

ЛАБОРАТОРНА РОБОТА 1

СЕРЕДА РОЗРОБКИ AVR-STUDIO

Мета роботи: вивчення системи команд мікроконтролера AVR ATmega 128 и принципів його програмування на мові асемблера в середовищі AVR Studio, способів виконання та отладки програм з використанням програмного стимулятора.

Завдання: набрати в середовищі програмування AVR Studio наведену програму на мові асемблера, провести асемблювання та виконання в різних режимах.

1 AVR Studio

AVR Studio 4 - нове професійне інтегроване середовище розробки (Integrated Development Environment - IDE), призначена для написання і налагоджування прикладних програм для AVR мікропроцесорів в середовищі Windows 9x/NT/2000/XP. AVR Studio 4 містить асемблер і емулятор. AVR Studio підтримує COFF, як формат вихідних даних для символного налагоджування. Інші програмні засоби третіх фірм також можуть бути конфігуровані для роботи з AVR Studio.

1.1 Основні елементи інтерфейсу AVR Studio

Головне вікно програми зображено на рисунку 1.1. Ключове вікно в AVR Studio - це вікно початкового тексту програми (1). Коли об'єктний файл відкритий, автоматично створюється вікно початкового тексту програм. У вікні відображувався код, який виконується в налагоджувальному оточенні (емуляторі або програмному симуляторі), а текстовий маркер (2) завжди знаходиться на рядку, який буде виконаний в наступному циклі.

Користувач може виконувати програму повністю в покроковому режимі, трасуючи блоки функцій, або виконуючи програму до місця, де є курсор. На додаток можна визначати необмежене число точок зупинок, кожна з яких може бути включена або вимкнена. Точки зупинок зберігаються між сесіями роботи.

