

Микропроцессорные Устройства

Отладочная плата VX MEGA-128

Руководство к выполнению лабораторных работ



Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)
Кафедра компьютерных систем в управлении и проектировании (КСУП)

УТВЕРЖДАЮ
Зав каф. КСУП
____ Ю.А. Шурыгин

Отладочная плата VX MEGA-128

Руководство к выполнению лабораторных работ
для бакалавров направления 27.03.04 Управление в технических системах
направленности Управление в роботехнических системах

Разработчики:
доцент каф. КСУП
____ В.П. Коцубинский
Инженер каф. КУСП
____ А.А. Изюмов
доцент каф. КСУП
____ В.М. Рулевский

Томск, 2018

1. Описание стенда

1.1. Микроконтроллер

IE-VX-MEGA128 – отладочный набор (см.рис.1) для разработки систем на базе микроконтроллера ATmega128. Плата оснащена 3-выводными разъёмами стандарта INEX, позволяющих подключать к ней различные периферийные модули производства Innovative Experiment, включая такие, как: драйвера моторов, приёмники диапазона 433МГц, ультразвуковые датчики дистанции, инфракрасные приёмники и приёмопередатчики.

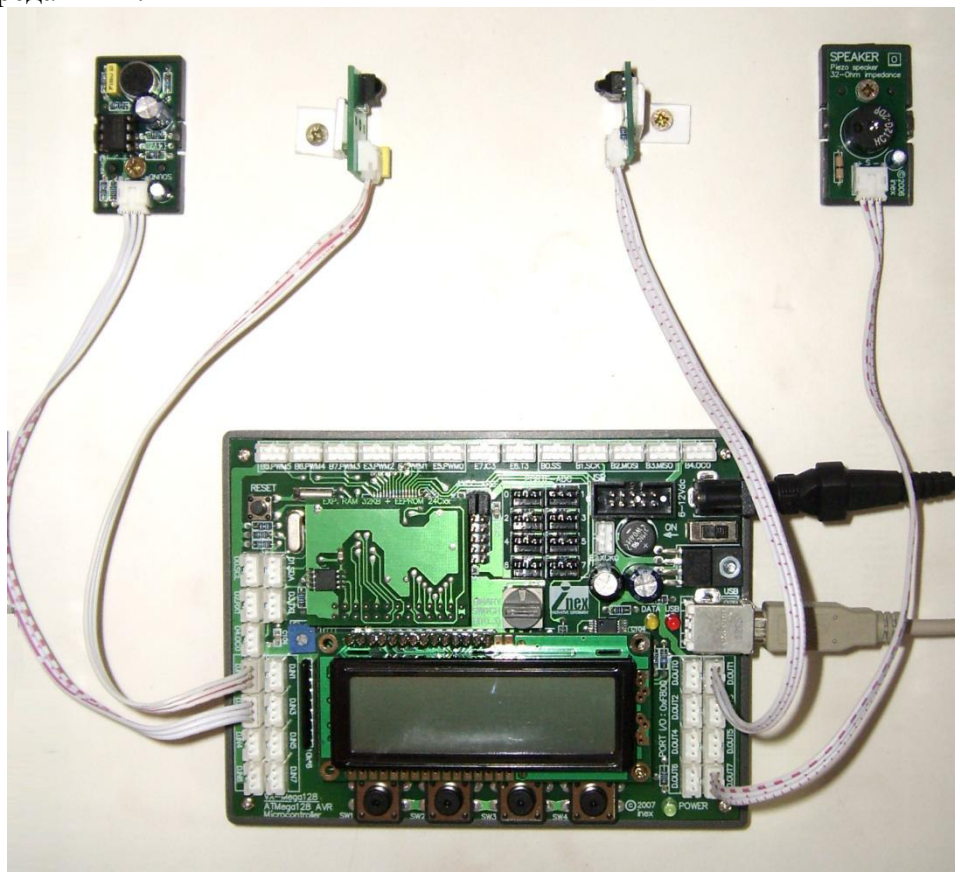


Рисунок 1 – Внешний вид отладочного набора

Отличительные особенности:

- установленный микроконтроллер ATMEGA128: 128кБ Flash-памяти программ, 64кБ ОЗУ, 4кБ EEPROM;
- все порты микроконтроллера доступны на трёхвыводных разъёмах (стандарт INEX);
- цепи питания;
- ISP разъём;
- USB-UART мост;
- 4 кнопки пользователя;
- символьный двухстрочный ЖКИ;
- 3-выводные разъёмы стандарта INEX для подключения различной периферии
- Plug-In модуль с установленной 32кБ RAM и 32кБ EEPROM.

Микроконтроллер ATMEGA128:

- тактовая частота до 16 МГц, время выполнения команды 62,5 нс;
- встроенный программируемый RC-генератор, частота 1, 2, 4, 8 МГц;
- Flash-ПЗУ программ, программируемое в системе, до 128 кбайт (10 000 циклов перезаписи);

- электрически перепрограммируемое ПЗУ данных до 4 кбайт (100 000 циклов перезаписи);
- внутреннее ОЗУ со временем доступа 1 такт, до 4 кбайт;
- мощный набор команд (более 120 инструкций);
- 6 аппаратных команд умножения (для семейства mega);
- развитая система адресации, оптимизированная для работы с C-компиляторами;
- 32 регистра общего назначения (аккумулятора);
- синхронный (USART) или асинхронный (UART) последовательные порты;
- синхронный последовательный порт (SPI);
- двухпроводный интерфейс TWI, совместимый с интерфейсом I2C;
- многоканальный 8-, 9-, 10-, 16-битный ШИМ-модулятор;
- 10-битный АЦП со временем преобразования 70 мкс и дифференциальными входами;
- программируемый коэффициент усиления — 1; 10 или 200;
- встроенный источник опорного напряжения 2,56 В;
- аналоговый компаратор;
- сторожевой таймер;
- настраиваемая схема задержки запуска после подачи питания;
- схема слежения за уровнем напряжения питания;
- JTAG-интерфейс для подключения эмулятора (в микросхемах с объемом ПЗУ от 16 кбайт);
- микромощное потребление (менее 100 мкА в активном режиме на частоте 32 кГц).

Питание от блока питания 12V/500mA.

1.2. Лабораторный стенд

Подготовка специалистов для проектирования и эксплуатации автоматизированных систем измерений требует организации лабораторных практикумов, позволяющих изучать компоненты этих систем, приобретать соответствующие практические навыки. Лабораторный стенд на базе микроконтроллера ATMEGA128 создан для проведения лабораторных работ у специальностей «Управление и информатика в технических системах» и «Управление в робототехнических системах». Функциональная схема стенда приведена на рисунке 2.

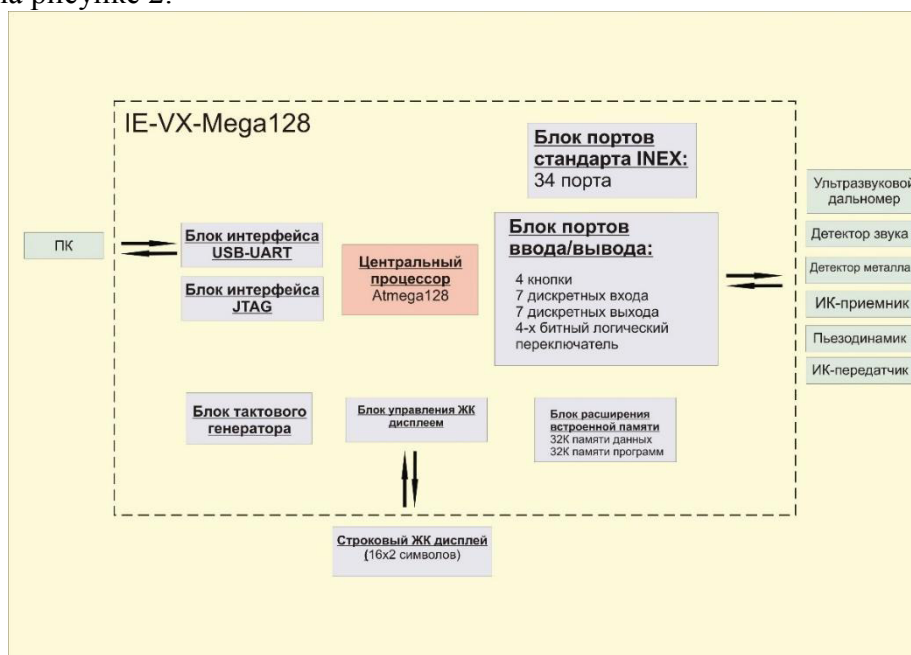


Рисунок 2 – Функциональная схема стенда

В составе стенда участвуют следующие функциональные блоки и элементы:

Функциональные элементы платы IE-VX-Mega128:

1. Блок 3-выводных разъемов стандарта INEX.
2. Блок дискретных входов (8 штук).
3. Блок дискретных выходов (8 штук).
4. Блок аналоговых входов (8 штук).
5. Блок кнопок управления (4 штуки).
6. Логический переключатель
7. Строковый 2 строчный ЖКИ дисплей.
8. Модуль расширения памяти (32 kB)
9. Кнопка Reset.
10. Разъем питания (12Вольт).
11. Переключатель питания платы.
12. Разъем USB.

Датчики и исполнительные механизмы (в зависимости от комплектации стенда):

13. Ультразвуковой дальномер.
14. Детектор звука.
15. Металлодетектор.
16. Инфракрасный приемник.
17. Инфракрасный передатчик.
18. Пьезодинамик.

2. Описание программной части

AVR Studio – интегрированная среда разработки (IDE) для разработки 8-ми и 32-х битных AVR приложений от компании Atmel, работающая в операционных системах Windows NT/2000/XP/Vista/7. AVR Studio содержит компилятор и симулятор, позволяющий отследить выполнение программы. Также среда разработки поддерживает различные вспомогательные программы – плагины.

Пакет WinAVR включает собственный редактор (Programmers Notepad) для написания исходных текстов программ и ряд утилит, облегчающих этот процесс. Однако значительно более удобным способом следует признать интеграцию WinAVR в бесплатную среду разработки программ для микроконтроллеров AVR фирмы Atmel - AVR Studio, самую свежую версию которой всегда можно скачать с сайта Atmel (<http://www.microchip.com/avr-support/atmel-studio-7>). Последние версии среды разработки получили название Atmel Studio.

WinAVR включает следующие компоненты:

- Programmers Notepad - удобный редактор программиста и интегрированная среда разработки (IDE);
- AVR GCC - компилятор языков C и C++ для AVR;
- avr-libc - стандартная C библиотека AVR для использования с GCC;
- avr-as - ассемблер для микроконтроллеров AVR;
- AVRDUDE - программатор (программа для загрузки и выгрузки кода микроконтроллеров);
- avrdude-gui - графический интерфейс пользователя для AVRDUDE (только в версиях до WinAVR-20060421 включительно);
- MFile - автоматический генератор Make-файлов for AVR GCC;
- GNU Binutils - утилиты для AVR (поддерживают преобразование форматов из ELF в AVR COFF или в AVR Extended COFF);
- GNU Debugger (GDB) - дебаггер (отладчик) с интерфейсом командной строки;
- Insight - дебаггер (отладчик) с графическим интерфейсом пользователя;

- AVaRICE (JTAG ICE interface) - программа для интерфейсов Atmel JTAG ICE (используется вместе с GDB);
- SimulAVR - симулятор GDB с поддержкой симулятора от AVR;
- SRecord - коллекция мощных утилит для загрузки файлов разных форматов в EPROM;

Несмотря на все достоинства утилит, входящих в WinAVR, они имеют один весьма существенный недостаток: не поддерживают отладку непосредственно в IDE и слабо интегрируются с аппаратными средствами разработки и отладки, называемых Starter Kit (стартовые комплекты). AVR Studio прекрасно работает со всеми аппаратными средствами Atmel, но поставляется лишь с поддержкой ассемблера. Объединение достоинств обоих пакетов позволяет достичь желаемого результата: разработка и интерактивная отладка с поддержкой всех средств Atmel на языке высокого уровня C++.

3. Изучение отладочного устройства и выполнение простых программ. Лабораторная работа №1.

Цель работы

Необходимо изучить особенности работы цифровой панели на базе МК VX-MEGA128, а также создать программу, работающую с портами ввода-вывода для работы с периферийными устройствами.

Ход работы

Для начала работы с цифровой панелью необходимо выполнить установку необходимого программного обеспечения, а именно:

1. AVR Studio (с сайта Atmel)
2. WinAVR (с сайта <http://winavr.sourceforge.net>)
При установке WinAVR необходимо учесть тот факт, что он должен быть установлен после AVR Studio, иначе IDE не распознает WinAVR как плагин.
3. Библиотека Домашней Лаборатории
Комплект содержит функции, облегчающие использование AVR. Последняя версия библиотеки находится на сайте Домашней Лаборатории (<http://home.roboticlab.eu>). Библиотека должна быть обязательно установлена в том же каталоге, что и WinAVR.
4. Драйвер USB-UART моста (виртуального COM-порта)
Драйвер позволяет подключить к компьютеру программатор JTAG ICE. Драйвер должен быть установлен до подключения к компьютеру программатора. Имя установочной программы 'CDM x.xx.xx.exe», где «x» обозначает версию. Затем необходимо подключить программатор к компьютеру через USB интерфейс, и позволить Windows выполнит автоматическую установку. В зависимости от того, сколько виртуальных последовательных портов на вашем компьютере уже установлено и введено в эксплуатацию, Windows автоматически установит следующий порядковый номер виртуального порта. Новый порядковый номер генерируется и при смене порта USB. Некоторые версии AVR Studio признают программатор ICE JTAG, только в последовательных портах COM1 ... COM9 и функцию отладки позволяют использовать только последовательные порты COM1 ... COM4. Пользователь имеет возможность изменить порядковый номер порта, используя инструмент Device Manager.



