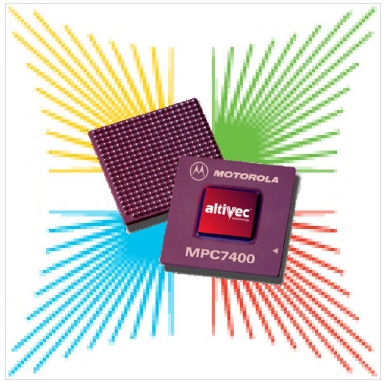


Цифровые Digital devices & Microprocessors МП устройства

Тема 3:

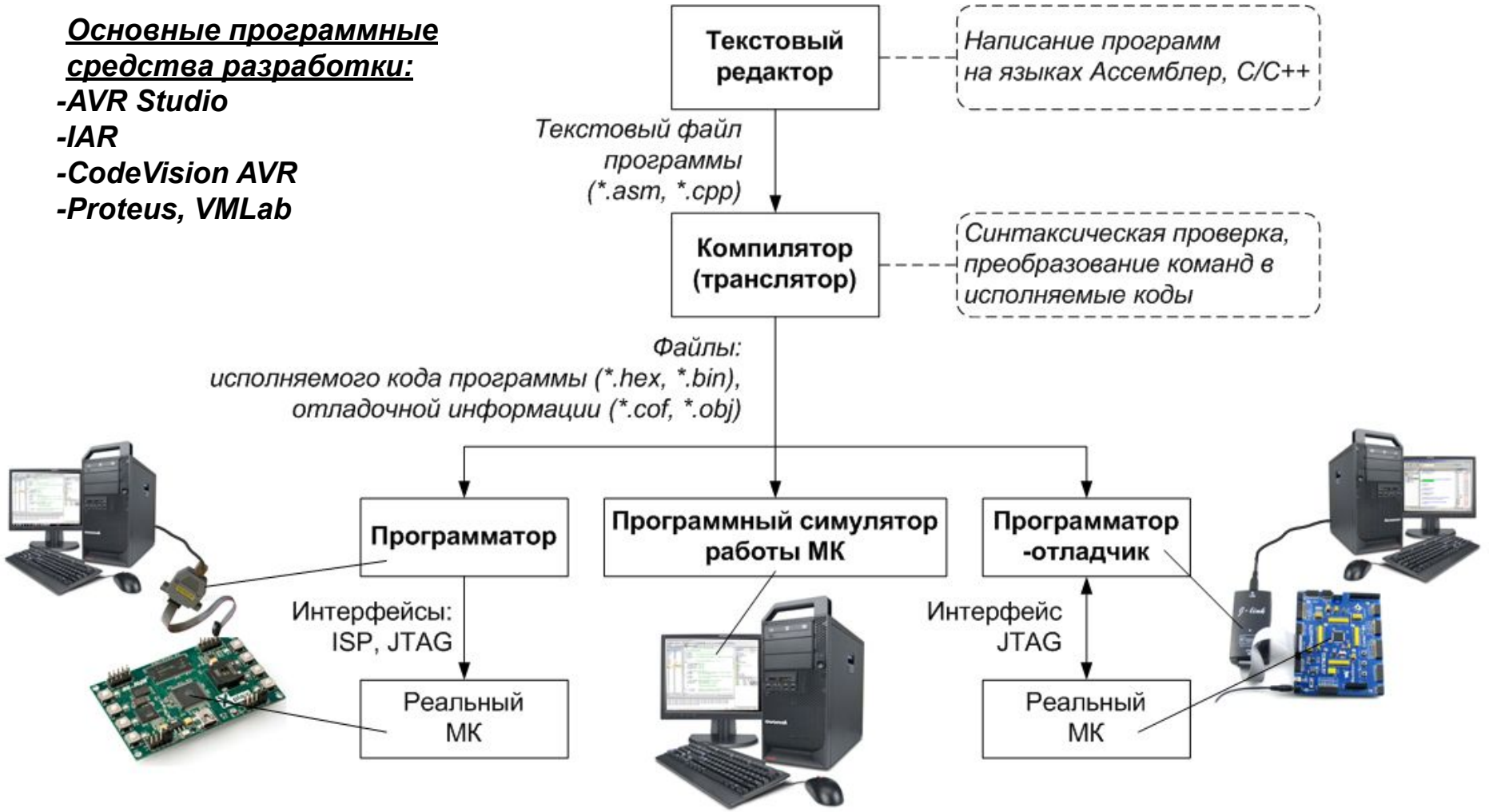
Написание программ AVR-МК. Основные команды и директивы ATmega16.