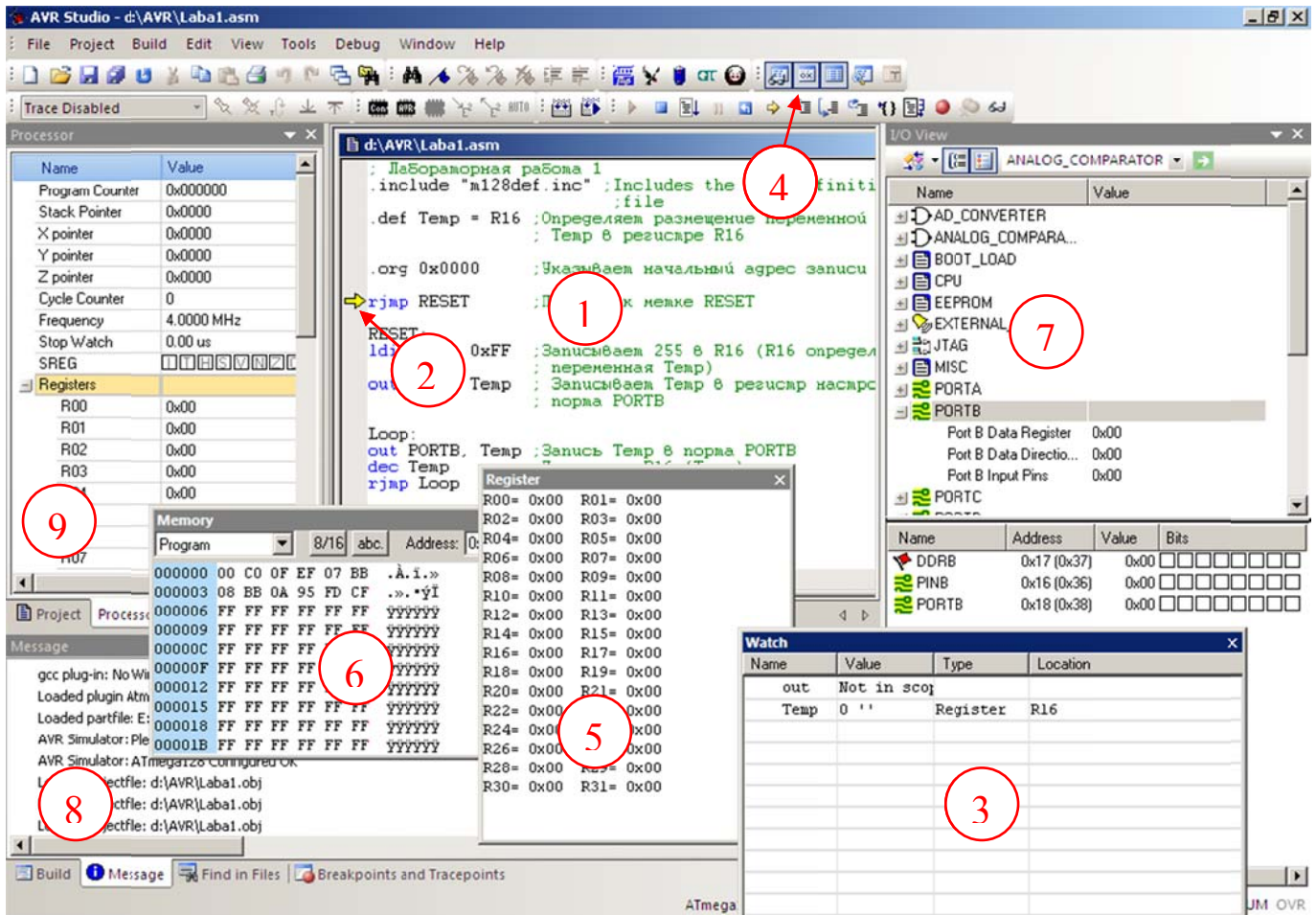


Рисунок 1.1 - Головне вікно програми

У вікні початкового тексту програми виводиться інформація про процес виконання програми. На додаток, AVR Studio має багато інших вікон, які дозволяють управляти і відображувати інформацію про будь-який елемент мікроконтролера.

Список доступних вікон:

1) Watch window (3): Вікно показує значення певних символів. В цьому вікні користувач може переглядати значення і адреси змінних; це та багато інших вікон визивається кнопкою на панелі інструментів (4) або меню View.

2) Trace window: Вікно показує хронологію програми, що виконується в даний час;

3) Register window (5): Вікно показує вміст регістрів. Регістри можна змінювати під час зупинки програми;

4) Memory windows (6): Вікна показують вміст пам'яті програм, даних, портів введення/висновку та енергонезалежного ПЗП. Пам'ять можна переглядати в HEX, двійковому або десятковому форматах. Вміст пам'яті можна змінювати під час зупинки програми;

5) I/O window (7): Показує вміст різних регістрів введення/виведення, EEPROM, I/O порти, таймери; та ін;

6) Message window (8): Вікно показує повідомлення від AVR Studio;

7) Processor window (9): У вікні відображувалася важлива інформація про ресурси мікроконтролера, включаючи програмний лічильник, показчик стека, регістр статусу і лічильник циклу. Ці параметри можуть модифікуватися під час зупинки програми.

Настройки рабочего окружения зберігаються при виході. При першому запуску вимагається набувати вікна для управління і висновку необхідної інформації. під час наступного завантаження настройки автоматично відновлюються.

1.2 Створення проекту

При запуску AVR Studio на екран виводиться вікно Створення/Відкриття проекту (рис 1.2).