Рисунок 3 - Подключение эмулятора COM порта

Создание проекта

Для написания AVR программы, необходимо создать новый проект, который обычно включает в себя много различных файлов: программный код, файлы заголовков и скомпилированные программы т.д. Для того чтобы проекты корректно между собой соотносились, необходимо для каждого проекта создавать новый каталог.

Для создания проекта необходимо пройти следующие этапы:

1. Открыть AVR Studio и нажать кнопку «Новый проект» (New Project). Если окно не открывается автоматически, выберите пункт меню Project - New project. Затем нажмите кнопку Next.

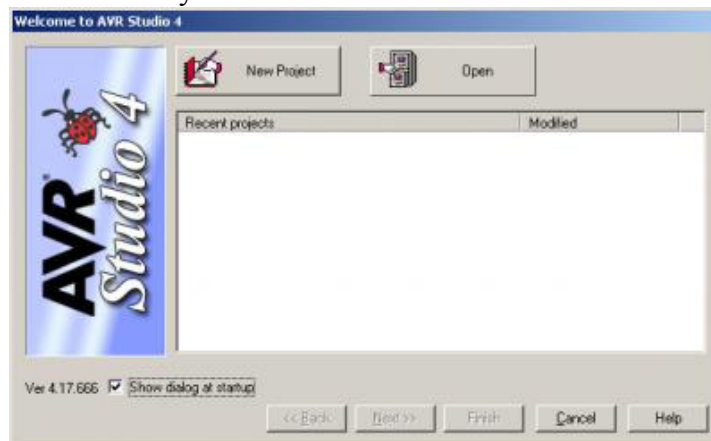


Рисунок 4 – Окно создания проекта

2. Откроется окно, в котором необходимо задать начальные настройки компилятора и файлов. В качестве компилятора выберите AVR GCC, введите имя проекта и имя исходного файла. Имя исходного файла должно обязательно иметь расширение «*.c». Необходимо так же указать директорию для хранения каталога проекта. После выбора нажмите кнопку Next. Если в списке выбора компилятора отсутствует AVR GCC, то WinAVR установлен неправильно. Его установка необходимо осуществить до написания основного кода программы на языке C++.



Рисунок 5 – Окно задания свойств проекта

3. Далее открывается окно, в котором отмечается платформа, используемая для отладки микроконтроллера. В Домашней Лаборатории в качестве платформы

для отладки микроконтроллера ATmega128 используется JTAG (Joint Test Action Group) ICE. Для завершения создания проекта нажмите кнопку Finish.



Рисунок 6 – Окно выбора типа контроллера

4. Далее открывается пользовательский интерфейс среды программирования, где пишется исходный код программы.

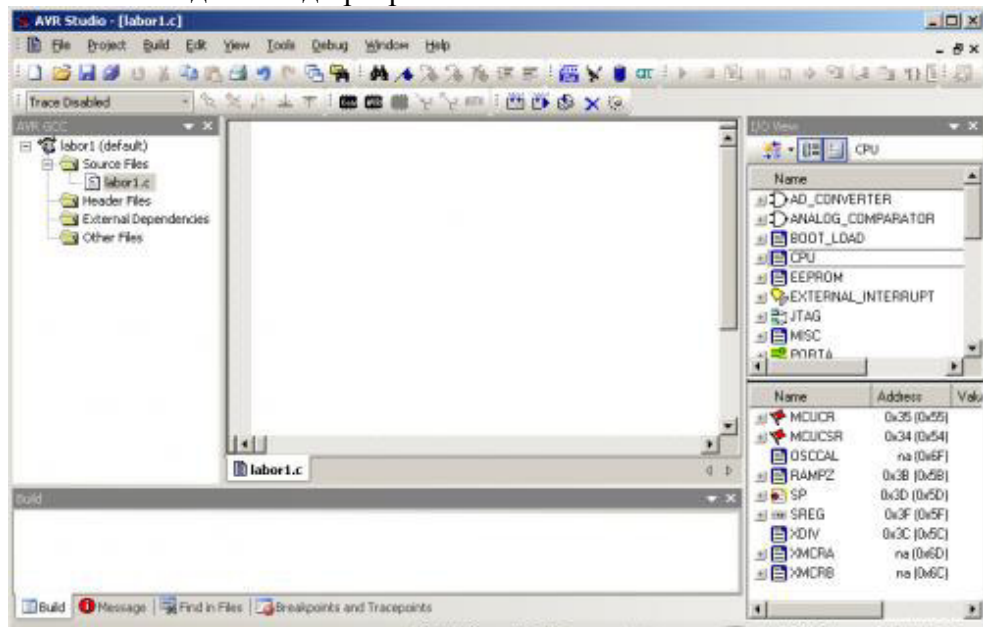


Рисунок 7 – Интерфейс программы

5. Перед компиляцией кода, необходимо определить настройки проекта. Наиболее важными параметрами является тактовая частота контроллера и метод оптимизации компилятора. Тактовая частота контроллера Домашней лаборатории - 14,7456 МГц (14745600 Гц). Частоту необходимо установить в свойствах проекта в Гц (не МГц) в окне Project → Configuration Options → General. Метод оптимизации должен остаться -Os, если нет необходимости использовать другие методы.
6. Для использования домашней лаборатории библиотека функций программного обеспечения должны быть установлены правильно. При каждом проекте необходимо из настроек Project → Configuration Options → Libraries добавить из списка нужную библиотеку проекта.

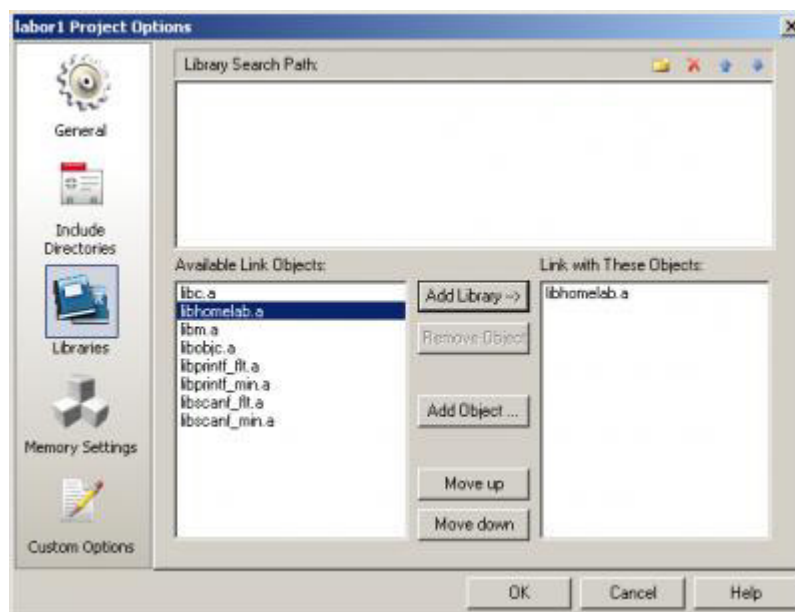


Рисунок 8 – Свойства проекта

Если объект libhomelab.a отсутствует в списке, то библиотека Домашней Лаборатории установлена неправильно.

7. Подключите программатор с платой ATmega128. При подключении проверьте, чтобы программатор был подключен в правильное гнездо (JTAG). После программатора подключите к плате контроллера питания (если источник питания подключен правильно, то должен загореться маленький зеленый светодиод). Также в Диспетчере устройств должно появиться в разделе Порты (COM и LPT) устройство с COM старших адресов, например, 5 или 6.
8. Подключить программатор с платой ATmega128.

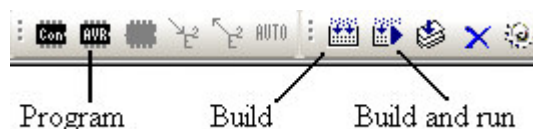


Рисунок 9 – Кнопки программирования

9. Скомпилировать программу приказом Build (клавиша F7). Должно появиться следующее сообщение:

Build succeeded with 0 Warnings...

10. Открыть окно загрузки кода с помощью Tools → Program AVR → Auto Connect. В результате должно открыться окно загрузки скомпилированного файла в микроконтроллер.

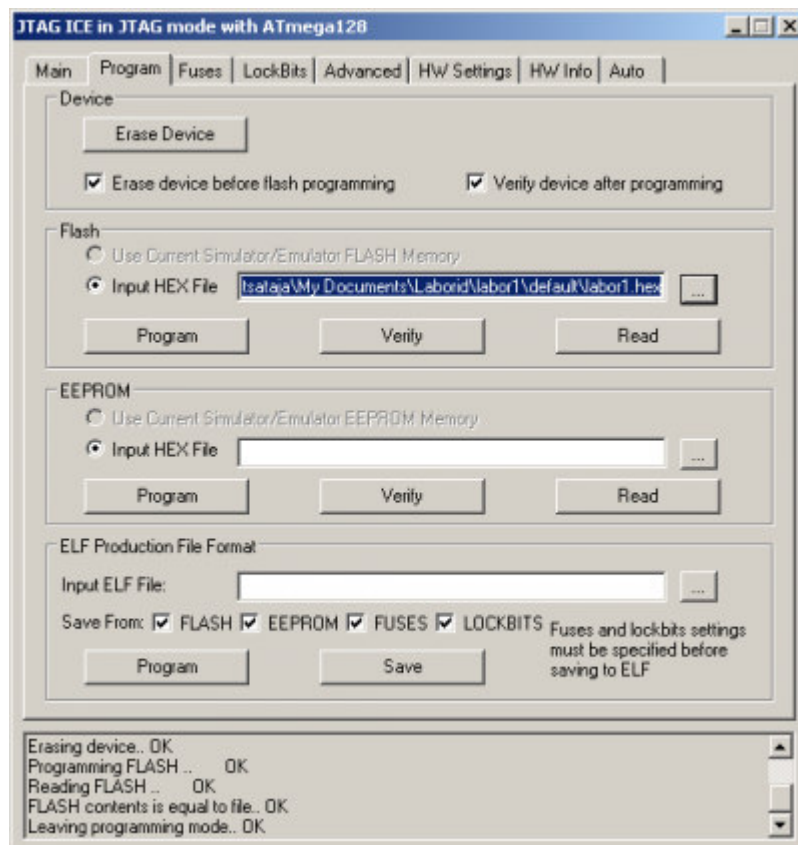


Рисунок 10 – Окно настройки

Обратите внимание на последние 6 строк: если в одной из них не стоит ОК, то программа не скомпилировала и не запрограммировала контроллер. При этом вы должны перевести плату VX-Mega 128 в режим BootLoader, а после, посредством кнопки Reset перевести в режим CODE (см. стр. 14 Приложения 1).

11. В окне программатора ввести в разделе Flash в текстовое поле Input HEX File путь к скомпированному файлу. Скомпированный файл обычно находится в подкаталоге проекта default и имеет такое же имя, как и проект, но с расширением *.hex, к примеру, «labor1.hex». После выбора правильного файла, нажмите на кнопку Program, который загрузит выбранную программу в контроллер. Если все удалось, то в нижней части окна среды программирования появится следующее сообщение:

```

OK
Reading FLASH input file.. OK
Setting device parameters for jtag programming ..OK
Entering programming mode.. OK
Erasing device.. OK
Programming FLASH ..    OK
Reading FLASH ..    OK
FLASH contents is equal to file.. OK
Leaving programming mode.. OK

```

Задание на лабораторную работу №1:

- 1) Подключить макет к компьютеру
- 2) Отладить и залить в память микроконтроллера программу для работы дальномера. Проверить правильность выполнения программы.

```

/*Для правильной работы программы необходимо указать все
нижеперечисленные директивы.*/
#define F_CPU 1000000UL
#include <inttypes.h>
#include <avr/io.h>
#include <util/delay.h>
#include <stdbool.h>
int GetDistance()
{
    DDRD &= ~(1<<3);
    int dist = 0;
    PORTB = 0x04;
    _delay_ms(2);
    PORTB = 0x02;
    _delay_ms(10);
    PORTB = 0x04;
    _delay_ms(2);
    PORTD |= (1<<3);
    dist = PIND;
    return dist;
}

int main()
{
    DDRB = 0xFF;
    DDRF = 0xFF;

    while(1)
    {
        if(getDistance() != 0x9F)
        {
            PORTF = 0x02;
            _delay_ms(500);
            PORTF = 0x04;
            _delay_ms(500);
        }
    }
}

```

- 3) Изменить данные программы и посмотреть, как будет вести себя микроконтроллер.

Требования к оформлению отчета

Отчет должен содержать структурную схему платы VX-Mega 128, описания результатов пунктов выполненного задания, полный текст программы с комментариями, а также общие выводы по работе.

