Badgercopter Debug Log

Companion Computer Setup: BadgerUpCore

5.31.22

This material will be used for pin assignment and protocol changes:
https://github.com/up-board/up-community/wiki/Ubuntu_20.04#enable-the-hat-functionality-from-userspace

Not sure whether need ubuntu toolchain for setup:
https://docs.px4.io/v1.12/en/dev_setup/dev_env_linux_ubuntu.html

6.12.22
Requirement 3 not fully satisfied with PX4-utopilot directory, didn’t change
https://docs.px4.io/v1.12/en/ros/ros2_offboard_control.html

Aerial Module

1. Mechanical design

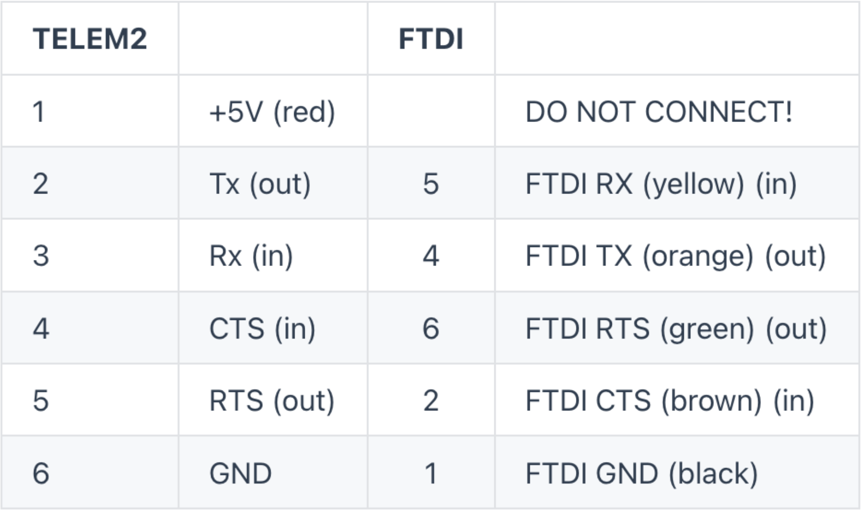
The aerial module of this project is designed based on the Holybro Pixhawk 4 Mini QAV250 quadcopter kit . Adding on to the structure is a 3D printed bracket to hold the companion computer, which is 4GB memory, 64 GB emmc Up Core single board computer . Along with it there is a wifi antenna and a usb dock. One of the USB port goes to an FTDI chip using a usb to serial mini usb cable. The pins of the FTDI chip is connected to px4 mini using jumper wire.

Bill of Material		
Item Name	Link	Price
Quadcopter Kit	https://banggood.onelink.me/zMT7/qcyiv0tm	$320.59
USB Dock	https://www.amazon.com/Hub,_VENTION-Ultra-Slim-Splitter-Supported-Compatible/dp/B08GY3GKRC/ref=sr_1_4?crid=1CSPQJDS54JB4&keywords=usb%2Bhub&qid=1652895544&s=electronics&sprefix=usb%2Bhub%2Celectronics%2C81&sr=1-4&th=1	$13.99
Up Core Computer	https://up-shop.org/up-core-series.html	$189.00
Wifi Antenna	https://up-shop.org/up-core-wifi-antenna-kit.html	$7.99
FTDI chip	https://www.amazon.com/HiLetgo-FT232RL-Converter-Adapter-Breakout/dp/B00IJXZQ7C/ref=sr_1_4?crid=3R	$ 5.99

	QICWVF43XZZ&keywords=ftdi+usb+to+serial&qid=1656528924&s=electronics&sprefix=ftdi+usb+to+serial%2Celectronics%2C66&sr=1-4	
Jumper Wires	/	/
3D printed PLA	/	/

2. Setup companion computer
    a. Install Linux and enable wifi connection
        i. Please check requirement of companion computer to install specific linux system. For Up Core, Ubuntu 20.04 can be installed here.
        ii. Following the tutorial, also enable wifi and bluetooth connection, and UART functionality.
    b. Install preliminaries
        i. Please install Java (JDK8), Gradle and Foonathan memory based on this link.
        ii. Not sure whether any package is installed, please use “-version” or “-v” parameter to check the version.
    c. Install fast DDS(RTPS)
        i. Follow the Fast DDS Installation Guide to install Fast RTPS(DDS) 2.0.2 (or later) and Fast-RTPS-Gen 1.0.4 (not later!) and their dependencies.
        ii. Please make sure which ROS2 version you are planning to install. Different versions will need to download different DDS/RTPS.
    d. Install ROS2
        i. It is recommended to Install ROS 2 Foxy
        ii. Follow the ROS2 User Guide to manually install some dependencies and tools
    e. Build ROS2 workspace
        i. Follow the ROS2 User Guide to build ROS2 workspace
3. Build Communication
    a. Flight Controller - Companion Computer
        i. Port Configuration
            1. For Pixhawk mini4, we used TELEM 1 port as the communication port with the companion computer
            2. We configured the TELEM 1 port in QGround control using the command: MAV_0_CONFIG (refer to the documentation). By configure TELEM 1 port using this command, Pixhawk mini4 can then communicate with the companion computer using Mavlink protocol via TELEM 1 port.
        ii. Hardware Wiring
            1. Since the companion computer is running ROS2, MAVROS is not needed.

