

PUMPKIN™

REAL-TIME SOFTWARE

750 Naples Street • San Francisco, CA 94112 • (415) 584-6360 • <http://www.pumpkininc.com>  
• Перевод: Андрей Шлеенков • <http://andromega.narod.ru> • <mailto:andromega@narod.ru> •

RM-ICCAVR

Справочное руководство

## Справочное руководство *Salvo* для компилятора *ImageCraft ICCAVR*

---



Salvo™

The RTOS that runs in tiny places.™

## Введение

Данное руководство предназначено для пользователей Salvo, использующих микроконтроллеры AVR®<sup>1</sup> и MegaAVR™ компании Atmel (<http://www.atmel.com/>) с компилятором Си ICCAVR компании ImageCraft (<http://www.imagecraft.com/>).

## Связанные документы

При создании приложений Salvo с компилятором Си ImageCraft ICCAVR, вместе с данным руководством должны использоваться следующие документы Salvo:

*Руководство пользователя Salvo (Salvo User Manual)*  
*Приложение AN-24 (Application Note AN-24)*

## Примеры проектов

Примеры проектов Salvo для использования с компилятором Си ImageCraft ICCAVR и средой разработки ImageCraft IDE могут быть найдены в следующих директориях каждого дистрибутива Salvo для Atmel AVR и MegaAVR:

```
\salvo\ex\ex1\sysv  
\salvo\tut\tu1\sysv  
\salvo\tut\tu2\sysv  
\salvo\tut\tu3\sysv  
\salvo\tut\tu4\sysv  
\salvo\tut\tu5\sysv  
\salvo\tut\tu6\sysv
```

## Свойства

Таблица 1 иллюстрирует основные особенности реализации Salvo для компилятора Си ImageCraft ICCAVR.

<b>ОСНОВНОЕ</b>	
доступные дистрибутивы	Salvo Lite, LE & Pro for Atmel AVR and MegaAVR
дополнительные дистрибутивы	Salvo tiny & SE for Atmel AVR and MegaAVR & ICCAVR
поддерживаемые устройства	семейства AVR и MegaAVR
заголовочные файлы	porticcavr.h
другие специфические для процессора файлы	porticcavr.s porticcatmega.s
имена поддиректорий проекта	SYSV
<b>salvocfg.h</b>	
автоопределение компилятора?	да <sup>2</sup>
<b>библиотеки</b>	
поддиректория \salvo\lib	iccavr
<b>переключение контекста</b>	
метод	на основе меток и функций OSDispatch() & OSCtxSw(label)
_OSLabel() требуется?	да
объем автоматических переменных и параметров функций в задачах	общий объем не должен превышать 254 8-битовых байт
<b>память и регистры</b>	
поддержка внутренней и внешней памяти RAM?	да, при помощи -bsalvoram:0xstart.0xend
R20...R23 используются?	нет
<b>прерывания</b>	
управляются через	I бит
статус прерывания сохраняется в критических секциях?	да
используемый метод	сохранение в стеке при помощи директивы #pragma monitor
степень вложенности	не ограничена
альтернативные методы возможны?	да <sup>3</sup>
<b>отладка</b>	
отладка в исходных кодах с библиотеками Salvo Pro?	да
<b>компилятор</b>	
поддержка упакованных битовых полей?	нет
printf() / %p поддерживается?	да / да
va_arg() поддерживается?	да

Таблица 1: Особенности реализации Salvo для компилятора Си ImageCraft ICCAVR

## Библиотеки

### Номенклатура

Имена библиотек Salvo для компилятора Си ImageCraft ICCAVR следуют соглашению, показанному на примере имени одной из библиотек на Рисунке 1.

Пример имени библиотеки: `libsficcavr-a.a`

СИМВОЛЫ	ЗНАЧЕНИЕ	ВОЗМОЖНЫЕ ВАРИАНТЫ
<code>lib</code>	библиотека	
<code>s</code>	Salvo	
<code>f</code>	тип	<code>f</code> : freeware <code>l</code> : standard
<code>icc</code>	Image Craft C	
<code>avr</code>	контроллер	<code>avr</code> : AVR <code>atmega</code> : MegaAVR
<code>-</code>	опция	<code>-</code> : нет опций <code>i</code> : с отладочной информацией
<code>a</code>	конфигурация	<code>a</code> : многозадачность с задержками и событиями <code>d</code> : многозадачность с задержками <code>e</code> : многозадачность с событиями <code>m</code> : многозадачность только <code>s</code> : библиотека Salvo SE <code>t</code> : многозадачность с задержками, событиями и ожиданиями с таймаутом <code>y</code> : библиотека Salvo tiny

