

**Goal:**

The goal of this document is to explain the entities involved in the Paparazzi project. I will first describe about how a Paparazzi build system is configured for building of the project and what is the directory structure of the code base. This document will be a good resource for anyone who is starting first time to look at the code base of the paparazzi. Moreover, this document will always be evolving and I will continue to add the details as I progress in understanding the Paparazzi system.

**Aircraft Configuration:**

The Paparazzi system is highly configurable and can be used with many boards and hardware peripherals. Each aircraft is configured for following five elements through XML files:

- Airframe (conf/airframes): This file configures the firmware part of the aircraft which will run on the aircraft.
- Radio(conf/radios): This file contains description of the radio control transmitter signals.
- Telemetry(conf/telemetry): The files in the telemetry folder describes the format of messages which can be send from the aircraft to the ground.
- Flight\_plan(conf/flight\_plans): A flight plan sets the way points for an aircraft to specify it's trajectory.
- Settings(conf/settings): This file contains parameters like GUI elements which can be seen in the Ground Control Station.

The Paparazzi build system parses the above files and automatically generates “C” header and source files necessary for overall build process. Whenever a new aircraft is prepared one need to add a separate file for each of the above section.

Below are the descriptions of some of the important XML configuration and make files.

**Airframe(conf/airframes):**

The airframe folders contains configuration for many airframes like Microjet,Booz,Bixler. This file basically defines various constant parameters required for hardware peripherals(senors,motors) and generates a file “airframe.h” . The main sections defined in a airframe files are following:

- Firmware(conf/firmware): The firmware sections defines the type of the firmware being used in the aircraft. There are currently two types of firmwares being used in the Paparazzi airframes. The one is for “fixedwing aircraft” and other is “rotorcraft”. This section basically choose an appropriate make file from the “conf/firmware” folder for the build system.
- Modules(conf/modules): Modules can be a hardware unit or an algorithm. But whenever a new module is added, one doesn't have to change in the source code.
- Commands: The commands section defines the “commands” by which an aircraft is controlled. In fixedwing aircraft there are three commands “Throttle”, ”Roll”, and ”Pitch”.
- Command\_Laws: Command laws defines expressions consists of the values of the commands above.
- Servos: The servos section defines the name of real physical connectors to which servos is connected on the autopilot board.

**firmware.makefile(conf/fixedwing.makefile or conf/rotorcraft.makefile):**

The firmware section which is defined in airframe section of “airframe.xml” file, selects following elements for a aircraft.

- Board(conf/board.makefile): There are many types of board or architecture supported by

Paparazzi like stm32xxx, lpc21xxx. This section selects an architecture and compile it for a particular firmware in the aircraft.

- Subsystem Name(conf/firmware/subsystems/firmware\_name/subsystem): Subsystems are basically hardware units which are required for proper functioning of the aircraft like various sensors(Infrared,accelerometer), IMU(Inertial measurement unit), and gps. Each aircraft contains at least one IMU, AHRS and GPS, RadioControl, and Telemetry subsystem.

#### **board\_name.makefile(conf/boards):**

This make file decides about the architecture of the micro-controller against which the code is compiled. This makefile also defines which configuration file to be included in the code. There might be several macros which can represent various ports on the aircraft like GPS\_PORT, MODEM\_PORT.

#### **subsystem\_name.makefile(conf/firmware/subsystems/firmware\_name/subsystem):**

Each hardware peripheral present on the airframe board represented as a subsystem in Paparazzi. Paparazzi supports many types of subsystem for sensing and balancing the aircraft. Therefore, which subsystem makefile is chosen depends upon the firmware configuration of the aircraft. We can assume a subsystem to be a generic interface which represent a specific hardware unit. For example “gps\_ublox” and “gps\_nmea” both represents “GPS” subsystems in Paparazzi.

#### **An example for configuration of an aircraft:**

##### **Boards:**

Board_Name	Architecture	Board_Configuration File
PC	SIM	\$(PAPARAZZI_HOME)/sw/airborne/ boards/tiny_sim.h
Tiny_Version1.1	lpc21	\$(PAPARAZZI_HOME)/sw/airborne/\$ (BOARD_Name)_\$( (BOARD_VERSION).h\"

**Subsystems:**

<b>radio_control_ppm.makefile</b>	<b>telemetry_xbee_api.makefile</b>	<b>control.makefile</b>	<b>imu_analog.makefile</b>	<b>gps_ublox_utm.makefile</b>
\$ (PAPARAZZI_HOME)/sw/airborne/ subsystem/radio_control.c	\$ (PAPARAZZI_HOME)/sw/airborne/ subsystems/datalink/downlink.c	\$ (PAPARAZZI_HOME)/sw/airborne/ firmware/\$ (SRC_FIRMWARE)/stabilization/stabilization_attitude.c	\$ (PAPARAZZI_HOME)/sw/airborne/ subsystems/imu.c	\$ (PAPARAZZI_HOME)/sw/airborne/ subsystems/gps/gps_ubx.h
subsystem/radio_control/ppm.c	\$ (PAPARAZZI_HOME)/sw/airborne/ subsystems/datalink/xbee.c	\$ (PAPARAZZI_HOME)/sw/airborne/ \$(SRC_FIRMWARE)/guidance/guidance_v.c	\$ (PAPARAZZI_HOME)/sw/airborne/ subsystems/imu/imu_analog.c	\$ (PAPARAZZI_HOME)/sw/airborne/ subsystems/gps/gps_ubx.h
\$ (PAPARAZZI_HOME)/sw/airborne/\$ (SRC_ARCH)/radio_control/ppm_arch.c	\$ (PAPARAZZI_HOME)/sw/airborne/ firmware/\$ (SRC_FIRMWARE)/datalink.c	\$ (PAPARAZZI_HOME)/sw/airborne/ firmwares/fixedwing/guidance/guidance_v.h\		\$ (PAPARAZZI_HOME)/sw/airborne/ subsystems/gps/gps_sim.c --For only board =PC
\$ (PAPARAZZI_HOME)/sw/airborne/ subsystem/radio_control/ppm.h				

