

## Installationsanleitung für MegapirateNG 3.0.1-R3

Quellen: [rcsky.de](http://rcsky.de) / [multicopter-rc.de](http://multicopter-rc.de)

Diskussionsthread: [megapirateng-3-0-1-r3-x-anleitung](http://megapirateng-3-0-1-r3-x-anleitung)

MegapirateNG 3.0.1-R3 ist kompatibel zu

- [AIOP V2](#)
- [HobbyKing AllInOnePro](#)
- [Hobbyking MultiWii Pro](#) mit ITG3205, BMA180, BMP085
- Black Vortex
- RCTimer Crius V2

Zum gegenwärtigen Zeitpunkt 19.12.2013 gilt die 3.0.1-R3 als aktuellste veröffentlichte Version.

### **Benötigte Software:**

Es wird eine spezielle Arduino Version benötigt, um MegaPirateNG sachgemäß installieren zu können.

Für Windows: [ArduPilot-Arduino-1.0.3](#)

Für MacOSX: [ArduPilot-Arduino-1.0.3](#)

gegenwertig aktuelle MegapirateNG Version: [MegapirateNG – APM 3.0.1-R3](#)

aktuellste Version ist hier downloadbar: [MegapirateNG – APM 3.x](#)

### **Vorbereitung:**

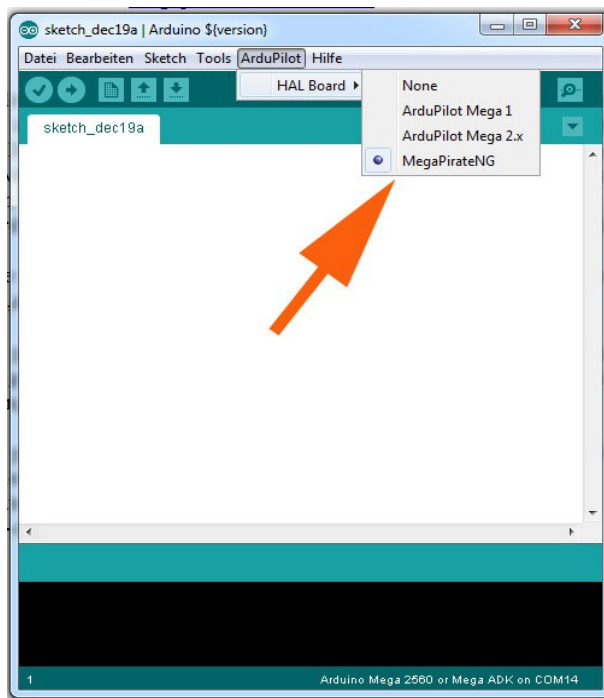
1. entpacke die ArduPilot-Arduino 1.0.3 Zip zum Bsp in den Ordner c:\megapirateng. In diesem Ordner sollte nun ein weiter Ordner ArduPilot-Arduino-1.0.3-windows vorzufinden sein. Bei der MacOSX Plattform eine Arduino Applikation im Ordner c:\megapirateng.
2. Nun kann die letzte Version von MegapirateNG in den Ordner c:\megapirateng entpackt werden. Es sollte nun ein weiterer Ordner, in unserem Bsp. mit der Version 1.0.3-R3, mit dem Archivnamen ardupilot-mpng-mpng-3.0.1-r3 im Ordner c:\megapirateng vorzufinden sein.
3. Sehr wichtig! Ersetze die pde.jar aus dem „ArduPilot-Arduino-1.0.3-windows“ Ordner mit der pde.jar aus dem „ardupilot-mpng-mpng-3.0.1-r3“ Ordner

Windows: Suche die pde.jar im Quellverzeichnis in unserem Bsp. hier c:\megapirateng\ardupilot-mpng-mpng-3.0.1-r3\pde.jar, kopiere diese und ersetze die pde.jar im folgenden Ordner c:\megapirateng\ArduPilot-Arduino-1.0.3-windows\lib

