

**AIRWOOD**

PROGRAMMING MODULE  
USER'S MANUAL

## PROGRAMMING INTRODUCTIONS

**AIRWOOD Programming Module is developed based on Arduino, which is an open-source hardware and software company designing and manufacturing single-board microcontrollers and microcontroller kits.**

### SUMMARY

The AIRWOOD Programming Module is based on the ATmega328. It supports all the same IDEs as Arduino Nano.

### SPECIFICATION

<b>Microcontroller</b>	ATmega328
<b>Architecture</b>	AVR
<b>Operating Voltage</b>	5 V
<b>Flash Memory</b>	32 KB of which 2 KB are used by bootloader
<b>SRAM</b>	2 KB
<b>Clock Speed</b>	16 MHz
<b>IDE</b>	Arduino IDE MIXLY Scratch for Arduino MDesigner

### RECOMMENDED GRAPHIC PROGRAMMING IDE - MIXLY

Mixly uses graphics programming to code and support Arduino systems.

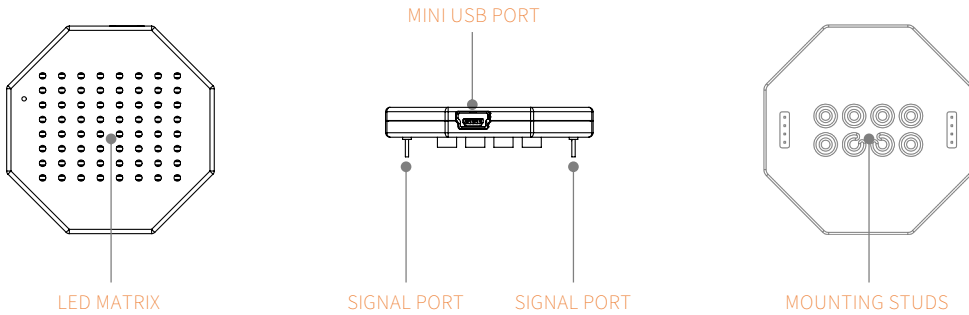
### Installation Instructions

Use the following link to Download Mixly Software and User Manual:

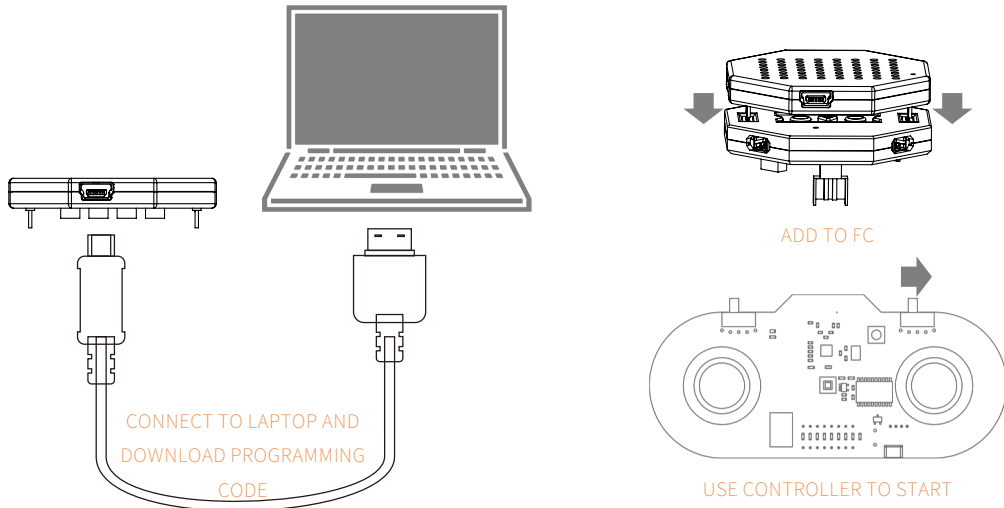
<http://www.uavi-tech.com/downloads>

## PROGRAMMING MODULE

For more information about programming, please refer to P18 **PROGRAMMING INSTRUCTIONS**.



