

Министерство науки и высшего образования Российской Федерации
ФГБОУ ВО АЛТАЙСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Институт цифровых технологий, электроники и физики
Кафедра вычислительной техники и электроники (ВТиЭ)

ДРЕБЕЗГ КНОПОК В УСТРОЙСТВАХ ПЕРСОНАЛЬНОГО ТИПА

Выполнил студент 595 группы:
_____ А. В. Лаптев

Проверил: ст. преп.
_____ И. А. Шмаков
«___» _____ 2024 г.

Барнаул 2024 г.

РЕФЕРАТ

Полный объём работы составляет 18 страниц, включая 6 рисунков и 0 таблиц.

Данная работа посвящена исследованию проблемы дребезга и методов ее решения.

В результате выполнения научно-исследовательской работы был проведен анализ различных методов погашения дребезга и выбран и реализован один из них в качестве решения для конкретного устройства – генератора персонажа DnD.

Ключевые слова: Дребезг, прерывание, кнопка.

Отчёт оформлен с помощью системы компьютерной вёрстки $\text{T}_\text{E}\text{X}$ и его расширения $\text{X}_\text{E}\text{L}_\text{A}\text{T}_\text{E}_\text{X}$ из дистрибутива *TeX Live*.

СОДЕРЖАНИЕ

1	Что такое дребезг контактов и причины возникновения дребезга . .	5
2	Устранение эффекта дребезга контактов	7
2.1	Программные решения	7
2.1.1	Использование задержек	7
2.1.2	Использование циклического опроса и счетчиков	7
2.1.3	Использование прерываний по состоянию вывода порта	8
2.2	Аппаратные решения	9
2.2.1	Сглаживающие фильтры	9
2.2.2	RS-триггер	10
2.2.3	Использование одновибратора (ждущего мультивибратора) . . .	11
3	Борьба с дребезгом в реальном проекте	13
	Список использованной литературы	14
	Приложение	15

ВВЕДЕНИЕ

Различные кнопки и переключатели встречаются в цифровой аппаратуре повсеместно, поэтому проблема дребезга контактов в них зачастую является существенной. Дребезг контактов может стать причиной неправильной работы устройства в следствие неверного распознавания переключения или нажатия/отжатия кнопки, а это в ряде случаев может стать критичным даже для человека.

Например, при использовании в медицинской или промышленной аппаратуре неправильное распознавание нажатий может стать причиной несчастного случая или ухудшения состояния здоровья человека.

1. ЧТО ТАКОЕ ДРЕБЕЗГ КОНТАКТОВ И ПРИЧИНЫ ВОЗНИКНОВЕНИЯ ДРЕБЕЗГА

Дребезг контактов — явление, происходящее в электромеханических коммутационных устройствах и аппаратах, длящееся некоторое время после замыкания электрических контактов [1; 2].

Дребезг наблюдается и при размыкании электромеханических контактов [3].

Пример схематичного изображения дребезга контактов изображен на Рис. 1.1.

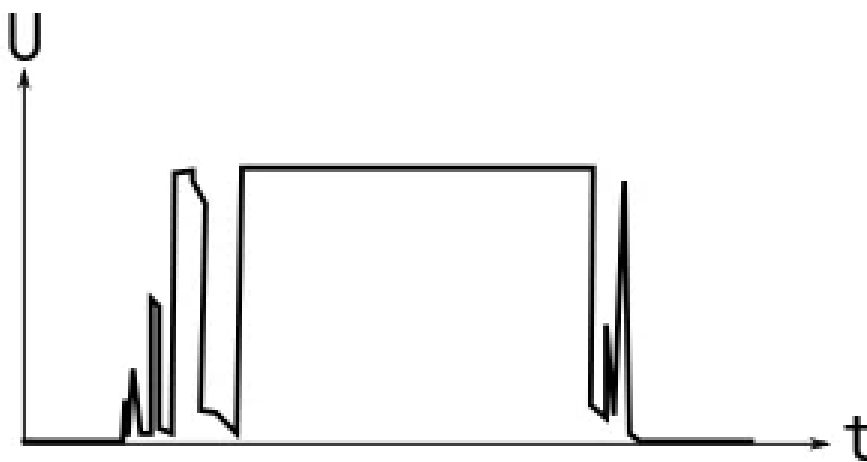


Рис. 1.1 Схематичное изображение дребезга на временной диаграмме

Как правило, считается, что контакт в переключателе является надежным и срабатывает мгновенно [4].