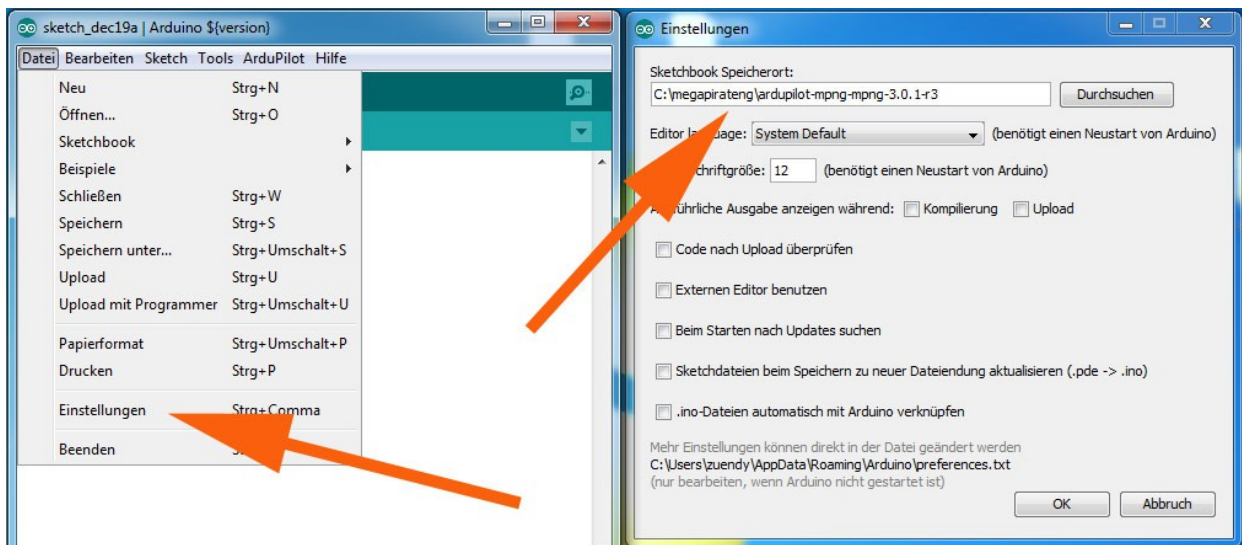
MacOSX: Suche die pde.jar hier: ~username/megapirateng/ardupilot-mpng-mpng-3.0.1-r3/pde.jar, kopiere diese und ersetze diese in  
~username/megapirateng/Arduino.app/Contents/Resources/Java

4. Starte nun ArduPilot-Arduino

5. Wähle über den Reiter ArduPilot\HAL Board das Flightcontrol MegaPirateNG aus.



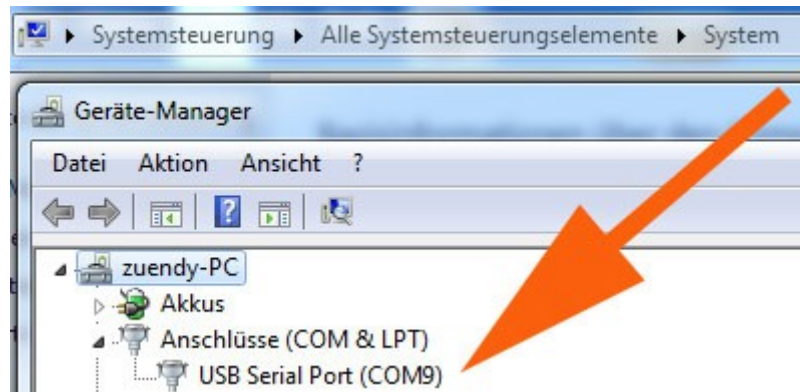
6. Öffne die Einstellungen und setze den Sketch Ordner als Verzeichnis, zum Bsp.:  
c:\megapirateng\ardupilot-mpng-mpng-3.0.1-r3



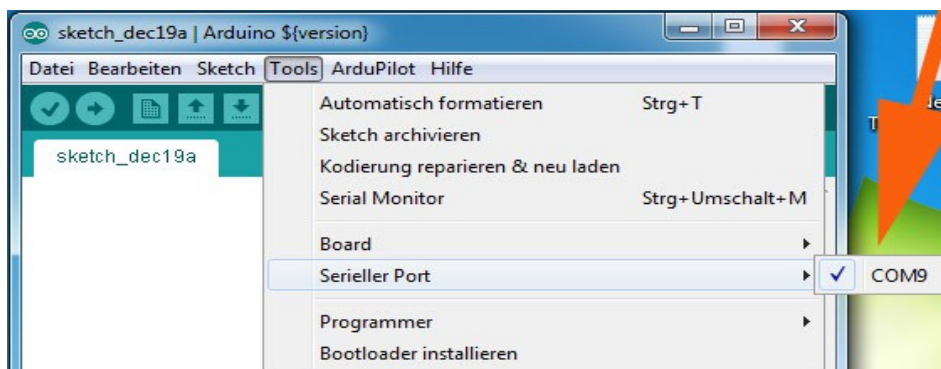
Es sollten alle Häkchen aus den Checkboxes entfernt werden.

