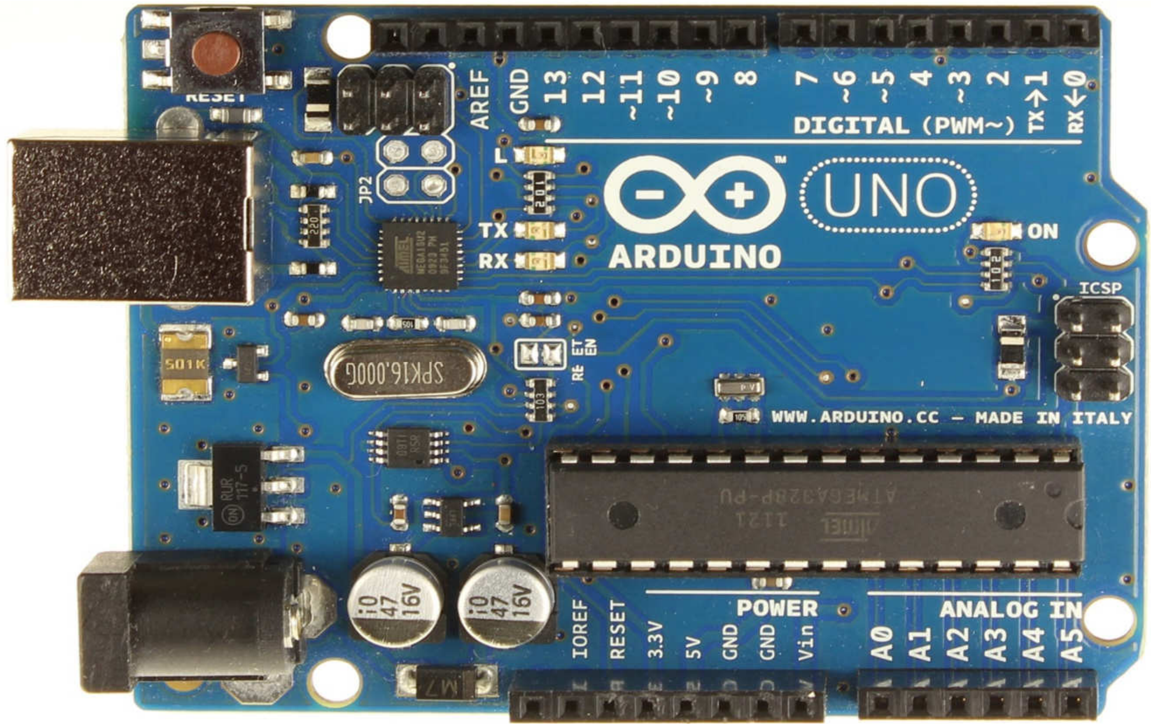
The following table describes the commands that the MLAT supports. This is a custom syntax created by Micro Lambda Wireless, Inc. All commands are sent and received in ASCII format. The commands are NOT case sensitive. These commands can be used with all forms of communication (USB and Serial).