4. Контрольные вопросы к лабораторной работе №1

- 1) Какой из типов датчиков (и исполнительных механизмов) не поставляются с лабораторным стендом IE-VX-Mega128?
 - а) Ультразвуковой дальномер

- б) Детектор звука
 - в) Металлодетектор
 - г) Инфракрасный приемник
 - д) Датчик давления
 - е) Инфракрасный передатчик
 - ж) Пьезодинамик
- 2) Расшифруйте аббревиатуру JTAG:
- а) Jack Test Action Group
 - б) Joint Test Action Group
 - в) Joint Thermal Action Grid
 - г) Jack Test And Grow
- 3) Как называется среда разработки, разработанная специалистами компании Atmel?
- а) VisualDSP
 - б) Microsoft Visual Studio
 - в) AVR Studio
 - г) WinRar

5. Ввод/вывод. Обращение к подпрограммам. Лабораторная работа №2.

Цель работы

Закрепление навыков работы с отладочным набором – подключения периферийных устройств и программирования микроконтроллера.

Ход работы

1) Описание интерфейсов VX-Mega 128

Плата отладочного набора ATmega128 оснащена 3-выводными разъёмами стандарта INEX, позволяющих подключать к ней различные периферийные модули производства Innovative Experiment: драйвера моторов, приёмники диапазона 433МГц, ультразвуковые датчики дистанции, инфракрасные приёмники и приёмопередатчики.

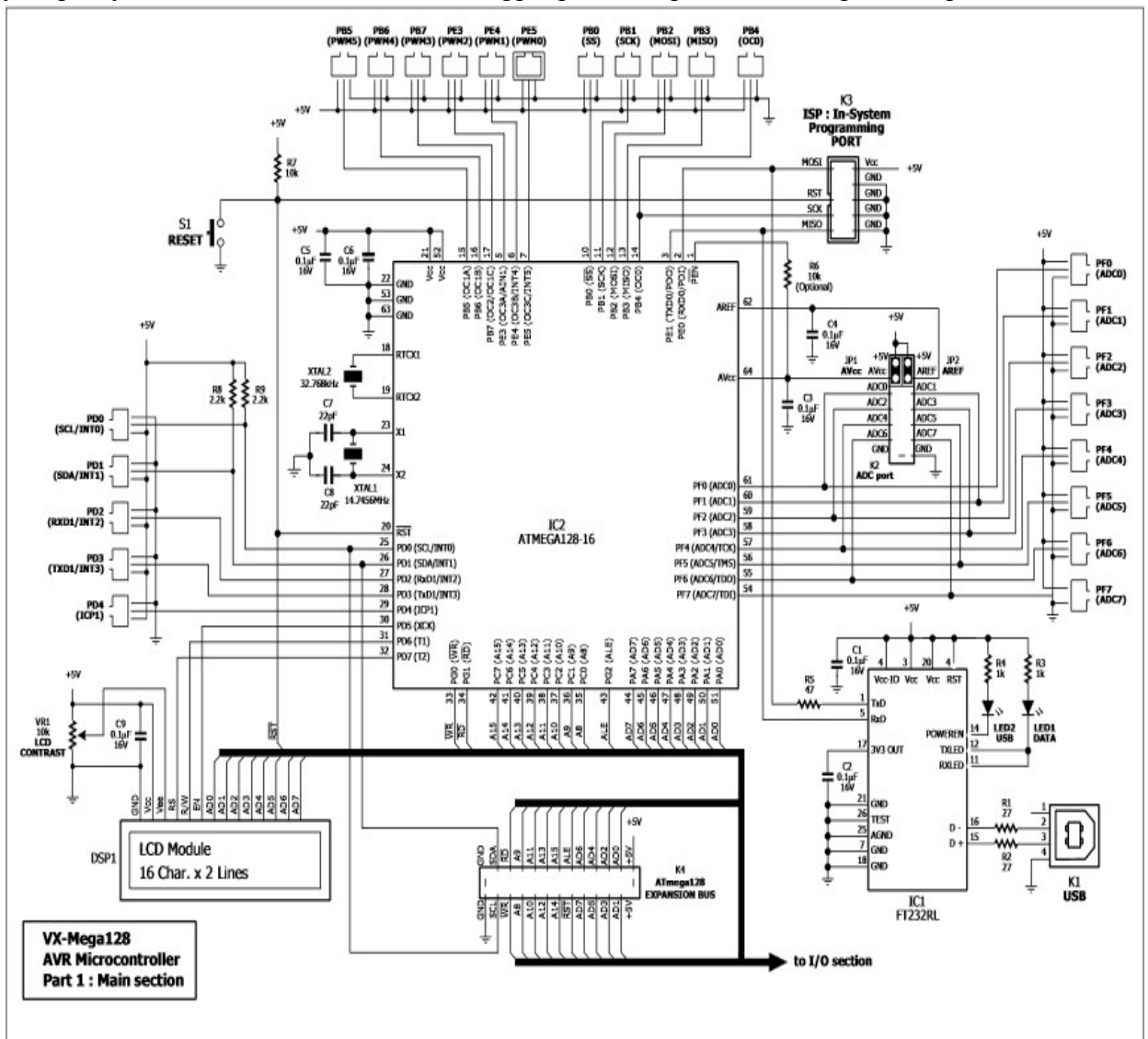


Рисунок 11 – Схема цифровой панели VX-Mega 128

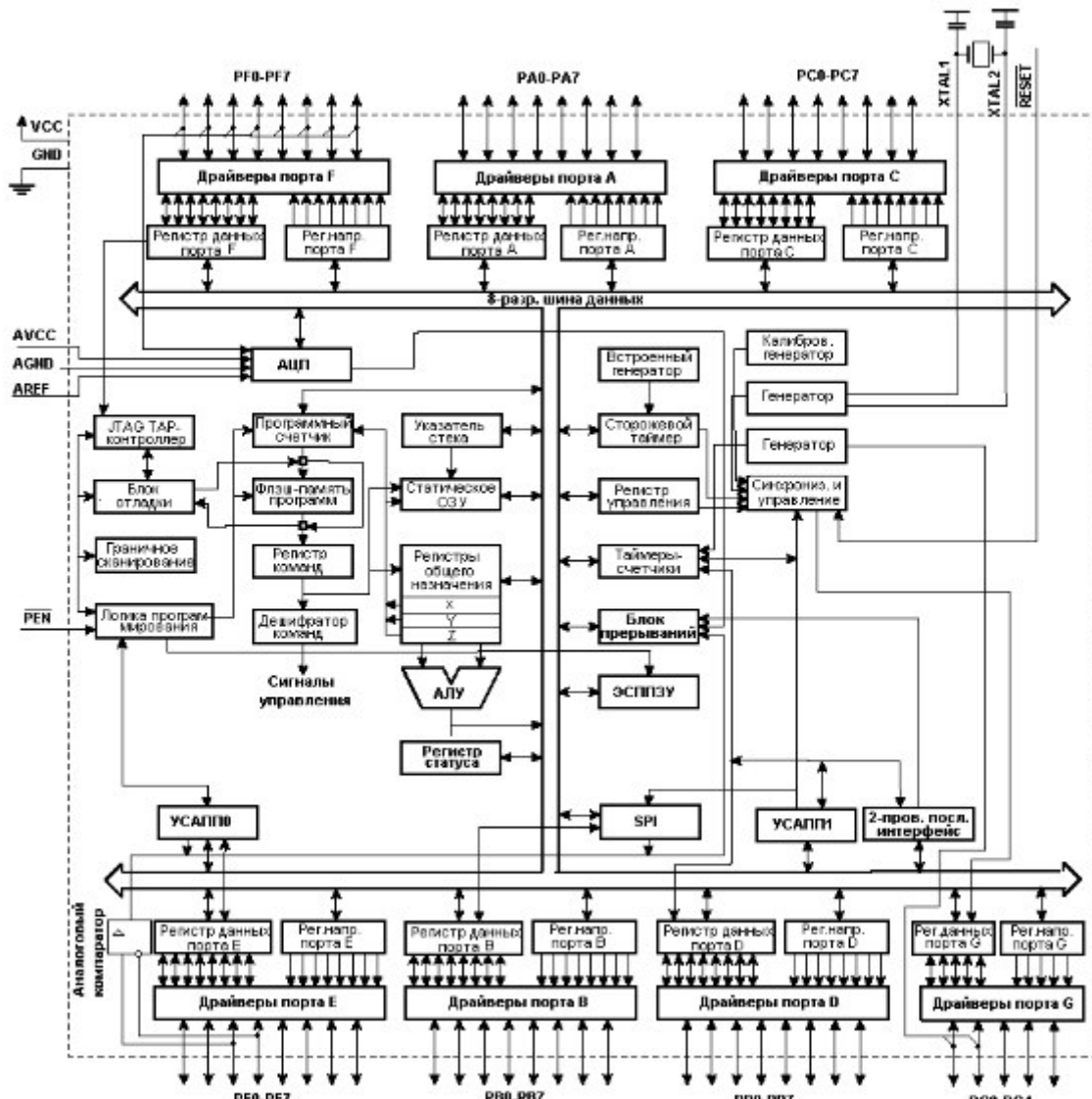


Рисунок 13 – Функциональная схема внутренней структуры ATmega 128

2) Металлодетектор ZX-METALL

Расстояние детектирования зависит от размеров металлического объекта. Центральным узлом модуля сенсора является ИМС TDA0161. Монолитные интегральные схемы предназначены для нахождения любого металлического тела, обнаруживая вариации высокочастотных потерь на вихревые токи. С внешним контуром они выступают в качестве генераторов. Уровень выходного сигнала изменяется при приближении металлического объекта. Выходной сигнал определяется текущими изменениями питания. Независимо от источника напряжения, этот ток высокий или низкий в зависимости от наличия или отсутствия металлического предмета.

Спецификация:

- построен на базе ИМС TDA0161
- регулировка чувствительности; цифровой выход, активный уровень «0»;
- светодиод состояния;
- напряжение питания +5В.

PIN CONNECTION (top view)

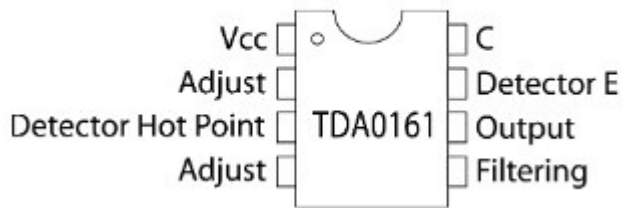


Рисунок 14 – Схема выводов на плате (вид сверху)

3) Ультразвуковой дальномер

Ультразвуковой датчик предназначен для определения расстояния до препятствий в диапазоне от 0,03-4 метра.

Спецификация:

- Питание 5В, 15 мА
- Частота передатчика 40 кГц
- Максимальная дальность 400 см
- Минимальный диапазон 3 см
- Размеры 43x20x15 мм

4) Микрофон

Печатная плата с электретным микрофоном и 100х усилителем.

Спецификация:

- Питание 2,7-5,5 В
- Размеры 12x18 мм



Рисунок 15 – Схема выводов на плате микрофона.

Задание на лабораторную работу №2

- 1) Подключить макет к компьютеру
- 2) На основании нижеследующего примера написать программу для работы дальномера. При достижении расстояния, равного/большого/меньшего выбранному – подать сигнал на пьезодинамик.

Программа 1:

*/*Для правильной работы программы необходимо указать все нижеперечисленные директивы.*/*

```
#define F_CPU 1000000UL
```

```
#include <inttypes.h>
```

```
#include <avr/io.h>
```

```
#include <util/delay.h>
```

```
#include <stdbool.h>
```

//Пример функции для калибровки датчика звука.

```
int getDistance()
```

```
{
```

```
    DDRD &= ~(1<<3);
```

```
    int dist = 0;
```

```

        PORTB = 0x04;
        _delay_ms(2);
        PORTB = 0x02;
        _delay_ms(10);
        PORTB = 0x04;
        _delay_ms(2);
        PORTD |= (1<<3);
        dist = PIND;
        return dist;
}

```

На выходе функции возвращается значение дистанции в hex коде. Например, для 159мм значение функции будет 0x9F.

Пример кода для вывода сигнала на порт динамика:

```

int main()
{
    DDRB = 0xFF;
    DDRF = 0xFF;

    PORTF = 0x02;
    _delay_ms(500);
    PORTF = 0x04;
    _delay_ms(500);
}

```

Для прерывной работы необходимо использовать цикл while. Пример:

```

while(1)
{
    Print_display("It Works");
}

```

3) Изменить данные программы и посмотреть, как будет вести себя микроконтроллер.

