

avr32, flags 0x00000112:  
EXEC\_P, HAS\_SYMS, D\_PAGED  
start address 0x00000000

Program Header:

```
LOAD off 0x00001000 vaddr 0x00000000 paddr 0x00000000 align 2**12
      filesz 0x0000a200 memsz 0x0000a200 flags r-x
LOAD off 0x0000c000 vaddr 0x10000000 paddr 0x0000a200 align 2**12
      filesz 0x000008c8 memsz 0x00000968 flags rw-
LOAD off 0x0000d000 vaddr 0x11fff000 paddr 0x0000ab68 align 2**12
      filesz 0x00000000 memsz 0x00001000 flags rw-
LOAD off 0x0000d000 vaddr 0x24007000 paddr 0x0000bb68 align 2**12
      filesz 0x00000000 memsz 0x00001000 flags rw-
```

private flags = 0:

Sections:

| Idx | Name                                  | Size     | VMA      | LMA      | File off | Algn |
|-----|---------------------------------------|----------|----------|----------|----------|------|
| 0   | .reset                                | 00000008 | 00000000 | 00000000 | 00001000 | 2**2 |
|     | CONTENTS, ALLOC, LOAD, READONLY, CODE |          |          |          |          |      |
| 1   | .rela.got                             | 00000000 | 00000008 | 00000008 | 00001008 | 2**2 |
|     | CONTENTS, ALLOC, LOAD, READONLY, DATA |          |          |          |          |      |
| 2   | .text                                 | 00009234 | 00000008 | 00000008 | 00001008 | 2**2 |
|     | CONTENTS, ALLOC, LOAD, READONLY, CODE |          |          |          |          |      |
| 3   | .rodata                               | 00000084 | 0000923c | 0000923c | 0000a23c | 2**2 |
|     | CONTENTS, ALLOC, LOAD, READONLY, DATA |          |          |          |          |      |
| 4   | .gcc_except_table                     | 00000e00 | 00009400 | 00009400 | 0000a400 | 2**9 |
|     | CONTENTS, ALLOC, LOAD, READONLY, CODE |          |          |          |          |      |
| 5   | .data                                 | 000008c8 | 10000000 | 0000a200 | 0000c000 | 2**2 |
|     | CONTENTS, ALLOC, LOAD, DATA           |          |          |          |          |      |
| 6   | .bss                                  | 000000a0 | 100008c8 | 0000aac8 | 0000c8c8 | 2**2 |
|     | ALLOC                                 |          |          |          |          |      |
| 7   | .comment                              | 00000ba6 | 00000000 | 00000000 | 0000c8c8 | 2**0 |
|     | CONTENTS, READONLY                    |          |          |          |          |      |
| 8   | .debug_aranges                        | 00000fe0 | 00000000 | 00000000 | 0000d46e | 2**0 |
|     | CONTENTS, READONLY, DEBUGGING         |          |          |          |          |      |
| 9   | .debug_pubnames                       | 0000182b | 00000000 | 00000000 | 0000e44e | 2**0 |
|     | CONTENTS, READONLY, DEBUGGING         |          |          |          |          |      |
| 10  | .debug_info                           | 00033baf | 00000000 | 00000000 | 0000fc79 | 2**0 |
|     | CONTENTS, READONLY, DEBUGGING         |          |          |          |          |      |
| 11  | .debug_abbrev                         | 00005881 | 00000000 | 00000000 | 00043828 | 2**0 |
