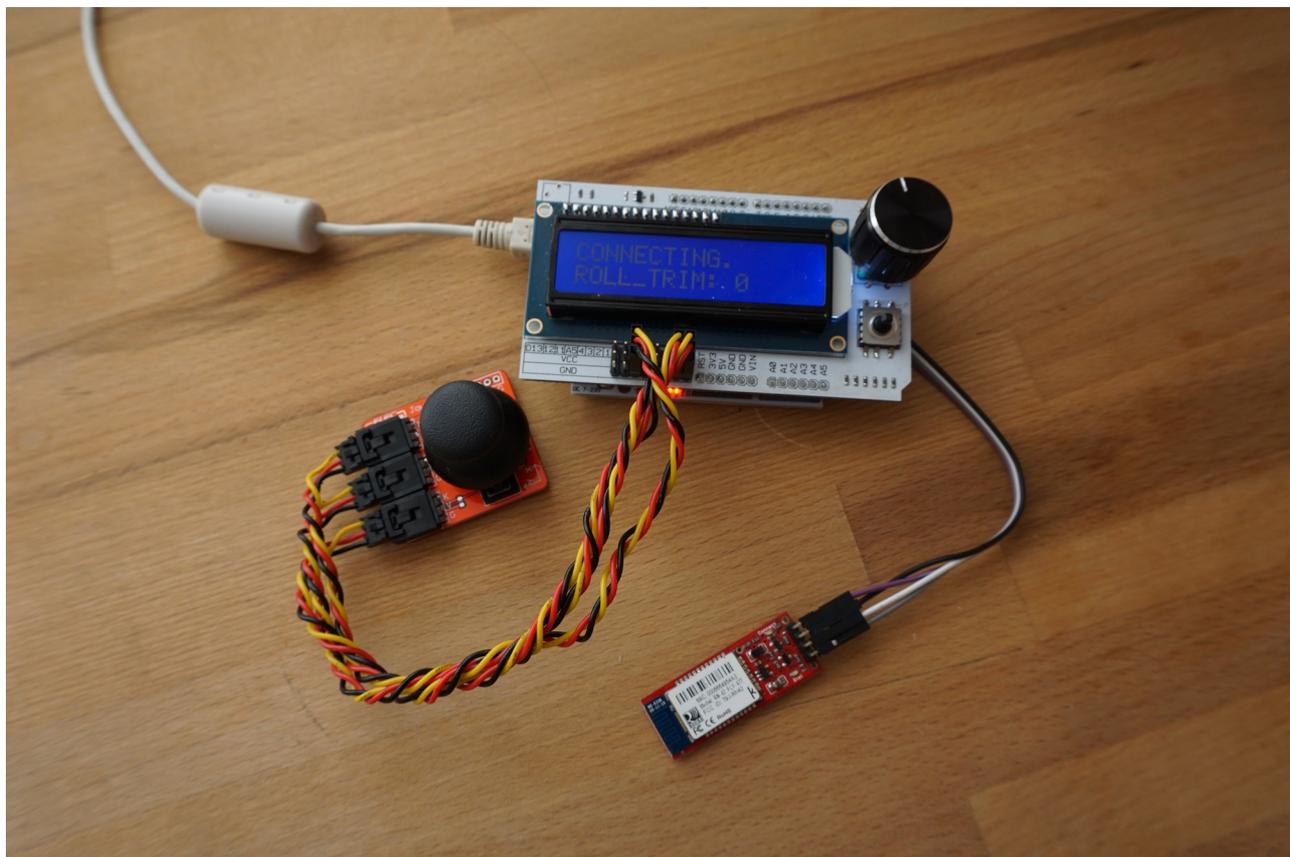


## Пример построения простого пульта управления подвесом с контроллером SimpleBGC на основе Arduino.



*Пульт управления в сборе*

В этой инструкции мы покажем, как можно сделать простой пульт управления подвесом с LCD-дисплеем. Пульт подключается к любому контроллеру SimpleBGC 32 бит посредством serial-интерфейса: кабель с 4-мя жилами, или беспроводной адаптер UART (Bluetooth, 3DR modem, Zigbee, WiFi-UART адаптеры и т. д.)

Базовые функции, реализованные в этом примере:

1. Отображение состояния системы на LCD-дисплее на нескольких страницах (листаются кнопками навигации Вверх/Вниз)
  - напряжение батареи
  - P: активный профиль
  - E: средняя ошибка стабилизации (0.001 градуса)
  - SE: ошибки коммуникации (Serial Error)
  - I2C\_ERR: Ошибки I2C-шины
  - Любая другая информация, например свободная память в Arduino, отладочные переменные, и т.д.
2. Управление подвесом при помощи джойстика по осям ROLL, PITCH
3. Управление наклоном по оси ROLL с шагом 0.1 градус (ROLL\_TRIM)
4. Кнопка джойстика дублирует кнопку меню на плате (на нее можно назначить переключение профилей и прочие команды в меню Service)
5. Изменение выбранного набора ключевых параметров системы при помощи энкодера. Параметры листаются кнопками навигации Вправо/Влево или кнопкой энкодера.

