

Лабораторна робота №2

Тема: ВСТАНОВЛЕННЯ ТА НАЛАШТУВАННЯ СЕРЕДОВИЩА РОЗРОБКИ

Мета: Розробити пристрій з дискретними входами та виходами.

Теоретичні відомості

Програмування на апаратному рівні вимагає налаштування кросплатформеної компіляції, коли код для виконання на процесорі формується на іншій платформі або архітектурі. Тут буде наведена інструкція налаштування середовища розробки для мікроконтролерів на базі CodeBlocks.

Для налаштування середовища розробки необхідно встановити в систему середовище розробки CodeBlocks. Завантажити IDE CodeBlocks можна за адресою www.codeblocks.org. Але це середовище розробки має лише редактор коду, компілятор використовується зовнішній. Тому для користувачів Windows потрібно завантажити та встановити WinAVR, який містить в собі все необхідне. Потрібно лише прослідкувати, щоб в шляху встановлення не було проміжків та знаків кирилиці. Необхідне програмне забезпечення можна також завантажити з серверу кафедри \\10.5.5.1. Для користувачів linux подібних систем потрібно встановити пакети gcc-avr binutils-avr avr-libc. Після цього можна перейти до налаштування середовища програмування. Також корисно завантажити та скопіювати в довільний каталог симулятор мікроконтролерів AVR avrsimul, інструкцію користування яким показано нижче.

Завдання:

1. Завантажити та встановити: WinAVR, CodeBlocks, simulavr. Все програмне забезпечення є вільним у використанні і доступне до завантаження з серверу кафедри. Установка є стандартною, але потрібно прослідкувати, щоб шляхи до встановлених програм не мали проміжків та кирилических символів. Також, якщо в першу чергу встановити саме WinAVR, налаштування CodeBlocks значно спрощується. Вам залишиться лише проконтролювати зазначені далі зміни в налаштуваннях компілятора та налагоджувача. Всі

прикладі та налаштування виконанні при встановленні програмних продуктів в кореневий каталог диску C.

2. Налаштувати CodeBlocks для роботи з компілятором avr-gcc.

2.1. Для цього в полях “Settings... Compiler...” потрібно вибрати компілятор для наших мікроконтролерів (рис 1.а)