Command	Function	Comments / Example
AP	Attenuator, preset to factory settings.	Copy NOVO Loc. 300-399 to 0-99
C0	Calibration mode OFF	Turns off cal. Mode. Attenuator operates normally.
C1	Calibration mode ON	Turns on cal. Mode, uses freq./level correction data from NOVO cal table.
C?	Request calibration state, returns ON or OFF	C? returns ON
F	Set Frequency in MHz (ASCII) (Dec. #), used in calibrated mode only.	ASCII freq in MHz: xxxxx.x (example: F10150.0), used for cal table index
FA	Start freq. for calibration table. MHz	FA5500.0 - updates NOVO location 16, used in cal. Mode only.
FB	End freq. for calibration table. MHz	FB16550.0 - updates NOVO location 17, used in cal. Mode only.
FC	Step size for calibration table. MHz (Range = 10 to 100 MHz) default = 50	FC50.0 - updates NOVO location 18, used in cal. Mode only.
F?	Request current frequency setting in MHz (same as NOVO Loc. 0003)	F? returns 2500.0
FA?	Request Min Cal frequency setting in MHz (same as NOVO Loc. 0016)	FA? returns 500.0
FB?	Request Max Cal frequency setting in MHz (same as NOVO Loc. 0017)	FB? returns 6000.0
FC?	Request Cal frequency Step size in MHz (same as NOVO Loc. 0018)	FC? returns 50.0
ID	Initialize Cal data to 0 (no correction), clears data table in memory	Clear calibration data; (example: ID) Requires 3 seconds to complete.
L	Set Attenuator Level in dB (ASCII) (Dec. #)	example: (L-12.5)
L?	Request current Attenuator setting in dB (same as NOVO Loc. 0002)	L? returns -6.5
LC	Set CAL. Attenuator Level in dB (ASCII) (Dec. #)	example: (LC-3.5)
LC?	Request current CAL. Attenuator setting in dB (same as NOVO Loc. 0004)	LC? returns -3.5
Rxxxx	Read NOVO Location (see locations below)	R0000 or R0 will read the model number of the unit.
SC	Store cal freq data point in nonvolatile memory	SC0 = first data point, Fcalend - Fcalstart / calstep (6000 - 500 / 50) 111 points
SCL	Set Source Level Cal. Point, in dBm. (Level that the cal will take place at)	SCL5.0, in dBm. Stored in NOVO location 19.
SCL?	Request current Source Cal. Level setting in dBm (= NOVO Loc. 0019)	SCL? Returns 5.0
SR	Soft Microprocessor Reset	Reset Micro., clear var. run code from start; (example: SR)
ST	Self-Test	Run internal test, ST returns Pass or Fail + error message. Result in NOVO 28
V1	Read internal +3.3V supply voltage	3.3V = normal; (example: V1, then read data)
V2	Read internal -3.3V supply voltage	-3.3V = normal; (example: V2, then read data)
V3	Read internal +5.0V supply voltage	5.0V = normal; (example: V3, then read data)
	(Example = R1) Read NOVO Location, R = 24 Bytes.	(Command requires 1 - 4 numeric digits.)
R0000	Model Number	MLAT-0522ABC
R0001	Serial Number	12345678
R0002	Current Attenuation setting in dB	-12.5
R0003	Current Set Frequency in MHz	2500.0
R0004	Current CAL. Attenuation setting in dB	-1.5 or N/A
R0005	Nominal impedance In/Out in Ohms	50.0
R0006	Insertion loss in dB @ Max Frequency	-3.5 dB or -7.0 dB or +?? dB with (option ?)
R0007	Attenuation Range in dB	31.5 or 63.0 or 94.5
R0008	Attenuation step size in dB	0.50
R0009	Attenuation accuracy w/ impedance match	0.20 +1.5% or 0.40 +3.0%
R0010	Max RF input, in dBm	+27.0
R0011	VSWR, Max.	2.2:1
R0012	****RF min, in dBm (for option?)	10.0 or N/A
R0013	****RF max, in dBm (for option?)	15.0 or N/A
R0014	Fmin, in MHz	50.0
R0015	Fmax, in MHz	22000.0
R0016	Start frequency for calibration table, in MHz (Only used when Cal mode on)	5500.0 (Start freq of level correction table) = current FA setting
R0017	End frequency for calibration table, in MHz (Only used when Cal mode on)	16550.0 (End freq of level correction table) = current FB setting
R0018	Frequency calibration resolution in MHz (Step Size) (for option?)	50.0 or N/A = current FC setting
R0019	Source cal Level (Level that the cal will take place at)	5.0, in dBm.
R0020	Cal Level resolution in dB (Step Size) (for option?)	0.50 or N/A
R0021	Cal Level Attenuation range in dB, Max. (for option A, B, and C)	31.5 only
R0022	Cal Level flatness Spec. in +/- dB (+/- 1.5 = 3.0 total) (for option?)	1.5
R0023	Level switching speed, to +/- 0.1dB, in microseconds	50.0
R0024	Temp min, in Deg. C	0.0