## INSTRUCTIONS FOR USE

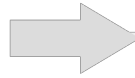


## MIXLY INTRODUCTIONS

What is **GRAPHICS PROGRAMMING** ?

```
#include "Wire.h"
uint8_t i,j,r,g,b,temp[4];
void setColor(uint8_t x, uint8_t ye)
Wire.beginTransmission(6);
temp[0] = 0x80|(y<<3)|x;
temp[1] = red;
temp[2] = 0x20|green;
temp[3] = 0x40|blue;
Wire.write(temp, 4);
Wire.endTransmission();
```

COMPLICATED CODE

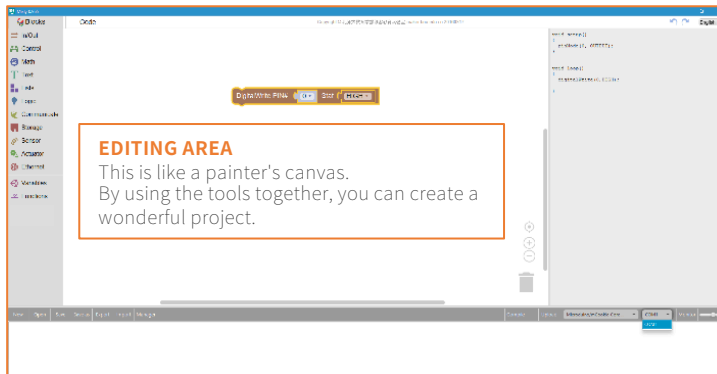


SIMPLE DRAG AND DROP CODE

### Interface

#### CODE BLOCKS

This is like an engineer's toolbox. All our tools can be found in this area.



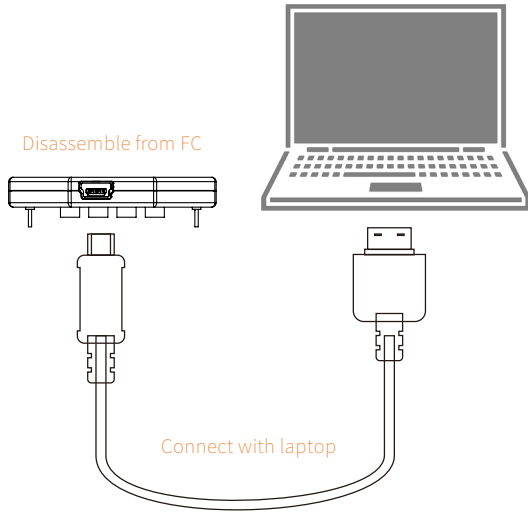
#### SOURCE CODE

This is like a scientist's microscope. We can see what are the source code that makes up the modules we are dragging into the editing area.

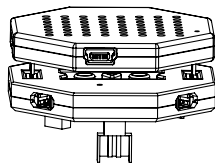
#### TOOLBAR

This area is for everything aside from coding, including upload, save, serial, selecting boards, etc.

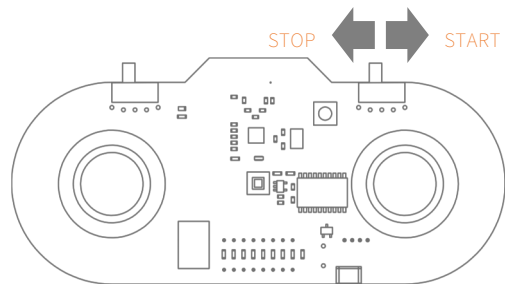
## UPLOAD YOUR PROGRAMMING CODE TO AIRWOOD



Choose “Arduino Nano ATmega 328”  
Then press upload



Add to Flight Controller  
**MAKE SURE FC IS POWERED OFF**



Turn Right Switch to RIGHT to start program  
Turn Right Switch to LEFT to stop program

## AIRWOOD PROGRAMMING GUIDE

### LED PART

Let the LED x **1** (1~8), y **1** (1~8) lit

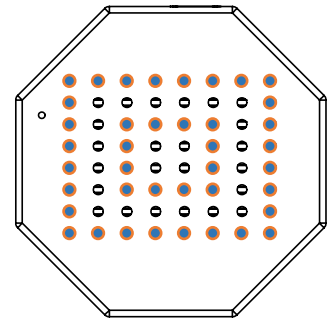
NAME	PARAMETERS	RETURN
Let the LED lit	Int, 1<=x<=8, 1<=y<=8	-
<b>FUNCTION</b>	Lets the x column y row led of the 8*8 LED matrix on the programming module light up.	