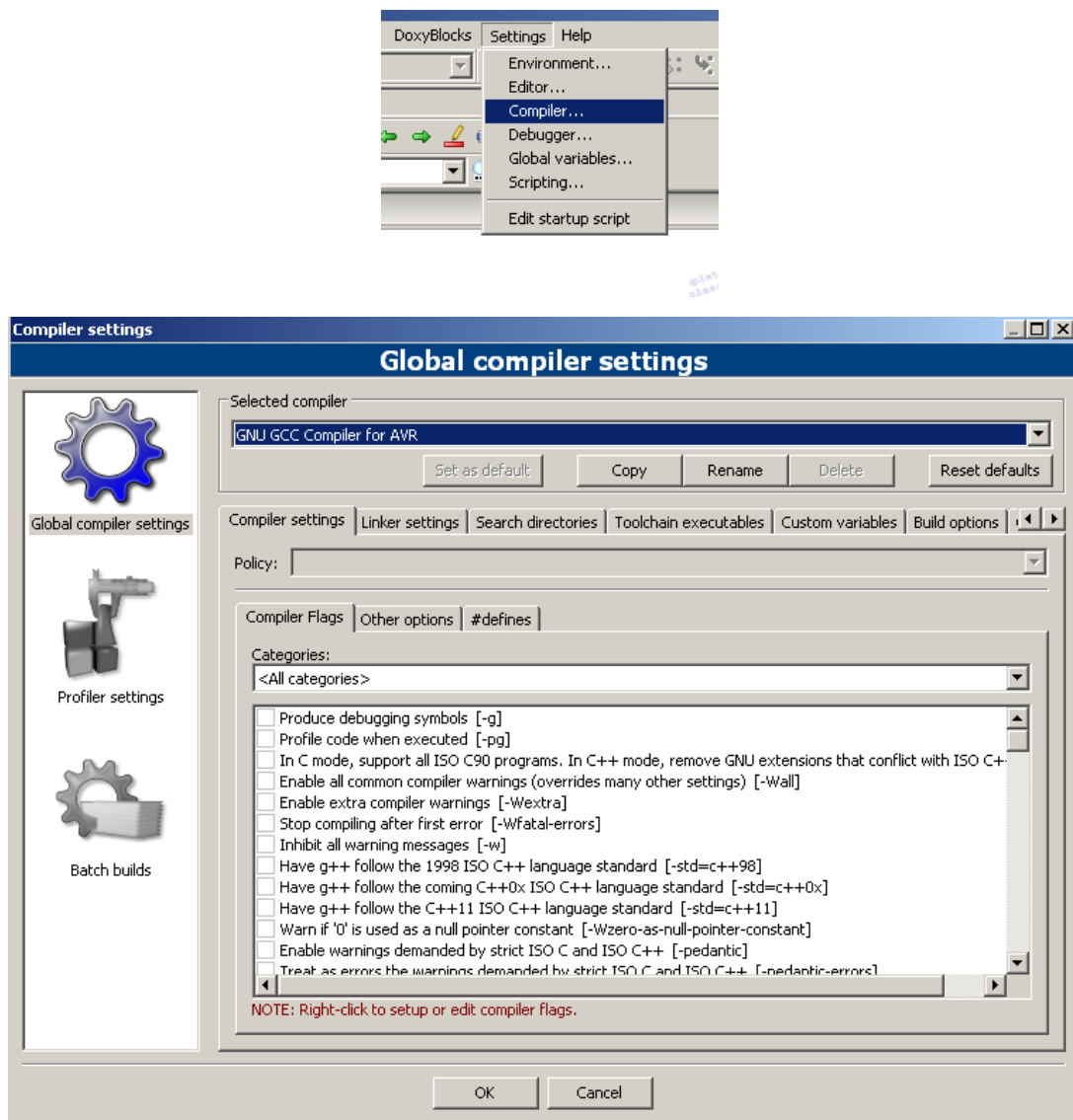


Рис. 1.а — Вікно налаштувань компілятора

2.2. В вкладці “Toolchain executables” додаємо до конфігурації шлях до самого компілятора та його програм (рис. 1.б). В разі відсутності потрібного дебагера в списку, залишіть поле невизначеним, налаштуємо його пізніше.