Однако на практике все несколько иначе. В каждом положении переключателя контакт между токопроводящими участками устанавливается или прерывается с помощью подвижных механических элементов.

Дребезг контактов возникает при нажатии на кнопку и переключатель из-за реальных вибраций контактной пластины при её перемещении. Как правило, пружинные компоненты применяются в качестве средства для перевода контакта из одного состояния в другое в виде либо металлической пластины, либо винтовой пружины. В тот момент, когда эти небольшие компоненты приводятся в движение, они перемещаются в требуемое положение. После срабатывания некоторое время происходят многократные неконтролируемые замыкания и размыкания контактов за счет упругости пружины и деталей контактной системы; при этом электрическая цепь размыкается и замыкается, пока движение полностью не прекратится [5].

В механических переключателях применяются механические же способы уменьшения дребезга контактов. Но полностью решить проблему они не могут. Наиболее часто применяют так называемый «механизм мгновенного действия». Пример того, как такой механизм может выглядеть представлен на Рис. 1.2 [6].

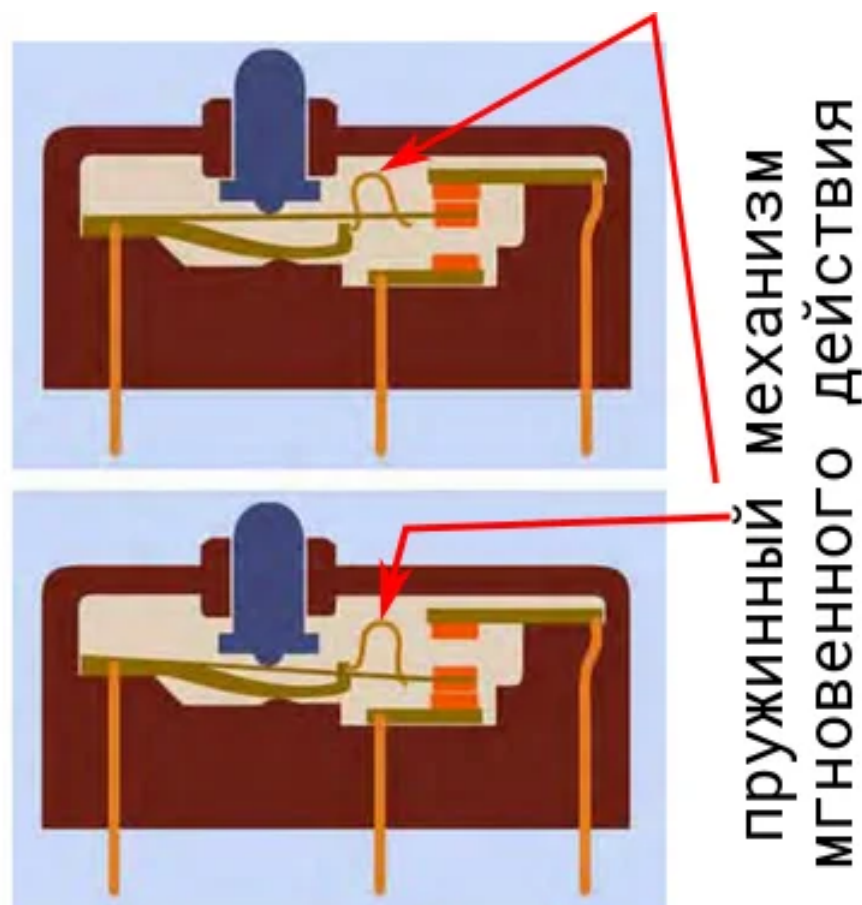


Рис. 1.2 Механизм мгновенного действия

При этом, в процессе эксплуатации кнопки «стареют». Контакты изнашиваются и загрязняются, а дребезг и переходное сопротивление возрастают.

Время и характер дребезга контактов очень сильно зависят от конкретного механического переключателя (размеры, конструкция, назначение) и степени его износа (время эксплуатации). Обычно, время дребезга составляет единицы, реже, десятки миллисекунд. Но может достигать и сотен миллисекунд для больших переключателей.