|     | CONTENTS, READONLY, DEBUGGING         |          |          |          |          |      |
| 12  | .debug_line                           | 0000d3c6 | 00000000 | 00000000 | 000490a9 | 2**0 |
|     | CONTENTS, READONLY, DEBUGGING         |          |          |          |          |      |
| 13  | .debug_frame                          | 00002170 | 00000000 | 00000000 | 00056470 | 2**2 |
|     | CONTENTS, READONLY, DEBUGGING         |          |          |          |          |      |
| 14  | .debug_str                            | 00002ac8 | 00000000 | 00000000 | 000585e0 | 2**0 |
|     | CONTENTS, READONLY, DEBUGGING         |          |          |          |          |      |
| 15  | .debug_loc                            | 0000b7ce | 00000000 | 00000000 | 0005b0a8 | 2**0 |
|     | CONTENTS, READONLY, DEBUGGING         |          |          |          |          |      |
| 16  | .debug_macinfo                        | 00b61658 | 00000000 | 00000000 | 00066876 | 2**0 |
|     | CONTENTS, READONLY, DEBUGGING         |          |          |          |          |      |
| 17  | .hstack                               | 00001000 | 11fff000 | 0000ab68 | 0000d000 | 2**0 |
|     | ALLOC                                 |          |          |          |          |      |
| 18  | .stack                                | 00001000 | 24007000 | 0000bb68 | 0000d000 | 2**0 |
|     | ALLOC                                 |          |          |          |          |      |
| 19  | .debug_ranges                         | 00000648 | 00000000 | 00000000 | 00bc7ece | 2**0 |
|     | CONTENTS, READONLY, DEBUGGING         |          |          |          |          |      |

SYMBOL TABLE:

```
00000000 1 d .reset 00000000 .reset
00000008 1 d .rela.got 00000000 .rela.got
00000008 1 d .text 00000000 .text
0000923c 1 d .rodata 00000000 .rodata
00009400 1 d .gcc_except_table 00000000 .gcc_except_table
10000000 1 d .data 00000000 .data
100008c8 1 d .bss 00000000 .bss
00000000 1 d .comment 00000000 .comment
00000000 1 d .debug_aranges 00000000 .debug_aranges
00000000 1 d .debug_pubnames 00000000 .debug_pubnames
00000000 1 d .debug_info 00000000 .debug_info
00000000 1 d .debug_abbrev 00000000 .debug_abbrev
00000000 1 d .debug_line 00000000 .debug_line
00000000 1 d .debug_frame 00000000 .debug_frame
00000000 1 d .debug_str 00000000 .debug_str
00000000 1 d .debug_loc 00000000 .debug_loc
00000000 1 d .debug_macinfo 00000000 .debug_macinfo
11fff000 1 d .hstack 00000000 .hstack
24007000 1 d .stack 00000000 .stack
00000000 1 d .debug_ranges 00000000 .debug_ranges
00000f78 1 .text 00000000 text_end_addr
00000f7c 1 .text 00000000 data_start_addr
00000f80 1 .text 00000000 data_end_addr
00000000 1 df *ABS* 00000000 system_call.c
00000000 1 df *ABS* 00000000 main_process_switch.c
10000000 1 O .data 00000004 nextPid
10000004 1 O .data 00000004 data_a
00000000 1 df *ABS* 00000000 port.c
00000000 1 df *ABS* 00000000 pcb.c
10000008 1 O .data 00000004 nextPid
1000000c 1 O .data 00000004 cnt
00000000 1 df *ABS* 00000000 init.c
00000000 1 df *ABS* 00000000 usart.c