Пример программы:

```

/*Для правильной работы программы необходимо указать все
нижеперечисленные директивы.*/
#define F_CPU 1000000UL
#include <inttypes.h>
#include <avr/io.h>
#include <util/delay.h>
#include <stdbool.h>
//Пример функции для калибровки датчика звука.
int getDistance()
{

    DDRD &= ~(1<<3);
    int dist = 0;
    PORTB = 0x04;
    _delay_ms(2);
    PORTB = 0x02;
    _delay_ms(10);
    PORTB = 0x04;
}

```

```

        _delay_ms(2);
        PORTD |= (1<<3);
        dist = PIND;
        return dist;
}

```

- 4) Разработать и отладить программу 2, позволяющую при приближении к металлодетектору металлического предмета увеличивать громкость издаваемую пьезодинамиком:

Программа 2:

```

define F_CPU 100000UL //тактовая частота микроконтроллера
#include <inttypes.h> //расширение для целочисленных типов
#include <avr/io.h> //доступ к портам
#include <util/delay.h> //задержка
#include <stdbool.h> //логические переменные

int Frequency()
{
    DDRD &= ~(1<<3); //настройка порта D на вывод
    int dist = 0;
    PORTB = 0x04; // высокий логический уровень на одной из
ножек порта B
    _delay_ms(10); //задержка 10 мс
    PORTD |= (1<<3); //высокий логический уровень на одной из
ножек порта D
    dist = PIND; //чтение и запись в dist состояния лог. уровня
порта D
    return dist;
}
int main()
{
    DDRB = 0xFF; //настройка порта B на вывод
    DDRF = 0xFF; //настройка порта F на вывод

    while (1) //бесконечный цикл

    {
        if (Frequency() != 0x9F) //условие, если
Frequency() не равно 0x9F то
        {
            PORTF = 0x02; //высокий лог. уровень на одной из ножек порта
F
            _delay_ms(500); //задержка 500 мс
            PORTF = 0x04; //высокий лог. уровень на одной из ножек порта
F
            _delay_ms(500); //задержка 500 мс
        }
    }
}

```

- 5) Разработать и отладить программу 3, позволяющую при приближении к дальномеру любого предмета увеличивать громкость издаваемую пьезодинамиком:

```

#define F_CPU 100000UL //тактовая частота микроконтроллера
#include <inttypes.h> //расширение для целочисленных типов

```

```

#include <avr/io.h> //доступ к портам
#include <util/delay.h> //задержка
#include <stdbool.h> //логические переменные

int getFrequency()
{
    DDRD &= ~(1<<3); //настройка порта D на вывод
    int dist = 0;
    PORTB = 0x04; // высокий логический уровень на одной из
ножек порта B
    _delay_ms(10); //задержка 10 мс
    PORTD |= (1<<3); //высокий логический уровень на одной
из ножек порта D
    dist = PIND; //чтение и запись в dist состояния лог.
уровня порта D
    return dist;
}

int main()
{
    DDRB = 0xFF; //настройка порта B на вывод
    DDRF = 0xFF; //настройка порта F на вывод
    while(1) //бесконечный цикл
    {
        if(getFrequency() != 0x9F) //условие, если Frequency()
не равно 0x9F то
        {
            for(int i = 0; i<10; i++) //цикл с 10 итерациями
            {
                PORTF = 0x02; //высокий лог. уровень на одной
из ножек порта F
                _delay_ms(500); //задержка 500 мс
                PORTF = 0x04; //высокий лог. уровень на одной
из ножек порта F
                _delay_ms(1000); //задержка 1000 мс
            }
        }
    }
}

```

- б) Самостоятельно разработать и отладить программу 4, позволяющую по хлопку выключать пьезодинамик.

Требования к оформлению отчета

Отчет должен содержать результаты выполнения пунктов работы (недостающие описания и фотографии Микрофона, Дальномера, Металлодетектора, Инфрокрасного датчика), тексты программ 1, 2, 3, 4, а также общие выводы по работе, в которых необходимо отразить в табличном виде используемые при подключении датчиков порты.

6. Контрольные вопросы к лабораторной работе №2

1. Уровни питания периферийных устройств?
 - а) 2-10В

- б) 110-220В
- в) 16-32В
- 2. На чем построен принцип обнаружения металлических предметов в монолитных интегральных схемах?
 - а) Обнаружение содержания меди
 - б) Обнаружение вариаций высокочастотных потерь на вихревые токи
 - в) Обнаружение падения напряжения при приближении проводника
- 3. При программировании микроконтроллера ATmega128 используется следующая система счисления:
 - г) Десятичная
 - а) Двоичная
 - б) Шестнадцатеричная
 - в) Восьмеричная
- 4. Для чего предназначен пьезодинамик?

ЛИТЕРАТУРА

1. Афонин Ю., Шарнин Л., Баран Е., Липницкий А., Лысов Н., Маслов А., Федоряк Р. Микропроцессорная техника для вузов// Современные технологии автоматизации. — 2001. — № 3.
2. Баран Е., Захаров П., Любенко А., Web-лаборатория «Микроконтроллеры и сигнальные процессоры» // СТА— 2005. — № 1. стр. 64-68
3. Шарапов, А. В. Основы микропроцессорной техники: Учебное пособие [Электронный ресурс] / А. В. Шарапов — Томск: ТУСУР, 2008. — 240 с. — Режим доступа: <https://edu.tusur.ru/publications/834> (дата обращения 05.05.2018)

СОДЕРЖАНИЕ

1. Описание стенда.....	3
1.1. Микроконтроллер	3
1.2. Лабораторный стенд	4
2. Описание программной части.....	5
3. Изучение отладочного устройства и выполнение простых программ. Лабораторная работа №1	6
4. Контрольные вопросы к лабораторной работе №1	11
5. Ввод/вывод. Обращение к подпрограммам. Лабораторная работа №2	13
6. Контрольные вопросы к лабораторной работе №2.....	19
Литература	20
Приложение 1. Описание отладочной платы VX-Mega128	21

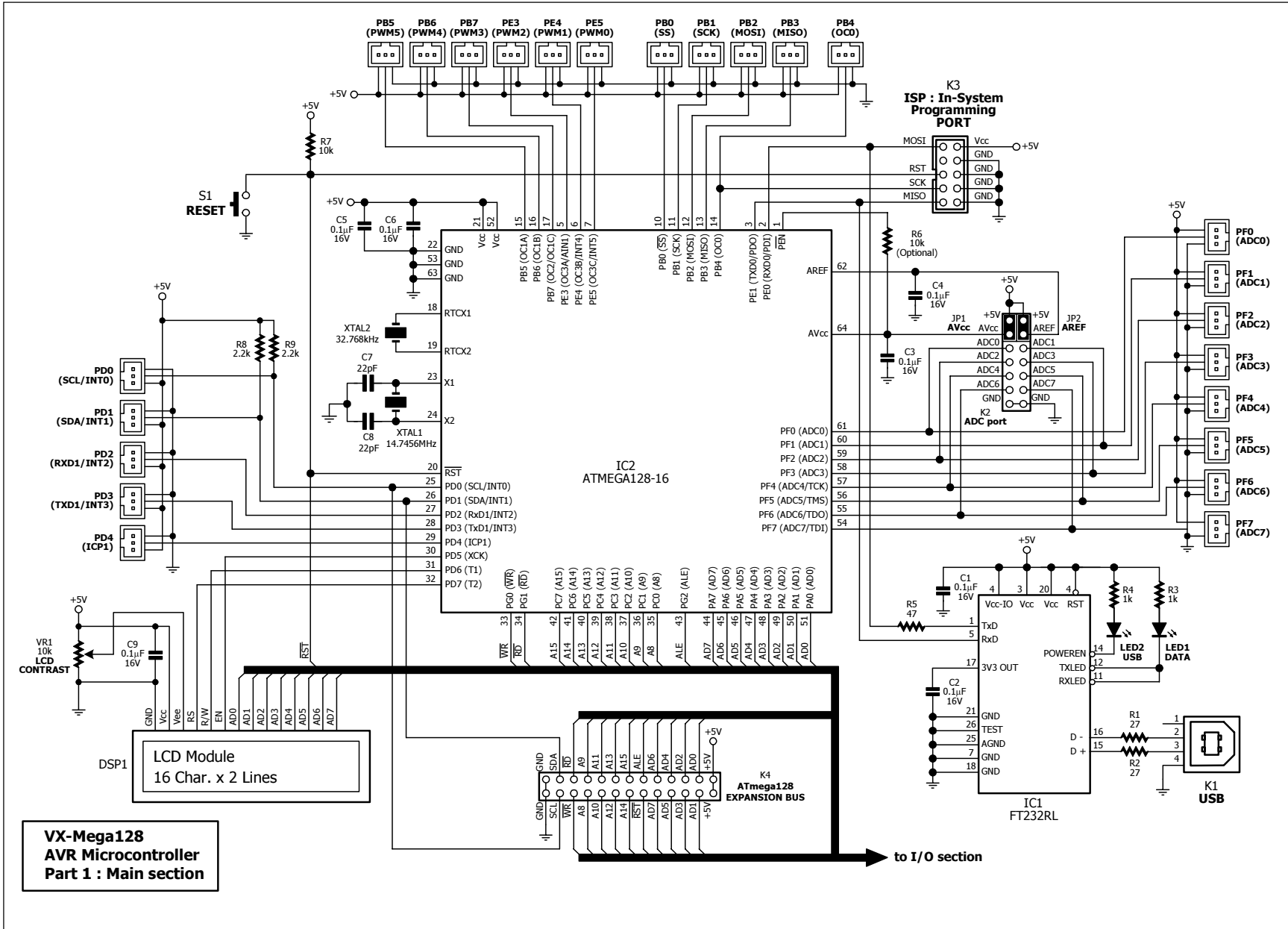
VX-Mega128

ATmega128 Microcontroller board

1. Features

- Support the ATmega128 with Flash memory of 128KB. Clock rate 14.7456MHz
- 34 I/O port of microcontroller are using INEX's standard PCB-3 pin connector.
- LCD16x2 module
- 4 of Button switches
- Binary Switch
- USB port with FT232RL USB to Serial converter circuit.
- ATmega128 pre-programmed Boot loader, programmable via USB port.
- 8-ch. Analog input with 10-bit Analog to Digital Converter
- 32KB RAM / 32KB EEPROM memory module plug on top
- Support In-system Programming via ISP connector with PX-400 Programmer
- Supply voltage +9V from external. +5V 500mA on-board regulator.

Includes : ATmega128 Microcontroller board, Plugable Memory module, USB cable, CD-ROM and Documentation.



VX-Mega128 AVR Microcontroller Part 1 : Main section

Figure 1 Microcontroller scetioh schematiuc diagram of VX-Mega128 board

2. Circuit and Memory map

The VX-Mega128 microcontroller board schematic diagram is splitted to 3 sections. The microcontroller section schematic diagram is shown in figure 1. Input/Output port and external device interface schematic diagram is shown in figure 2 and figure 3 shows the circuit of Memory extension module.

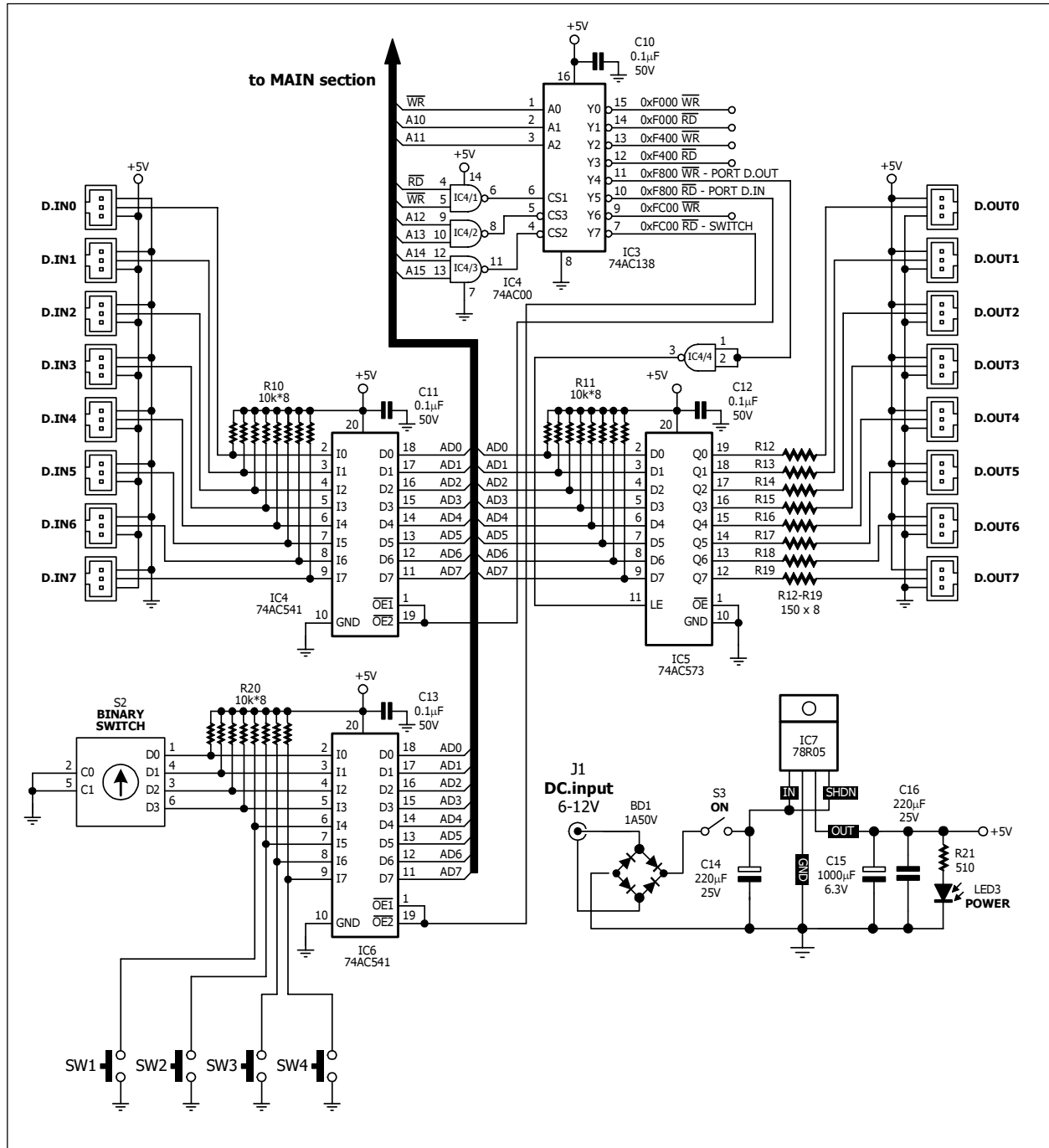


Figure 2 I/O port and external device interface schematic diagram of VX-Mega128 microcontroller board

