# File Formats

* [Generic file format](#Generic_file_format)
* [Object file format](#Object_file_format)

## Generic file format

The Assembler can generate three different output file formats:

* Generic
* Motorola S-Records
* Intel Intellec 8/MDS

The formats of the latter two are assumed known. The Generic file format is a simple, self defined format, where each line has the following format:

ADR:OPCODE

Where ADR is a 6 digit (24 bit) hexadecimal number and OPCODE is a 4 digit (16 bit) hexadecimal number. ADR defines an address in the Program memory, and OPCODE defines the contents of this address.

### Example

Given the following assembly file gen\_demo.asm:

; Demonstration of the Generic file format   
        mov r0,r1   
        inc r1   
        call oursub   
.org 0x50                 ; Set Program space   
                          ; address to 50 (Hex)   
oursub: add r1,r2         ; Do something   
ret

Then the following output file gen\_demo.rom will be produced:

000000:2c01   
000001:9413   
000002:940e   
000003:0050   
000050:0c12   
000051:9508

Note that the two-word instructions (CALL and JMP) need two lines of coding.

## Object file format

The object (.obj) file produced by the Assembler is also represented in a self defined format. The object file contains some limited debug information, and can be used together with AVR Studio

The object file has a header section, a record section and a trailer section. The header section has the following format:

* Offset to source file names (4 bytes)
* Offset to object records (4 bytes)
* Number of bytes in each record (1 byte)
* Number of file names stored in the Trailer (1 byte)
* The string “AVR Object File\0” (\0 means zero terminated)

The records are currently 9 bytes long. Each record has the following format:

* Program memory address (3 bytes)
* Opcode (2 bytes)
* Source file number of the instruction (1 byte, first file numbered 0)
* Line number in the source file (2 bytes, first line numbered 1)
* Macro indicator (1 byte, 1 if instruction is in a macro, 0 if not)

Finally, the trailer section has the following format:

* File names (Zero terminated, number of file names in header)
* ASCII 0

### Example

Given the following assembly file obj\_demo.asm:

; Demonstration of the Object file format (obj\_demo.asm)   
.equ const1=0x15   
.equ const2=0x40   
.macro SWIN               ; SWIN - swap and increment   
        swap @0   
        inc @0   
.endmacro                 ; End macro   
start:  ldi r16,const1   
        SWIN r16          ; Call macro   
        ldi r16,const2   
        SWIN r16          ; Call macro   
        rjmp start   
.include "delay.asm"      ; Include another assembly file delay.asm

; Include file, demonstration of the Object file format   
; (delay.asm)   
delay:  dec r16           ; Decrement counter   
        breq delay        ; If not zero branch to delay   
        ret               ; Return from subroutine

Then the following output file obj\_demo.obj would be produced (the file is a binary file which has been converted into hexadecimal representation, the offset column and the line shifts are manually inserted for reasons of clarity):

Offset:         File contents:                        Comment:   
00000000:       00000074                              Offset to file names   
00000004:       0000001A                              Offset to records   
00000008:       09                                    #Bytes/record   
00000009:       02                                    #File names   
0000000A:       415652204F626A6563742046696C6500      AOF string   
0000001A:       000000E10500000B00                    First record   
00000023:       000001950200000C01   
0000002C:       000002950300000C01   
00000035:       000003E40000000D00   
0000003E:       000004950200000E01   
00000047:       000005950300000E01   
00000050:       000006CFF900000F00   
00000059:       000007950A01000400   
00000062:       000008F3F101000500   
0000006B:       000009950801000600                    Last record   
00000074:       4F424A5F44454D4F2E41534D00            “OBJ\_DEMO.ASM\0”   
00000081:       44454C41592E41534D00                  “DELAY.ASM\0”   
0000008B:       00                                    End of object file