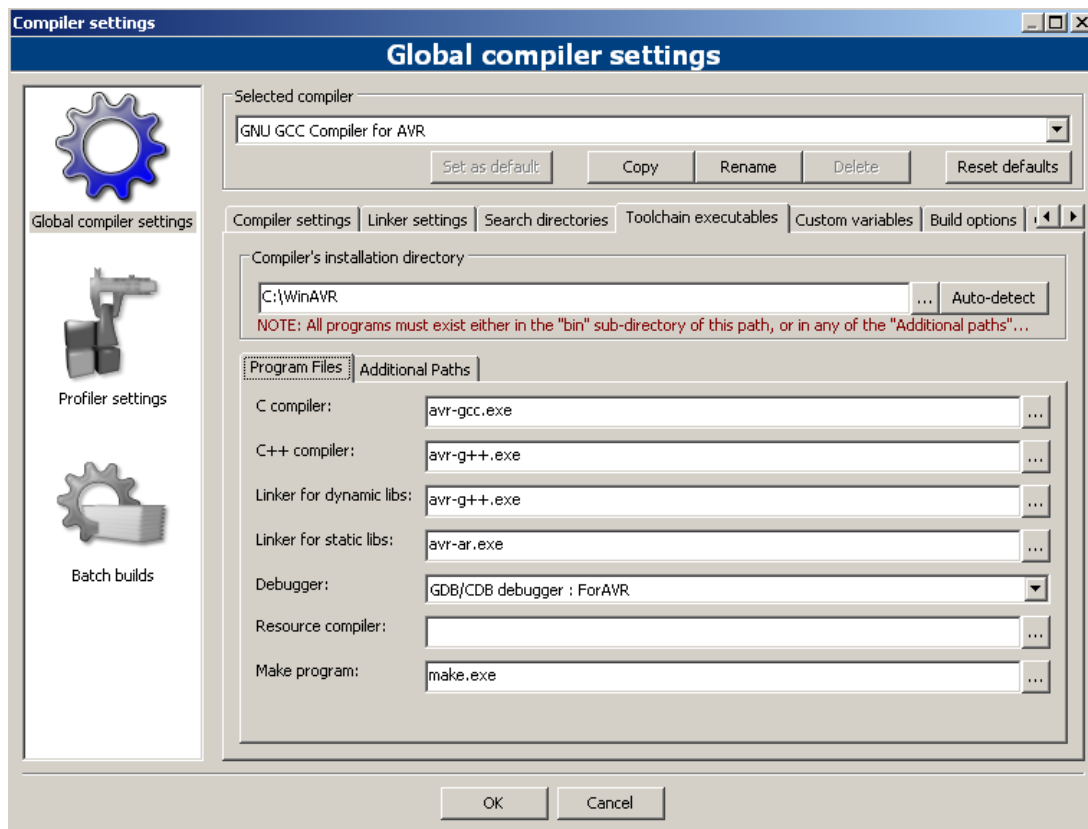


Рис. 1.б — Шлях до компілятора та визначення файлів компілятора

2.3. Потрібно вказати шляхи до бібліотек та модулів (рис. 1.в).

