**Сравнение Arduino Uno R3 и Arduino Uno R4**

|  |  |  |
| --- | --- | --- |
| Параметр | Arduino Uno R3 | Arduino Uno R4 |
| Тактовая чистота | 16 МГц | 48 МГц |
| Разрешение (micros()) | 4 мкс | 1 мкс |
| Скорость (digitalWrite) | ~3,5 мкс | ~ 0,1 мкс |
| Время выполнения (loop()) | ~50-100 мкс | ~ 10-20 мкс |

**Анализ работы светодиодов на Arduino Uno R3 и R4**

**LED1 (10 мс):** На R3 функция millis() с разрешением 1 мс обеспечивает стабильные 50 Гц, тогда как на R4 более точное millis() сохраняет ту же частоту ~50 Гц без визуальных отличий.

**LED2 (1 мс):**  
 R3 из-за ограничения millis() в 1 мс даёт нестабильные 2-3 мс (~250-333 Гц), в то время как R4 с улучшенным millis() достигает 1-2 мс (~500-1000 Гц).

**LED3 (500 мкс):**  
 На R3 медленный цикл loop() вызывает пропуски (~500-800 Гц), тогда как R4 с быстрым процессором стабильно держит 500 мкс (1 кГц).

**LED4 (100 мкс):**  
 R3 показывает дрожание 5-10 кГц из-за задержек обработки, в то время как R4 демонстрирует стабильные 10 кГц с минимальными отклонениями.

**LED5 (50 мкс):** На R3 светодиод практически не мигает из-за превышения времени обработки, тогда как R4 успевает отрабатывать интервал, выдавая 10-20 кГц.

**Ключевые различия платформ:**

* R3 стабильно работает только с интервалами ≥500 мкс
* R4 корректно обрабатывает даже 50 мкс интервалы
* Для R3 критически важна оптимизация кода при работе с микросекундными интервалами

**Вывод**

Для точного временного контроля на R3 следует использовать:

* Прерывания таймеров
* Прямую работу с регистрами
* Аппаратный PWM для светодиодов

Данный анализ показывает, как аппаратные различия между версиями Arduino влияют на выполнение временно-критичных задач.