2. УСТРАНЕНИЕ ЭФФЕКТА ДРЕБЕЗГА КОНТАКТОВ

Для подавления дребезга контактов в электронных схемах существует два пути:

1. Решение программным путем
 - 1.1. Использование задержек
 - 1.2. Использование счетчиков
 - 1.3. Использование прерываний по состоянию вывода порта
2. Решение аппаратным путем
 - 2.1. Сглаживающие фильтры
 - 2.2. Триггер Шмидта
 - 2.3. Использование одновибратора (ждущего мультивибратора)

2.1. Программные решения

2.1.1. Использование задержек

Самым простым способом справиться с проблемой дребезга кнопки является выдерживание паузы. Микроконтроллер останавливается и ждет, пока переходный процесс не завершится. Для этого можно использовать любую встроенную функцию задержки, интегрированную в язык программирования. 10-50 миллисекунд обычно для большинства случаев.

Проблема с дребезгом настолько популярна, что есть специальные библиотеки, в которых не надо организовывать ожидание и паузы вручную — это все делается внутри специального класса. Пример популярной библиотеки для борьбы с дребезгом кнопок — библиотека Bounce.

В приложении приведен пример программы, которая реализует данный метод подавления дребезга.

2.1.2. Использование циклического опроса и счетчиков

Данный способ представляет собой циклическое считывание состояния кнопки с установленным интервалом считывания (обычно примерно 0,5-5 мс). Эта операция может выполняться как внутри основного цикла программы, так и в процедуре обработки прерываний таймера, которые будут происходить с той же частотой.

Основу этого метода составляет счетчик связанный с циклом опроса кнопки. В начальном состоянии он равен 0. Если в очередном цикле опроса

кнопки на выводе порта высокий уровень, а значение счетчика еще не достигло порогового значения, счетчик инкрементируется. Если в очередном цикле опроса кнопки на выводе порта низкий уровень, а счетчик не равен 0, то он декрементируется.

Достижение счетчиком порогового значения свидетельствует о том, что за последние циклы опроса состояние кнопки не изменялось и она нажата. Достижение счетчиком нулевого значения свидетельствует о том, что за последние циклы опроса состояние кнопки не изменялось и она отпущена.

2.1.3. Использование прерываний по состоянию вывода порта

Это более сложный способ обработки кнопки. К тому же, не в каждом микроконтроллере есть прерывания вообще, и по изменению состояния порта, в частности. Зато этот метод позволяет сразу фиксировать моменты нажатия и отпускания. Кроме того, программе не нужно тратить время на сканирование кнопки.

Собственно говоря, это одна из реализаций способа с использованием задержек.

Микроконтроллер запоминает состояние, 0 или 1, для заданного вывода порта. Когда состояние меняется, не важно, из 0 в 1, или из 1 в 0, микроконтроллер генерирует прерывание. В некоторых случаях можно задавать какое изменение (0-1 или 1-0) генерирует прерывание. Дополнительно может потребоваться выполнять чтение порта, для обновления сохраненной информации о состоянии вывода.

В начальный момент времени считается, что кнопка не нажата. При этом прерывания по изменению состояния вывода порта разрешены.

Как только состояние вывода изменится, микроконтроллер сгенерирует прерывание. Если состояние вывода изменилось из 0 в 1, мы фиксируем нажатие кнопки. Если состояние изменилось из 1 в 0, мы фиксируем отпускание кнопки.

При этом прерывания по изменению состояния вывода порта нужно заблокировать на время гашениядребезга контактов. Это исключает влияниедребезга на состояние кнопки в программе.

По истечении заданного времени гашениядребезга снова разрешаются прерывания по изменению состояния вывода порта.

Далее, обработка кнопки будет проходить так же, как и в начальный момент времени. С той лишь разницей, что теперь у нас не predetermined (отпущена), а фактическое состояние кнопки.

В приложении приведен алгоритм работы такого метода и программный код в качестве примера.

2.2. Аппаратные решения

2.2.1. Сглаживающие фильтры

Один из аппаратных способов устранения дребезга основан на использовании сглаживающих фильтров [7].

Это очень популярный метод гашения дребезга. Большинство используемых кнопок имеют лишь контакт на замыкание, что не позволяет использовать RS-триггер.