10000010 1 O .data 00000004 usart_base
00000000 1 df *ABS* 00000000 syscalls.c
100008e0 1 O .bss 00000004 cur_heap_pos.2852
00000000 1 df *ABS* 00000000 interrupts.c
00000000 1 df *ABS* 00000000 exception_handling.c
00000000 1 df *ABS* 00000000 usart.c
00001994 1 F .text 000000c2 usart_set_baudrate
00000000 1 df *ABS* 00000000 pio.c
00000000 1 df *ABS* 00000000 uart_driver.c
10000018 1 O .data 00000004 usart1
1000001c 1 O .data 00000004 cpu_hz
00001f5c 1 O .text 00000004 C.0.9432
10000900 1 O .bss 0000000c opt
100008f0 1 O .bss 00000010 device_uid
00000000 1 df *ABS* 00000000 leds_driver.c
1000090c 1 O .bss 00000008 device_uid
00000000 1 df *ABS* 00000000 device_driver.c
100000b0 1 O .data 00000004 new_dev_id
00000000 1 df *ABS* 00000000 process2.c
00000000 1 df *ABS* 00000000 process1.c
00000000 1 df *ABS* 00000000 errno.c
00000000 1 df *ABS* 00000000 exit.c
00000000 1 df *ABS* 00000000 fflush.c
00000000 1 df *ABS* 00000000 findfp.c
0000260c 1 F .text 00000038 std
```

```
00000000 1      df *ABS* 00000000 fwalk.c
00000000 1      df *ABS* 00000000 impure.c
100000b8 1      O .data 00000400 impure_data
00000000 1      df *ABS* 00000000 malloc.c
00000000 1      df *ABS* 00000000 mallocr.c
00000000 1      df *ABS* 00000000 memset.c
00000000 1      df *ABS* 00000000 mlock.c
00000000 1      df *ABS* 00000000 printf.c
00000000 1      df *ABS* 00000000 sbrkr.c
00000000 1      df *ABS* 00000000 stdio.c
00000000 1      df *ABS* 00000000 strcpy.c
00000000 1      df *ABS* 00000000 vfprintf.c
000030fc 1      F .text 0000042c get_arg
00003528 1      F .text 00000028 __sprint_r
00002f04 1      O .text 00000100 chclass
00003004 1      O .text 0000006c state_table
00003070 1      O .text 0000006c action_table
000030dc 1      O .text 00000010 blanks.3643
000030ec 1      O .text 00000010 zeroes.3644
00000000 1      df *ABS* 00000000 writer.c
00000000 1      df *ABS* 00000000 wsetup.c
00000000 1      df *ABS* 00000000 __call_atexit.c
00000000 1      df *ABS* 00000000 closer.c
00000000 1      df *ABS* 00000000 dtoa.c
00005cc8 1      F .text 0000014c quorem
00000000 1      df *ABS* 00000000 fclose.c
00000000 1      df *ABS* 00000000 mallocr.c
00000000 1      df *ABS* 00000000 fvwrite.c
00000000 1      df *ABS* 00000000 locale.c
00007224 1      O .text 00000004 charset
00007228 1      O .text 00000030 lconv
00000000 1      df *ABS* 00000000 lseekr.c
00000000 1      df *ABS* 00000000 makebuf.c
00000000 1      df *ABS* 00000000 memchr.c
00000000 1      df *ABS* 00000000 memmove.c
00000000 1      df *ABS* 00000000 mprec.c
000075d0 1      O .text 0000000c p05.2619
00000000 1      df *ABS* 00000000 readr.c
00000000 1      df *ABS* 00000000 mallocr.c
00000000 1      df *ABS* 00000000 s_isinfd.c
00000000 1      df *ABS* 00000000 s_isnand.c
00000000 1      df *ABS* 00000000 strlen.c
00000000 1      df *ABS* 00000000 mallocr.c
00000000 1      df *ABS* 00000000 fstatr.c
00008418 1      .text 00000000 __avr32_f64_mul_op1_zero
000082f6 1      .text 00000000 __avr32_f64_mul_op1_subnormal
0000833a 1      .text 00000000 __avr32_f64_mul_op2_subnormal
0000837e 1      .text 00000000 __avr32_f64_mul_op_nan_or_inf