к.т.н., доцент каф.501
Мазуренко А.В.

Мнемоника – сокращение (3-5 символов) от слова или выражения, обозначающего действие, соответствующее команде.

Основные программные средства разработки:

-AVR Studio
-IAR
-CodeVision AVR
-Proteus, VMLab



Технология написания и отладки программ в ИСР AVR Studio

DEVICE **- Определить устройство для которого компилируется программа**

Синтаксис: ***.DEVICE*** <имя модели AVR-МК>

CSEG **– Определить начало программного сегмента**

Синтаксис: ***.CSEG***

DSEG **- Определить начало сегмента данных в ОЗУ**

Синтаксис: ***.DSEG***

ESEG **– Определить начало сегмента данных в ЭСППЗУ**

Синтаксис: ***.ESEG***

ORG **- Установить положение в сегменте**

Синтаксис: ***.ORG*** **<**значение или математическое выражение>

DEF **- Назначить регистру символическое имя**

Синтаксис: ***.DEF*** *<символическое_имя> = <стандартное имя РОН>*

Обозначения, использованные при описании команд:
Rd – результирующий (и исходный) регистр общего назначения;
Rr – исходный регистр общего назначения;
b – константа (3 бита), может быть константное выражение;
AdrIO – константа (5-6 бит), может быть константное выражение;
K8 – константа (8 бит), может быть константное выражение;
AdrRAM – константа (размер зависит от инструкции и модели МК), может быть константное выражение;
k – константа (размер зависит от инструкции), может быть константное выражение;
X,Y,Z – регистры косвенной адресации (X=R27:R26, Y=R29:R28, Z=R31:R30).

Флаги регистра статуса:
C – флаг переноса;
Z – флаг нулевого значения;
N – флаг отрицательного результата операции;
V – флаг переполнения дополнительного кода;
S – флаг для проверок со знаком (N⊕V);
H – флаг полупереноса;
T – бит хранения пользовательского флага;
I – бит глобального разрешения\запрещения прерываний.

Обозначения систем счисления числовых величин:
0b – префикс для обозначения двоичной системы счисления;
0 – префикс для обозначения восьмеричной системы счисления;
0x (или $) – префиксы для обозначения шестнадцатеричной системы счисления;
Без префиксов – обозначение десятичной системы счисления.

LDI – Load Immediate – Загрузить константу в РОН

Операция: Rd ← K8

Синтаксис:	Операнды:	Счетчик команд:
ldi Rd,K8	16≤ d ≤31	PC ← PC+1

Флаги на которые воздействует команда: не воздействует
Количество тактов выполнения операции: 1.

LDS – Load Direct from data space – Прямая загрузка из памяти данных

Операция: Rd ← RAM(AdrRAM)

Синтаксис:	Операнды:	Счетчик команд:
lds Rd, AdrRAM	0≤ d ≤31, 0≤ AdrRAM ≤65535	PC ← PC+2

Флаги на которые воздействует команда: не воздействует
Количество тактов выполнения операции: 2.

STS – Store Direct to data space – Прямая загрузка в память данных
Операция: RAM(AdrRAM) ← Rr

Синтаксис:	Операнды:	Счетчик команд:
sts AdrRAM,Rr	0≤ r ≤31, 0≤ AdrRAM ≤65535	PC ← PC+2

Флаги на которые воздействует команда: не воздействует
Количество тактов выполнения операции: 2.