4 ● VX-Mega128 : ATmega128 Microcontroller board documentation

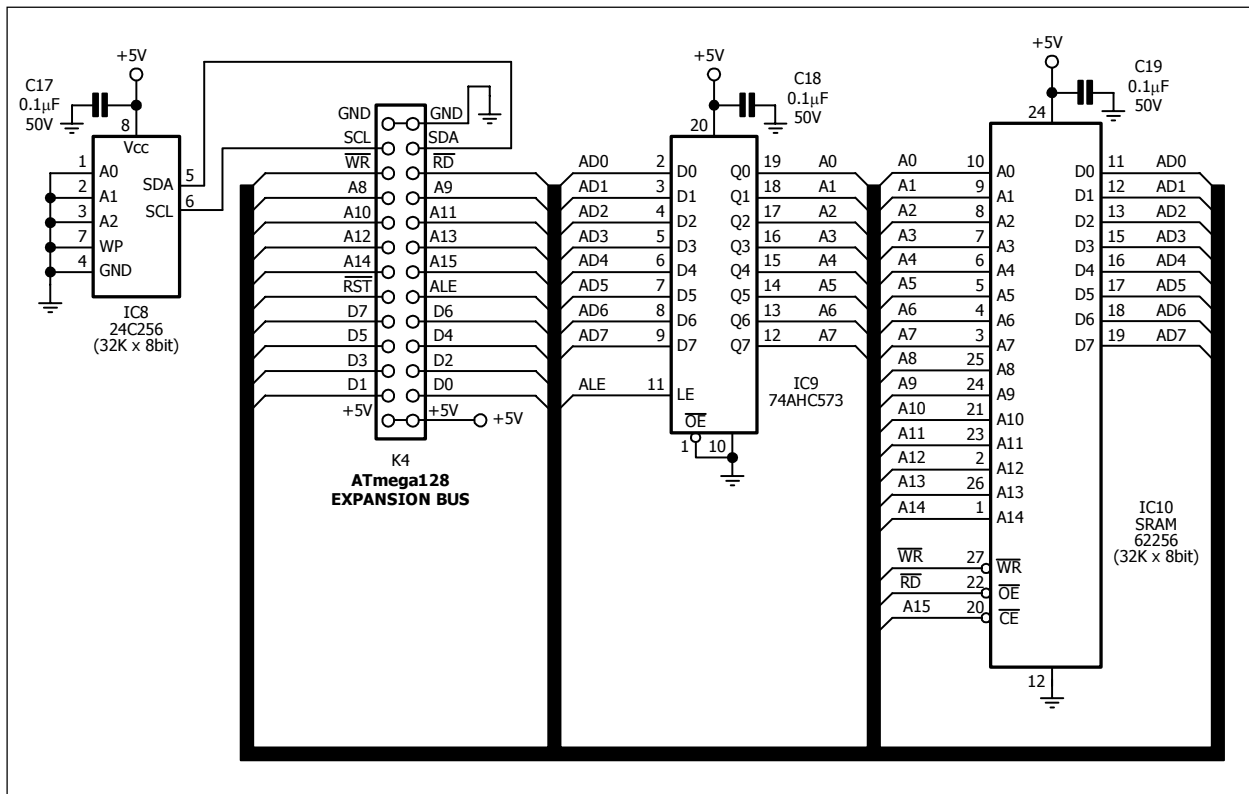


Figure 3 Data memory extension module schematic diagram

ATmega128 operates with clock frequency 14.7546MHz from external crystal circuit. The crystal 32.768kHz is connected to ATmega128 for setting the clock frequency to support internal Real-time counter. Supply voltage +5V comes from external DC adaptor +9V (+16V maximum) via +5V regulator circuit with 78R05 ICs.

VX-Mega128 supports both in Single chip microcontroller mode and Microprocessor mode. ATmega128 interfaces with all external device via address assignment. In figure 4 shows the memory and I/O map of VX-Mega128 board.

External device that connected with ATmega128 are assigned in address F000H to FFFFH included :

(1) **LCD module 16x2** : connect 8-bit data bus with AD0 to AD7 of ATmega128. It causes reading and writing via data bus instead direct interface.

(2) **4 of Input Switches (SW1 to SW4)** : address at upper nibble of FC00H. Signal is buffered via 74AC541 IC.

(3) **4-bit binary Switch** : this switch gives 4 binary bits. It means 16 different data 0000 to 1111 in binary base or 0 to 15 in decimal and 0 to F in HEX. Data from Binary switch are transferred to data bus ; AD0 to AD3 via 74AC541 buffer IC and assigned at lower nibble of FC00H.

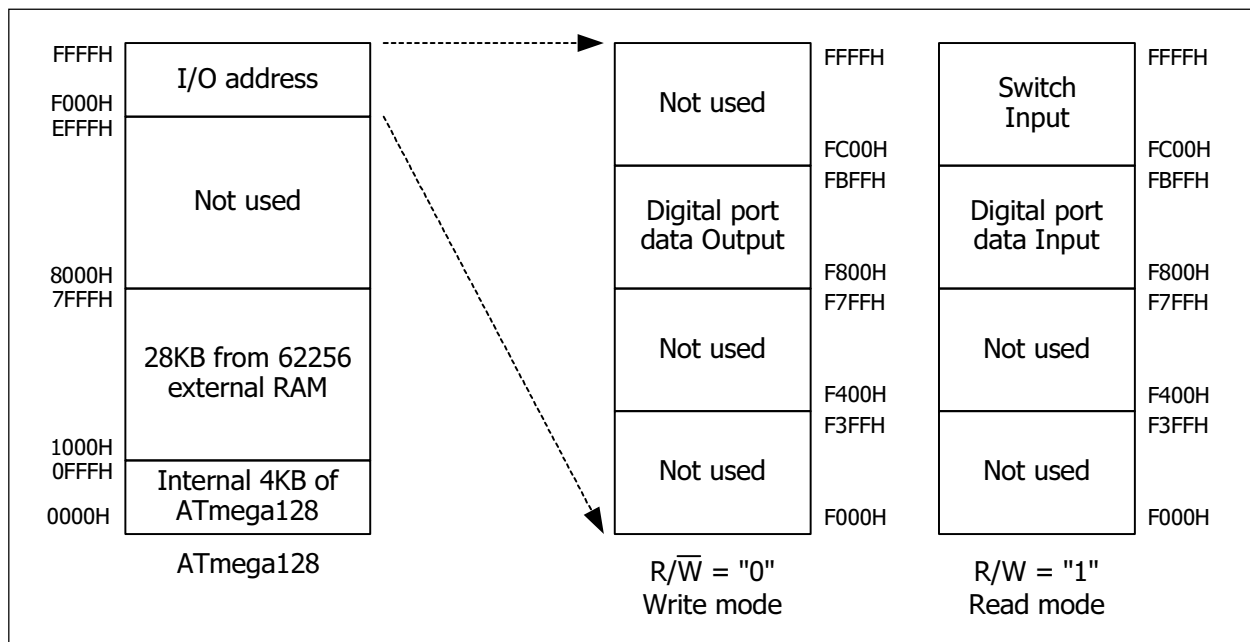


Figure 4 Memory and Input Output port organization of VX-Mega128 microcontroller board

Also the digital input and output port of VX-Mega128 board are assigned via address map system as follows :

Digital input port D.IN0 to D.IN7 are assigned at F800H address.

Digital output port D.OUT0 to D.OUT7 are assigned at F800H address too, but active with \overline{WR} signal instead.

All interface port include Analog to Digital converter module input are connect to INEX 3-pin PCB connector. User can connect with INEX many sensors and modules.

VX-Mega128 board supports 8-channel of Analog to digital converter. User can select supply voltage and reference for ADC module with 2 jumpers JP1 and JP2. If fit all, selects ADC module to use supply and reference voltage at +5V. If remove all, user can apply the voltage at these points. The limitation is reference voltage must lower or equal the supply voltage of ADC module. Normally, fit both jumpers on the board.

Computer interface of VX-Mega128 board via USB. The heart of this function is FT232RL the USB to Serial converter chip. Thus, interface signal is serial and connect to UART1 module in ATmega128. The programming shall be easier and supports all modern computer includes laptop which not serial port available.

Addition user can program microcontroller's flash memory via USB interface. Because ATmega128 microcontroller will be pre-programmed the bootloader ready. It means this microcontroller board can program the flash 2 ways. One is via ISP connector with external In-System Programmer such as PX-400. Another is via USB with bootloader (ATmega128 must pre-programmed ready).

3. How to program the VX-Mega128 board

Flash programming ATmega128 on VX-Mega128 board has 2 methods.

(1) via ISP connector with external In-System Programmer : The suggestion programmer is INEX's PX-400 and Atmel's AVR ISP In-System Programmer

(2) via USB with bootloader : This method ATmega128 must pre-programmed bootloader firmware. INEX prepare ready to using. However user can re-program the bootloader with yourself by external programmer via ISP connector. In CD-ROM contain the HEX file of this bootloader. On VX-Mega128 board has FT232RL USB to Serial converter chip to interface the USB but in software will look as COM port. Signal from FT232RL are connected to SPI programming pin PDI (PE0) and PDO (PE1) of ATmega128. With this method, user no need the external programer to flash programming. Only plug USB cable between board and computer. Open the suitable software. You can work with this microcontroller board.

4. Extension memory board

The memory board of VX-Mega128 is 64KB. Divides 2 parts. One is Static RAM 32KB ; 62256. However user can work only lower 28KB. Because upper 4KB is reserved by internal RAM data memory of ATmega128. Interfacing of this RAM work via the EXPANSION BUS. This bus provides 8-bit data/address lines; AD0 to AD7 and Address line A8 to A15, and all control signal.

Another memory part is serial EEPROM 32KB; 24C256. ATmega128 interfaces via I²C bus or Two-Wire Interface (TWI).

If you need to use this memeory board with VX-Mega128 board, must plug this module on top at EXPANSION connector.

5. Software in VX-Mega128 board

The related software work with VX-Mega128 microcontroller board that suggestion has 4 softwares as follows :

(1) IDE development tool - introduce **AVR Studio**. It support both Assembly and C programming. Download the latest version at www.atmel.com

(2) C compiler - suggest **WinAVR**. It is powerful GCC compiler and important Free of charge. It can work with AVR Studio. User can develop C program under AVR Studio and use WinAVR to compile. Download the latest version at <http://sourceforge.net/projects/winavr>

(3) Flash programming software - suggest 2 softwares; **AVR Prog** and **AVR-OSP II**. The AVR Prog will install with AVR Studio. Avr-Osp II is separated. But user can add Avr-OSP II into AVR Studio for easier to using. Download the latest version at <http://esnips.com/web/AtmelAVR>

(4) Boot loader - It is small firmware that program into Flash program memory of ATmega128. Its function is write the HEX code with itself via controlling from computer via serial port. Programming this code into ATmnga128 must use external In-System Programmer in first time. However the manufacturer pre-program the bootloader ready.

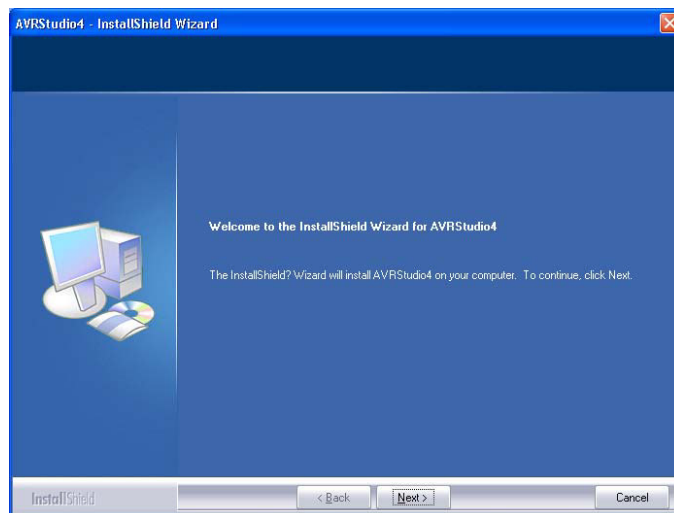
The installation must do following these step as follows

- (1) Install AVR Studio
- (2) Install WinAVR compiler
- (3) Install Virtual COM port (VCP) driver
- (4) Install Avr-Osp II software if require

5.1 AVR Studio installation

5.1.1 Insert the AVR CD-ROM and look for this file in the AVR Studio directory; aStudio4b460.exe. Double-click this file.