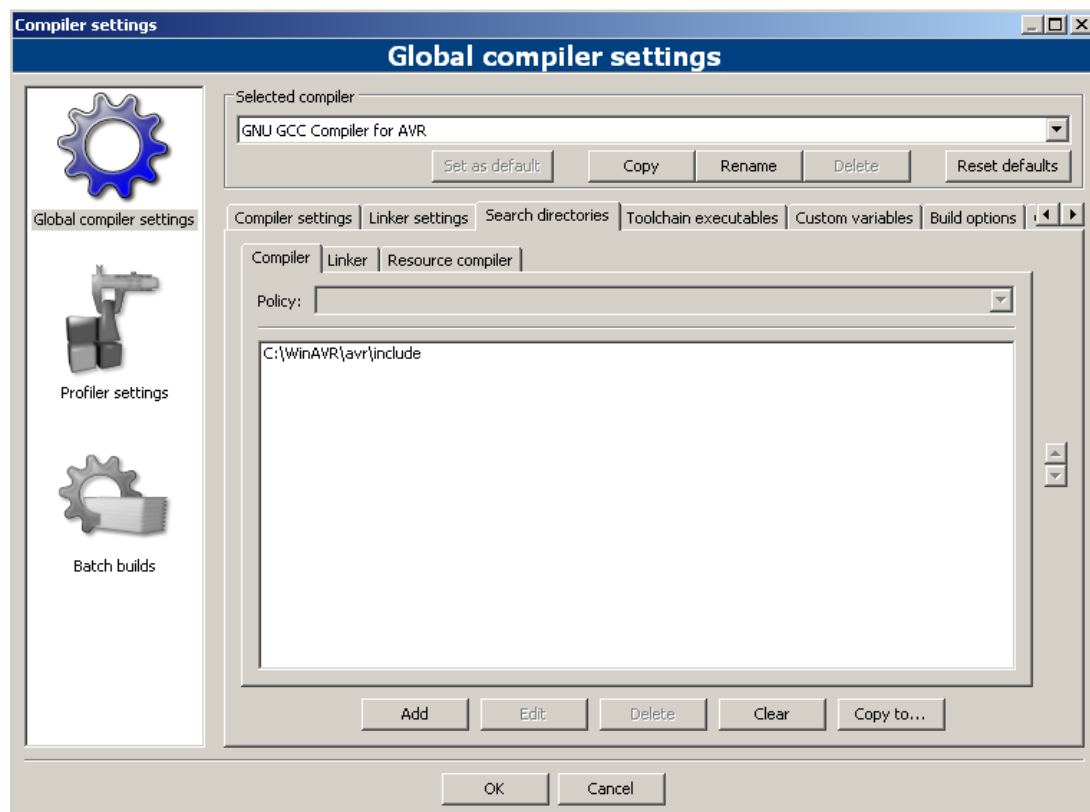


Рис. 1.в — Встановлення шляхів до бібліотек

2.4. Тепер можна вказати програму, за допомогою якої можна відлагоджувати Ваш код. Якщо ще не вказано дебагер, поверніться до позиції як на рис. 1.б і додайте щойно створений дебагер для AVR процесорів.

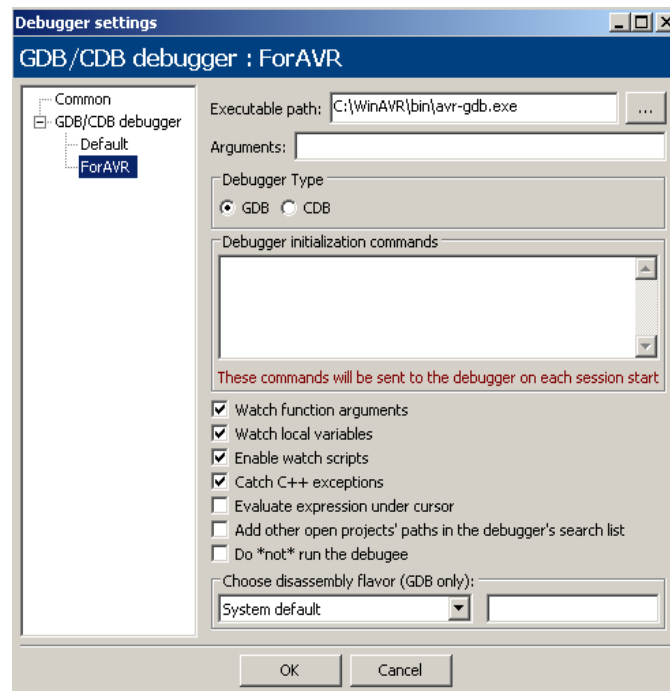


Рис. 1.г — Вибір програми-дебагера

Попередні налаштування можна вважати завершеними. Тепер приступимо до створення проекту для мікроконтролера Atmega48.

3. Створюємо перший проект.

3.1. Відкриваємо вікно створення проекту, обираємо “AVR Project”. Далі зручно для всіх проектів виділити спільну теку, де для кожного проекту буде створено окремі теки (рис. 2.а).

3.2. Для коректної адресації та врахування обсягів оперативної та постійної пам’яті, потрібно вказати цільовий процесор. В прикладі вказаний процесор ATMega48 (рис. 2.б).