CLR – Clear Register – Очистить регистр

Операция: Rd ← $00

Синтаксис:	Операнды:	Счетчик команд:
clr Rd	0≤ d ≤31	PC ← PC+1

Флаги на которые воздействует команда: S←0, V←0, N←0, Z←1
Количество тактов выполнения операции: 1.

SER – Set Register – Установить регистр

Операция: Rd ← $FF

Синтаксис:	Операнды:	Счетчик команд:
ser Rd	16≤ d ≤31	PC ← PC+1

Флаги на которые воздействует команда: не воздействует
Количество тактов выполнения операции: 1.

SBR – Set Bits in Register – Установить биты в регистре
Операция: Rd ← Rd V K8

Синтаксис:	Операнды:	Счетчик команд:
sbr Rd,K8	16≤ d ≤31	PC ← PC+1

Флаги на которые воздействует команда: S, V←0, N, Z
Количество тактов выполнения операции: 1.

ADD – Add without Carry – Сложить без учета переноса

Операция: Rd ← Rd + Rr

Синтаксис:	Операнды:	Счетчик команд:
add Rd,Rr	0≤ d, r ≤31	PC ← PC+1

Флаги на которые воздействует команда: H, S, V, N, Z, C
Количество тактов выполнения операции: 1.

ADC – Add with Carry – Сложить с учетом переноса

Операция: Rd ← Rd + Rr +C

Синтаксис:	Операнды:	Счетчик команд:
adc Rd,Rr	0≤ d, r ≤31	PC ← PC+1

Флаги на которые воздействует команда: H, S, V, N, Z, C
Количество тактов выполнения операции: 1.

SUB – Subtract without Carry – Вычесть без учета переноса
Операция: Rd ← Rd - Rr

Синтаксис:	Операнды:	Счетчик команд:
sub Rd,Rr	0≤ d,r ≤31	PC ← PC+1

Флаги на которые воздействует команда: H, S, V, N, Z, C
Количество тактов выполнения операции: 1.

SBC – Subtract with Carry – Вычесть с учетом переноса
Операция: Rd ← Rd – Rr-C

Синтаксис:	Операнды:	Счетчик команд:
sbc Rd,Rr	0≤ d,r ≤31	PC ← PC+1

Флаги на которые воздействует команда: H, S, V, N, Z, C
Количество тактов выполнения операции: 1.

RJMP – Relative jump – Относительный переход

Операция: PC ← PC+1+k

Синтаксис:	Операнды:	Счетчик команд:
rjmp k	-2K≤ k ≤+2K	PC ← PC+1+k

Флаги на которые воздействует команда: не воздействует
Количество тактов выполнения операции: 2.

JMP – Jump – Длинный относительный переход

Операция: PC ← PC+1+k

Синтаксис:	Операнды:	Счетчик команд:
jmp k	0≤ k ≤4M	PC ← PC+1+k

Флаги на которые воздействует команда: не воздействует
Количество тактов выполнения операции: 3.

Задание:

Написать ПО МК ATmega16, которое бы выполняло:

1)суммирование двух двухбайтных беззнаковых чисел N и K, которые размещаются в ячейках ОЗУ с адресами N - RAM($60:$61) и K - RAM($62:$63);

2)запись всех байтов результата в ячейки ОЗУ с адресами, начиная с $6A.

Решение:

1)создать проект ПО и исходный текстовый файл программы AVR-МК на языке Ассемблер в AVR Studio (смотри файл «***Описание AVR Studio 4.18 (rev.1.1).pdf***» - стр.16-18);

2)написать исходный текстовый файл программы в текстовом редакторе (смотри файл: «***Subject_3.asm»***);

3)компилировать программу компилятором AVR Studio (смотри файл «***Описание AVR Studio 4.18 (rev.1.1).pdf***» - стр.24-25);

4)выполнить отладку программы с помощью программного симулятора AVR Studio (смотри файл «***Описание AVR Studio 4.18 (rev.1.1).pdf***» - стр.25-31);

Понятие «бесконечного» цикла работы программы!