**Рисунок 1: Номенклатура библиотек Salvo для компилятора Си ImageCraft ICCAVR**

### Тип

Дистрибутив Salvo Lite содержит *свободные (freeware)* библиотеки. Все остальные дистрибутивы Salvo содержат *стандартные (standard)* библиотеки. Дополнительную информацию о типах библиотек см. в главе *Библиотеки (Libraries)* документа *Руководство пользователя Salvo (Salvo User Manual)*.

### Целевой процессор

Каждая библиотека предназначена для одного или нескольких определенных процессоров. Таблица 2 перечисляет корректные библиотеки для каждого поддерживаемого процессора AVR или MegaAVR.

код процессора	процессор(ы)
avr :	AT90S2313, AT90S2323, AT90S2333, AT90S2343, AT90S4433, AT90S4434, AT90S4444, AT90S8515(A), AT90S8534, AT90S8535, ATmega8(L), ATtiny26
atmega :	AT94K05, AT94K10, AT94K40, ATmega103, ATmega128(L), ATmega16, ATmega161(L), ATmega162, ATmega163(L), ATmega169, ATmega32, ATmega323(L)

Таблица 2: Процессоры для библиотек Salvo для компилятора Си ImageCraft ICCAVR

**Замечание:** Результирующий код для отсутствующего в списке процессора в общем случае может быть получен, если процессор является членом семейства AVR или MegaAVR.

## Опция

Пользователи Salvo Pro могут выбирать между двумя наборами библиотек – стандартные библиотеки и стандартные с информацией для отладки в исходных кодах. Последние могут быть получены с помощью опции командной строки `+g` компилятора Си ImageCraft ICCAVR. Это добавит отладочную информацию в библиотеки, делая их идеальными для прогона и пошаговой отладки в отладчике ICCAVR. Для использования этих библиотек в проекте, необходимо выбрать ту, которая включает отладочные коды (например, `libsliccavrit.a`) вместо обычной (например, `libsliccavr-t.a`).

## Конфигурация

Для различных дистрибутивов Salvo предусмотрены различные конфигурации библиотек, позволяющие пользователю минимизировать код ядра Salvo. Дополнительную информацию о конфигурации библиотек см. в главе *Библиотеки* документа *Руководство пользователя Salvo*.

## Установки компиляции

Библиотеки Salvo для компилятора Си ImageCraft ICCAVR построены, используя установки по умолчанию, описанные в главе *Библиотеки* документа *Руководство пользователя Salvo*. Специфические для процессоров установки и их замены перечислены в Таблице 3.

ограничения компиляции	
макс. число задач	3
макс. число событий	5
макс. число флагов событий	1
макс. число очередей сообщений	1
специфические для процессора установки	
размер задержки	8 бит
сторожевой таймер	очищается в <code>OSSched()</code>
счетчик системного времени	доступен, 32 бита

Таблица 3: Установки и замены для библиотек Salvo для компилятора Си ImageCraft ICCAVR

**Замечание:** Ограничения компиляции библиотек Salvo могут быть изменены в меньшую сторону (все дистрибутивы Salvo) или в большую сторону (все дистрибутивы Salvo кроме Salvo Lite) относительно значений по умолчанию. См. главу *Библиотеки* документа *Руководство пользователя Salvo*.

## Доступные библиотеки

Всего доступно 34 библиотеки Salvo для компилятора Си ImageCraft ICCAVR. Каждый тип дистрибутива Salvo для Atmel AVR и MegaAVR включает также библиотеки Salvo из младших версий дистрибутивов.

## Специфичные для процессора исходные файлы Salvo

Для компиляции с исходным кодом Salvo Pro требуется один из двух отличающихся исходных файлов – `porticcavr.s` или `porticcatmega.s`. Для определенного целевого процессора необходимо использовать файл в соответствии с кодом процессора из номенклатуры Таблицы 2.

**Замечание:** `porticcavr.s` будет работать с любым процессором AVR, имеющим не более 8 КВ программной памяти. `porticcatmega.s` требуется для процессоров MegaAVR, адресующих более 8 КВ программной памяти.

