

Всем привет, в этом выпуске уроков ардуино я расскажу вам о временных функциях и особенностях их использования.

Начнем с задержек, они очень просты в понимании. Функция дилей приостанавливает работу программы на указанное число миллисекунд. Я напому, что в одной секунде 1000 миллисекунд. Максимальное значение, которое можно подать на функцию дилей, это 2 байта, то есть чуть больше 32000 миллисекунд, то есть 32 секунды. Давайте выведем в порт несколько строчек с интервалом в 2 секунды. Как работать с портом я рассказывал во 2 уроке. Вот так вот, строчки выводятся с заданным интервалом.

Вторая похожая функция дилей микросекондс. Она делает все тоже самое, только в микросекундах. Максимальное значение может быть 16383, что есть всего 16 миллисекунд. Обе функции дилей удобны и просты в использовании, например классическое мигание светодиодом, или вывод строчек в порт. Но взрослые ардуинцтки очень не любят функцию дилей, так как она полностью тормозит микроконтроллер. Программа полностью застывает на месте. Для мигалок этот вариант подходит, но для больших серьезных проектов нет, так как лишние перебои в работе нам не нужны. Как решить эту проблему вы узнаете чуть позже. А сейчас рассмотрим еще одну пару функций, миллис и микрос.

Миллис возвращает количество миллисекунд, прошедших с момента запуска микроконтроллера. Микрос делает то же самое, но в микросекундах. Счетчик миллис переполняется примерно через 50 суток, счетчик микрос - через 70 минут. При переполнении счетчик просто сбрасывается в 0. Кстати разрешение функции микрос составляет 4 микросекунды. Как это использовать, спросите вы? Довольно просто. Сейчас будем решать проблему функции дилей.

Объявим большую переменную, она будет хранить предыдущее обращение к функции миллис, то есть число может достигать 4 миллиардов. По умолчанию эта переменная равна 0. Создадим условие: если предыдущее время меньше чем текущее на 5000 миллисекунд, то. Я описал это как разность, надеюсь вам понятно. Наша переменная неизменна, а вот миллис всё время растёт. Значит растёт и разность между ними. И при достижении порогового значения условие выполняется. Вот теперь точно понятно. Итак, прошло 5000 миллисекунд, то есть 5 секунд, условие выполнилось. Первым делом перезаписываем в нашу переменную текущее значение времени миллис.

Вот этот блок кода будет повторяться каждые 5 секунд. Получается, эта конструкция аналогична команде дилей(5000), но в отличие от нее не тормозит остальной код, то есть наш блок будет выполняться каждые 5 секунд независимо от остального кода прошивки. Эта конструкция очень и очень полезна, обязательно в ней разберитесь. Что касается времени, то задавать можно произведением. Например хочу исполнять функцию каждые 23 дня. Переводим в миллисекунды: $23 * 24 * 60 * 60 * 1000$. Всё! Теперь помещенный сюда набор команд будет выполняться раз в 23 дня, а остальной код программы будет спокойно себе выполняться. Таким образом можно организовать полив растения, кормёжку животных и многое многое другое, грубо говоря это просто циклический таймер.

Есть более короткий вариант, который я придумал пару дней назад. Берём миллис, делим его на тысячу, получаем секунды. Округляем чтобы получать целые секунды. И затем проверяем кратность например числу 2. Если кратно – делаем одно действие, если не кратно – то можно ничего не делать или делать другое действие. Получается очень удобный циклический таймер, который не требует переменных и работает чисто на функции миллис и математике, и

умещается в одну строчку кода. Изменяя число 100, можно регулировать задержку. Увеличение приведёт к увеличению времени одного цикла, а уменьшение к уменьшению.

Есть еще одна крутая штука, использование встроенного таймера ардуино, уход самой ардуины в спячку с низким энергопотреблением и просыпание по вотчдогу, то бишь по таймеру. Но это тема для отдельного выпуска. А на сегодня все. С вами был канал заметки ардуинщика, всем пока.