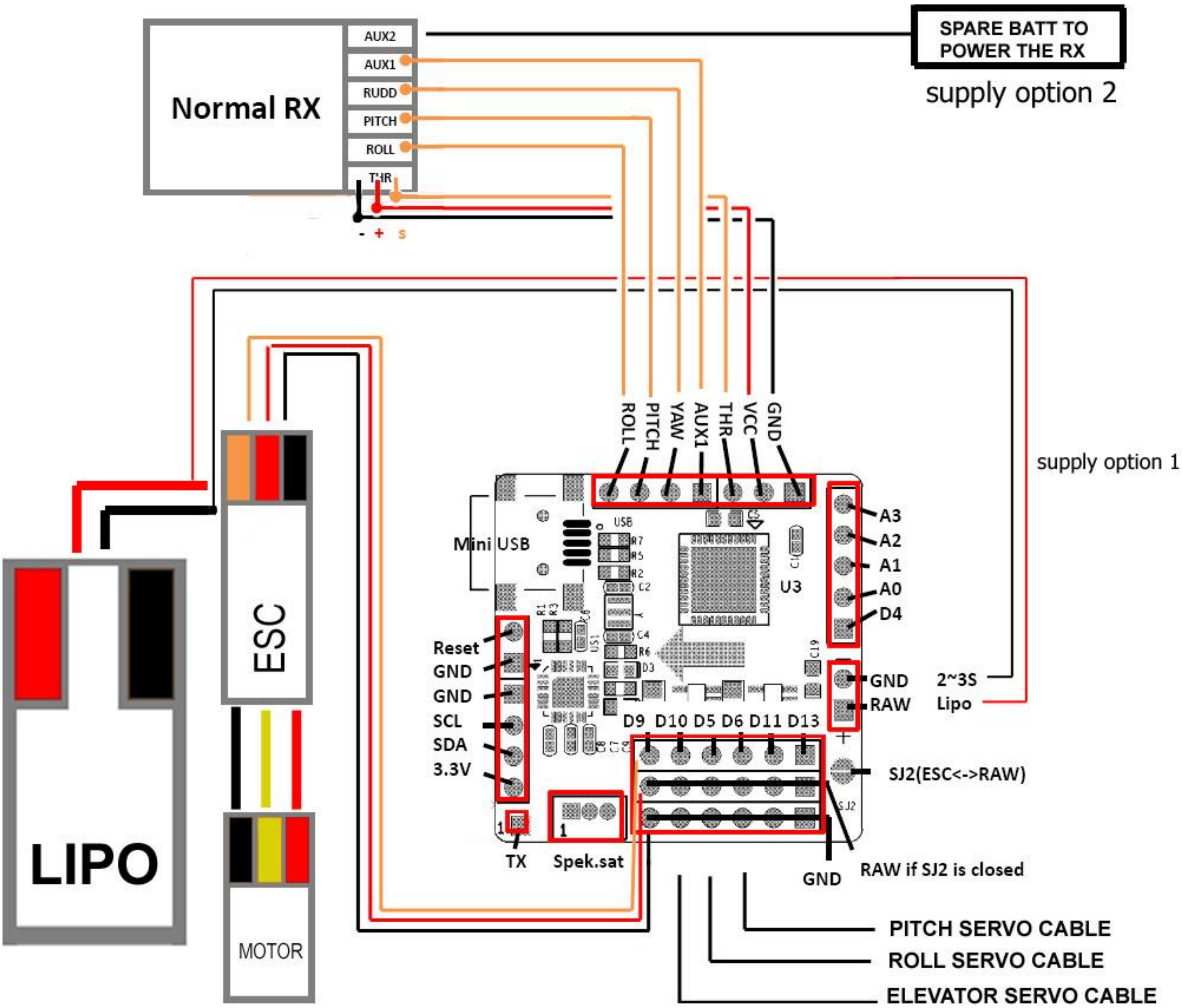
Nanowii ATMega32U4 for Dummies

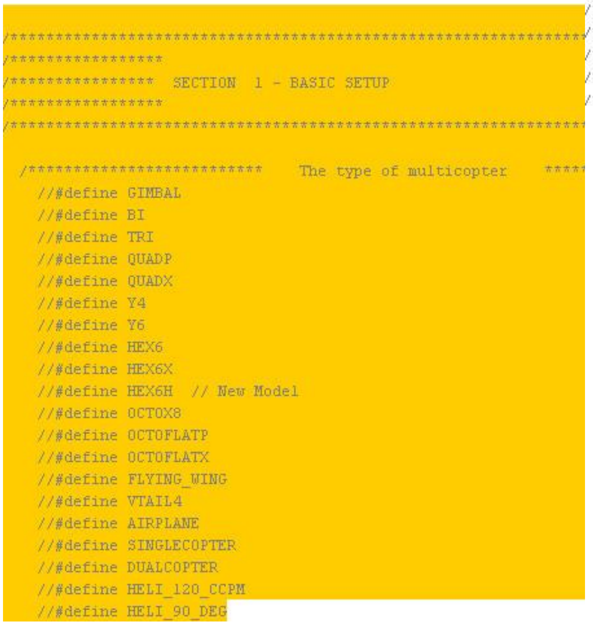
– How to config in AIRPLANE MODE (basic)

(by Pietro Iacubino after sleepless nights)

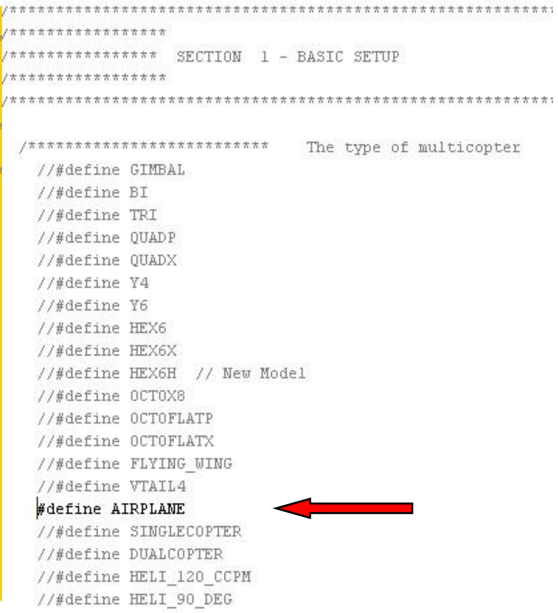


1. **Basic Knowledge**
   1.1. The Arduino board come pre-setted from manufacturing as multicopter (maybe). You need to reconfigure to Airplane Mode throught Arduino software and adjust/match the configuration via Multiwii software
   1.2. The main file is “multiwii.ino” that permits to read/modify the file “config.h” where we can activate (uncomment) or deactivate (comment) a specific row command.
   1.3. After open/modify files you need to upload those files into board and after end a double flash means the board have stored the file and is ready with new configuration.
   1.4. “//” are the first 2 characters to comment a row (deactivate a command line). Eliminating “//” to uncomment the line command and make active that row.
   1.5. Every uncommented row only, will be execute from the main programme.

2. **Upload the EEPROM**
   2.1. Download the software “arduino-1.0.5-windows” or latest release to get access/upload to the EEPROM from website http://www.arduino.cc/en/Main/Software
   2.2. Download the software “MultiWii_2_3” or latest release that enclosed 2 folders: MultiWii (where is located the basic arduino board configuration file, and MultiWii_Config (is the GUI where to look the parameter in live mode (how really works TX/board when moving the stick or moving the board) from website https://code.google.com/p/multiwii/downloads/list
   2.3. After this, install the “arduino-1.0.5-windows” and at the end of installation, connect the board to your computer with mini USB cable provided.
   2.4. Go to the folder MultiWii_2_3 → MultiWii and run MultiWii file with arduino brand icon. This action wake up the “arduino-1.0.5-windows” software (basic configuration file) and automatically the arduino IDE appear on the desktop.
   2.5. From Tools Menù →Arduino Type, choose Arduino Leonardo (specific for ATMega 32U4). The software is ready now to modify the “basic configuration file”.
   2.6. We have to do 2 things only (for beginners):
      2.6.1.Go to “Config.h” tab on SECTION 1 – BASIC SETUP eliminate double “//” from the row that you want (in this case we configure AIRPLANE mode)