Let the **1** (1~8) line LED lit as **1** **1** **1** **1** **1** **1** **1** **1** (1 for lit,0 for off)

NAME	PARAMETERS	RETURN
Let one line LED lit	8 Char	-
<b>FUNCTION</b>	Lets one line of the 8*8 LED matrix light	

### EXAMPLE

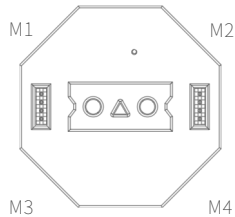
Let the <b>8</b> (1~8) line LED lit as <b>1</b> <b>1</b> <b>1</b> <b>1</b> <b>1</b> <b>1</b> <b>1</b> <b>1</b> (1 for lit,0 for off)
Let the <b>7</b> (1~8) line LED lit as <b>1</b> <b>0</b> <b>0</b> <b>0</b> <b>0</b> <b>0</b> <b>0</b> <b>1</b> (1 for lit,0 for off)
Let the <b>6</b> (1~8) line LED lit as <b>1</b> <b>0</b> <b>1</b> <b>1</b> <b>1</b> <b>1</b> <b>0</b> <b>1</b> (1 for lit,0 for off)
Let the <b>5</b> (1~8) line LED lit as <b>1</b> <b>0</b> <b>1</b> <b>0</b> <b>0</b> <b>1</b> <b>0</b> <b>1</b> (1 for lit,0 for off)
Let the <b>4</b> (1~8) line LED lit as <b>1</b> <b>0</b> <b>1</b> <b>0</b> <b>0</b> <b>1</b> <b>0</b> <b>1</b> (1 for lit,0 for off)
Let the <b>3</b> (1~8) line LED lit as <b>1</b> <b>0</b> <b>1</b> <b>1</b> <b>1</b> <b>1</b> <b>0</b> <b>1</b> (1 for lit,0 for off)
Let the <b>2</b> (1~8) line LED lit as <b>1</b> <b>0</b> <b>0</b> <b>0</b> <b>0</b> <b>0</b> <b>0</b> <b>1</b> (1 for lit,0 for off)
Let the <b>1</b> (1~8) line LED lit as <b>1</b> <b>1</b> <b>1</b> <b>1</b> <b>1</b> <b>1</b> <b>1</b> <b>1</b> (1 for lit,0 for off)



## MOTOR CONTROL

Let the motor **1** (1~4) rotate with speed **10** (0~1000)

NAME	PARAMETERS	RETURN
MotorControl	whichMotor : Motor Selection, 1~4,int; speed :0~1000, int	-
<b>FUNCTION</b>	Lets the selected motors rotate at a specific speed	



## EXAMPLE

Makes the speed of the no. 1 motor slowly increase from 0~100, then slowly decrease.

```
programModel initial
if switch is on right?
do
  count with i from 0 to 100 step 1
  do
    Let the motor 1 (1~4) rotate with speed i (0~1000)
    Delay ms 10
  count with i from 100 to 0 step -1
  do
    Let the motor 1 (1~4) rotate with speed i (0~1000)
    Delay ms 10
else if switch is on left?
do
  Let the motor 1 (1~4) rotate with speed 0 (0~1000)
```

## ATTITUDE PARAMETERS

the angle(-90~90) of pitch

NAME	PARAMETERS	RETURN
getPitch	-	-90~90
<b>FUNCTION</b>	Gets the angle of the pitch.	

the angle(-90~90) of roll

NAME	PARAMETERS	RETURN
getRoll	-	-90~90
<b>FUNCTION</b>	Gets the angle of the roll.	

the angle(0~360) of yaw

NAME	PARAMETERS	RETURN
getYaw	-	0~360
<b>FUNCTION</b>	Gets the angle of the yaw.	

### EXAMPLE

Control the flow of led lights according to the horizontal angl

```
Declare r as int value 0
Declare p as int value 0
programModel initial
if switch is on right?
do
  r the angle(-90~90) of roll
  p the angle(-90~90) of pitch
  Let the LED x (r ÷ 10) + 4 (1-8) , y (p ÷ 10) + 4 (1-8) lit
```







## CONTROLLER

the value(0~1000) of  1 (1~4) channel

NAME	PARAMETERS	RETURN
getChannel	whichChannel : channelSelection, 1~4,int;	0-1000
<b>FUNCTION</b>	Get the channel value.	