000083a6 1      .text 00000000 __avr32_f64_mul_res_subnormal
00008408 1      .text 00000000 __avr32_f64_mul_res_inf
000083f8 1      .text 00000000 __avr32_f64_mul_res_zero
000083a0 1      .text 00000000 __avr32_f64_mul_op1_not_naninf
00008402 1      .text 00000000 __avr32_f64_mul_res_nan
00008438 1      .text 00000000 __avr32_f64_sub_from_add
000085d4 1      .text 00000000 __avr32_f64_add_from_sub
00008506 1      .text 00000000 __avr32_f64_sub_opL_subnormal
00008488 1      .text 00000000 __avr32_f64_sub_opL_subnormal_done
0000852c 1      .text 00000000 __avr32_f64_sub_opH_nan_or_inf
```

```
000084b4 1      .text 00000000 __avr32_f64_sub_shift_done
00008594 1      .text 00000000 __avr32_f64_sub_longshift
000084da 1      .text 00000000 __avr32_f64_sub_longnormalize_done
000085b2 1      .text 00000000 __avr32_f64_sub_longnormalize
0000854a 1      .text 00000000 __avr32_f64_sub_subnormal_result
000084ec 1      .text 00000000 __avr32_f64_sub_round
00008542 1      .text 00000000 __avr32_f64_sub_return_nan
000086cc 1      .text 00000000 __avr32_f64_add_op2_subnormal
00008664 1      .text 00000000 __avr32_f64_add_opH_nan_or_inf
0000863c 1      .text 00000000 __avr32_f64_add_shift_done
00008692 1      .text 00000000 __avr32_f64_add_longshift
000086ae 1      .text 00000000 __avr32_f64_add_res_of
00008648 1      .text 00000000 __avr32_f64_add_res_of_done
0000864e 1      .text 00000000 __avr32_f64_add_round
0000868a 1      .text 00000000 __avr32_f64_add_return_nan
00008676 1      .text 00000000 __avr32_f64_add_opL_nan_or_inf
000087fe 1      .text 00000000 __floatsidf_return_op1
00008b44 1      .text 00000000 __avr32_f64_div_res_subnormal
00008a5a 1      .text 00000000 __avr32_f64_div_round_subnormal
00008bf4 1      .text 00000000 __avr32_f64_div_op1_zero
00000000 1      df *ABS* 00000000 libgcc2.c
00000000 1      df *ABS* 00000000 libgcc2.c
0000a154 1      .gcc_except_table 00000000 _handle_exception
0000a132 1      .gcc_except_table 00000000 _handle_breakpoint
0000a180 1      .gcc_except_table 00000000 _handle_scall
0000a14a 1      .gcc_except_table 00000000 _breakpoint_unhandled
0000a16a 1      .gcc_except_table 00000000 _exception_return
0000a176 1      .gcc_except_table 00000000 _exception_unhandled
0000a18c 1      .gcc_except_table 00000000 _int_disable
0000a19c 1      .gcc_except_table 00000000 _int_enable
00000000 1      df *ABS* 00000000 system_call_api.c
00000000 1      df *ABS* 00000000 sched.c
00000000 1      df *ABS* 00000000 halos.c
00000000 1      df *ABS* 00000000 cpu.c
00000000 1      df *ABS* 00000000 gdi_api.c
00000000 1      df *ABS* 00000000 pm.c
00000000 1      df *ABS* 00000000 mmu_exceptions.c
00000000 1      df *ABS* 00000000 mmu.c
00000000 1      df *ABS* 00000000 gdi_driver.c
00000000 1      df *ABS* 00000000 halosUtil.c
00000000 1      df *ABS* 00000000 Time.c
00000000 1      df *ABS* 00000000 strlcpy.c
00000000 1      df *ABS* 00000000 reent.c
10000020 g      O .data 00000090 actDevices
0000a11a g      .gcc_except_table 00000000 _int2
00007604 g      O .text 00000028 __mprec_tinytens
