

Привет друзья, сегодня мы с вами научимся создавать собственные функции и использовать их в своих проектах.

Функция это набор команд, которые можно вызвать обратившись к этой функции, можно сказать это маленькая библиотека.

Функции объявляются вне других функций, я обычно делаю это под функцией луп, чтобы глаза не мозолили.

Функции можно разделить на два типа: функции которые что то возвращают, и те которые ничего не возвращают. Если функция ничего не возвращает, она объявляется словом void, что в переводе с английского значит бездна, пустота. Уже знакомые вам функции seta и lup объявлены именно так. Функции, возвращающие значения, объявляются типом данных, соответствующим возвращаемому значению. То есть байт, инт, Лонг и так далее.

На втором месте идёт название функции, на английском языке без пробелов. Далее в скобках указываются входные переменные, если их нет то не указываются. Теперь в фигурных скобках указываем непосредственно набор команд нашей функции. Создадим функцию, которая будет просто выводить чего нибудь в монитор порта. Void май фанкшн, пустые скобки, фигурные скобки, и здесь пишу вывод в порт. Теперь обратимся к нашей функции в seta, чтобы один раз вывести и все. Название, пустые скобки и точка запятой. Чтобы обратиться к функции, пишем ее название, скобки, точка запятой, все.

Усложним код. Теперь на вход функции будем подавать значение, чтобы она работала с ним, а именно, выводила его в порт. Для этого в объявлении функции объявляем переменную. Эта переменная будет действовать только в пределах этой функции. Пусть инт. И эту переменную используем для вывода в порт. Теперь при обращении к функции указываем в скобках переменную, которую хотим вывести. Эта переменная при вызове функции представится вот сюда, и будет использована дальше, в нашем случае выведено в порт. Вот так вот, надеюсь это было понятно. Давайте ещё немного усложним функцию, заставим ее считать. В качестве входных данных используем два потенциометра, подключенных к нулевому и первому аналоговому пину. Объявляем две переменные. присваиваем им значения с потенциометров. Теперь подаём их на вход нашей функции через запятую. Значит нашу функцию нужно научить работать с двумя переменными. Окей, не вопрос, добавляем ещё одну переменную. Давайте выведем в порт произведение этих двух переменных. Причем сначала я выведу сами переменные, а затем произведение. Вот такая получилась машина для перемножения чисел.

Алекс Алекс, расскажи нам про функции , которые возвращают значение! Хорошо, слушайте. Пусть функция будет возвращать результат перемножения чисел. Навскидку число будет большое, ставим тип данных лонг. Теперь внимание: команда ретерн возвращает значение функции. Что это значит. Представьте себе: в том месте, где мы обратились к функции, вместо названия функции, то есть вот этого текста, появится число, которое функция возвращает. То есть функция теперь не просто что то делает, она вместо себя подставляет значение, которое можно складывать вычитать умножать или просто вывести в порт. Возвращаю результат перемножения двух переменных и подставляю это в вывод в порт. И вот оно работает. На этом простом примере мы с вами рассмотрели все типы функций и варианты их использования. Остался главный вопрос: зачем? Все очень просто. Функции нужно использовать в том случае, если у вас во первых есть повторяющийся кусок кода, который удобнее один раз записать в функцию и к ней обращаться. Так будет проще и вам, и ардуине. Во вторых, если какой то кусок кода просто мозолит вам глаза и вы хотите чтобы он не мешал работать, его можно просто вынести в отдельную функцию, главное не забывать про область действия переменных, это один из первых уроков. Как пример - скетч с режимами для адресной светодиодной ленты. Вот почти полсотни режимов, вызываются при помощи своих функций. Почему? Да просто потому что это удобно. Гораздо удобнее чем работать с таким объемом текста.

Взять тот же ультразвуковой дальномер, для получения расстояния используется несколько команд, которые можно взять и заменить функцией, которая будет возвращать расстояние. Всё. Никаких лишних глобальных переменных, все четко и быстро.

Таким образом функции сильно упрощают жизнь и облегчают работу с ардуино. Напоминаю что скачать конспект с правой части всех уроков можно по ссылке в описании под видео. На этом сегодня все, Спасибо за внимание и до новых встреч!