## EXAMPLE

Using the right joystick ( Channel 1,2) to the LED flow.

```
Declare channel1 as int value 0
Declare channel2 as int value 0
programModel initial
if switch is on right?
do
  channel1 the value(0~1000) of 1 (1~4) channel
  channel2 the value(0~1000) of 2 (1~4) channel
  Let the LED x  channel2  100 (1~8), y  channel1  100 (1~8) lit
```

**FLIGHT CONTROL (LIGHT FLOW Sensor needed for a stable flight controlling.)**



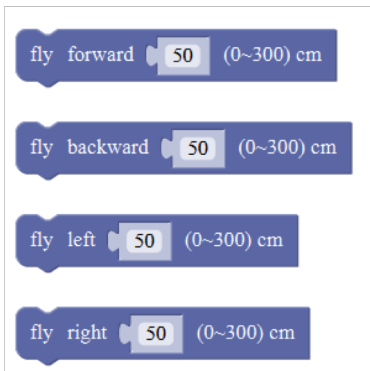
NAME	PARAMETERS	RETURN
take off	-	-
<b>FUNCTION</b>	Lets the drone fly to a steady altitude automatically.	



NAME	PARAMETERS	RETURN
land	-	-
<b>FUNCTION</b>	Lets the drone fly down and land	

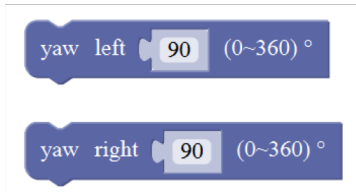


NAME	PARAMETERS	RETURN
setAltitude	altitude:0 - 300,int	-
<b>FUNCTION</b>	Lets the drone fly to a specified altitude.	

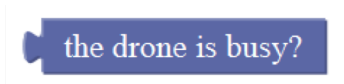


NAME	PARAMETERS	RETURN
fly forward fly backward fly left fly right	distance:0 - 300,int distance:0 - 300,int distance:0 - 300,int distance:0 - 300,int	-
<b>FUNCTION</b>	Lets the drone fly forward or backward or left or right with specified distance.	

## FLIGHT CONTROL (LIGHT FLOW Sensor needed for a stable flight controlling.)



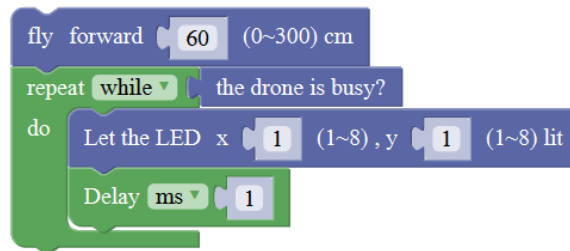
NAME	PARAMETERS	RETURN
yaw left yaw right	distance:0 – 360,int distance:0 – 360,int	-
<b>FUNCTION</b>	Lets the drone rotate with a specified yaw angle.	



NAME	PARAMETERS	RETURN
the drone is busy?	-	1 for ture 0 for false
<b>FUNCTION</b>	Determine whether the drone is executing flight instructions	

### EXAMPLE

if the drone is still busy to fly forward to a special distance, you can let the led light as you want, so as the fly backward,fly left,fly right, yaw left,yaw right and set altitude.



## EXAMPLE

```
programModel initial
if switch is on right?
do
  take off
  Delay ms 6000
  set altitude to 180 (0-300) cm
  repeat while the drone is busy?
  do
    Delay ms 1
  Delay ms 6000
  fly forward 80 (0-300) cm
  repeat while the drone is busy?
  do
    Delay ms 1
  Delay ms 6000
  fly backward 80 (0-300) cm
  repeat while the drone is busy?
  do
    Delay ms 1
  Delay ms 6000
  fly left 80 (0-300) cm
  repeat while the drone is busy?
  do
    Delay ms 1
  Delay ms 6000
  fly right 80 (0-300) cm
  repeat while the drone is busy?
  do
    Delay ms 1
  Delay ms 6000
  land
end program
```

- (1) Let drone fly to 180cm altitude.
- (2) Check if the drone is still working. If not working, delay 6 seconds.
- (3) Let drone fly forward with 80cm.
- (4) Check if the drone is still working. If not working, delay 6 seconds.
- (5) Let drone fly backward with 80cm.
- (6) Check if the drone is still working. If not working, delay 6 seconds.
- (7) Let drone fly left with 80cm.
- (8) Check if the drone is still working. If not working, delay 6 seconds.
- (9) Let drone fly right with 80cm.
- (10) Check if the drone is still working. If not working, delay 6 seconds.
- (11) Let drone land automatically.