## Примеры `salvocfg.h`

Ниже приводятся примеры файлов конфигурации проекта `salvocfg.h` для различных дистрибутивов Salvo для Atmel AVR и MegaAVR, использующих целевой микроконтроллер AT90S8515.

**Замечание:** При изменении заданных по умолчанию числа задач, событий и т.п. при построении с библиотеками Salvo в файле проекта `salvocfg.h` *должны быть определены* `OSTASKS` и `OSEVENTS` соответственно. При отсутствии определений, будут использованы значения по умолчанию (см. Таблицу 3).

## Компиляция с библиотеками Salvo Lite

```
#define OSUSE_LIBRARY           TRUE
#define OSLIBRARY_TYPE         OSF
#define OSLIBRARY_CONFIG       OSA
```

Листинг 1: Пример `salvocfg.h` для компиляции с библиотеками, используя `libsliccavr-a.a`

## Компиляция с библиотеками Salvo tiny

```
#define OSUSE_LIBRARY           TRUE
#define OSLIBRARY_TYPE         OSL
#define OSLIBRARY_CONFIG       OSY
```

Листинг 2: Пример `salvocfg.h` для компиляции с библиотеками, используя `libsliccavr-y.a`

## Компиляция с библиотеками Salvo SE

```
#define OSUSE_LIBRARY           TRUE
#define OSLIBRARY_TYPE         OSL
#define OSLIBRARY_CONFIG       OSS
```

Листинг 3: Пример `salvocfg.h` для компиляции с библиотеками, используя `libsliccavr-s.a`

## Компиляция с библиотеками Salvo LE & Pro

```
#define OSUSE_LIBRARY           TRUE
#define OSLIBRARY_TYPE         OSL
#define OSLIBRARY_CONFIG       OSA
```

Листинг 4: Пример `salvocfg.h` для компиляции с библиотеками, используя `libsliccavr-a.a` или `libsliccavria.a`

## Компиляция с исходными кодами Salvo Pro

```
#define OSENABLE_IDLE_HOOK     TRUE
#define OSENABLE_SEMAPHORES    TRUE
#define OSEVENTS                1
#define OSTASKS                  3
```

Листинг 5: Пример `salvocfg.h` для компиляции с исходным кодом

## Эффективность

### Использование памяти

учебные примеры <sup>4</sup>	всего ROM <sup>5</sup>	всего RAM <sup>6</sup>
tu1lite	239 / 232	22
tu2lite	325 / 317	22
tu3lite	358 / 349	24
tu4lite	695 / 654	33
tu5lite	1058 / 961	45
tu6lite	1144 / 1038 <sup>7</sup>	47 <sup>8</sup>
tu6pro <sup>9</sup>	1025 / 947 <sup>10</sup>	43 <sup>11</sup>

Таблица 4: Требования памяти ROM и RAM для приложения Salvo, создаваемого компилятором Си ImageCraft ICCAVR

## Специальные вопросы

### Размер стека

Компилятор Си ImageCraft ICCAVR использует два отдельных стека – один для адресов возврата (аппаратный стек, управляемый SP) и один для передачи параметров и хранения локальных данных (программный стек, управляемый Y).

По сравнению с не Salvo не многозадачным приложением со схожей структурой вызовов, соответствующее приложение Salvo потребует дополнительно 4-х байт (т.е. двух адресов возврата) в аппаратном стеке.<sup>12</sup>

Размер аппаратного стека может быть установлен в ICCAVR IDE в меню *Project>Options>Target>Advanced>Return Stack Size* или в командной строке компоновщика iccavr, например:

```
iccavr ... -dhwstk_size:20 ...
```

Приложения, использующие вложенные прерывания, плавающую точку или типы long, потребуют большего аппаратного стека, чем определено по умолчанию – см. ICCAVR Help для получения дополнительной информации.



## Внешняя память SRAM

Глобальные объекты Salvo<sup>13</sup> могут быть размещены как во внутренней, так и во внешней памяти данных RAM. В ImageCraft ICCAVR IDE размещение объектов (например, переменных) в области программы `data` контролируется через меню *Project>Options>Target>Device Configuration* (Internal SRAM), и т.д. В командной строке компоновщика `iccavr` расположение этих объектов определяется при помощи команды `-bdata:start,end`, например:

```
iccavr ... -bdata:0x260,0xffff ...
```

определяет, что область `data` начинается по адресу `0x260` (конец внутренней памяти SRAM) и продолжается до `0xFFFF` (граница 64 КБ).

