ОПИСАНИЕ СКЕТЧЕЙ

Для работы с RaCEBoard представлены следующие скетчи:

*BMP280\_work.py* – пример работы барометра BMP280, позволяет получать данные:

* температура окружающей среды, *оС*;
* атмосферное давление, *кПа*;
* высота над уровнем моря в *м*.

*Button\_LED.py* – пример работы светодиода, который загорается от нажатия кнопки.

*Buzzer\_work.py* – пример работы пищалки, представлены функции:

* *BuzzerOn*() – включает пищалку с частотой *frequency\_buzzer*;
* *BuzzerOff*() - выключает пищалку.

*DCMotor\_work.py* – пример работы двигателя постоянного тока, представлены функции:

* *DCMotorSetSpeed(speed)* – функция задания скорости в процентах от -100 % до +100%.

*GNSS\_work.py* – пример работы *GNSS*-модуля, позволяет получить данные:

* широта, *градусы и минуты*;
* долгота, *градусы и минуты*
* высота, *м*;
* скорость передвижения, *узлы*;
* азимут, *градусы;*
* расхождение по горизонтали.

Обязательное условие: необходимо как можно чаще вызывать *gnss.update()* в цикле *loop*, т.е. с частотой вызовов не менее 5-10 кГц.

*LED\_blink.py* – пример мигания светодиодом.

*LSM6DSL\_work.py* – пример работы акселерометра-гироскопа *LSM6DSL,* позволяет получить данные:

* линейные ускорения по трём осям (*X, Y, Z*), *м/с2*;
* угловые скорости по трём осям (*X, Y, Z*), *град/с2*;
* температура датчика, *оС*.

*MUX\_work.py* – пример работы аналогового переключателя 1 к 8 (1 двунаправленный вход с одной стороны, и 8 с другой стороны).

*Линия 0* позволяет измерить напряжение аккумулятора;

*Линии 1, 2* имеют подтяжку на *GND* 10 *кОм*;

*Линии 3-7* имеют состояние *float*.

Представлены следующие функции:

* *muxSelectLine(line)* – позволяет присоединить пин микроконтроллера к выбранной линии (0-7);
* *mux2In()* – позволяет установить, что пин является цифровым входом;
* *mux2Analog()* – позволяет установить, что пин является аналоговым входом;
* *mux2Out()* – позволяет установить, что пин является цифровым выходом.

*PCF8563\_work.py* – пример работы часов реального времени, представлены следующие функции:

* *PCF8563.setDate(YYYY,MM,DD)* – задать начальную дату;
* *PCF8563.setTime(HH,MM,SS)* – задать начальное время;
* *PCF8563.getDate(YYYY,MM,DD)* – получить текущую дату;
* *PCF8563.getTime(HH,MM,SS)* – получить текущую время.

*QMC5883L\_work.py* – пример работы цифрового магнетометра QMC5883L, представлены следующие функции:

*qmc5883l.magnetometer()* – получить значение индукции магнитного поля в *мТл*.

*Ra01S\_TX\_work.py* – пример работы радиопередачи с помощью радиомодуля *Ra01S*, представлены следующие функции:

*Ra01S.on()* – включение радио;

*Ra01S.SetLowPower()* – задание минимальной мощности радиопередатчика для отладки внутри помещения;

*Ra01S.SetMaxPower()* – задание максимальной мощности радиопередатчика для полёта;

*Ra01S.SetChannel(channel)* – задание канала (0-6);

*Ra01S.SendS(string)* – отправка строки по радиоканалу.

Обязательное условие: Чтобы использовать радио, необходимо инициализировать *CS*-пин карты памяти – даже если не планируется её использование.

*RA01S\_RX\_work.py* – скетч для проверки приёма радио.

Обязательное условие: перед проверкой радиопередачи, в данном скетче необходимо назначить свой канал.

*SD\_files.py* – пример работы *SD*-карты в режиме чтения и записи файлов. После успешного монтирования карты памяти, работа с файлами идёт, как на стандартном *Python*.

*SD\_folders.py* - пример работы *SD*-карты в режиме чтения и записи файлов. Показывает, как создавать и удалять папки с файлами. Поддерживается только один уровень вложенность – нельзя удалить папку в папке.

*Servo\_work.py* – пример работы сервопривода, представлены следующие функции:

* *ServoSetAngle(servo, angle)* – функция задания угла поворота сервы (от 0 до 180 градусов).