**Ход работы:**

1. **Инициализация ШИМ**:
   * ШИМ реализован на базе таймера Timer0 в режиме **FastPWM** (режим 3).
   * Установлен неинвертирующий режим: при совпадении значения регистра OCR0 с таймером сигнал сбрасывается в 0.
   * Частота сигнала определяется делителем (CS00, CS02) и значением в регистре OCR0.
2. **Изменение скважности**:
   * В массиве my\_pwn[] хранятся значения для изменения коэффициента заполнения (скважности). Пример:
     + 25%: OCR0 = 256/4
     + 50%: OCR0 = 256/2
     + 75%: OCR0 = 3\*256/4
     + 100%: OCR0 = 256
   * Прерывание таймера используется для последовательного переключения между значениями массива.
3. **Рассчитанные коэффициенты для OCR0**:
   * Для получения скважности 25%, 50%, 75% и 100% рассчитаем OCR0:
     + 25%: 63OCR0=4256​−1=63
     + 50%: 127OCR0=2256​−1=127
     + 75%: 191OCR0=43×256​−1=191
     + 100%: 255OCR0=256−1=255
4. **Прерывания и обработка**:
   * В обработчике ISR(TIMER0\_OVF\_vect) значения OCR0 изменяются по циклу из массива, создавая эффект изменения скважности.

**Ответы на вопросы:**

1. **Рассчитать коэффициенты OCR0 для получения осциллограммы:**
   * Для генерации сигнала с заданной скважностью расчет приведен выше. Пример:
     + 50%: OCR0=127OCR0 = 127OCR0=127
     + 75%: OCR0=191OCR0 = 191OCR0=191
2. **Как изменится работа таймера, если биты предделителя частоты установить в нули?**
   * При установке всех битов делителя частоты (CS02, CS01, CS00) в нули, таймер перестанет работать, так как его тактирование будет отключено.
3. **В каких режимах работы можно использовать таймеры, описанные в данной работе?**
   * **Режимы работы таймеров:**
     + **Normal mode:** обычный счет без автоматического сброса.
     + **CTC (Clear Timer on Compare):** таймер сбрасывается при совпадении с заданным значением.
     + **FastPWM:** генерация ШИМ-сигнала с высокой частотой.
     + **Phase Correct PWM:** генерация ШИМ-сигнала с двойным счетом (вверх и вниз).
4. **Разработка обработчика прерывания для синусоидальной формы:**
   * Для генерации синусоидального сигнала можно использовать таблицу значений амплитуды, представляющую один период синуса. В обработчике прерывания нужно циклически загружать значения из этой таблицы в OCR0.

volatile uint8\_t sine\_wave[32] = {127, 150, 173, 196, 218, 238, 255, 270, 283, 293, 300, 304, 305, 304, 300, 293, 283, 270, 255, 238, 218, 196, 173, 150, 127, 104, 81, 58, 36, 16, 0, 0};

volatile uint8\_t sine\_index = 0;

ISR(TIMER0\_OVF\_vect) {

OCR0 = sine\_wave[sine\_index++];

if (sine\_index >= 32) sine\_index = 0;

}

**Доработка для управления скважностью через UART:**

* Добавьте прием данных через UART. В зависимости от введенного значения (1–100), рассчитайте новое значение для OCR0.

ISR(USART\_RXC\_vect) {

uint8\_t duty\_cycle = UDR; // Принимаем значение от 1 до 100

if (duty\_cycle >= 1 && duty\_cycle <= 100) {

OCR0 = (duty\_cycle \* 256 / 100) - 1;

}

}

#### Выводы:

1. В ходе работы изучены основные режимы работы таймеров микроконтроллера AVR, включая генерацию ШИМ.
2. Реализовано переключение скважности сигнала и управление им через прерывания.
3. Определены способы управления параметрами ШИМ через UART.