Сглаживающий фильтр занимается сглаживанием всплесков сигналов за счет добавления в схему элементов, имеющих своеобразную «инерцию» по отношению к таким электрическим параметрам как ток или напряжение. Самым распространенным примером таких «инерционных» электронных компонентов является конденсатор. Он может медленно накапливать и отдавать энергию, что позволяет добиться сглаживания [6].

Пример простого фильтра на базе RC-цепочки представлен на Рис. 2.1.

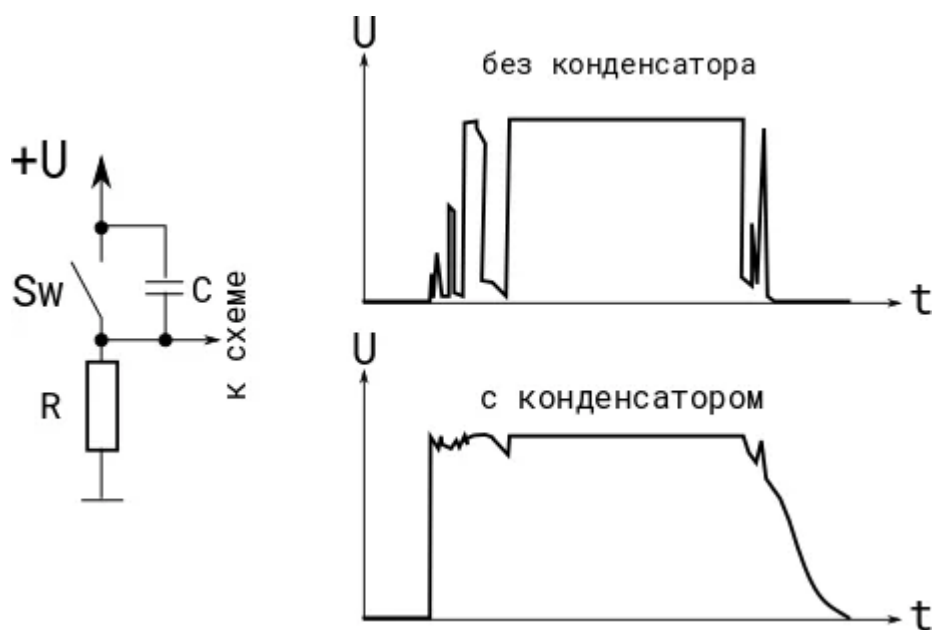


Рис. 2.1 Сглаживающий фильтр

Когда кнопка не нажата конденсатор заряжен, поэтому ток в цепи не протекает и не создает падения напряжения на резисторе. На входе схемы низкий уровень напряжения. Нажатие кнопки разряжает конденсатор. При этом на входе схемы появляется напряжение высокого уровня, через контакты кнопки. Короткие периоды разомкнутого состояния контактов во время

дребезга позволяют конденсатору немного зарядиться через резистор, при этом протекающий через резистор ток заряда конденсатора создает на нем падение напряжения, а это поддерживает на входе схемы напряжение высокого уровня, но плавно спадающее.

Если постоянная времени RC цепи выбрана правильно, конденсатор не успевает заметно зарядиться за времядребезга, что позволяет избежать ошибочной работы схемы, к которой кнопка подключена [8].

Этот метод гашениядребезга позволяет точно отследить момент замыкания кнопки, но момент размыкания отслеживается с задержкой. А это может быть критично. Кроме того, быстрое отпускание и повторное нажатие кнопки может быть воспринято как единичное длительное нажатие, так как конденсатор не успеет зарядиться до уровня, когда будет зафиксирован логический 0.

Сделать квадратную форму сигнала с помощью простой RC цепочки невозможно. Для «ограничения» сглаженных форм используется триггер Шмидта, который срабатывает при достижении определенного уровня сигнала. На выходе триггера Шмидта мы получим или высокий или низкий уровень сигнала. Ниже представлена схема и результат работы с триггером Шмидта [9].

Устранениедребезга с помощью триггера Шмидта представлено на Рис. 2.2.