00002ea8 g      F .text 0000005c strcpy
00000260 g      F .text 00000024 start_first_process
00000870 g      F .text 00000128 ClkSelect
00007264 g      F .text 00000034 _lseek_r
00002db4 g      F .text 00000024 printf
00002e18 g      F .text 0000002c __sseek
00002648 g      F .text 00000054 __sinit
00002db2 g      F .text 00000002 __malloc_unlock
00008896 g      F .text 00000000 __avr32_f64_cmp_lt
10000014 g      O .data 00000004 __intc_base__
000006bc g      F .text 0000002c clk_switch
100008d0 g      O .bss 00000004 pcb2
```

```
0000746e g F .text 00000098 memmove
000077b8 g F .text 00000058 _Balloc
00000814 g F .text 0000005c GClkSetup
0000249c g F .text 0000000c __errno
000022fc g F .text 00000020 generate_dev_uid
00000008 g F .text 000000d8 scall_Handler
0000213c g F .text 00000038 leds_driver_init
100008d4 g O .bss 00000004 pcb3
00008228 g F .text 00000034 _fstat_r
10000960 g O .bss 00000004 errno
00001d48 g F .text 00000090 pioGetHandle
10000948 g O .bss 00000004 processStarted
0000825c g F .text 00000000 __avr32_f64_mul
00007420 g F .text 00000000 memcpy
00001c3a g F .text 00000060 usart_read_char
00001f00 g F .text 0000005c pio_enable_module
00001454 w F .text 00000090 _handle_interrupt
0000269c g F .text 00000014 _cleanup_r
000088f8 g F .text 00000000 __avr32_f64_div
11fff000 g .hstack 00000000 _hstack
00002798 g F .text 00000018 malloc
00002050 g F .text 00000070 uart_driver_read_byte
000010ee w F .text 00000016 isatty
000005c8 g F .text 000000f4 init_ngw100
00001104 w F .text 0000001a _fstat
0000133c g F .text 00000020 set_interrupts_base
00007508 g O .text 000000c8 __mprec_tens
10000914 g O .bss 00000004 __malloc_top_pad
100008d8 g O .bss 00000004 flag
1000094c g O .bss 00000004 global1
0000012c g F .text 000000b8 rtc_int_handler
0000a104 g .gcc_except_table 00000000 _int0
11fff000 g *ABS* 00000000 __heap_end__
00007258 g F .text 0000000c _localeconv_r
00007bb8 g F .text 0000001c __i2b
00006f5c g F .text 000002c8 __sfvwrite_r
00002dd8 g F .text 00000030 _sbrk_r
00007d1c g F .text 00000034 _read_r
10000950 g O .bss 00000004 globalVar2
00006bd8 g F .text 000000c4 _fclose_r
000025dc g F .text 00000030 fflush
10000940 g O .bss 00000004 __malloc_max_sbrked_mem
00008f1c g F .text 00000320 __umoddi3
0000a000 w .gcc_except_table 00000000 _evba
0000277c g O .text 00000004 _global_impure_ptr
00007d50 g F .text 000003d0 _realloc_r
00000ce4 g F .text 00000098 Get_HSB_Clk
00008c0c g F .text 00000310 __udivdi3
000075dc g O .text 00000028 __mprec_bigtens
00001120 w F .text 00000078 _sbrk
0000777e g F .text 00000038 __mcmp
100008ec g O .bss 00000004 handler_table
00001bf8 g F .text 00000042 usart_write_char
100008dc g O .bss 00000004 flag2
0000883c g F .text 00000000 __avr32_f64_cmp_ge
000001e4 g F .text 0000007a InitRTC
00007640 g F .text 00000088 __hi0bits
00002100 g F .text 0000003c uart_driver_close
```

```
10000954 g O .bss 00000004 globalVar1
100008c8 g O .bss 00000004 ptrCurrentPCB
000020c0 g F .text 00000040 uart_driver_write_byte
000086ec g F .text 00000000 __avr32_f64_to_s32
000086e8 g F .text 00000000 __avr32_f64_to_u32