**Практическая работа с осциллографом.**

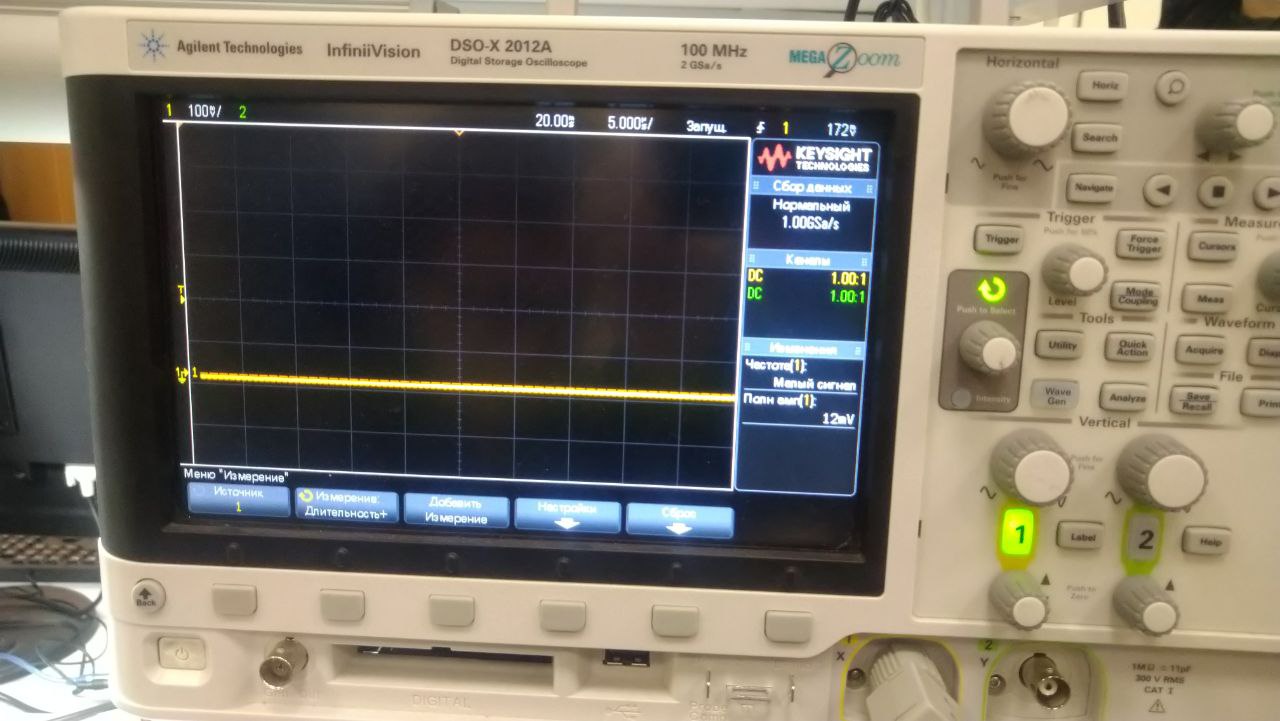
В ходе данной работы были приобретены практические навыки работы с осциллографом, который является основным инструментом для анализа электрических сигналов в реальном времени. Осциллограф позволяет визуализировать форму сигнала, измерять его параметры, такие как амплитуда, частота и длительность, а также обнаруживать возможные искажения.

Основные функции, изученные в ходе работы, включают:

* **Измерение параметров с помощью курсоров**: позволяет точно определить временные и амплитудные характеристики сигнала.
* **Автоскейл (автоматическое масштабирование)**: функция для оптимального отображения сигнала на экране при изменении его параметров.
* **Команда "Stop/Run"**: используется для фиксации сигнала на экране, что облегчает анализ полученных данных.
* **Настройка горизонтальной развертки**: регулирует временной масштаб отображаемого сигнала для более детального анализа его периодических характеристик.

Перед началом работы был произведен тест осциллографа без подключения к микроконтроллеру. На рисунке ниже представлены его показания. Видимая горизонтальная линия свидетельствует об отсутствии входного сигнала, что подтверждает корректность базовой настройки прибора.

В отсутствие сигнала мы наблюдаем входное напряжение осциллографа равное нулю, экране отображается линия, совпадающая с уровнем базовой настройки. Это является стандартным результатом при работе с осциллографами, не подключенными к источнику сигнала.