7. Verbinde den Flight Controller mit deinem Computer. Achte darauf den Copter vom Akku genommen zu haben. Setze nun den richtigen Serial Port in der Arduino IDE.

Unter Systemsteuerung\Alle Systemsteuerungselemente\System\Geräte-Manager\Anschlüsse\USB Serial Port kannst du den richtigen Serial Port auslesen.

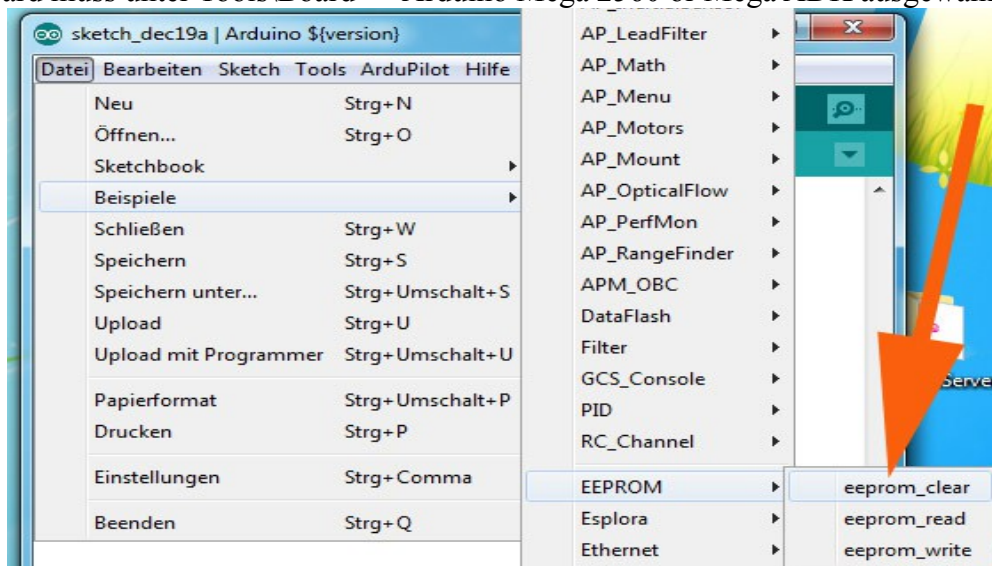


Setze den richtigen Serial Port in der Arduino IDE



Nun ist alles korrekt für die Erstbespielung des Flight Controls eingestellt. Doch meistens war schon ein andere Software aufgespielt. Um evtl. Rückstände komplett auszuschließen, sollte ein eeprom clear durchgeführt werden.

Dafür benötigen wir jedoch eine frische Version von Arduino, da die hier vorbereitete Version bei mir einen Fehler ausspuckte! Herunterladbar [hier](#). Empfohlen wird auch hier die Version 1.0.3. Als Arduinoboard muss unter Tools\Board → Arduino Mega 2560 or Mega ADK ausgewählt sein.



## **Kompilierung**

Nun öffnen wir die präferierte APM Autopilot Plattform (ArduCopter, Arduplane...) über Datei\Sketchbook.

Um den Quellcode kompilieren zu lassen klickt man das erste Icon oder drückt ctrl+R.

Ist alles ohne weitere Probleme kompiliert ist nun alles bereit um das UAV zu konfigurieren.

## **Generelle Konfiguration von MegaPirateNG**

Nach der Anpassung der Konfiguration muss alles gespeichert werden und eine erneute Kompilierung + Upload erfolgen.

### **Einstellung Flight Control**

MegapirateNG unterstützt unterschiedliche Modelle des Flight Controls. Das AIO V2 welches auf Ebay zu finden ist, ist identisch mit dem RCTimer Crius V2! Das AIOP V2 Board von Hobbyking ist kein !!!kein!!! AIO V2! Es ist die Revision V1.1 und gilt hier als Crius V1. Muss man nicht verstehen, sollte man jedoch akzeptieren.