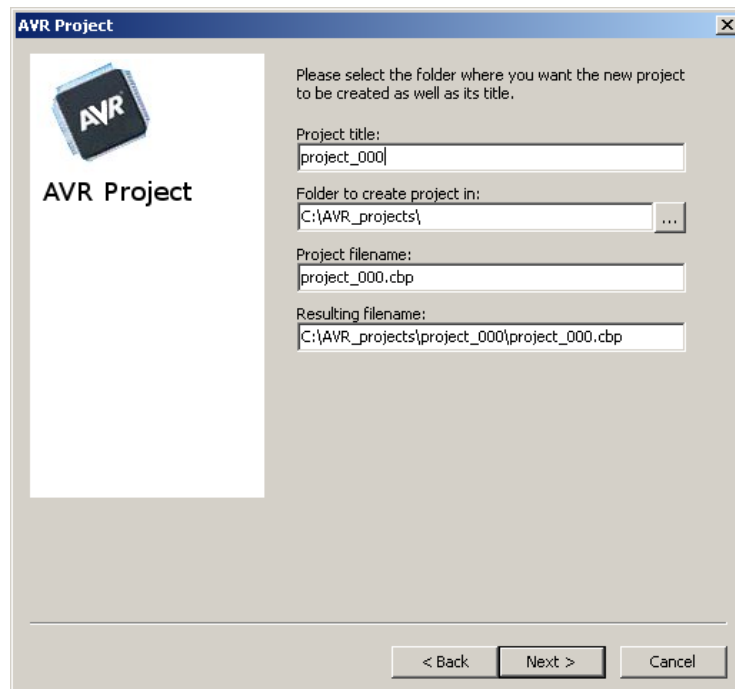
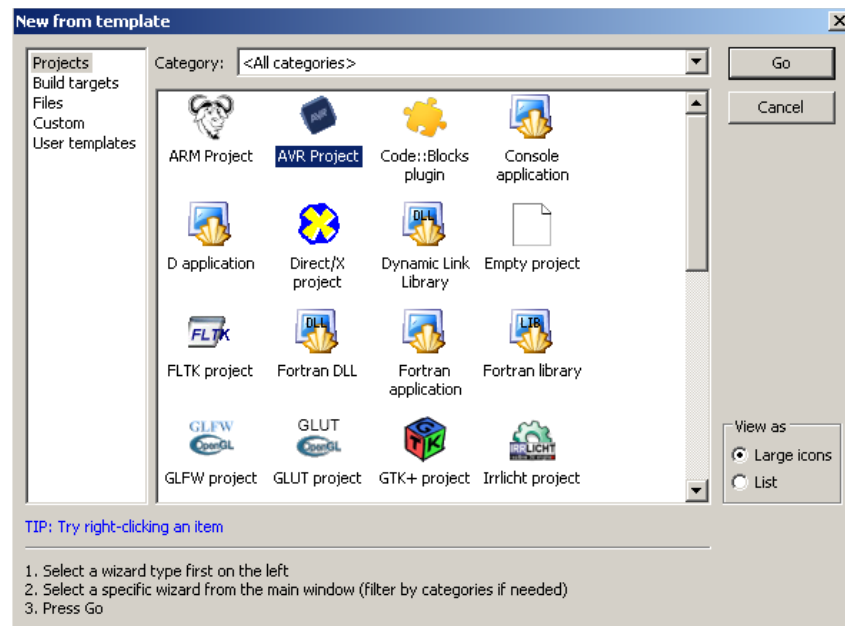



Рис. 2.а — Вікно створення проекту

3.3. Робимо пробну компіляцію проекту за допомогою кнопки збірки проекту на панелі задач: . Результатом є повідомлення з пустим списком зауважень та помилок, яке показано на рис. 2.в.