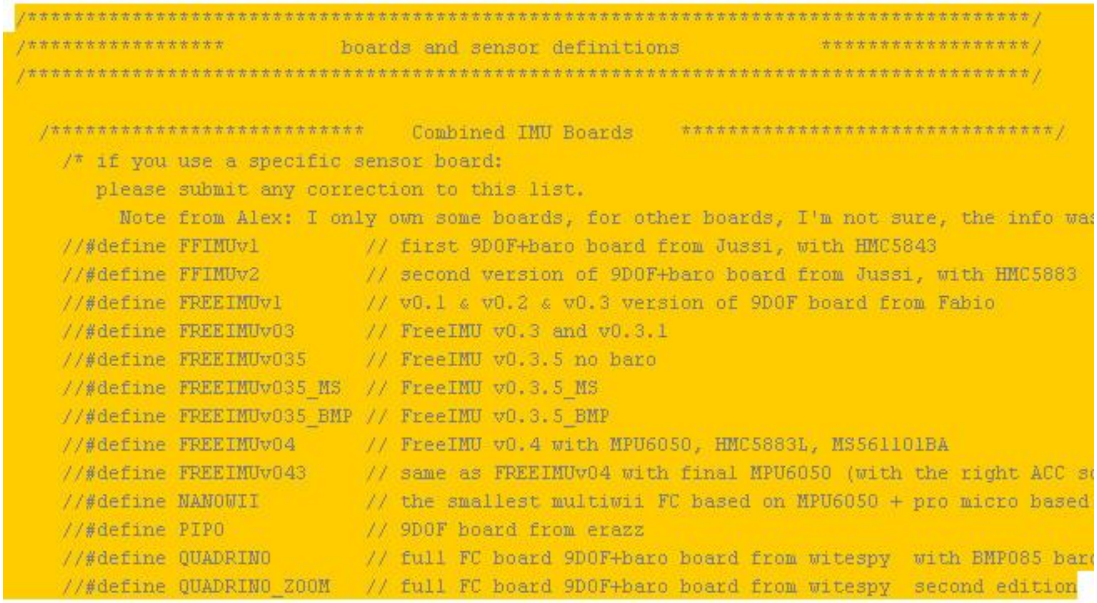


```
/**************************************************************************/
/****************
/****************  SECTION  1 - BASIC SETUP
/****************
/**************************************************************************/

  /**************************    The type of multicopter    ****
    //#define GIMBAL
    //#define BI
    //#define TRI
    //#define QUADP
    //#define QUADX
    //#define Y4
    //#define Y6
    //#define HEX6
    //#define HEX6X
    //#define HEX6H  // New Model
    //#define OCTOX8
    //#define OCTOFLATP
    //#define OCTOFLATX
    //#define FLYING_WING
    //#define VTAIL4
    //#define AIRPLANE
    //#define SINGLECOPTER
    //#define DUALCOPTER
    //#define HELI_120_CCPM
    //#define HELI_90_DEG
```

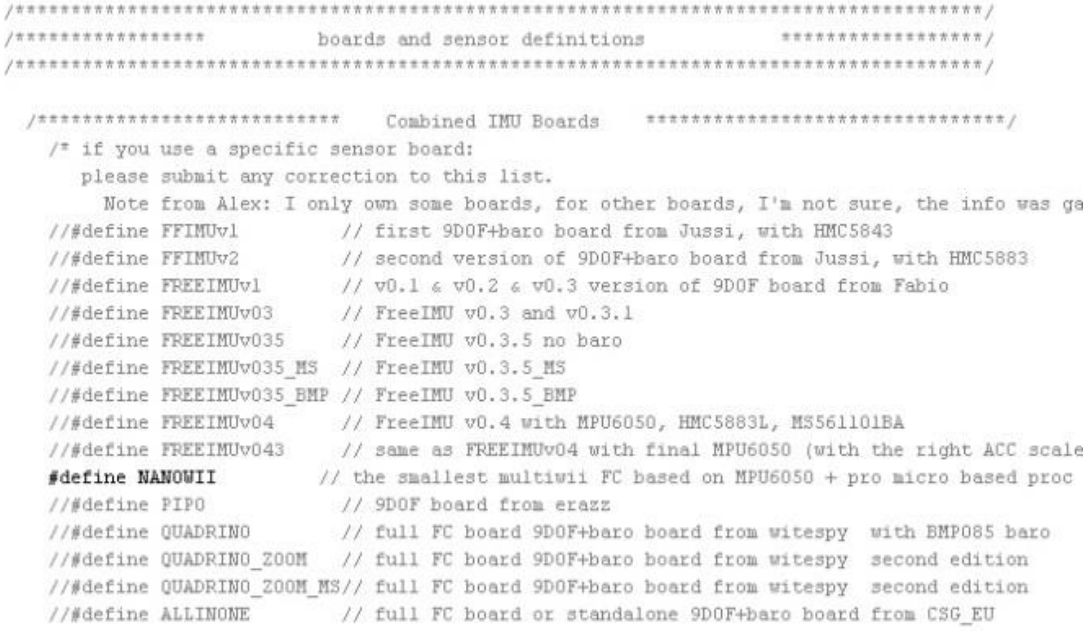


```
/*****************************************************************/
/****************
/****************  SECTION  1 - BASIC SETUP
/****************
/*****************************************************************/

  /**************************    The type of multicopter
    //#define GIMBAL
    //#define BI
    //#define TRI
    //#define QUADP
    //#define QUADX
    //#define Y4
    //#define Y6
    //#define HEX6
    //#define HEX6X
    //#define HEX6H  // New Model
    //#define OCTOX8
    //#define OCTOFLATP
    //#define OCTOFLATX
    //#define FLYING_WING
    //#define VTAIL4
    #define AIRPLANE
    //#define SINGLECOPTER
    //#define DUALCOPTER
    //#define HELI_120_CCPM
    //#define HELI_90_DEG
```