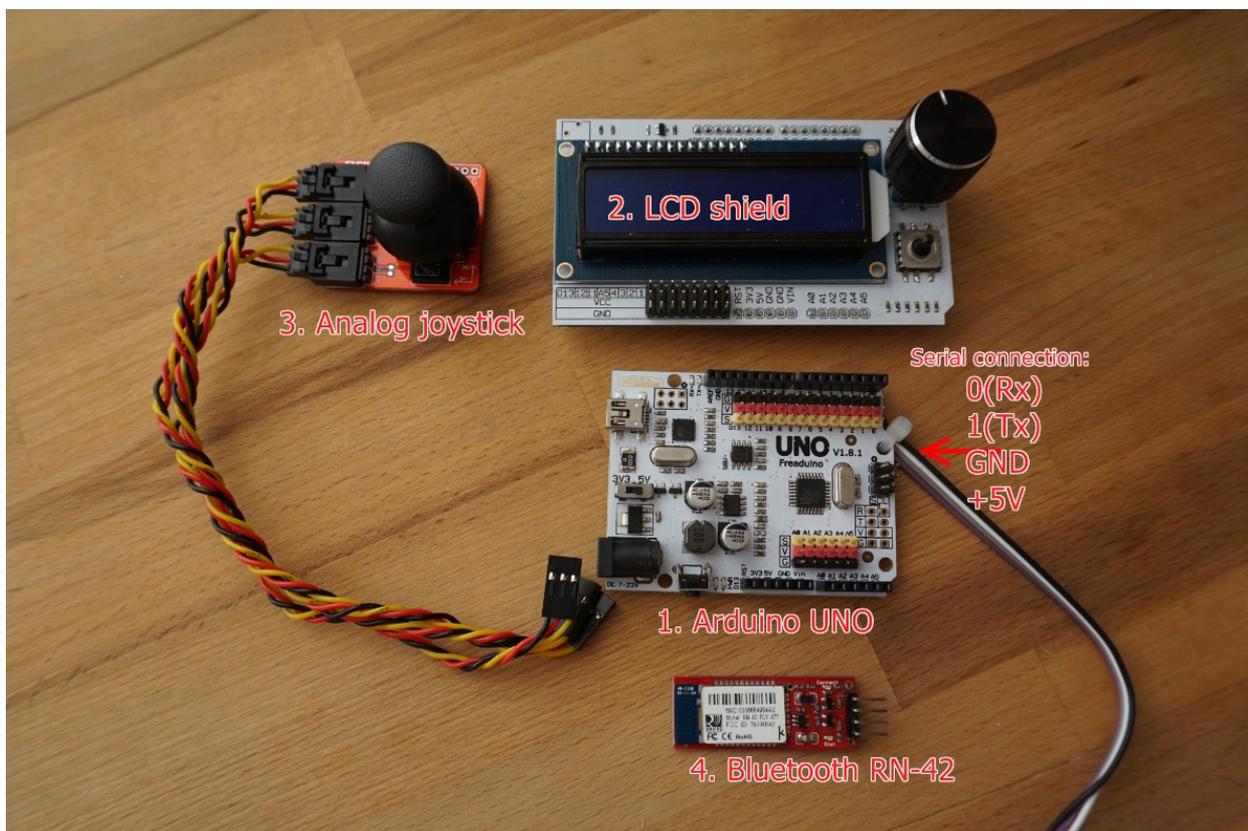
6. Нажатие кнопки Select включает/отключает моторы.

Демо-видео работы пульта: <https://www.youtube.com/watch?v=a4uw7QhJKNM>

## Набор оборудования

Пример написан для платформы Arduino и протестирован на Arduino UNO. Из железа вам понадобится:

1. Arduino UNO
2. LCD Key shield от elfreaks или аналогичный набор компонентов:  
[http://www.electfreaks.com/wiki/index.php?title=LCD\\_Key\\_Shield](http://www.electfreaks.com/wiki/index.php?title=LCD_Key_Shield)
  - 1602 LCD display
  - Инкрементальный энкодер управления с А,В выходами и кнопкой
  - Кнопочный джойстик для навигации (5 степеней свободы: Left, Right, Up, Down, Select), заведенный через резистивный делитель на один ADC-вход.
3. 2-х осевой аналоговый джойстик с кнопкой
4. Bluetooth-адаптер в ролии Master (опционально, если используется прямое подключение, отключите его в секции конфигурации в скрипте). В данном примере написан код для модуля RN-42. Также в режиме «мастер» может работать HC-05. На стороне подвеса можно использовать любой блютуз-модуль совместимый со стандартом 2.1.
5. Подвес с контроллером SimpleBGC 32, в котором доступен UART-разъем. Возможна работа и со старой 8-битной версией контроллера, но нужно будет переписать часть команд и не все функции будут доступны.