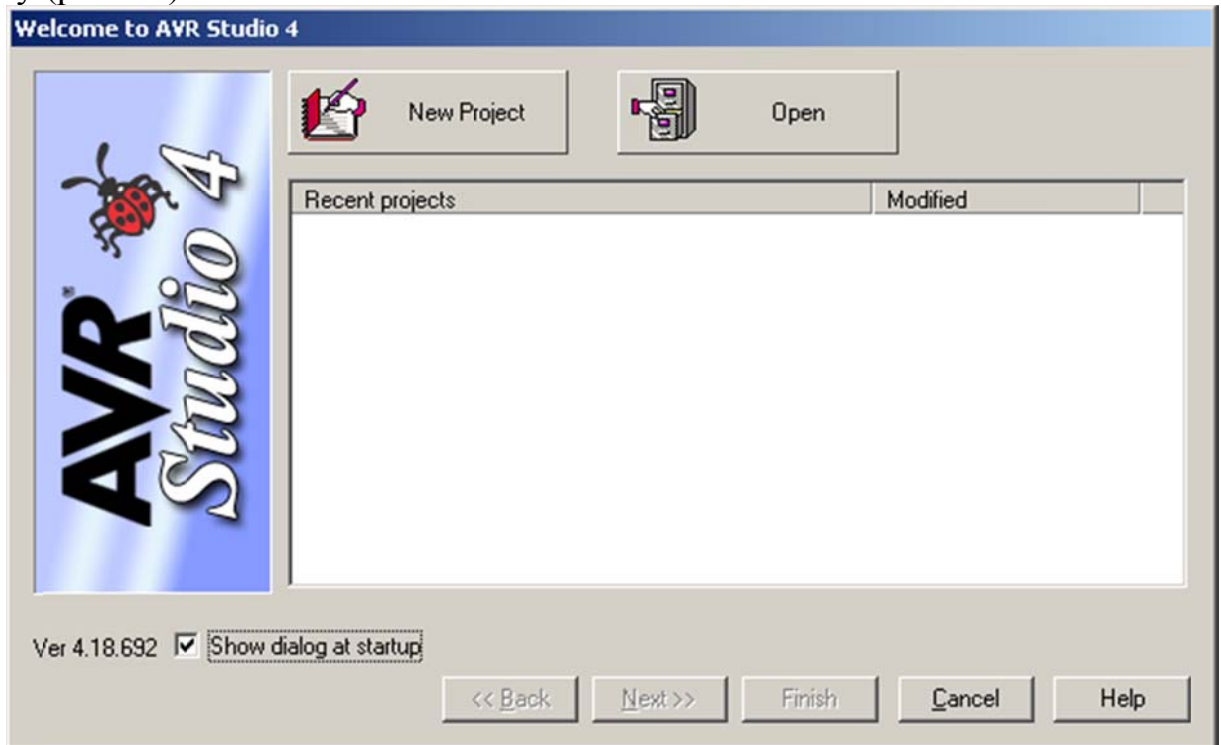


Рисунок 1.2 - Вікно Створення/Відкриття проекту

Вразі вибору створення проекту (натискання кнопки New Project) на екрані з'являється вікно зображене на рисунку 1.3.

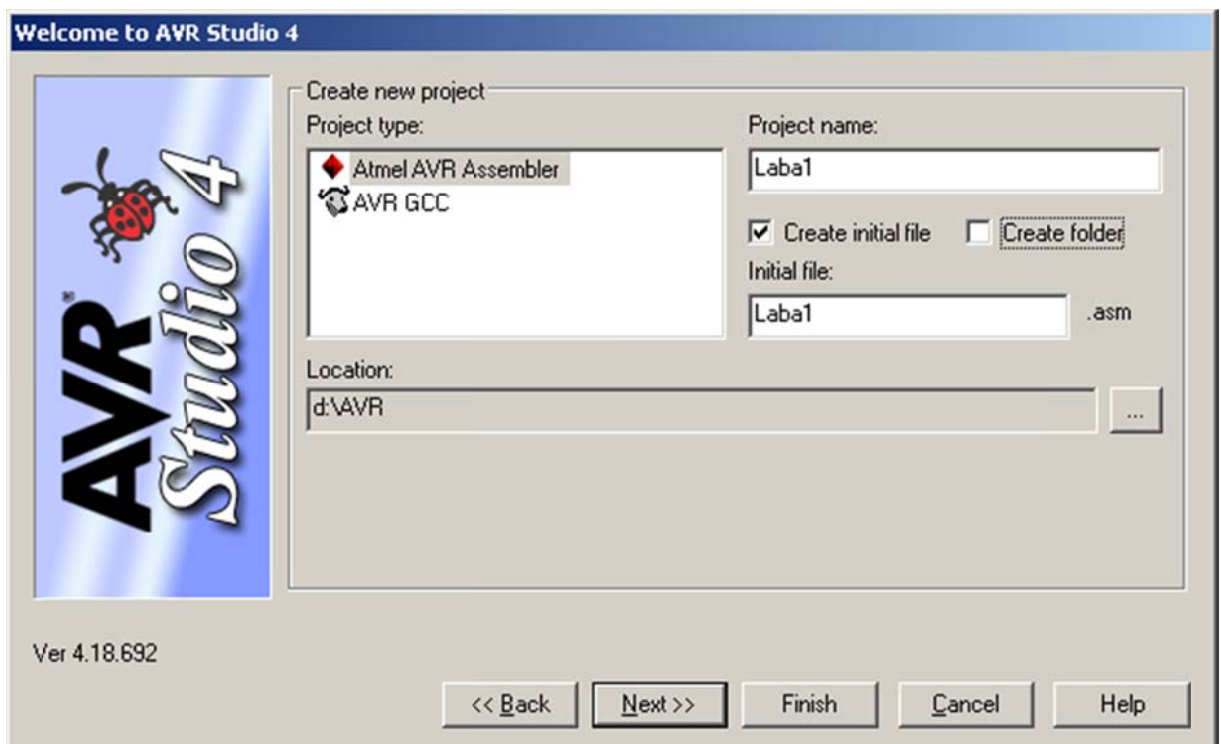


Рисунок 1.3 – Вікно початкових налаштувань проекту

У вікні пропонується вибрати мову програмування, зазначити ім'я проекту на ім'я основного файлу проекту.