2.6.2. Go to “Config.h” tab on BOARDS AND SENSOR DEFINITIONS eliminate double “//” from the row that you want (in this case we configure NANOWII)



```
/**************************************************************************************/
/******************          boards and sensor definitions          ******************/
/**************************************************************************************/

  /****************************    Combined IMU Boards    ********************************/
    /* if you use a specific sensor board:
       please submit any correction to this list.
         Note from Alex: I only own some boards, for other boards, I'm not sure, the info wa
    //#define FFIMUv1         // first 9DOF+baro board from Jussi, with HMC5843
    //#define FFIMUv2         // second version of 9DOF+baro board from Jussi, with HMC5883
    //#define FREEIMUv1       // v0.1 & v0.2 & v0.3 version of 9DOF board from Fabio
    //#define FREEIMUv03      // FreeIMU v0.3 and v0.3.1
    //#define FREEIMUv035     // FreeIMU v0.3.5 no baro
    //#define FREEIMUv035_MS  // FreeIMU v0.3.5_MS
    //#define FREEIMUv035_BMP // FreeIMU v0.3.5_BMP
    //#define FREEIMUv04      // FreeIMU v0.4 with MPU6050, HMC5883L, MS561101BA
    //#define FREEIMUv043     // same as FREEIMUv04 with final MPU6050 (with the right ACC s
    //#define NANOWII         // the smallest multiwii FC based on MPU6050 + pro micro based
    //#define PIPO            // 9DOF board from erazz
    //#define QUADRINO        // full FC board 9DOF+baro board from witespy  with BMP085 bar
    //#define QUADRINO_ZOOM   // full FC board 9DOF+baro board from witespy  second edition
```



```
/**************************************************************************************/
/******************          boards and sensor definitions          ******************/
/**************************************************************************************/

  /****************************    Combined IMU Boards    ********************************/
    /* if you use a specific sensor board:
       please submit any correction to this list.
         Note from Alex: I only own some boards, for other boards, I'm not sure, the info was ga
    //#define FFIMUv1         // first 9DOF+baro board from Jussi, with HMC5843
    //#define FFIMUv2         // second version of 9DOF+baro board from Jussi, with HMC5883
    //#define FREEIMUv1       // v0.1 & v0.2 & v0.3 version of 9DOF board from Fabio
    //#define FREEIMUv03      // FreeIMU v0.3 and v0.3.1
    //#define FREEIMUv035     // FreeIMU v0.3.5 no baro
    //#define FREEIMUv035_MS  // FreeIMU v0.3.5_MS
    //#define FREEIMUv035_BMP // FreeIMU v0.3.5_BMP
    //#define FREEIMUv04      // FreeIMU v0.4 with MPU6050, HMC5883L, MS561101BA
    //#define FREEIMUv043     // same as FREEIMUv04 with final MPU6050 (with the right ACC scale
    #define NANOWII         // the smallest multiwii FC based on MPU6050 + pro micro based proc
    //#define PIPO            // 9DOF board from erazz
    //#define QUADRINO        // full FC board 9DOF+baro board from witespy  with BMP085 baro
    //#define QUADRINO_ZOOM   // full FC board 9DOF+baro board from witespy  second edition
    //#define QUADRINO_ZOOM_MS// full FC board 9DOF+baro board from witespy  second edition
    //#define ALLINONE        // full FC board or standalone 9DOF+baro board from CSG_EU
```