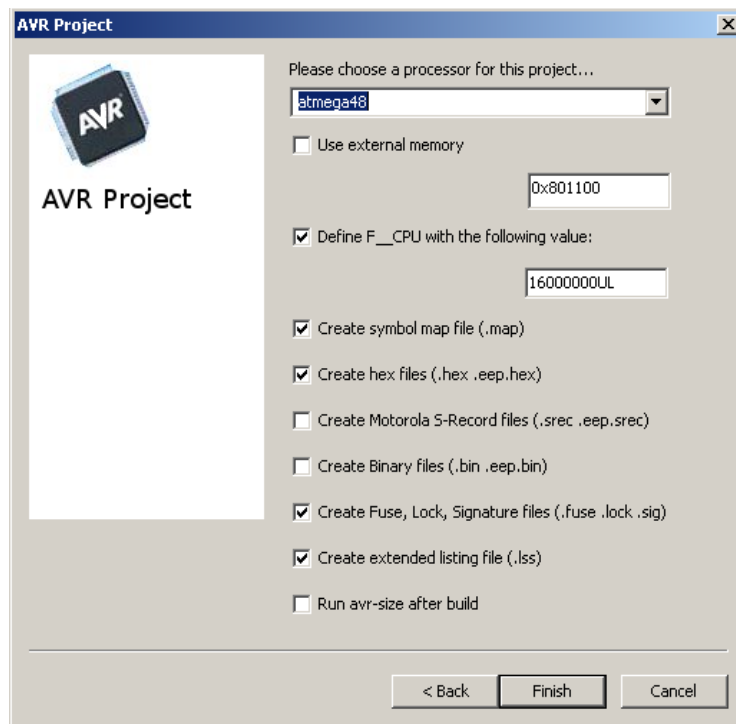


Рис. 2.6 — Вибір процесору

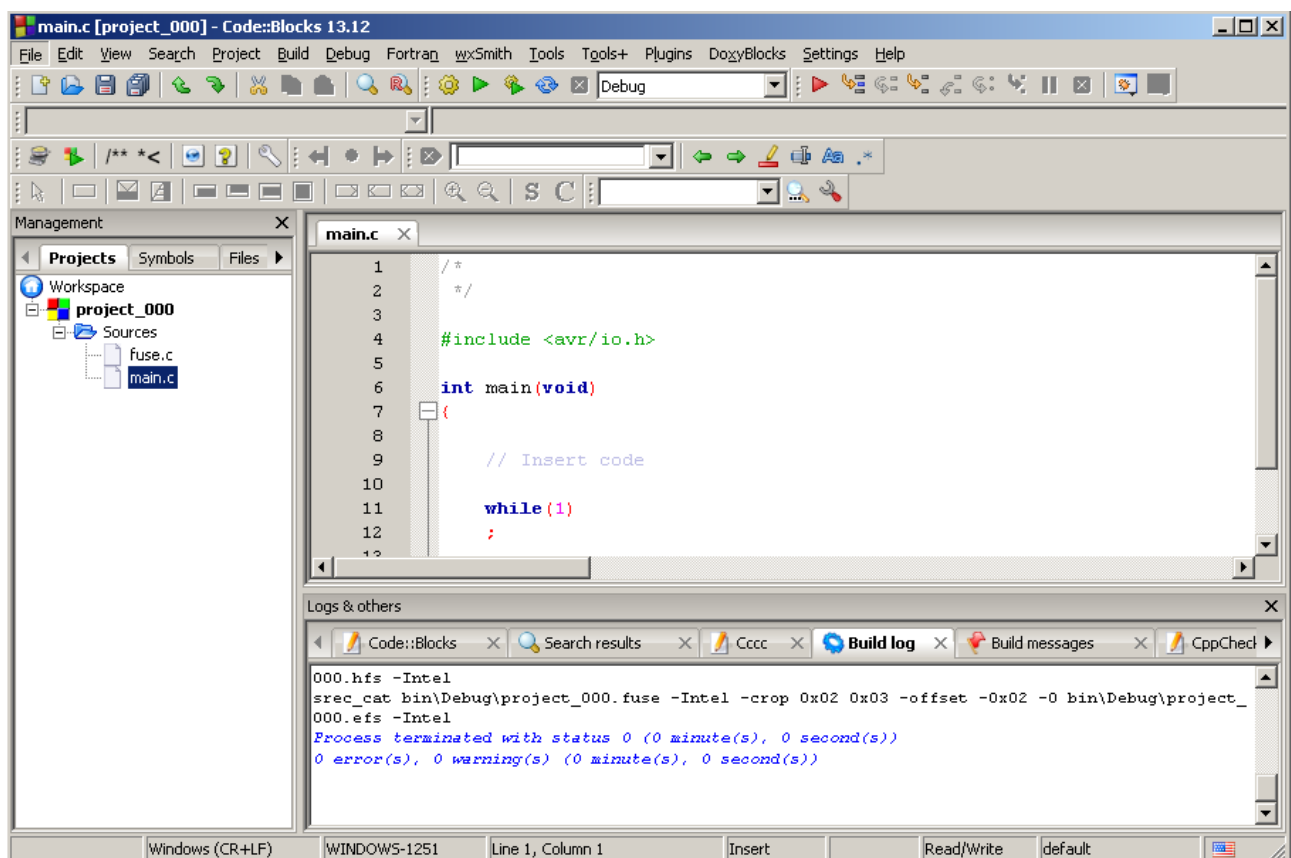


Рис. 2.в — Результат пробної компіляції проекту

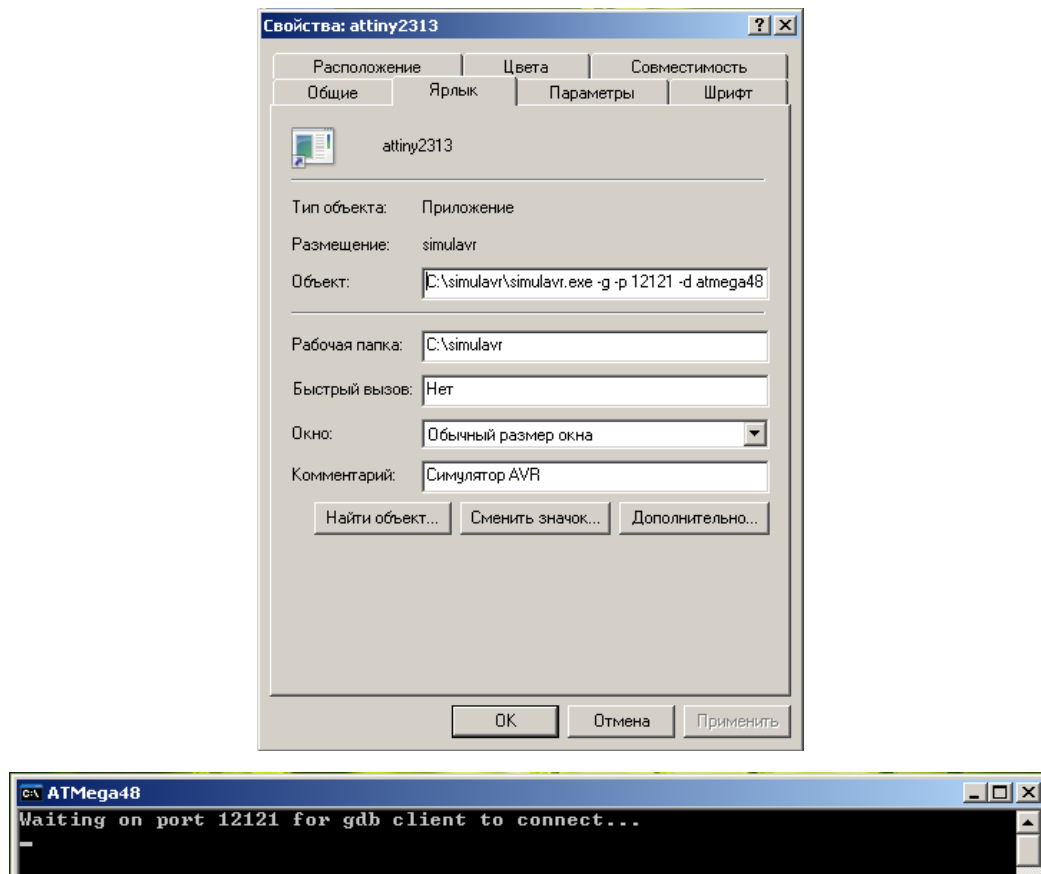



Рис. 3 — Властивості ярлика для завантаження програмного симулятора процесора

4. Для повноцінної розробки програмного забезпечення потрібно мати можливість покрокового виконання коду з моніторингом змін змінних. Це можна досягти використавши симулятор та відладчик-дебагер. На рис. 3. показано налаштування ярлику для завантаження програми симулятора, який має налаштування для процесору ATmega48 з інтерфейсом на 12121 порту мережі. Запустивши програму, ви повинні побачити текстове вікно з повідомленням, що симулятор чекає на з'єднання.