Hat man den Sketch geöffnet sollte man auf den Reiter APM\_Config.h klicken und unter der Zeile

```
// Select Megapirate board type:
```

nachschauen, welches Flight Control ausgewählt ist.

Hier wird es am Bsp. Crius V1 beschrieben.

```
// Select Megapirate board type:
#define MPNG_BOARD_TYPE    CRIUS_V1
/*
  RCTIMER_CRIUS_V2      -- (DEFAULT!!!) Use ONLY for RCTimer CRIUS V2 board
  CRIUS_V1              -- Use this define for RCTimer CRIUS V1(1.1) board and
  HK_RED_MULTIWII_PRO  -- HobbyKing MultiWii Pro board with ITG3205 and BMA180,
  BLACK_VORTEX
*/
```

Besitzt man ein anderes Board einfach CRIUS\_V1 mit z.B. RCTIMER\_CRIUS\_V2 ersetzen

## Auswahl des Rahmen Typus

Standardmäßig ist hier der Quadrocopter Rahmen ausgewählt. Sollte ein anderer Rahmen verwendet werden wieder unter APM\_Config.h aber diesmal unter

```
#define FRAME_CONFIG
```

schauen.

Bsp.:

```
#define FRAME_CONFIG QUAD_FRAME
/*
 * options:
 * QUAD_FRAME
 * TRI_FRAME
 * HEXA_FRAME
 * Y6_FRAME
 * OCTA_FRAME
 * OCTA_QUAD_FRAME
 * HELI_FRAME
 */
```

Bei einem anderen Rahmentyp das Quad\_Frame durch das Entsprechende ersetzen.

## Einstellung der Fernsteuerung

Abhängig davon welcher Empfänger verwendet wird, stehen folgende Auswahlmöglichkeiten zur Verfügung:

1. Serial PPM on A8 pin selected (default)
2. Serial PPM (CPPM) on PL1 pin
3. Regular PWM inputs A8-A15

Dies kann unter `libraries\AP_HAL_MPNG\RCInput_MPNG.cpp` angepasst werden

### **Serial PPM (CPPM) auf PL1 pin**

Ändere `#define SERIAL_PPM SERIAL_PPM_ENABLED` zu `#define SERIAL_PPM SERIAL_PPM_ENABLED_PL1`

### **Standard PWM inputs A8-A15**

Ändere `#define SERIAL_PPM SERIAL_PPM_ENABLED` zu `#define SERIAL_PPM SERIAL_PPM_DISABLED`

## Ändern der Reihenfolge der Fernsteuerungs Eingänge

MegapirateNG benutzt die standard Einstellung der APM Hardware, es jedoch möglich auch diese anzupassen.

Die Konfiguration ist unter `libraries\AP_HAL_MPNG\RCInput_MPNG.cpp` zu finden.

Bsp.:

Unter

```
#define RC_MAPPING RC_MAP_STANDARD
```

steht folgendes:

```
#define RC_MAPPING RC_MAP_MULTIWII
/*
    RC_MAP_STANDARD 1
    RC_MAP_GRAUPNER 2
    RC_MAP_HIITEC 3
    RC_MAP_MULTIWII 4
    RC_MAP_JR 5
*/
```

Nun seine präferierte Steuerungsart auswählen.

## Anschlusspins für Erweiterungen

- Serial 0 (RX0,TX0) – USB, OSD (Minim OSD) oder Telemetrie (3DR, Xbee, Bluetooth)
- Serial 1 (RX1,TX1) – Serial1 wird aktuell (Stand 3.0.1-R3) nicht unterstützt geplant Mavlink
- Serial 2 (RX2,TX2) - GPS
- Serial 3 (RX3, TX3) – OSD (Minim OSD oder Telemetrie (3DR, XBEE, Bluetooth)

## Sonar

Wird in der Version 3.0.1-R3 noch nicht unterstützt.

## Kameraauslöser

Der Auslöser kann über PIN D46 am Flight Controller verbunden werden. Aktiviert wird er in den Kamera Gimbal Einstellungen im Mission Planner.

Weiterführende Links:

- [Flight Controls](#)
- [Multicopter](#)