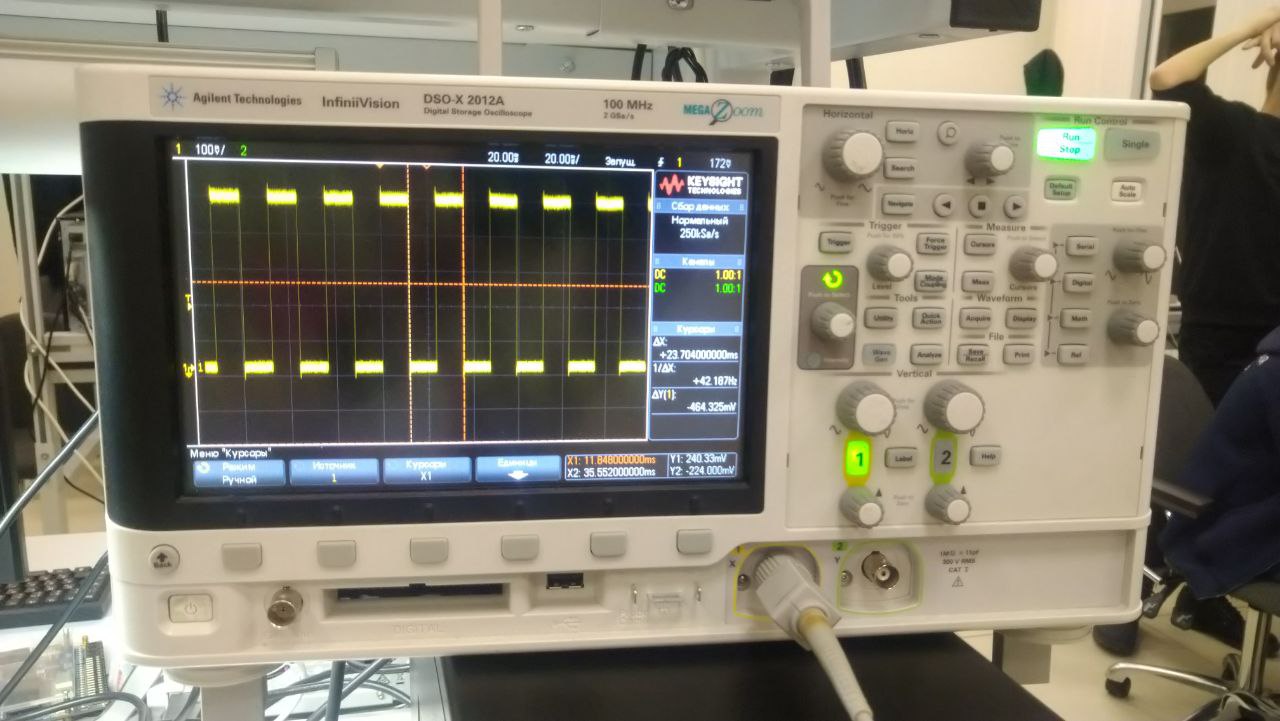


Далее мы наблюдали сигнал после подключения осциллографа к микроконтроллеру. Полученный сигнал имеет форму меандра — периодической последовательности прямоугольных импульсов. Такой сигнал характерен для работы модуля ШИМ (широтно-импульсной модуляции), где длительность высокого уровня регулируется для изменения скважности.

В процессе работы были настроены параметры осциллографа, такие как:

* **Чувствительность по вертикали** (вольты на деление) для корректного отображения амплитуды сигнала;
* **Частота горизонтальной развертки** (секунды на деление) для отображения нескольких периодов сигнала;
* **Фильтрация помех** для устранения нежелательных шумов и улучшения четкости изображения.