2.6.3. At the end of the modifying, save the configuration files in a new folder to preserve a clear basic file for future reference. Remember: the new folder must have the same name of the arduino Main File (.ino). Example: filename → ” Airplane_mode.ino” and containing folder name will be “Airplane_mode”

2.6.4. Save to save the file on the computer , Check to check the file for maximum dimension and presence of errors, Upload to store the file in the board EEPROM (double flashing means FILE RECEIVED and board operating)

3. Connecting parts

3.1. Connect single pin wire (what color you prefer) from RX Roll output signal to Board Roll input signal single socket

3.2. Connect single pin wire (what color you prefer) from RX Pitch output signal to Board Pitch input signal single socket

3.3. Connect single pin wire (last color) from RX Yaw output signal to Board Yaw input signal single socket

3.4. Connect 3 pins wire from RX Throttle channel output to Board GND/VCC/THR input signal 3 pins socket (as show by picture)

3.5. Connect your servos in order: to Board D10 socket (elevator), Board D5 socket (roll) and Board D6 socket (pitch)

3.5.1. For analog servos (50 KHz) you must modify the file “config h” at specific row as you are using ANALOG servos

3.5.2. For digital servos (300 KHz) you must modify the file “config h” at specific row as you are using DIGITAL servos

3.6. Connecting motor

3.6.1. If you have an ESC connect ESC 3 pins wire to Board D9 socket, but you must provide feed to your RX by spare battery connected to the RX (supply option 2) or from lipo (3s) directly on the board to GND/RAW pins (supply option 1)

3.6.2. If you have ESC with BEC functions connect ESC/BEC 3 pins wire to Board D9 socket only (because the VCC is provided by BEC side to the Board D9 output socket (D10-D5-D6-D11-D13 also)