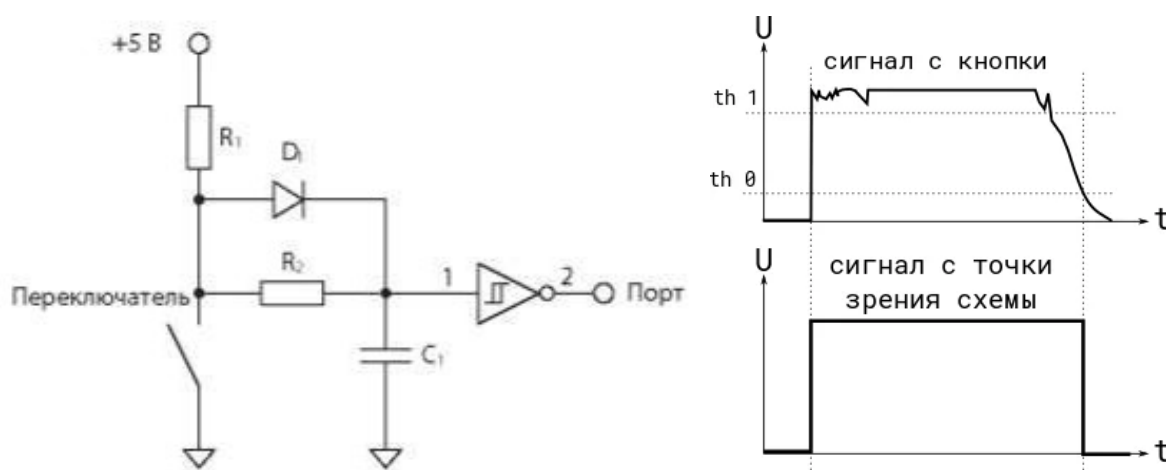


Рис. 2.2 Устранениедребезга с помощью триггера Шмидта

2.2.2. RS-триггер

Это классический и давно используемый метод. Схема для борьбы сдребезгом представлена на Рис. 2.3.

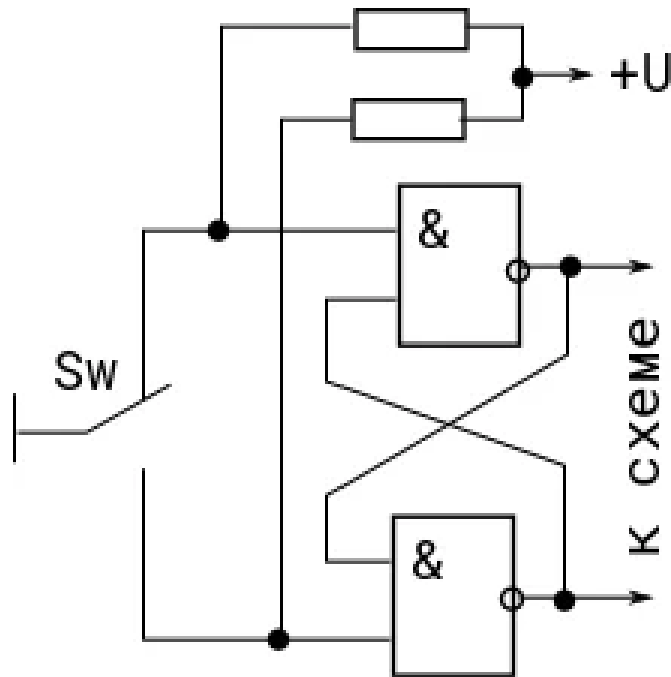


Рис. 2.3 Схема для подавления дребезга с использованием RS-триггеров

Эта схема гарантированно подавляет дребезг. Но она требует использования кнопки с переключающим контактом. Конструкция кнопки с переключающим контактом такова, что дребезг сразу двух половин контактной группы исключен. Именно это исключает многократное опрокидывание триггера при нажатии и отпускании кнопки [6].

Эта схема не фиксирует момент размыкания верхнего контакта, она фиксирует момент первого замыкания нижнего контакта. Точно так же, она не фиксирует момент размыкания нижнего контакта, но фиксирует момент замыкания верхнего. Учет этой особенности иногда может быть важным, когда кнопка используется для отслеживания положения объекта, а не нажимается человеком.

2.2.3. Использование одновибратора (ждущего мультивибратора)

Этот метод тоже встречается, хотя и реже двух ранее описанных. Пример схемы и ее работы представлен на Рис. 2.4.