Глобальные объекты Salvo могут быть помещены группой в любом месте памяти RAM (внутренней или внешней), специфицируя стартовый и конечный адреса программной области `salvoram`. Это применимо к исходному коду и к компиляции с библиотеками. Например, для размещения всех глобальных объектов Salvo в 256-байтном блоке внешней памяти RAM в самом конце 32 КБ границы, при компоновке приложения используется следующее<sup>14</sup>:

```
iccavr ... -bsalvoram:0x7F00,0x7FFF ...
```

**Замечание:** Если не использовать аргумент командной строки компоновщика `-b`, программная область `salvoram` будет размещена непосредственно после области `bss` в области данных программы. Поэтому эта опция требуется только при желании разместить глобальные объекты Salvo отдельно от переменных программы и т.п. Порядок областей программы может быть изменен директивой ассемблера `.area`.

## Сегменты данных

Регистр RAMPD обычно используется для доступа ко всему пространству данных в процессорах с более чем 64 КБ пространством памяти данных. Не имеется никаких средств для доступа к глобальным объектам Salvo вне текущего сегмента данных размером в 64 КБ.

## Компрессор кода

Salvo обладает совместимостью с компрессором кода ImageCraft ICCAVR Code Compressor<sup>15</sup> как в библиотеках, так и при компиляции с исходным кодом.

## Косвенный вызов функций

Для правильной работы компрессора кода, все косвенные вызовы функций должны выполняться при помощи `xicall`. Метод переключения контекста, используемый Salvo с компилятором Си ImageCraft ICCAVR использует `xicall` для всех своих косвенных вызовов функций.

## Регистры R20..R23

ICCAVR может быть проинструктирован не использовать регистры R20..R23. На практике, это оказывает незначительный эффект на код Salvo – это может немного повысить скорость работы и уменьшить размер памяти ROM.

Библиотеки Salvo сгенерированы без использования R20..R23, поэтому контроль над этими регистрами остается за программистом.

Пользователи Salvo Pro могут управлять использованием этих регистров в генерируемом исходном коде.

## Расположение библиотек

Компилятор Си ImageCraft ICCAVR предполагает размещение библиотек в директории `\icc\lib`. Поэтому инсталлятор Salvo размещает библиотеки для ICCAVR в двух местах: `\salvo\lib\iccavr` и `icc\lib`.

<sup>1</sup> Микроконтроллеры tinyAVR не поддерживаются из-за отсутствия памяти RAM.

<sup>2</sup> Выполняется автоматически при помощи символов `_IMAGECRAFT_` и `_AVR_`, определяемых компилятором.

<sup>3</sup> Так как сохранение и восстановление бита I тесно связано с `#pragma monitor` компилятора, альтернативные методы в общем случае не рекомендованы.

<sup>4</sup> Salvo v.3.2.0 с ICCAVR v.6.28.

<sup>5</sup> В словах. Второй номер отражает размер ROM с разрешенным компрессором кода. Включает векторы прерываний и таблицу `func_lit` для функций, вызываемых косвенно через `xicall`. R20..R23 не используются.

<sup>6</sup> В байтах. Не включает память RAM, резервируемую для адресов возврата (аппаратный стек) и передачи параметров и хранения локальных переменных (программный стек).

<sup>7</sup> Включает 2 байта из секции `idata`.

<sup>8</sup> Включает 2 байта из секции `data`.

<sup>9</sup> Компиляция с Salvo Pro несколько отличается от Salvo Lite при конфигурировании – см. учебный пример `salvocfg.h`.

<sup>10</sup> Включает 2 байта из секции `idata`.

<sup>11</sup> Включает 2 байта из секции `data`.

<sup>12</sup> Приложение Salvo Pro может уменьшить это на 2 байта (адрес возврата) встраиванием кода `OSSched()` в исходный код проекта способом `inline`.

<sup>13</sup> Например, блоки управления задачами, указатели очередей, счетчики и т.д.

<sup>14</sup> Ошибка выделения достаточной памяти RAM для области `salvoram` приведет к ошибке компоновщика `area 'salvoram' not large enough`.

<sup>15</sup> Компрессор кода включен в ICCAVR Professional.