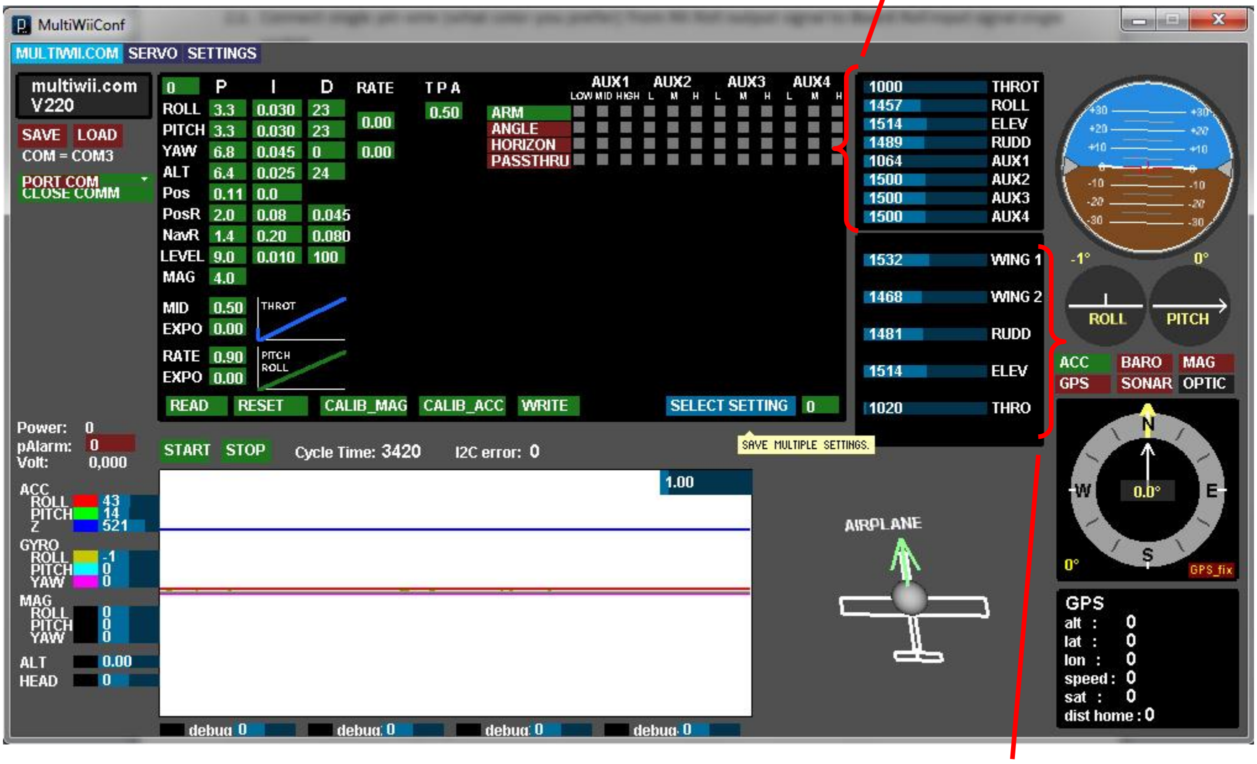
4. Testing

4.1. The first important thing is “Board Full Function activate/deactivate” moving YAW stick full right (opposite if you have the RUD command inverted) to activate the board all functions (engine will start). Moving YAW stick full left to deactivate the board (engine will stop). Remember when you power the board you feed the servos and the Gyro/Accelerators only. To activate or deactivate the motor you need to play with YAW stick full right or left at minimun throttle settings (Pay attention, cause injury)

4.2. On your computer open folder MultiWii_2_3 → MultiWiiConf → application.windows32 (if your OS is MS Windows 7) and double click to run MultiWiiConf

4.3. MultiWiiConf is the GUI that shows the relations between components (TX/RX-Board-Servos-Motor).

4.4. Click on RECONNECT to connect the TX/RX to the Board and servos/motor via COM port (in this case alias is USB port)



multiwii.com V220

COM = COM3

0	P	I	D	RATE	TPA
ROLL	3.3	0.030	23	0.00	0.50
PITCH	3.3	0.030	23		
YAW	6.8	0.045	0	0.00	
ALT	6.4	0.025	24		
Pos	0.11	0.0			
PosR	2.0	0.08	0.045		
NavR	1.4	0.20	0.080		
LEVEL	9.0	0.010	100		
MAG	4.0				

MID 0.50
EXPO 0.00
RATE 0.90
EXPO 0.00

SELECT SETTING 0

1000	THROT
1457	ROLL
1514	ELEV
1489	RUDD
1064	AUX1
1500	AUX2
1500	AUX3
1500	AUX4

1532	WING 1
1468	WING 2
1481	RUDD
1514	ELEV
1020	THRO

-1° 0°

Power: 0
pAlarm: 0
Volt: 0,000

Cycle Time: 3420 I2C error: 0

1.00

ACC	ROLL	43
	PITCH	14
	Z	521
GYRO	ROLL	-1
	PITCH	0
	YAW	0
MAG	ROLL	0
	PITCH	0
	YAW	0
ALT		0.00
HEAD		0

0.0°

0°

GPS
alt : 0
lat : 0
lon : 0
speed : 0
sat : 0
dist home : 0

debug 0 debug 0 debug 0 debug 0

COMMAND APPLIED GRID BY TX

OUTPUT RESULTS GRID FROM BOARD

4.5. For each movement on the TX sticks, corresponds the related channel value variation on the Command Applied Grid (if you have the servos connected to the Board look its moving)

4.6. For each movement applied to the Board (simulating the airplane in flight), corresponds the related channel/direction value variation on the Output Results Grid (if you have the servos connected to the Board look its moving according to ACC/GYRO functions)