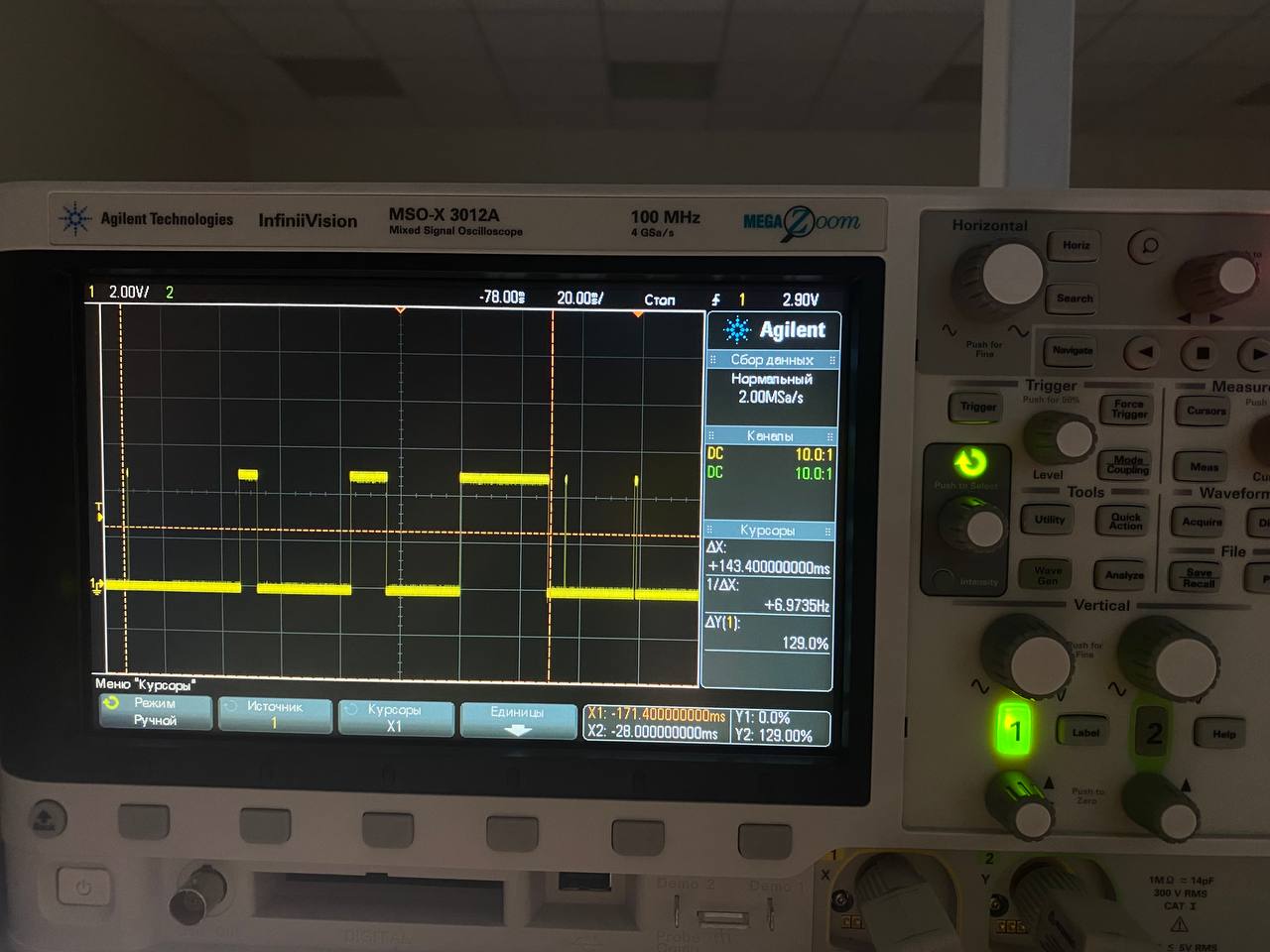
Благодаря правильно подобранным настройкам удалось наблюдать сигнал без искажений и помех, что позволило получить точные данные для анализа.



На рисунке ниже продемонстрированы циклические изменения скважности ШИМ-сигнала, которые были реализованы с использованием микроконтроллера ATmega16. Скважность сигнала изменялась итерационно, что позволило наблюдать переходы между различными уровнями заполнения: от 0% до 100% с заданным шагом.

Управление скважностью осуществлялось программно, с помощью модуля таймера/счетчика микроконтроллера. Изменение значения регистра OCR0 (Output Compare Register 0) определяло продолжительность высокого уровня сигнала в каждом цикле.

Этот этап работы позволил изучить принципы программного управления параметрами ШИМ, а также продемонстрировал применение прерываний для обеспечения точного изменения параметров сигнала. Полученный сигнал визуализировался с помощью осциллографа, что подтвердило корректность реализации.



В ходе выполнения лабораторной работы мы освоили работу с микроконтроллером ATmega16 и вспомогательным оборудованием, таким как осциллограф. Написанный код позволил успешно реализовать управление параметрами сигнала ШИМ, включая его скважность. Особое внимание было уделено программированию циклического изменения скважности **Вывод**:, что продемонстрировало возможности использования микроконтроллера для динамического управления сигналами.

Полученные навыки могут быть применены в разработке систем управления, где требуется точное регулирование мощности или состояния исполнительных устройств.

**Листинг кода**:

#define *F\_CPU* 11059200UL

#include <avr/io.h>

#include <util/delay.h>

#include <avr/interrupt.h>

volatile int i=0;

volatile int my\_pwn[5]={0,64,128,192,256};

ISR(TIMER0\_OVF\_vect){

OCR0=my\_pwn[i++];

if(i>5)i=0;

TCNT0=10000;

}

void Timer1\_Init() {

DDRA=0xff;

TCCR1B|=(1<<CS12)|(1<<CS10);

TIMSK = (1 << TOIE1);

}

void SHIM\_init(){

DDRB |=(1<<3);

TCCR0|= (1<<CS00)|(1<<CS02);

TCCR0|= (1<<WGM01)|(1<<WGM00);/\*Mode3 Режим генерации сигнала FastPWM. \*/

TCCR0|= (1<<COM01);/\*Режим работы компаратора СОМ0[1:0]=[1,0]. Не-инвертирующий режим когда выполняется сравнение установить ОСО в 0.\*/

OCR0=127;

}

void main(void) {

SHIM\_init();

Timer1\_Init();

TIMSK|=(1<<TOIE0);

sei();

/\* Replace with your application code \*/

while (1)

{

}

}