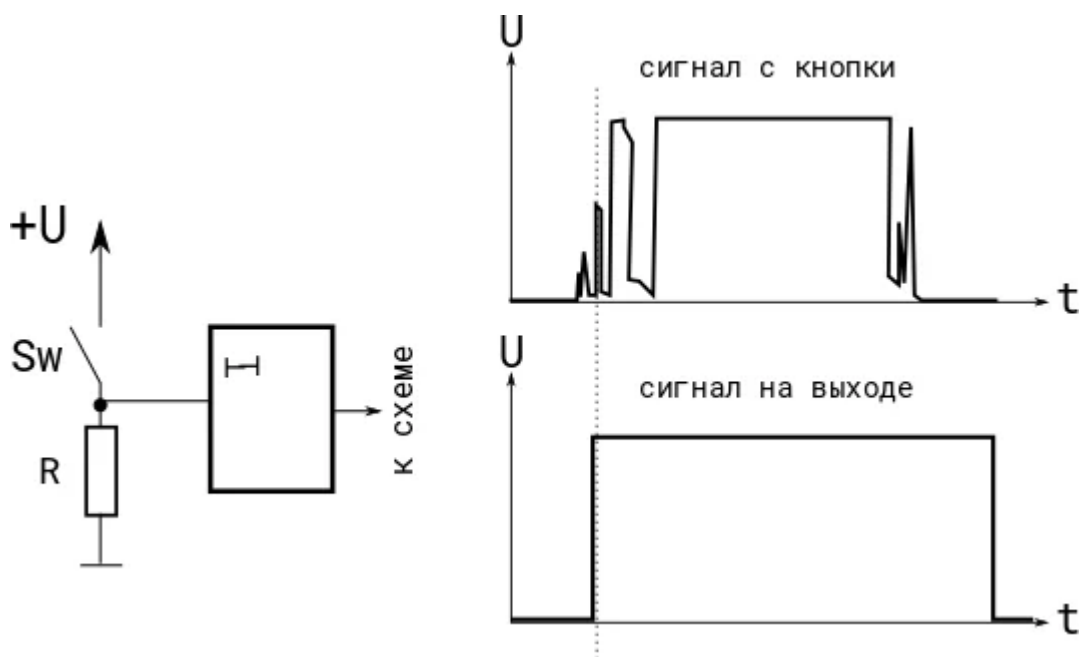


Рис. 2.4 Использование одновибратора для подавления дребезга

Одновибратор может быть перезапускаемым, или не перезапускаемым. В данном случае, он запускается по переднему фронту первого, распознанного как переход от низкого к высокому уровню, импульса. И выдает на выходе импульс фиксированной длительности, не зависящей от длительности нажатия кнопки [6].

Одновибратор может быть и специализированной микросхемой, и реализован, например, на D-триггере.

Однако, данный метод гашения дребезга имеет один, но существенный, недостаток. Если длительность нажатия кнопки больше длительности формируемого импульса, то возможны ложные срабатывания дребезга при отпуске кнопки. Кроме того, можно отследить только момент нажатия кнопки, но не момент отпускания.

Схему можно усложнить, что позволит устранить некоторые недостатки. Можно использовать несколько одновибраторов, что позволит сформировать отдельный сигнал длинного нажатия, а это невозможно с приведенными выше методами. Однако, в настоящее время такие решения применяются довольно редко.

3. БОРЬБА С ДРЕБЕЗГОМ В РЕАЛЬНОМ ПРОЕКТЕ

Среди рассмотренных ранее методов для конкретного проекта наиболее подходящим является аппаратный метод подавления дребезга с использованием прерываний по состоянию вывода порта.

Преимуществом данного метода по сравнению с любым аппаратным является тот факт, что для использования данного метода не нужно вносить никаких изменений в схему устройства, что в свою очередь не ведет к ее удорожанию и другим проблемам, к которым может привести добавление новых элементов.

Преимущество данного метода перед другими программными реализациями заключается в том, что, по сравнению с использованием обычной задержки, не требуется останавливать работу микроконтроллера, чтобы дождаться окончания дребезга, а также программа не будет тратить время на постоянный опрос состояния кнопки, как при использовании счетчика.

В Arduino для таких целей предусмотрен метод `attachInterrupt`, одним из параметров которого является функция, в которой прописываются действия, которые должны выполняться для обработки прерывания после его генерации.