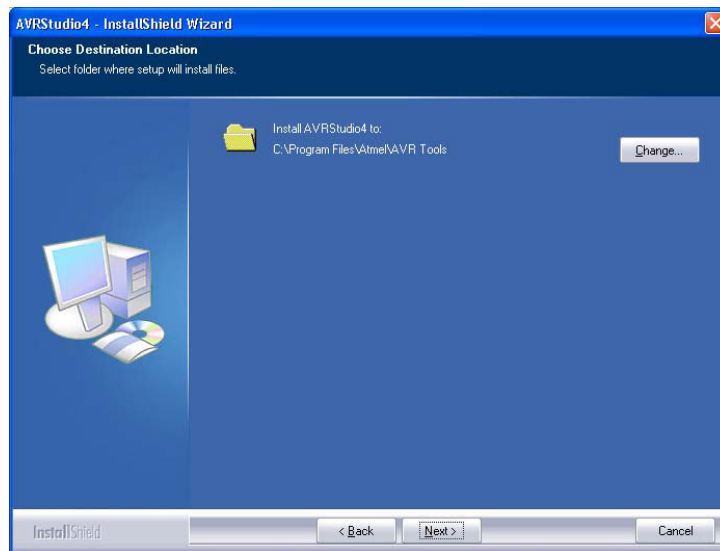
5.1.2 Enter the Installation Wizard. Click on the Next button to continue.



5.1.3 In the license agreement window, Select the box : **I accept the terms of the license agreement** and Click on the **Next** button.

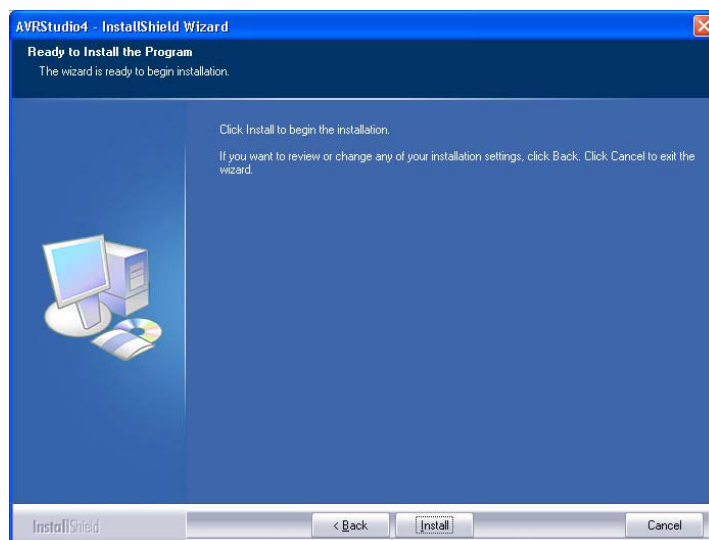
8 ● VX-Mega128 : ATmega128 Microcontroller board documentation

5.1.4 Choose Destination Location windows will appear. You can change the path by clicking on the **Change** button and setting a new path. After this, click on the **Next** button.



5.1.5 The Driver USB Upgrade window will now appear. Click on the **Next** button to pass this step.

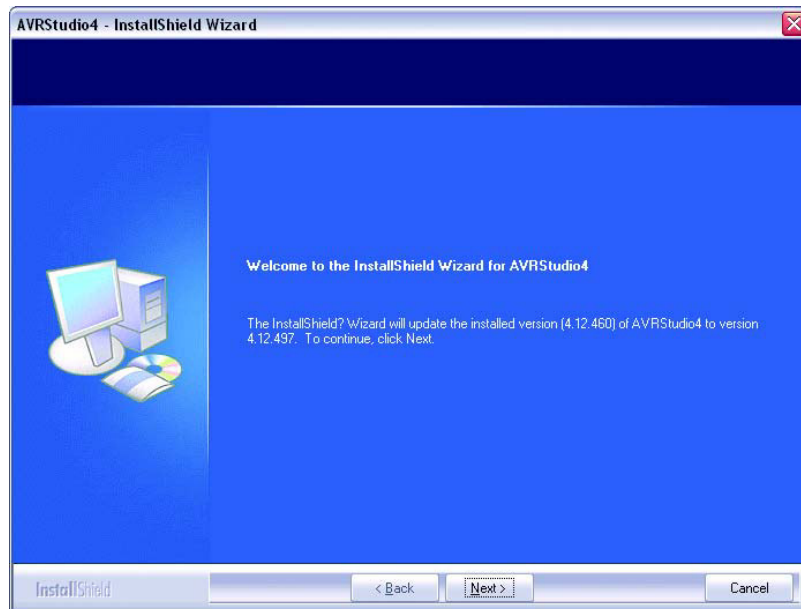
5.1.6 In the begin installation window, click on the **Install** button to start installation.



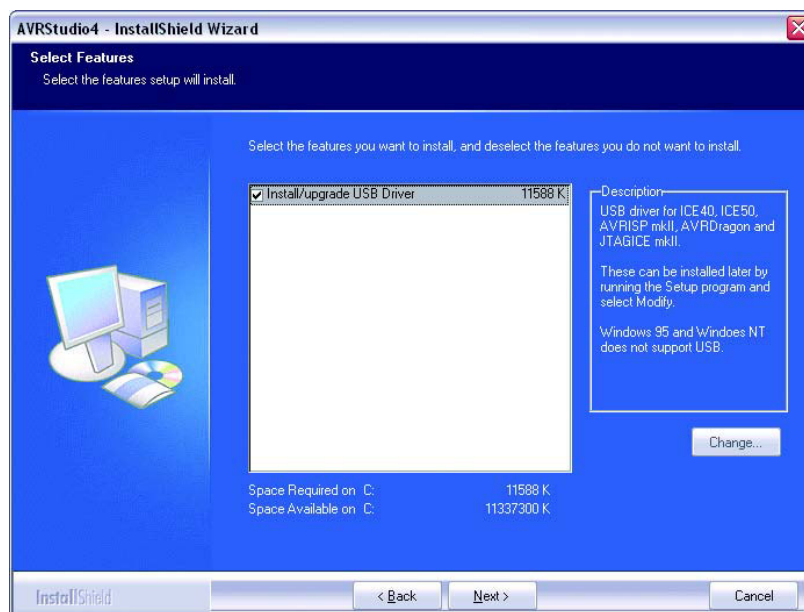
5.1.7 After the installation is complete, click on the **Finish** button to end the installation of AVR Studio.

VX-Mega128 : ATmega128 Microcontroller board documentation ● 9

5.1.8 Next step is updating the software with service pack 4. Open CD-ROM . Enter to **Software folder** → **AVR Studio 4.12**. Look for this file in the Update folder ; **aStudio412SP4b497.exe**. Double-click this file. The welcome installation windows will appear. Click the **Next** button to continue.

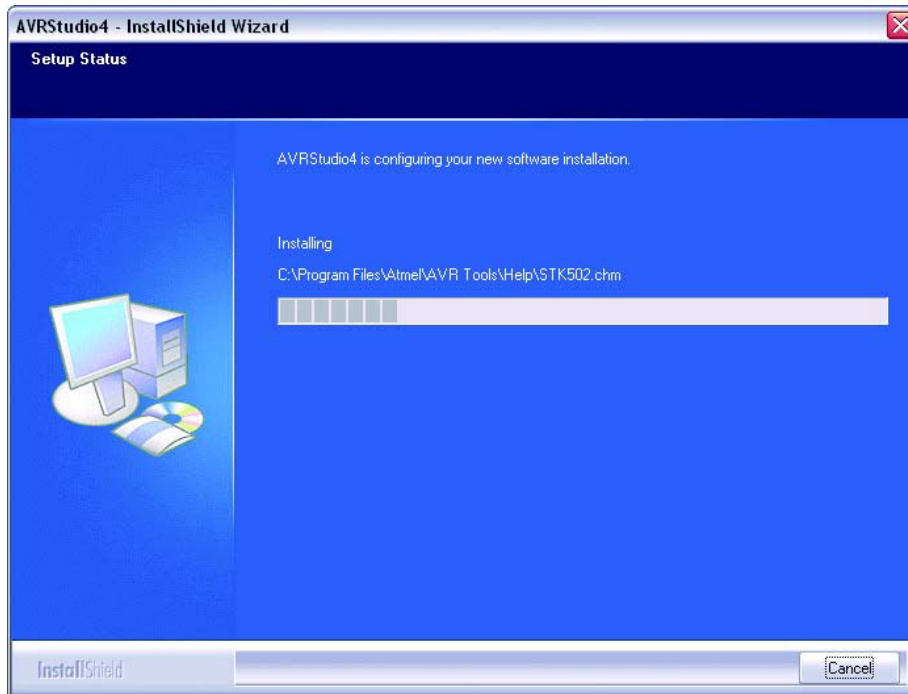


5.1.9 The Select Features window appear. Click the Next button to install the update files.

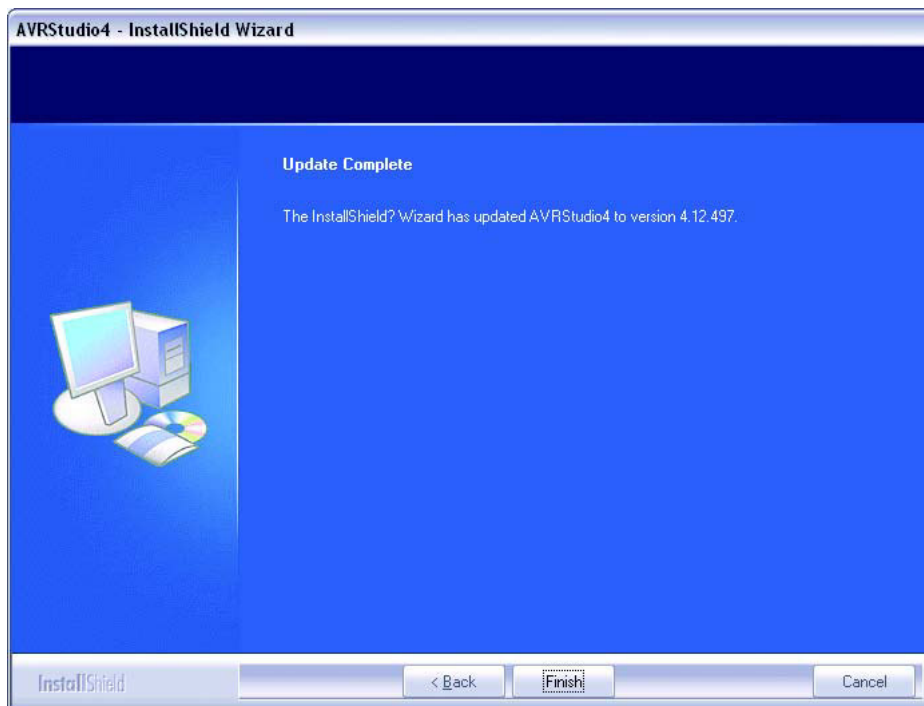


10 ● VX-Mega128 : ATmega128 Microcontroller board documentation

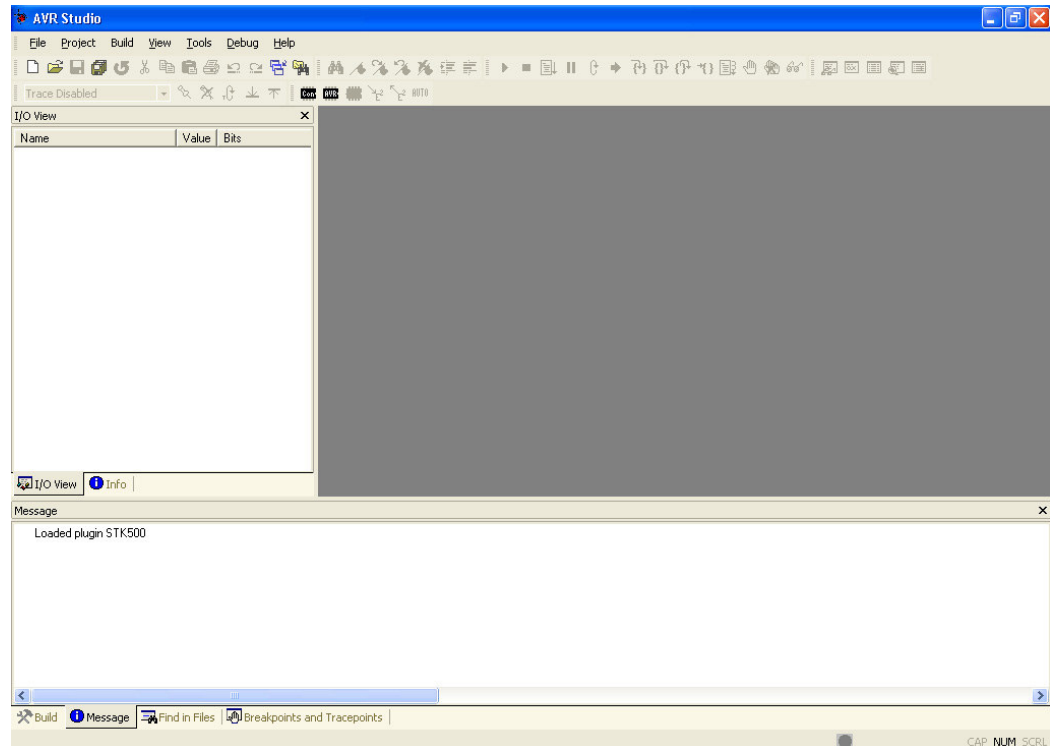
5.1.10 The Setup Status window appear and shows the installation progress bar. Wait until finish.