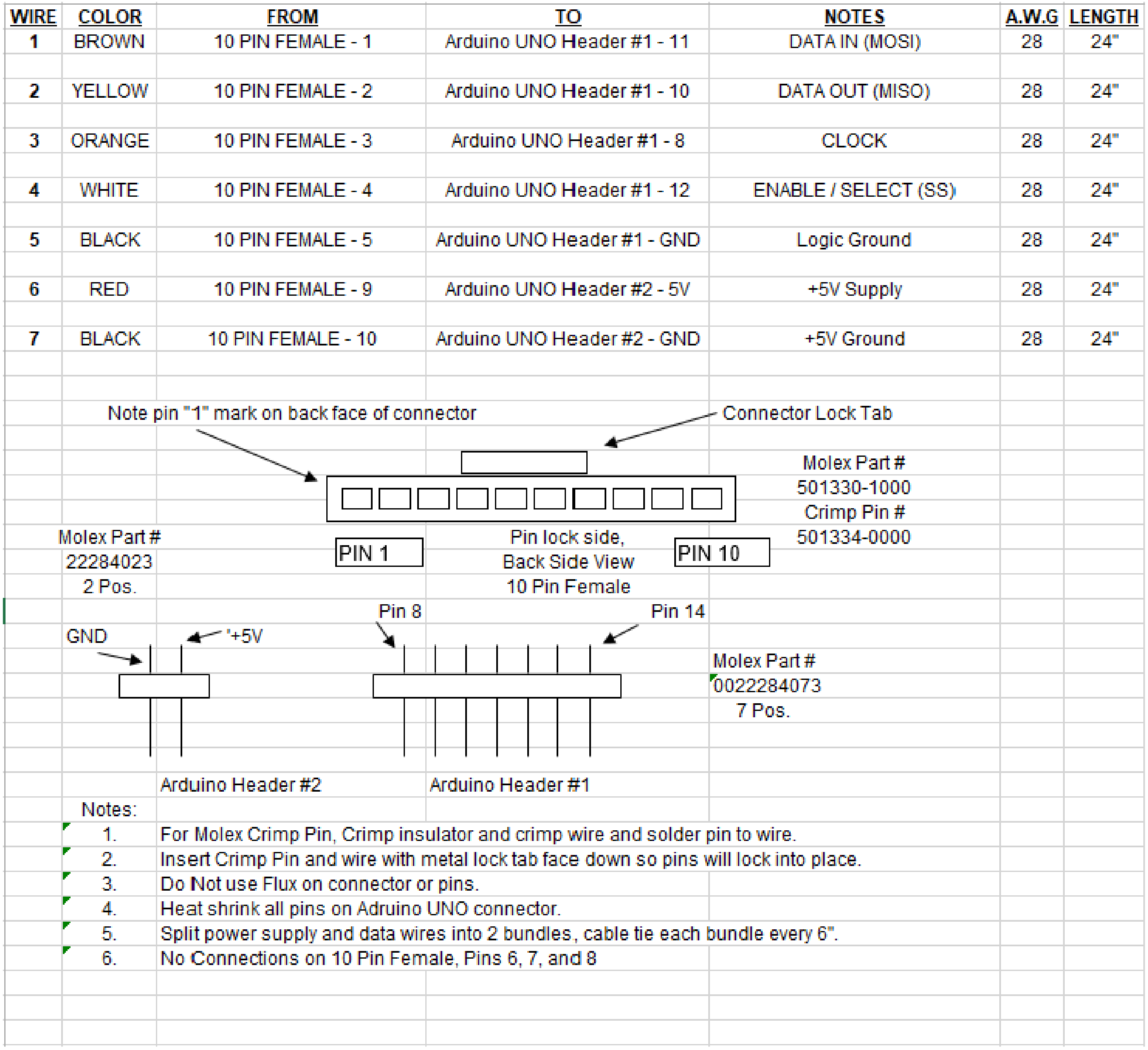
R0025	Temp max, in Deg. C	60.0
R0026	Level vs Frequency Cal status; Yes/No	Yes (ID command clears table & resets this to NO!)
R0027	Unit Calibration Status - ON/OFF	ON (uses freq./Level correction data from user calibration table)
R0028	Self-Test Results - Pass/Fail	Pass
R0029	+5V Supply current Max, in mA	150
R0030	Level CAL vs Frequency option installed.	Yes
R0031	Module Gain option installed. Yes/No	Yes
R0032	Customer part number, if shown on P.O.	123-45-6789 (Shown on unit label as PN:)
R0033	Firmware Version & date	1.5 Nov 16 2019
R0034	NOVO State - Locked/Unlocked	Locked
R0035	MLWI Sales (Job) number	07-0024
R0036	MLWI Product Outline Drawing # and Revision	99-0221-027 A
R300-399	Config data backup safe area, SF - save factory stores data here.	Backup copy of NOVO location 0000 to 0040 (Config Data)
R500-2000	Freq. cal data, stored in XX MHz increments, FA to FB	0 = no correction. 50-22000 / 50 = 439 cal points / locations