В приложении приведен код программы с реализацией данного метода. Алгоритм подавления дребезга представлен на Рис. 3.1.

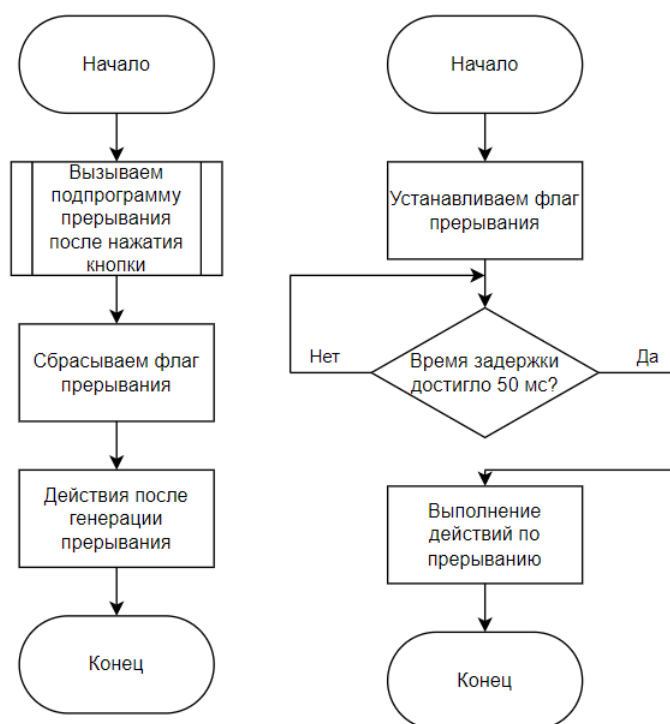


Рис. 3.1 Алгоритм подавления дребезга с помощью прерываний

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. *Г. Н. А. И. Т. В.* Подключение кнопок, дискретных сигналов к микроконтроллеру STM32F103. — 2018. — URL: <http://kb-au.ru/wp-content/uploads/AaSI-2018-4-4.pdf>.
2. [Электронный ресурс] Дребезг контактов — wikipedia.org. — URL: https://ru.wikipedia.org/wiki/%D0%94%D1%80%D0%B5%D0%B1%D0%B5%D0%B7%D0%B3_%D0%BA%D0%BE%D0%BD%D1%82%D0%B0%D0%BA%D1%82%D0%BE%D0%B2.
3. *Титце У. Ш. К.* Полупроводниковая схемотехника. Том I. — 2007. — URL: <http://lazysmart.ru/wp-content/uploads/2016/07/Tittse-U.-SHenk-K.-Poluprovodnikovaya-shemotekhnika.-Tom-I-2007.pdf>.
4. [Электронный ресурс] Схемы для устранения дребезга контактов механических переключателей - symmetron.ru. — URL: <https://www.symmetron.ru/articles/skhemy-dlya-ustraneniya-drebezga-kontaktov-mekhanicheskikh-pereklyuchateley/>.
5. [Электронный ресурс] Что такое дребезг контактов и как его устранить - samelectrik.ru. — URL: <https://samelectrik.ru/cto-takoe-drebezg-kontaktov-i-kak-ego-ustranit.html>.
6. [Электронный ресурс] Работаем с кнопками или о дребезге контактов и не только о нем — dzen.ru. — URL: <https://dzen.ru/a/XOITtUZcMACzcgsU>.
7. *Благодаров Л. Л. Владимиров А. В.* Программирование микроконтроллеров. — 2016. — URL: https://mipt.ru/upload/medialibrary/229/programmirovanie_mikrokontrollerov.pdf.
8. [Электронный ресурс] Дребезг контактов — asutpp.ru. — URL: <https://www.asutpp.ru/drebezg-kontaktov.html>.
9. [Электронный ресурс] Устранение дребезга конаткта кнопки - arduinomaster.ru. — URL: <https://arduinomaster.ru/datchiki-arduino/ustranenie-drebezg-kontaktov-knopki/#i-5>.