5.1.11 After installation complete, click the **Finish** button.



5.1.12 To Launch the AVR Studio program, click on **Start → Programs → Atmel AVR Tools → AVR Studio 4**. The main window of the AVR Studio program will appear.



5.2 WinAVR installation

Please note that installation of WinAVR is done after the installation of AVR Studio. Please ensure this is being done before proceeding.

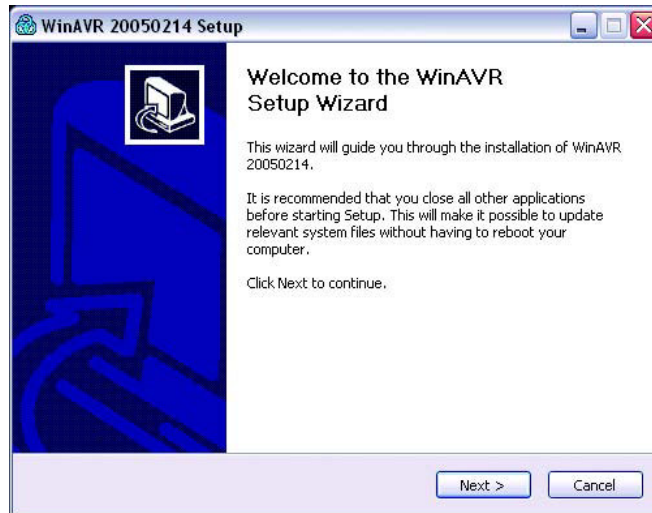
5.2.1 Insert the AVR CD-ROM, and find the installation file of WinAVR ; **WinAVR-20060421-install.exe** in folder **Software → WINAVR 20060421** Double-click this file.

5.2.2 Installation language dialog box will appear for selection the language of this installation. Select your preferred language from the sliding bar. After that click on the **OK** button.



12 ● VX-Mega128 : ATmega128 Microcontroller board documentation

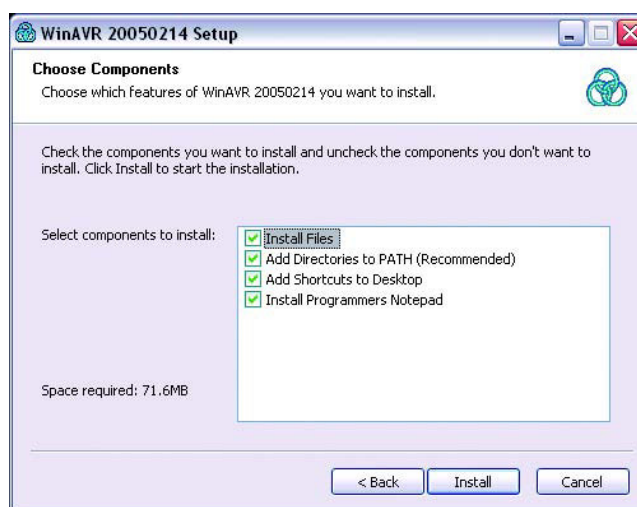
5.2.3 The Welcome installation software window appears and shows the installation information. Click on the **Next** button.



5.2.4 In the License agreement window, Click on the **I agree** button.

5.2.5 Choose Install Location window appears. User can change the path and the folder for the installation of WinAVR by clicking at the **Browse** button and selecting the respective folder. The proposed folder is **C:\WinAVR**. After selection, click **Next** button to continue to the next step.

5.2.6 In the Choose Components window, select the components which you want to install or follow according to the below diagram. Click on the **Install** button to begin the installation.



5.2.7 The installation process starts and reports the status back on the screen. The User needs to wait until the installation is complete. Click on the **Finish** button to end once its done.

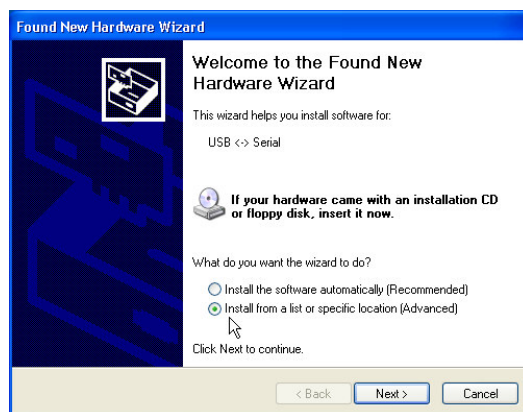
5.3 Install Virtual COM port (VCP) driver

5.3.1 Connect USB cable between VX-Mega128 board and computer's USB port. Put the ACR CD-ROM in CD-ROM drive. Apply the supply voltage to VX-Mega128 board.

5.3.2 Computer will detect the new hardware.



5.3.3 After that the New Hardware window appear. Select **Install from a list or specific location (Advanced)** header and click the **Next >** button to continue.



5.3.4 Next, click to select at **Search for the best driver in these location** and **Include this location in the search** header. After that click **Browse** button to look for Driver directory in CD-ROM. The driver file will contain in folder **USB-VCP_driver → WinNT_XP** Click the **Next** button to continue.

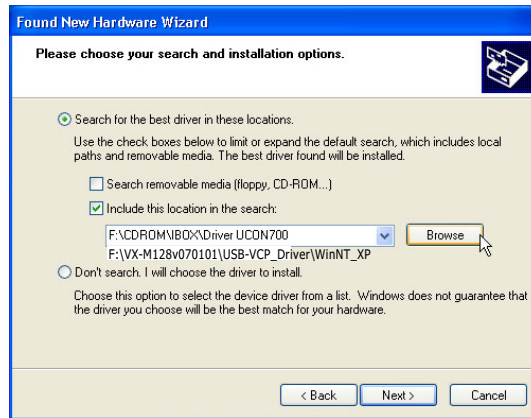
5.3.5 All driver files will install to computer.

5.3.6 The USB High Speed Serial Converter installation complete window appear. Click the **Finish** button. USB indicator on VX-Mega128 board lights.

5.3.7 Wait a moment, the Virtual COM port installation driver window will appear. Select **Install from a list or specific location (Advanced)** header similar step 5.1.3.3 and click the **Next >** button.

14 ● VX-Mega128 : ATmega128 Microcontroller board documentation

5.3.8 Select **Search for the best driver in these location** and **Include this location in the search** header. Click **Browse** button to looks for Driver directory in CD-ROM. Must select the same destination from step 5.1.3.4. Click the **Next >** button to continue.



5.3.9 All driver files will install to computer.

5.3.10 The USB Serial port installation complete window appear. Click the **Finish** button to finish installation.

5.3.11 After installation complete, user can check the position of the Virtual COM port or USB Serial port. Open **Control Panel**. Select **System** → **Hardware** → **Device Manager** and see **Port (COM & LPT)**. You will found **USB Serial Port** position. Remeber it for interfacing later.

6. Flash programming on VX-Mega128 board

6.1 Via USB port with Bootloader

6.1.1 Apply the supply voltage in range +6 to 12Vdc to VX-Mega128 board. Press the SW1 switch and turn-on power switch. The green POWER LED lights. Release the SW1. At LCD screen shows message below :

```
VX-Mega128
Run > Bootloader
```

Now the VX-Mega128 board enter to Bootloader mode. Ready for flash programming via USB port.

6.1.2 Connect USB cable between PC's USB port and VX-Mega128 board. Check the USB Serial port position from Device Manager following the figure 5.

6.1.3 Open AVR Studio to develop C program and compile to HEX file ready.

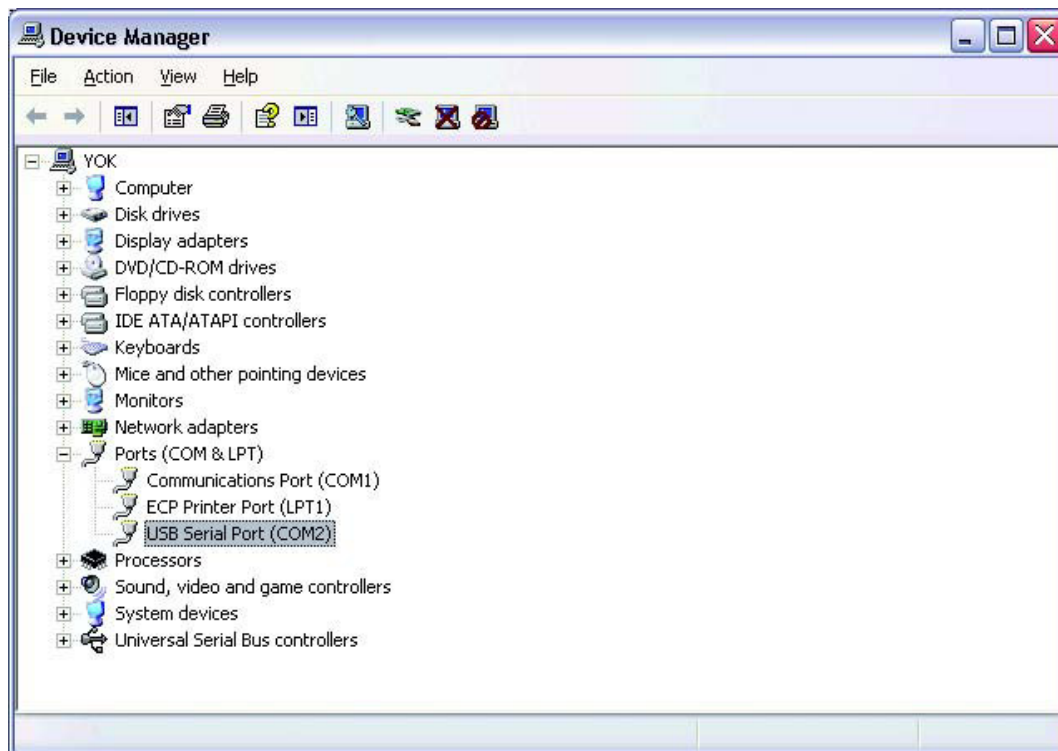
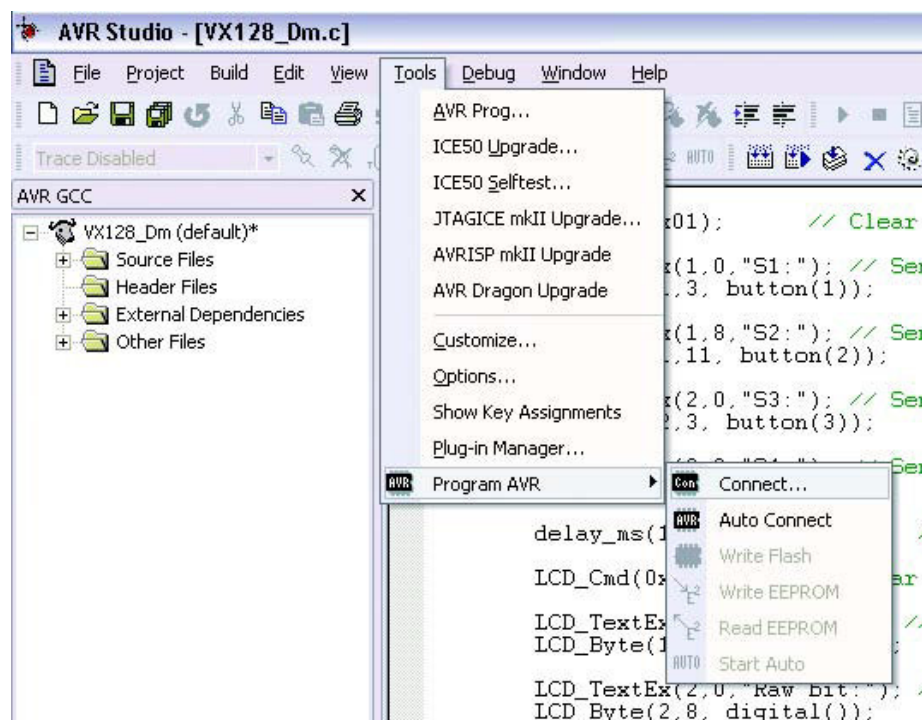


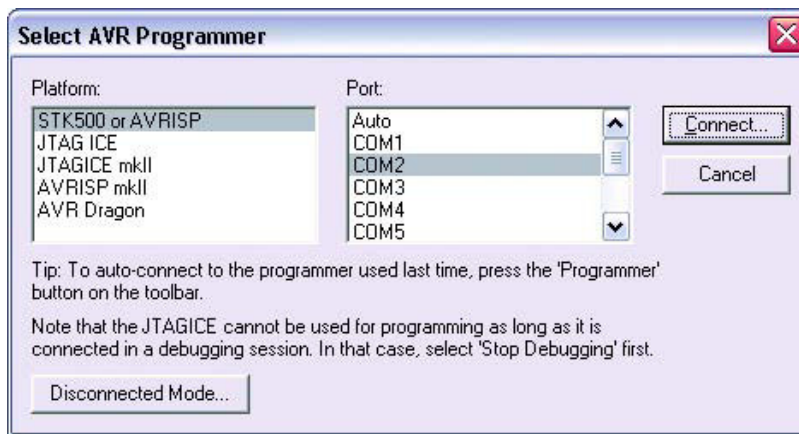
Figure 5 The Device Manager window shows the USB Serial port that installed in computer

6.1.4 At menu, selects **Tool** → **Program AVR** → **Connect...** following the illustrated below for setting the VX-Mega128 board to connect with AVR Studio.