9.0 Arduino to MLAT Serial Interface Example

In this section we offer an example of serial interface communication using an Arduino UNO microcontroller, coded in C language. The Arduino series of educational prototype boards is a simple and effective way to communicate with the MLAT Attenuator. They are readily available for purchase on many web sites, including Amazon.com, for under $15.00.



The wiring interconnection between the Arduino and the MLAT is as follows:



WIRE	COLOR	FROM	TO	NOTES	A.W.G	LENGTH
1	BROWN	10 PIN FEMALE - 1	Arduino UNO Header #1 - 11	DATA IN (MOSI)	28	24"
2	YELLOW	10 PIN FEMALE - 2	Arduino UNO Header #1 - 10	DATA OUT (MISO)	28	24"
3	ORANGE	10 PIN FEMALE - 3	Arduino UNO Header #1 - 8	CLOCK	28	24"
4	WHITE	10 PIN FEMALE - 4	Arduino UNO Header #1 - 12	ENABLE / SELECT (SS)	28	24"
5	BLACK	10 PIN FEMALE - 5	Arduino UNO Header #1 - GND	Logic Ground	28	24"
6	RED	10 PIN FEMALE - 9	Arduino UNO Header #2 - 5V	+5V Supply	28	24"
7	BLACK	10 PIN FEMALE - 10	Arduino UNO Header #2 - GND	+5V Ground	28	24"

Notes:

1. For Molex Crimp Pin, Crimp insulator and crimp wire and solder pin to wire.
2. Insert Crimp Pin and wire with metal lock tab face down so pins will lock into place.
3. Do Not use Flux on connector or pins.
4. Heat shrink all pins on Adruino UNO connector.
5. Split power supply and data wires into 2 bundles, cable tie each bundle every 6".
6. No Connections on 10 Pin Female, Pins 6, 7, and 8