ПРИЛОЖЕНИЕ

Листинг 3.1 Исходный код для иллюстрации метода задержек

```

1  int currentValue, prevValue;
2  void loop() {
3      currentValue = digitalRead(PIN_BUTTON);
4      if (currentValue != prevValue) {
5          // Место, где возможна зона неопределенности
6          // Делаем задержку 50 мс
7          delay(50);
8          // Считываем значение, считая, что нестабильность
           ↳ исчезла
9          currentValue = digitalRead(PIN_BUTTON);
10     }
11
12     prevValue = currentValue;
13     Serial.println(currentValue);
14 }

```

Листинг 3.2 Исходный код для иллюстрации метода циклического опроса и счетчиков

```

1  // Состояния кнопки
2
3  bool KeyPressed; // Кнопка нажата
4  int KeyCounter; // Счетчик кнопки
5  int LongCounter; // Счетчик длительного нажатия
6
7  // События кнопки
8
9  bool ShortPress; // короткое нажатие
10 bool LongPress; // Длинное нажатие
11
12 // Пороги состояний кнопки
13
14 int Press_th; // Порог гашениядребезга

```

```

15  int Long_th; // Порог длинного нажатия
16
17  // Вывод порта к которому подключена кнопка
18
19  bool key;
20
21  // Начальные значения переменных
22
23  KeyPressed = false;
24  KeyCounter = 0;
25  LongCounter = 0;
26  ShortPress = false;
27  LongPress = false;
28  ShortPress = false;
29
30  if(Key) {
31      if (KeyConter < Press_th) {
32          KeyCounter++;
33      } else {
34          KeyPresed=true;
35          if (LongCounter < Long_th) {
36              LongCounter++;
37          } else{
38              Longpress=true;
39          }
40      } else {
41          if (KeyCounter) {
42              KeyCounter--;
43          } else {
44              KeyPressed=false;
45              LongCounter=0;
46              if(!LongPress) {
47                  ShortPress=true;
48              } else {
49                  LongPress=false;
50              }

```



```

51     }
52 }

```

Листинг 3.3 Исходный код для иллюстрации метода прерываний по состоянию вывода порта

```

1  volatile boolean chatterFlag = false;
2
3  void setup() {
4      Serial.begin(9600);
5      pinMode(2, INPUT_PULLUP);
6      attachInterrupt(0, chatter, FALLING);
7  }
8
9  volatile uint32_t chatter_time;
10 void chatter() {
11     chatterFlag = true;
12     // 50 подавляемдребезгкнопки, читаем состояние
13     ↪ порта
14     if (millis() - chatter_time >= 50 && digitalRead(2))
15     {
16         chatter_time = millis();
17         // Дальнейший код прерывания
18     }
19 }
20 void loop() {
21     if (chatterFlag) {
22         chatterFlag = false;
23         // Действия после генерации прерывания
24     }
25     delay(1000);
26 }

```

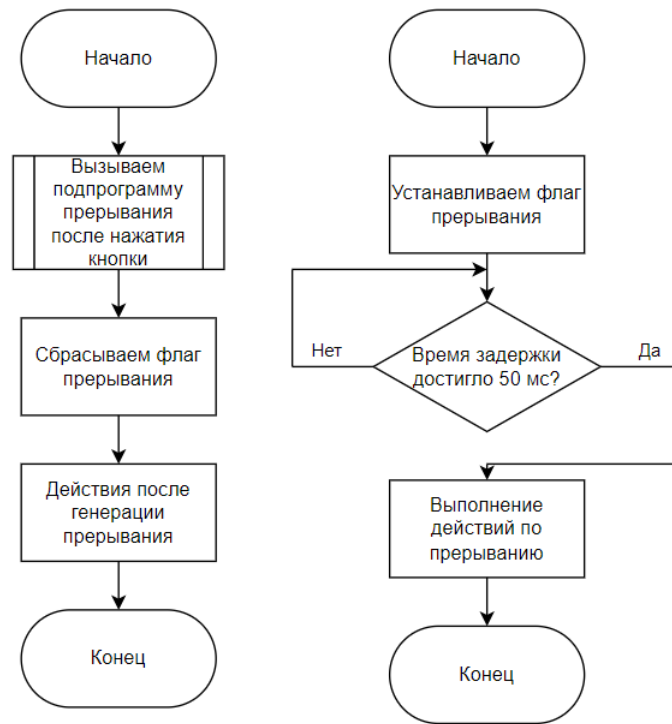


Рис. 3.2 Алгоритм подавлениядребезга с помощью прерываний