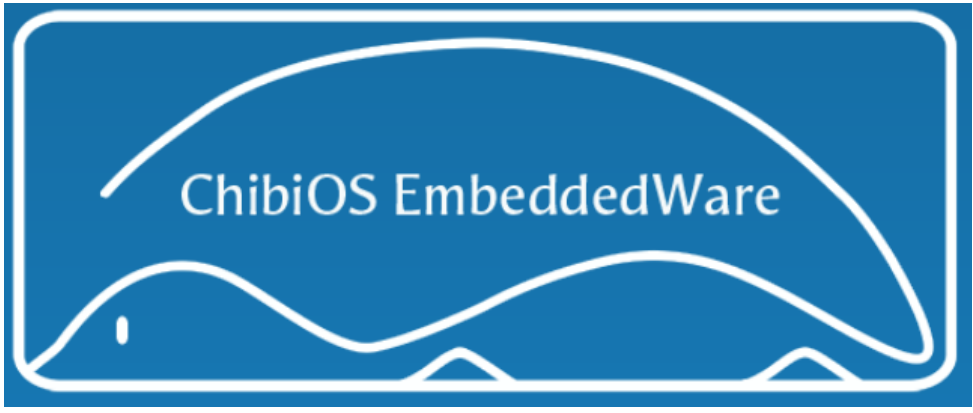
# ArduPilot on ChibiOS

Andrew Tridgell
ArduPilot Dev Team

# History

- Started on 8bit AVR
  - APM1/APM2
- Add HAL abstraction
  - allowed for multiple platforms
- Added new HALs
  - HAL_Linux, HAL_QURT, HAL_SITL, HAL_VRBrain, HAL_PX4
- Most used HAL is HAL_PX4
  - partnered with PX4 on NuttX
  - supported Pixhawk, Pixhawk2 Cube, Pixracer, P4Pro

# HAL_ChibiOS

- ChibiOS used in CAN peripherals
  - work by Pavel, Jon and others
- HAL_ChibiOS started in late 2017
  - Started by Siddharth Purohit
  - Aiming for smaller, faster HAL for STM32
  - Aim to support boards with 512k flash

# ChibiOS RTOS

- Started in 2007
  - Lead developer Giovanni Di Sirio
  - Current stable release 18.2.0 (Feb 2018)
  - GPLv3 license (with commercial licenses also available)
  - very widely used and respected RTOS
  - focus on efficiency and small size
  - very active community
  - supports wide range of MCUs
  - main site http://www.chibios.org
  - extensive documentation
  - good online book, highly recommended
    - ChibiOS/RT 3.0 “Ultimate Guide”

# ChibiOS Structure

- Minimal layer above hardware
  - ‘lld’ (low level driver) for each peripheral
  - MCU driver for each MCU
  - OSAL (Operating System Abstraction Layer) above low level drivers
  - Locking zones and task states can be strictly enforced
  - Not Posix compliant, but some posix-inspired features

# HAL_ChibiOS

- New HAL, same structure as other HALs
  - waf wrapper around ChibiOS makefile build
  - uses ‘hwdef.dat’ abstraction for board definition
  - adds shared DMA abstraction
  - uses DMA for all peripherals when possible

# Hardware Definitions

- Porting to new boards was hard
  - new ports on HAL_PX4 required special expertise
  - required changes in many layers
  - new ports were rarely done
- Solved with hwdef.dat
  - single file to define hardware setup of a board
  - generates ChibiOS config headers
  - auto-fills from STM32 database

# Excerpt from FMUv3 hwdef.dat

```
# MCU class and specific type
MCU STM32F4xx STM32F427xx

# crystal frequency
OSCILLATOR_HZ 24000000

# USB support
PA11 OTG_FS_DM OTG1
PA12 OTG_FS_DP OTG1

# USART2 telem1
PD3 USART2_CTS USART2
PD4 USART2_RTS USART2
PD5 USART2_TX USART2
PD6 USART2_RX USART2

# a SPI bus
PA5 SPI1_SCK SPI1
PA6 SPI1_MISO SPI1
PA7 SPI1_MOSI SPI1

# some PWM channels
PE14 TIM1_CH4 TIM1 PWM(1) GPIO(50)
PE13 TIM1_CH3 TIM1 PWM(2) GPIO(51)
PE11 TIM1_CH2 TIM1 PWM(3) GPIO(52)
PE9  TIM1_CH1 TIM1 PWM(4) GPIO(53)
PD13 TIM4_CH2 TIM4 PWM(5) GPIO(54)
PD14 TIM4_CH3 TIM4 PWM(6) GPIO(55)
```

# MCU Support

- MCU database
  - extracted from STM32 datasheet
  - support for STM32F427, STM32F405 and STM32F412 so far
- Scripts to create database files
  - uses tabula tool to parse STM32 datasheets
  - python scripts to parse csv from tabula to create databases

# Shared DMA

- STM32F4xx has DMA mapping limits
  - cannot always assign exclusive DMA for peripheral
  - using DMA greatly lowers CPU load
  - dma_resolver in build finds optimal DMA mapping
  - allows for sharing DMA channels if required
  - locking with shared DMA adds consierable complexity

# Bonus Features

- Support for custom USB IDs for boards
  - control strings, vendor IDs, product IDs
  - will be able to control via apj_tool in the future
  - Per-board USB IDs
- Derived boards
  - use ‘include’ in hwdef, then undef and define new pins
  - allows for cut down boards, with optimal peripheral usage
- Developer Friendly
  - faster builds!

# Performance on a Solo

ArduCoper master 3.6-Dev // Nuttx & PX4
PERF: 1316/4000 max=6659 min=2119 avg=2953 sd=597
PERF: 1280/4000 max=8157 min=1882 avg=2952 sd=616
PERF: 1275/4000 max=7370 min=2014 avg=2958 sd=637
PERF: 1313/4000 max=7256 min=1905 avg=2960 sd=607
PERF: 1328/4000 max=6748 min=1942 avg=2958 sd=600
PERF: 1313/4001 max=6854 min=1934 avg=2953 sd=584

ArduCopter 3.6-Dev // ChibiOS
PERF: 1/4000 max=3004 min=2200 avg=2501 sd=86
PERF: 2/4000 max=3308 min=2125 avg=2501 sd=96
PERF: 1/4000 max=3102 min=2197 avg=2501 sd=90
PERF: 2/4000 max=3102 min=2098 avg=2501 sd=88
PERF: 5/4000 max=3795 min=2104 avg=2505 sd=99
PERF: 0/4000 max=2998 min=2104 avg=2501 sd=94
PERF: 0/4000 max=2888 min=2100 avg=2500 sd=82

# Faster Main Loops

- Better timing
  - able to support faster main loop
  - demonstrated flights up to 1kHz
  - plans in place for up to 4kHz rate loop

# Current Status

- Support for lots of boards
  - fmuv3, fmuv4, crazyflie2, skyviper-v2450, skyviper-f412, revo-mini, mini-pix
  - adding new boards very fast (usually a few hours)
- Missing features
  - ports for fmuv1, vrbrain and p4pro
  - port of IOMCU firmware to ChibiOS
  - serial5 support on fmuv3
  - pwm based rangefinders
  - mavlink serial device for GPS and radio update