Arduino Code example:

```
/*
  Communicate with Micro Lambda Wireless, MLAT Attenuator, using the Arduino UNO USB serial monitor.
  For wire harness diagram see - MLAT User Manual, or MLAT Attenuator harness for Arduino interface 24
  inch.xls
  D.S. 7/15/20
*/
//UNO pin connections
const int clockPin = 8; //Serial Clock line (SCLK)
const int dataInPin = 10; //Serial Data (IN) line RX (MISO)
const int dataOutPin = 11; //Serial Data (OUT)line TX (MOSI)
const int selectPin = 12; //Serial Select line (SS)
//Pin 14(GND) = Serial Logic Ground

char sendArray[24]; //24 bits max sent
char recvArray[24]; //24 bits max received
int counter;
int b = 0;
int i = 0;
int x = 0;

void setup() {
  //set pins to output/input serial data
  pinMode(clockPin, OUTPUT);
  pinMode(dataInPin, INPUT);
  pinMode(dataOutPin, OUTPUT);
  pinMode(selectPin, OUTPUT);
  digitalWrite(selectPin, HIGH); //Init the selectPin high
  Serial.begin(9600);
  while (! Serial); // Wait untilSerial is ready - "For Leonardo"
  Serial.println("MLAT Attenuator Interface - Enter a Command");
}

void loop() {
  while (Serial.available() > 0) {
    //read all char's to send from serial monitor
    delay(1); //wait for serial port
    sendArray[i] = Serial.read();
    i = i + 1;
    delay(1); //wait for serial port

    if(i >= 24) //keeps loop happy
    {
      i = 23;
    }

    //when all char's received, call serial send frequency routine (sends ASCII #'s)
    if (Serial.available() == 0) {
      sendRecData();
    }
    //prints the command sent / received to the serial monitor window
    if (Serial.available() == 0) {
      printComm();
    }
  }
```

```
}

//send / receive command routine
void sendRecData() {
  digitalWrite(dataOutPin, LOW); //set data out low
  digitalWrite(clockPin, LOW); //set clock low
  digitalWrite(selectPin, LOW); //selectPin Low and hold low for as long as you are transmitting
  delayMicroseconds(2); //front porch

  //loop for array length of chars to send
  for (counter = 0; counter < 24; counter++) //24 bytes
  {
    //shift out 8 bits with xx uS delay (Bit Bang)
    for (b = 0; b < 8; b++)  {
      { digitalWrite(dataOutPin, !!(sendArray[counter] & (1 << (7 - b))));
      }
      //toggle clock line
      digitalWrite(clockPin, HIGH);
      digitalWrite(clockPin, LOW);
    }
    if(sendArray[counter] == 0)
    {
      break; //break out of loop if NULL sent (end of command)
    }
    delayMicroseconds(10); //delay between chars for byte visibility on Dig. scope
  }
  delayMicroseconds(2); //back porch

  digitalWrite(selectPin, HIGH); //pull the selectPin to save the data
  digitalWrite(dataOutPin, LOW); //set data out low when done
  digitalWrite(clockPin, LOW); //set clock low when done

  delayMicroseconds(400); //delay between write and read (ST - selftest command can take a long time to
reply)

  //read back data from unit if available
  digitalWrite(selectPin, LOW); //selectPin Low and hold low for as long as you are receiving
  delayMicroseconds(2); //front porch

  //read back data from unit if available, 24 bytes
  for (counter = 0; counter < 24; counter++) //receive 24 bytes
  {
    //shift in 8 bits with xx uS delay (Bit Bang)
    for (b = 0; b < 8; b++)  {
      //toggle clock line
      digitalWrite(clockPin, HIGH);
      { recvArray[counter] |= digitalRead(dataInPin) << (7 - b); //shift in bits MSB first
      }
      //toggle clock line
      digitalWrite(clockPin, LOW);
    }
    if(recvArray[counter] == 0)
    {
      break; //break out of loop if NULL received (end of data returned)
    }
    delayMicroseconds(10); //delay between chars for byte visibility on Dig. scope
  }
  delayMicroseconds(2); //back porch

  digitalWrite(selectPin, HIGH); //pull the selectPin High to finish
  digitalWrite(dataOutPin, LOW); //set data out low when done
  digitalWrite(clockPin, LOW); //set clock low when done

 }

//format and print command sent/received to serial monitor window
void printComm() {
  Serial.print("Command Sent = ");
  for (x = 0; x < 24; x++) {
    Serial.print(sendArray[x]);
  }
  Serial.println();
  Serial.print("Info Received = ");
  for (x = 0; x < 24; x++) {
    Serial.print(recvArray[x]);
  }

  Serial.println();
  //clear var.
  i = 0;
  x = 0;
  b = 0;
  memset(&sendArray[0], 0, sizeof(sendArray)); //clear array
  memset(&recvArray[0], 0, sizeof(recvArray)); //clear array
}
```

10.0 Hardware installation information

The unit may be installed into a system using four #2-56 X 0.875” long flat head screws, inserted from the top, in the 4 corner mounting holes. Alternately, four #4-40 X 0.25” screws can be attached from the bottom side through the mounting surface, in the 4 corner mounting holes. The Molex connector (J1) and the mating connector information are shown in the drawing number 99-0221-001 in this document. The USB connector is the standard Micro-B style.

11.0 Technical Support

For Technical support please contact:

Micro Lambda Wireless, Inc.
46515 Landing Pkwy.
Fremont, CA 94538
Ph: (510) 770-9221
Fax: (510) 770-9213

Email: sales@microlambdawireless.com

You can visit our website at http://www.microlambdawireless.com for updated information, specifications and downloads.

12.0 Warranty

Seller warrants for a period of twelve (12) months from the date of original shipment that the products will be free from defects in material and workmanship and design (if of Micro Lambda Wireless, Inc. design) and will be in conformity with applicable specifications and drawings and all other contractual requirements. However, this warranty shall not apply to any product which that has been subjected to misuse, misapplication, accident, improper installation, neglect, unauthorized repair, alteration, adjustment, inundation or fire. See the complete warranty and return policy document number 201-005 Rev- at our website at http://www.microlambdawireless.com.