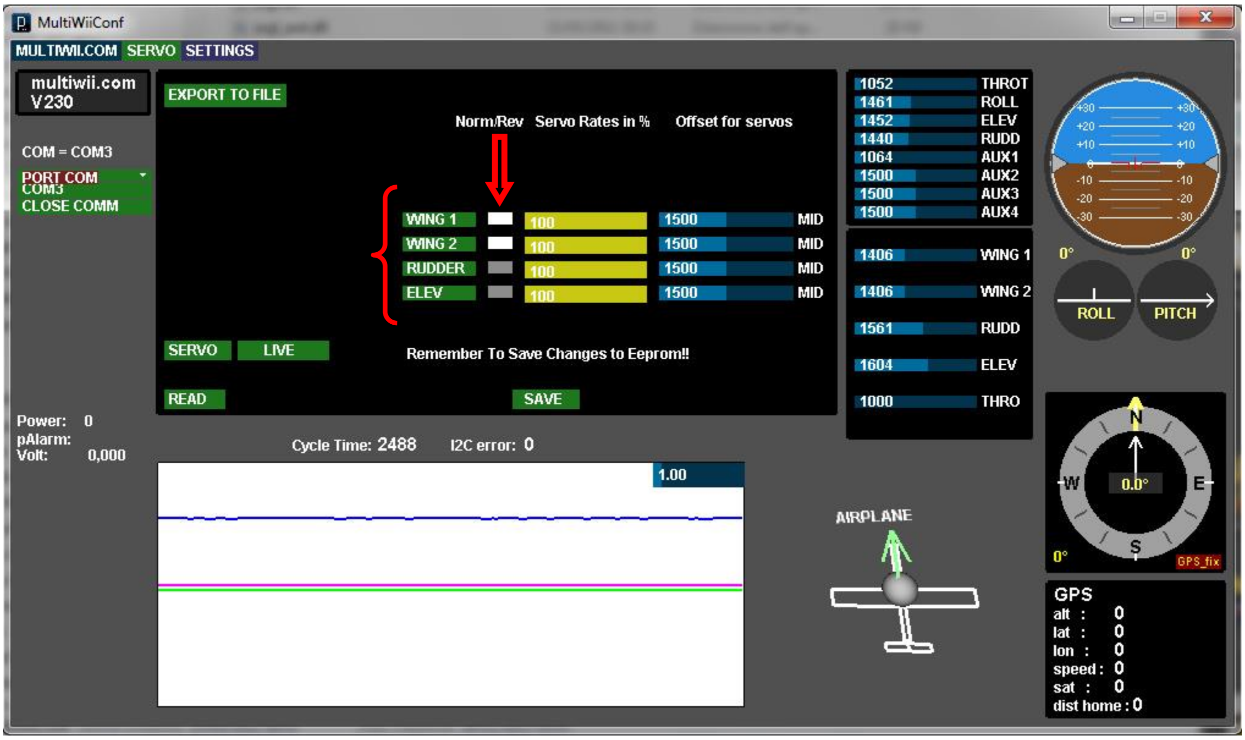
5. Enjoying

5.1. You can enjoy modifying the file “config.h” to get best options you needed.

5.2. Just in case of excessive RC model vibrations, you can modify the “gyro filters”
→ **#define GYRO_SMOOTHING {20, 20, 3}** (activated) to decrease the gyro sensibility.

6. Extremely care

6.1. Ensuring the right direction of gyro corrections: go to SERVO TAB and set NOR or REV to adjust gyro corrections direction +/-, in this manner:



	Norm/Rev	Servo Rates in %	Offset for servos	
WING 1	□	100	1500	MID
WING 2	□	100	1500	MID
RUDDER	□	100	1500	MID
ELEV	□	100	1500	MID

Remember To Save Changes to Eeprom!!

Cycle Time: 2488   I2C error: 0

COM = COM3

Power: 0
pAlarm:
Volt: 0,000

1052	THROT
1461	ROLL
1452	ELEV
1440	RUDD
1064	AUX1
1500	AUX2
1500	AUX3
1500	AUX4

1406	WING 1
1406	WING 2
1561	RUDD
1604	ELEV
1000	THRO

GPS
alt : 0
lat : 0
lon : 0
speed : 0
sat : 0
dist home : 0

6.1.1.Scrolling the airplane PITCH ND (nose down) → gyro response ELEV UP to get NOSE UP

6.1.2.Scrolling the airplane PITCH NU (nose up) → gyro response ELEV DN to get NOSE DN

6.1.3.Scrolling the airplane ROLL RH → gyro response RH AILERON DN and LH AILRON UP

6.1.4.Scrolling the airplane ROLL LH → gyro response RH AILERON UP and LH AILRON DN

6.1.5.Scrolling the airplane YAW RH → gyro response RUDDER LH

6.1.6.Scrolling the airplane YAW LH → gyro response RUDDER RH

6.2. Saving: press SAVE and EXPORT TO FILE to store all settings in EEPROM

6.3. At this step DO NOT push RESET, otherwise you lost all settings