Набор компонентов

## Сборка и подключение

Все подключение описано в начале скрипта LCDRemote.ino в секции «Hardware

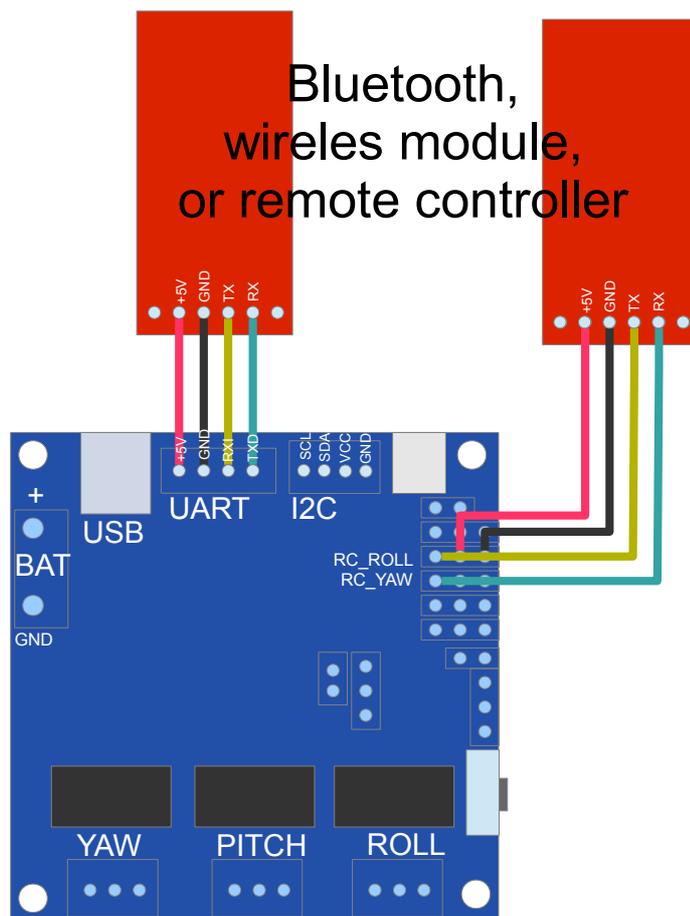
Configuration». Если используется LCD Key shield, как на картинке - просто наденьте его на Arduino. Подключите джойстик к разъемам на шилде: A1 (ось X), A2 (ось Y), D11 (кнопка). При установке шилда становятся недоступными выходы Serial RX, TX, так что нужно припаять их с задней стороны платы. Также не забудьте GND и +5V.

*ЗАМЕЧАНИЕ: Так как этот серийный порт используется и для заливки прошивки, перед заливкой нужно отключать Rx, Tx от блютуз-модуля или контроллера SimpleBGC 32.*

Для питания можно использовать USB кабель, брать питание с UART-разъема контроллера (если используется проводное подключение по 4-м жилам), или использовать внешний источник питания.

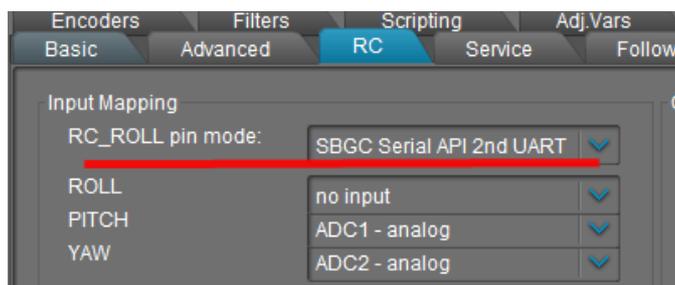
*ЗАМЕЧАНИЕ: в моем экземпляре Arduino UNO при питании от внешнего источника DC-преобразователь создавал существенные наводки на входы ADC, что делало работу джойстика практически невозможной. Нужно предпринимать дополнительные шаги по фильтрации ADC-входов, к примеру установкой конденсаторов 1nF. При питании от USB или напрямую от 5V такой проблемы нет.*

## Подключение и настройка контроллера SimpleBGC32



*Варианты подключение к контроллеру SimpleBGC 32*

Для подключения можно использовать основной UART-разъем (в этом случае одновременное подключение пульта управления и GUI невозможно), или дополнительный UART на входах RC\_ROLL (Rx), RC\_YAW (Tx). Для его включения выберите «RC\_ROLL Pin mode» = «SBGC Serial API 2<sup>nd</sup> UART»