16 ● VX-Mega128 : ATmega128 Microcontroller board documentation

6.1.5 The Select AVR Programmer window appears below.



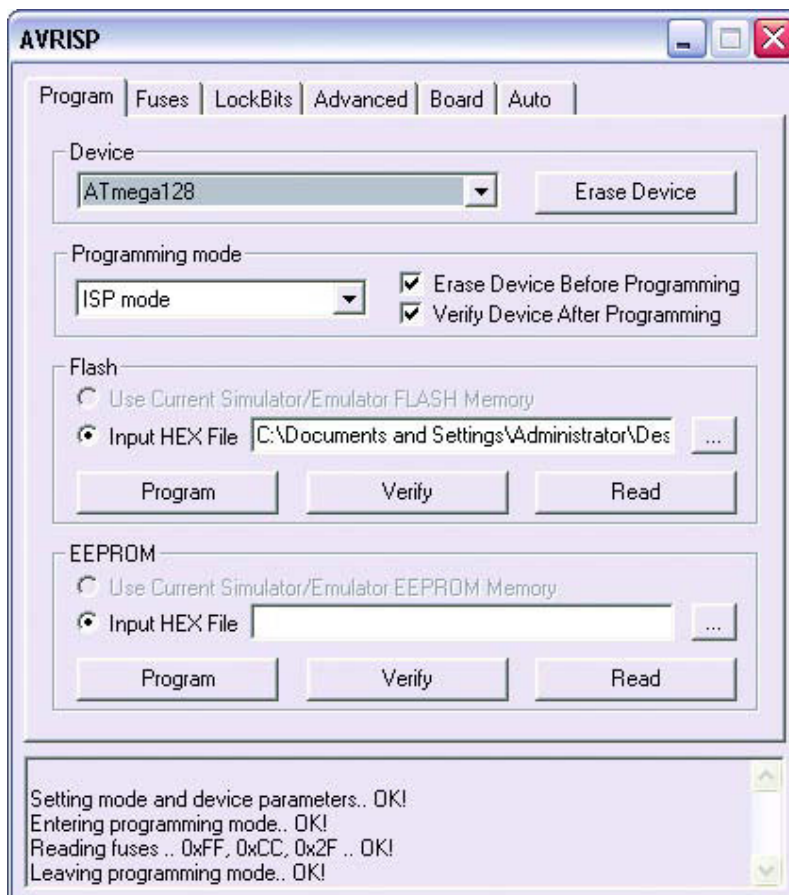
Set all parameter as follows

Platform: STK500 or AVRISP

Port: COM2 (the USB Serial port position that set by computer)

Click on the Connect button to connecting.

6.1.6 The AVRISP window appears below.



Set all parameter as follows

Device :	Atmega128
Programming mode :	ISP mode
Flash :	Input Hex File

After that click on the Browse button to select HEX file which required to download. Example file is **VX128_Dm.HEX**. Its path is **C:\Documents and Settings\Administrator\Desktop\VX128_Dm\default** (The result file - HEX file from compiler will be store in default folder)

6.1.7 Click on the Program button in Flash header for downloading VX128_Dm.HEX file to Atmega128 microcontroller on the VX-Mega128 board.

The programming steps are :

6.1.7.1 Erase

6.1.7.2 Program the HEX file. It is VX128_Dm.HEX file.

6.1.7.3 Verify programming

At AVRISP window will show the status at the bottom left corner of AVR Studio main window. Until the donwload finish, screen will show the message **Leaving programming mode.. OK!**

6.1.8 Now the flash programming ready. Press **RESET** switch to run the program. Observe the operation of VX-Mega128 board.

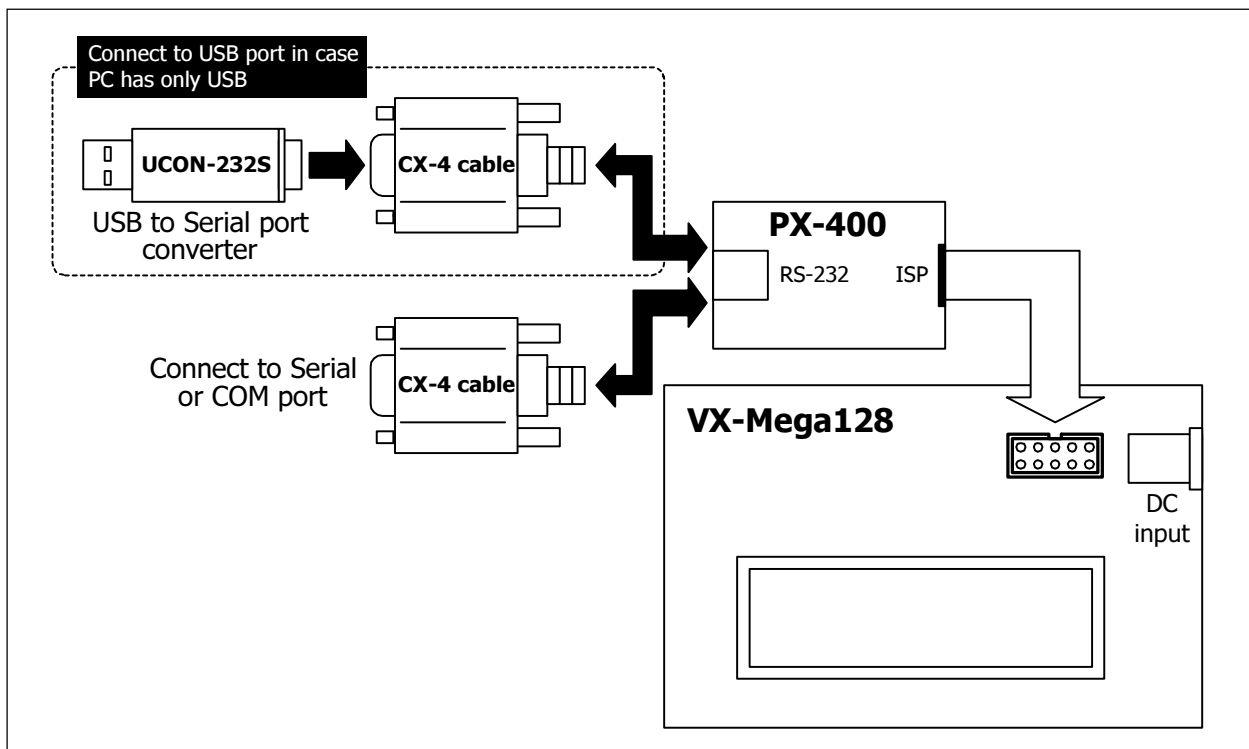


Figure 5 The diagram of Flash programming via ISP connector with PX-400 programmer

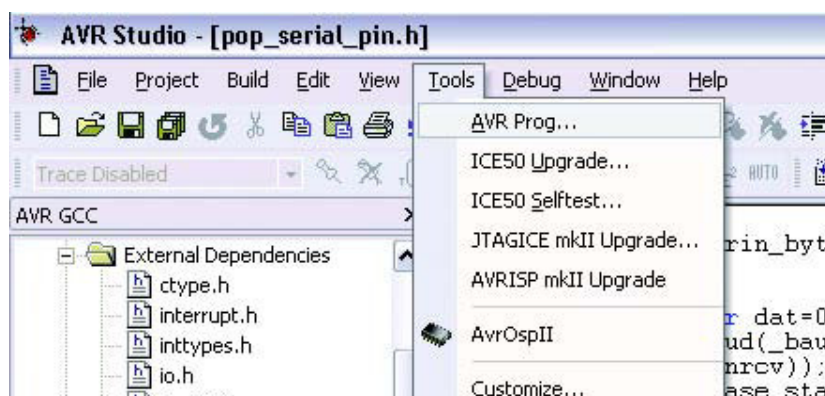
6.2 With PX-400 programmer via ISP connector

6.2.1 Apply the supply voltage during 6 to 12Vdc to VX-Mega128 board. Turn-on power switch. The green POWER indicator lights.

6.2.2 Connect PX-400 between Serial port and ISP connector on VX-Mega128 board. If your computer has only USB port, the USB to Serial port converter required. The UCON-232S is recommended. See the diagram in figure 5.

6.2.3 Open AVR Studio to develop C program and compile to HEX file ready.

6.2.4 At menu, Select **Tool → AVR Prog...**



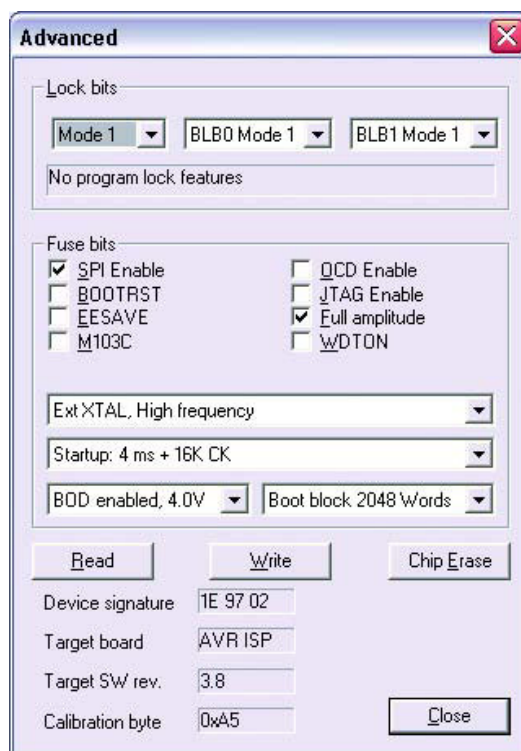
6.2.5 The AVRprog window appear.



6.2.6 For first time using AVRProg program, developer must set some parameter of flash programming for AVR Prog. The step is as follows :

6.2.6.1 **Device** box select **ATmega128**

6.2.6.2 Click on the **Advance..** button. The Advance window appear.



20 ● VX-Mega128 : ATmega128 Microcontroller board documentation

6.2.6.3 Set all parameters identical the illustrate in step 6.2.6.2. After that click on the Write button once for writing the parameters. These parameters will store and set to default value for next working.

6.2.6.4 Click on the Close button to exit this window.

6.2.7 Back to the main window of AVRProg, at **Hex file** border, click on the **Browse** button to find the HEX file. For example, load file **VX128_Dm.HEX** from this path **C:\Documents and Settings\Administrator\Desktop\VX128_Dm\default**

6.2.8 Click on the **Program** button in **Flash** border. The progress bar will active to shows the progress status.

6.2.9 After downloading finished, the program will run immediately. *The file **VX128_Dm.HEX** is Hardware testing program for VX-Mega128 board.*

6.3 How to program bootloader with your own.

If you make the flash programming of the VX-Mega128 board with the external In-System Programmer, possible to delete the Bootloader firmware. If you need to program Flash memory with bootloader again. You can re-program the bootloadr by your own with the external programmer via ISP connector.

The bootloder HEX file is **VX-M128_Boot.HEX**. **It is contained in folder Bootloader in CD-ROM that bundled with VX-Mega128 set.** Programming procedure is similar the flash programming with ant HEX file with extrenal programmer.

After the bootloader is programmed ready, you can program ATmega128 flash memory with Bootloader procedure at topic 6.1

7. Summary of Teting file of VX-Mega128 board

The VX128_Dm.HEX is HEX file for testing the operation of VX-Mega128 board. Normally, maker will pre-programmed ready to test. In first time operation, after apply the supply voltage. LCD screen on the board will show this message :

```
VX-Mega128
Run > User code
```

and change to

```
VX-Mega128
Demo
```

After that MCU will read data from 4-bit Binary Switch to display on the LCD screen. The binary switch can supply 4-bit digital data and is called Knob. User can change the shaft at Binary switch and see the correct position.

Summay of all functuion can show as follows :

- 0 : Display the Knob's value "000" on LCD16x2 module
- 1 : Testing the switches SW1 to SW4
- 2 : ADC module testing
- 3 : Test D.IN0 to D.IN7 input digital port
- 4 : Test the D.OUT0 to D.OUT7 for output port
- 5 : Read port B (PB0 to PB7)
- 6 : Read port D (PD2 to PD4)
- 7 : Read port E (PE2 to PE7)
- 8 : Test the RAM on the memory extension module (must plug the memory board on top of VX-Mega128 board at EXPANSION connector).
- 9 : Test the EEPROM on the memory extension module (must plug the memory board on top of VX-Mega128 board at EXPANSION connector).
- A : Display the Knob's value "010" on LCD16x2 module
- B : Display the Knob's value "011" on LCD16x2 module
- C : Display the Knob's value "012" on LCD16x2 module
- D : Display the Knob's value "013" on LCD16x2 module
- E : Display the Knob's value "014" on LCD16x2 module
- F : Display the Knob's value "015" on LCD16x2 module

22 ● VX-Mega128 : ATmega128 Microcontroller board documentation

8. Resource

All sourcecode in C program is contained in Example code > VX128_Dm. You can see and edit by your own to make the new code.

For more information of ATmega128 microcontroller, can see and download datasheet and many applicatiobn notes from www.atmel.com.

Special thanks for

1. Atmel Corp. about the perfect IDE software; AVR Studio.
2. Many people who develops the nicely C compiler in name WinAVR.



www.inex.co.th
www.inexglobal.com