Racer Firmware

Compiler

- gcc-arm-none-eabi-10-2020-q4-major-x86_64-linux

Supported boards

- Pixracer
- OpenPilot Revo (planned)

The firmware should be easily portable to similar STM32F4-based boards.

Architecture

There are mainly 3 primary layers built on top of the hardware:

(STM32F4 board) <-> mcal <-> racer hal <-> fw logic

- mcal: microcontroller abstraction layer (src/target/mcal)
- racer hal: racer hardware abstraction layer (src/racer)
- fw logic: firmware control logic (board-indepdent application)

Tasks

ID	Task Name	Priority	Period	HW Data In	Q Data In	Q Data Out	HW Data Out
1	Sensor	Low	20ms@50Hz	SPI/Hall		SensorData	
2	Receiver	Mid	20ms@50Hz	UART@100kbps		ReceiverState	
3	Datalink	Mid	20ms@50Hz	UART@100kbps		UserCommand	
4	System	High	20ms@50Hz	CAN@500kbps	ReceiverState,UserCommand	ControlTarget	
5	Control	High	20ms@50Hz		ControlTarget	ActuatorCommand	
6	Actuator	Highest	triggered		ActuatorCommand		PWM@50Hz (HW)

Data flow

```
                    Receiver -->
                               \
                                --> System --> Control --> Actuator
                               /
```

IPC/Linux -- (CAN Bus) -- Datalink -->

<-- Sensor