2. If the companion computer and Pixhawk mini4 is connected wirelessly, then you should install MAVLink Router (recommended) or MAVProxy to route MAVLink between serial and UDP
3. We connect TELEM 1 and companion computer USB port via a FTDI chip, which is a serial-to-USB adapter board



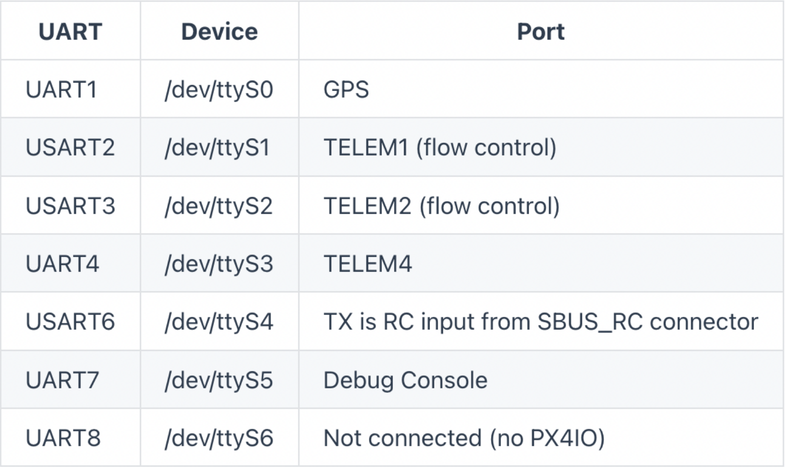
TELEM2		FTDI	
1	+5V (red)		DO NOT CONNECT!
2	Tx (out)	5	FTDI RX (yellow) (in)
3	Rx (in)	4	FTDI TX (orange) (out)
4	CTS (in)	6	FTDI RTS (green) (out)
5	RTS (out)	2	FTDI CTS (brown) (in)
6	GND	1	FTDI GND (black)

4. In addition to FTDI cable, use another cable to connect the usb port of the flight controller in order access into its interface.
5. Refer to this document for detail for setups. You can ignore the “Serial Port Software setup on Linux” part if you are using ROS2. We couldn’t figure out a way to see out pixhawk using lsusb command in out companion computer, but the communication was still established after setting up client and agent

iii. Setup Client and Agent

1. Refer to the RTPS/DDS Interface: PX4-Fast RTPS(DDS) Bridge
2. To build client in px4, you need to access to the PX4 console first, please refer to this document (USB wire from companion computer to the microUSB port on PX4) (for PX4 mini port info, refer to this link)
3. Use “make px4_fmu-v4_rtps upload” command in companion computer to build and upload RTPS to PX4. Different hardware is using fmu, only the correct version is able to build communication. If the upload process stop and ask for “Bootloader”, simply stop the process and take off and replug in all connection to let the flight controller reboot. It will start again at where you stop.
4. Enter PX4 console with the extra usb connection other than ftdi using “screen /dev/ttyXXX BAUDRATE 8N1” (use ls /dev/tty* and watch what changes when unplugging / replugging the USB device), ours is /dev/ttyACM0; BAUDRATE is 57600.

        5. In the PX4 console, enter “micrortps_client start -d /dev/ttyXXX -b BAUDRATE”, for TELEM1 port on PX4, it’s /dev/tty/S1 (for PX4 mini port info, refer to this link), and BAUDRATE is 921600. This is connected to the ftdi cable, which is connected to telem1.



UART	Device	Port
UART1	/dev/ttyS0	GPS
USART2	/dev/ttyS1	TELEM1 (flow control)
USART3	/dev/ttyS2	TELEM2 (flow control)
UART4	/dev/ttyS3	TELEM4
USART6	/dev/ttyS4	TX is RC input from SBUS_RC connector
UART7	/dev/ttyS5	Debug Console
UART8	/dev/ttyS6	Not connected (no PX4IO)

        6. Turn on the RTPS agent using “micrortps_agent start -t UART -d /dev/ttyXXX -b 921600”. Please check the port of FTDI using “ls /dev/tty*” to replace XXX. (try to unplug and then plug wire again if there is any problem)
        7. Right now, both RTPS agent and client is initiated, and you can test the connection using “micrortps_client status” in px4 console.
        8. If the console terminates each time after you run the RTPS client, please modify a parameter in PX4-Autopliote workspace and re-upload the RTPS client firmware.
    b. Companion Computer - Base Station
4. Quadcopter Control

Ground Module

1. Mechanical Design
2. Build Communication
    a. Hardware wiring
    b. Arduino Uno code
    c. Companion Computer - Arduino Uno
3. Ground Module Control
    a. Arduino Uno code
    b. Node