5. У властивостях проекту необхідно зазначити, що налагодження буде йти по TCP протоколу через порт 12121. Необхідні налаштування показані на рис. 4(а, б). Після цього можна починати відлагодження за допомогою . В процесі відлагодження можна відслідковувати зміни регістрів процесора, його пам'яті та окремих змінних.

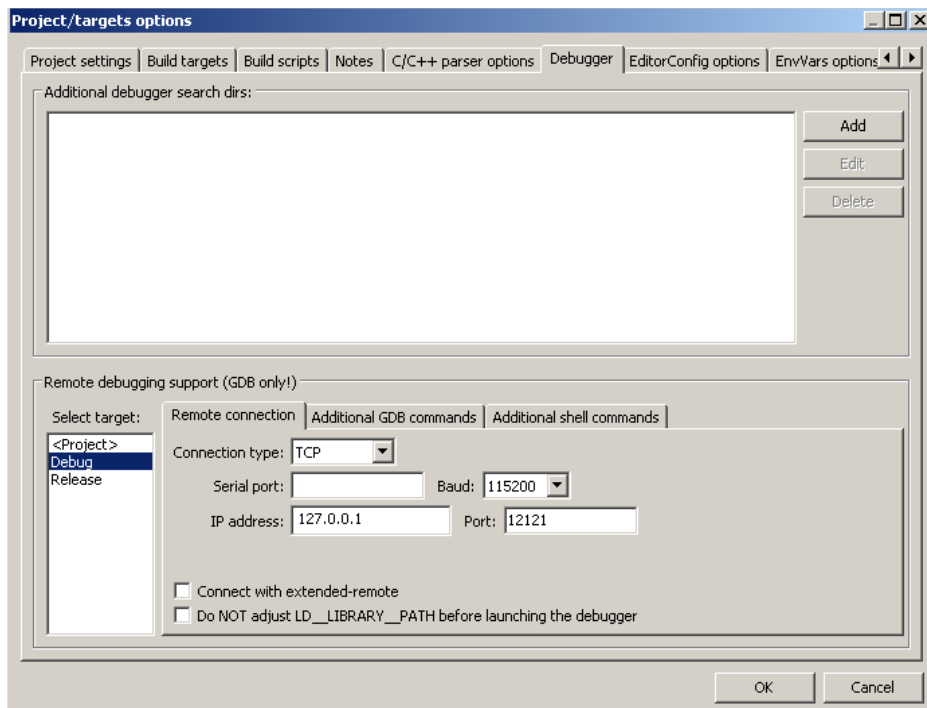


Рис. 4.а — Налаштування проекту для використання відладки проекту на симуляторі

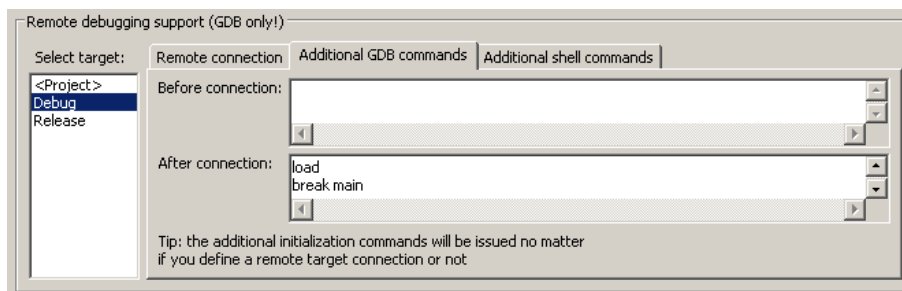


Рис. 4.б — Налаштування завантаження коду в симулятор