Після натиснення кнопки "Next" відкривається вікно "Select debug platform and device" (рис.1.4), де вибирається налагоджувальна платформа (симулятор або емулятор) і тип мікроконтролеру. Можна вибрати один з пропонованих внутрішньо схемних емуляторів, помітимо, що у кожного емулятора свій список підтримуваних мікросхем. Для даного прикладу ми вибираємо, як налагоджувальну платформу AVR Simulator і мікросхему ATmega128. Натиснути кнопку «Finish».

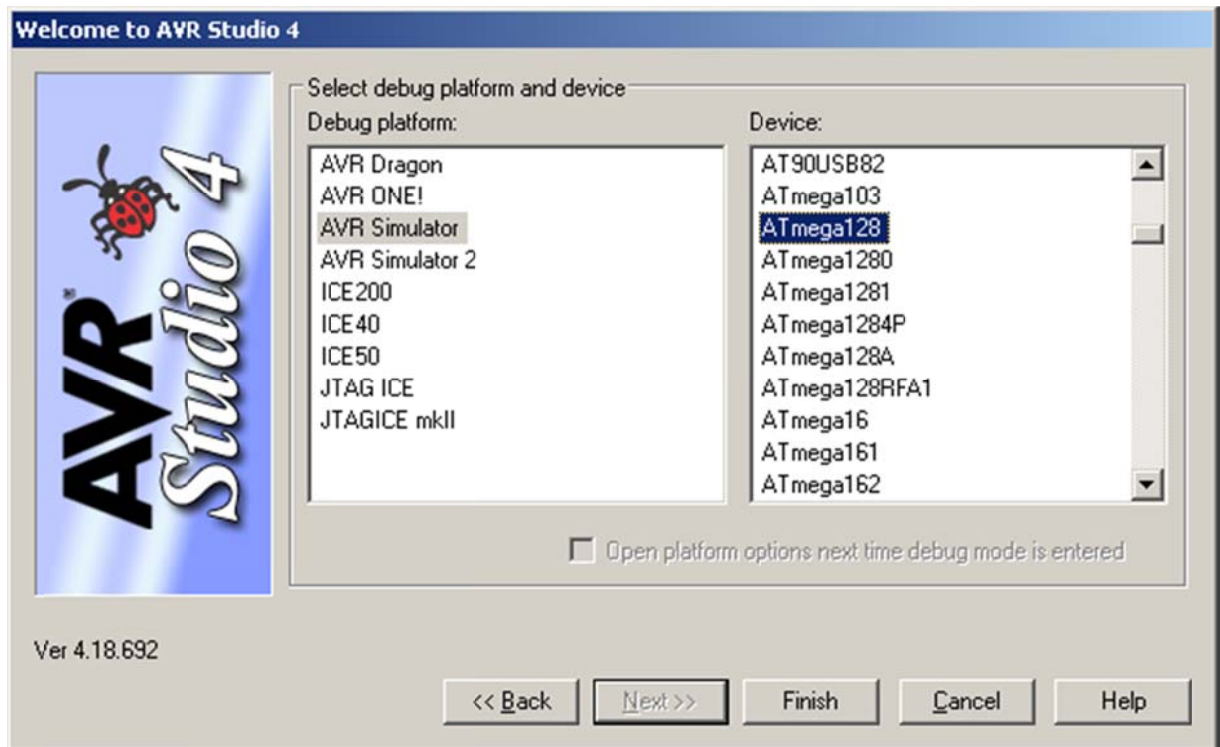


Рисунок 1.4 - Вікно вибору платформи та пристрою

2 Прграма на мові ассемблера

; Лабораторна робота 1

```
.include "m128def.inc" ; включення файлу визначень ATmega128  
.def Temp = R16 ; Визначаємо розташування змінної Temp  
.org 0x0000 ; Початкова адреса  
rjmp RESET ;перехід на позначку RESET
```

RESET:

```
ldi Temp, 0xFF ; Запис 255 в R16 (R16 пов'язаний з змінною Temp)  
out DDRB, Temp ; запис значення Temp в регістр налаштування порта  
PORTB
```

Loop:

```
out PORTB, Temp ; запис Temp в порт PORTB  
dec Temp ; декремент (зменшення на 1) R16 (Temp)  
rjmp Loop ;перехід на позначку Loop
```