00007c6c g F .text 000000b0 __pow5mult
00001dd8 g F .text 00000128 pio_setup_pin
10000958 g O .bss 00000004 i
00000368 g F .text 0000014a ptr_uc_init_stack
00002644 g F .text 00000002 __sfp_lock_acquire
0000737c g F .text 000000a4 memchr
0000168c w F .text 00000048 handle_scall
00001958 g F .text 0000003c usart_reset
00006d5c g F .text 00000200 _free_r
100008e8 g O .bss 00000004 int_initialized
000006e8 g F .text 00000048 ClkSwitch
00005bd4 g F .text 000000c4 __call_exitprocs
100008c0 g O .data 00000004 __malloc_sbrk_base
00000000 g F .reset 00000000 _start
100008e4 g O .bss 00000004 int_handler_table
00000534 g F .text 00000094 allocatePCBAndStack
00002354 g F .text 00000080 open_driver
000011bc w F .text 0000002c _lseek
000079c0 g F .text 000000bc __lshift
00008722 g F .text 00000000 __avr32_s32_to_f64
00001a58 g F .text 000001a0 usart_init_rs232
00002492 g F .text 00000008 user_process1
12000000 g .hstack 00000000 _hstack
000016d4 w F .text 000001fc unhandled_exception
00007a7c g F .text 0000013c __multiply
00002228 g F .text 0000007c leds_driver_set_bit
10000918 g O .bss 00000028 __malloc_current_mallinfo
00007810 g F .text 000000d4 __d2b
00000fb0 g F .text 0000003c usart_putc
00000c4c g F .text 00000098 Get_CPU_Clk
00005c98 g F .text 00000030 _close_r
00005b08 g F .text 000000cc __swsetup_r
100004b8 g O .data 00000408 __malloc_av_
00002e80 g F .text 00000028 __sread
00000dcc g F .text 00000078 delay_us
00002db0 g F .text 00000002 __malloc_lock
000024d4 g F .text 00000108 _fflush_r
000081b8 g F .text 00000070 _calloc_r
00001668 w F .text 00000024 _register_scall_handler
000022a4 g F .text 00000058 leds_driver_close
100008c8 g *ABS* 00000000 __bss_start
00002d50 g F .text 00000060 memset
00000284 g F .text 000000e4 main
00001370 w F .text 000000e4 init_interrupts
00001f60 g F .text 0000007c uart_driver_init
10000944 g O .bss 00000004 __malloc_max_total_mem
00008c0c g F .text 00000310 __avr32_udiv64
00002e08 g F .text 00000010 __sclose
00006c9c g F .text 00000018 fclose
00005e14 g F .text 00000dc4 _dtoa_r
000027b0 g F .text 000005a0 _malloc_r
10000968 g *ABS* 00000000 __heap_start__
0000843c g F .text 00000000 __avr32_f64_sub
```

```
00002718 g      F .text 00000064 _fwalk
000014e4 w      F .text 00000184 register_interrupt
00006cb4 g      F .text 000000a8 _malloc_trim_r
000004b4 g      F .text 00000080 processCreate
0000241a g      F .text 00000032 write_driver
000010d8 w      F .text 00000016 unimplemented_syscall
00005ad4 g      F .text 00000034 _write_r
000085d8 g      F .text 00000000 __avr32_f64_add
100000b4 g      O .data 00000004 _impure_ptr
100008cc g      O .bss 00000004 pcb1
10000000 g      *ABS* 00000000 _data
00008120 g      F .text 0000002c __isinf
24008000 g      .stack 00000000 _estack
000012ac w      F .text 00000090 _write
100008c8 g      *ABS* 00000000 _edata
00000e44 g      F .text 000000f0 SDRAMC_Init
10000968 g      *ABS* 00000000 _end
00002174 g      F .text 000000b4 leds_driver_open
0000a200 g      .data 00000000 _data_lma
0000871a g      F .text 00000000 __avr32_u32_to_f64
00002e44 g      F .text 0000003c __swrite
```