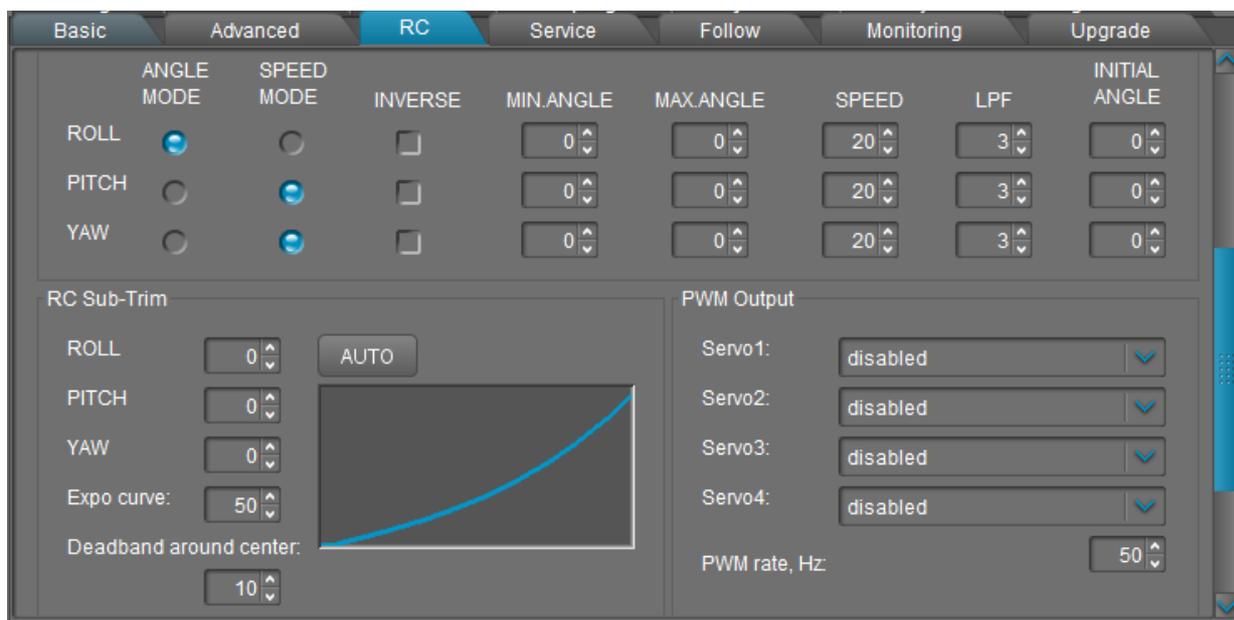


Во втором случае возможна параллельная работа GUI и пульта управления, что может быть удобно для его отладки.

В контроллере должна быть залита прошивка 2.55b7 или более поздняя.

Контроллер настраивается как обычно, настройки управления на вкладке RC можно не менять (при подключении пульта любой другой источник сигнала автоматически отключится). Если вы ещё не настраивали управление, можете использовать такие параметры:

PITCH, YAW: “SPEED MODE”, ROLL: “ANGLE MODE”  
 MIN.ANGLE=0, MAX.ANGLE=0  
 SPEED=20  
 LPF=3  
 INITIAL ANGLE=0  
 DEADBAND=10  
 Expo curve=50



Если вы планируете переключать профили, не забудьте сделать соответствующие настройки в каждом профиле.

Вы можете назначить различные действия нажатиям кнопки джойстика, в меню «Service» (от

1 до 5 нажатий подряд).

## **Компиляция и заливка прошивки**

Актуальная версия примеров и библиотек расположена по адресу:

<https://github.com/alexmos/sbgc-api-examples>

Следуйте инструкциям в файле Readme.md.

***ЗАМЕЧАНИЕ:** для данного примера (и любого другого, где предполагается получать данные с контроллера), необходимо увеличить размер буфера приема Serial в Arduino IDE. Для этого найдите файл `path-to-arduino-ide\hardware\arduino\cores\arduino\HardwareSerial.cpp` и замените значение константы `BUFFER_SIZE` с 64 на 150:*

```
#define SERIAL_BUFFER_SIZE 150
```

*(расположение файла и имя константы может меняться в зависимости от версии IDE)*

***ЗАМЕЧАНИЕ:** Arduino крайне ограничен в ресурсах. Обращайте внимание на расход оперативной (RAM) памяти, так как в данном примере (для сохранения читабельности кода и наглядности) нет специальных оптимизаций для сокращения ее потребления. Если вы хотите развивать этот код дальше, используйте стандартные техники по выносу констант в `PROGMEM` и уменьшения числа статических переменных.*