

BEFORE void* mem1 = _malloc(1024);

The screenshot shows a debugger interface with three main panels:

- Registers:** A list of registers R0 through R15. R0 is 0x20005848, R1 is 0x00000028, R2 is 0x42413938, R3 is 0x200057F8, R4 is 0x20005820, R5 is 0x4A494847, R6 is 0x46454443, R7 is 0x00000000, R8 is 0x00000000, R9 is 0x00000000, R10 is 0x000008B8, R11 is 0x00000000, R12 is 0x56555453, R13 (SP) is 0x200057F8, R14 (LR) is 0x00007D0C, R15 (PC) is 0x00007ACC. The xPSR register is 0x61000000.
- Disassembly:** Shows assembly code for the function `driver_keilc`. Lines 42-45 show `_free(mem6);`, `_free(mem5);`, `_free(mem1);`, and `_free(mem7);`. Lines 34-41 show memory allocation: `mem1 = _malloc(1024);`, `mem2 = _malloc(1024);`, `mem3 = _malloc(8192);`, `mem4 = _malloc(4096);`, `mem5 = _malloc(512);`, `mem6 = _malloc(1024);`, and `mem7 = _malloc(512);`.
- Memory 1:** A memory dump starting at address 0x20006800, showing a large block of zeroed-out memory.

The status bar at the bottom indicates "Simulation" mode, with a timestamp of "t1: 0.00208867 sec" and other system information.

void* mem1 = _malloc(1024);

The screenshot shows a debugger interface with three main panels:

- Registers:** A list of registers R0 through R15. R0 is 0x20001000, R1 is 0x00000028, R2 is 0x00000028, R3 is 0x42413938, R4 is 0x200057F8, R5 is 0x20005820, R6 is 0x4A494847, R7 is 0x46454443, R8 is 0x00000000, R9 is 0x00000000, R10 is 0x20001000, R11 is 0x00000000, R12 is 0x56555453, R13 (SP) is 0x200057F8, R14 (LR) is 0x00007D0C, R15 (PC) is 0x00007D0C. The xPSR register is 0x61000000.
- Disassembly:** Shows assembly code for the function `driver_keilc`. Lines 29-32 show `_bzero(stringB, 40);`, `_strcpy(stringB, stringA, 40);`, and `_bzero(stringA, 40);`. Lines 34-41 show memory allocation: `mem1 = _malloc(1024);`, `mem2 = _malloc(1024);`, `mem3 = _malloc(8192);`, `mem4 = _malloc(4096);`, `mem5 = _malloc(512);`, `mem6 = _malloc(1024);`, and `mem7 = _malloc(512);`. Lines 42-46 show `_free(mem6);`, `_free(mem5);`, `_free(mem1);`, `_free(mem7);`, and `_free(mem2);`. Line 48 shows `mem8 = _malloc(4096);`.
- Memory 1:** A memory dump starting at address 0x20006800, showing a large block of zeroed-out memory.

The status bar at the bottom indicates "Simulation" mode, with a timestamp of "t1: 0.00212592 sec" and other system information.

Registers

Register	Value
Core	
R0	0x20001400
R1	0x00000028
R2	0x00000028
R3	0x24139338
R4	0x200057f8
R5	0x20005820
R6	0x4A494847
R7	0x46454443
R8	0x00000000
R9	0x20001400
R10	0x20001000
R11	0x00000000
R12	0x56555453
R13 (SP)	0x200057f8
R14 (LR)	0x000007df
R15 (PC)	0x000007e0
xPSR	0x61000200
Banked	
System	
Internal	
Mode	Thread
Privilege	Privileged
Stack	PSP
States	26004
Sec	0.00216700
FPU	

Disassembly

Address	Instruction	Comment
0x000007D0	F7FFFD82 BL.W	0x000002D8 _malloc
0x000007D4	4682 MOV	r10,r0
35:	mem2 = _malloc(1024);	
0x000007D6	F4F6080 MOV	r0,#0x400
0x000007DA	F7FFFD7D BL.W	0x000002D8 _malloc
0x000007DE	4681 MOV	r9,r0
36:	mem3 = _malloc(8192);	
0x000007E0	F4F5000 MOV	r0,#0x2000

driver_keil.c

```

30  _strncpy( stringB, stringA, 40 );
31  _bzero( stringA, 40 );
32
33
34  mem1 = _malloc( 1024 );
35  mem2 = _malloc( 1024 );
36  mem3 = _malloc( 8192 );
37  mem4 = _malloc( 4096 );
38  mem5 = _malloc( 512 );
39  mem6 = _malloc( 1024 );
40  mem7 = _malloc( 512 );
41
42  _free( mem6 );
43  _free( mem5 );
44  _free( mem1 );
45  _free( mem7 );
46  _free( mem2 );
47
48  mem8 = _malloc( 4096 );
49

```

The screenshot displays the Disassembly window in Immunity Debugger. The left pane shows the Register list with R3, R1, R2, R3, R4, R5, R6, R7, R8, R9, R10, R11, R12, R13 (SP), R14 (LR), and R15 (PC) selected. The main pane shows the disassembly of the 'driver_kell.c' file. The assembly code includes instructions like 'BL.W', 'MOV', and 'MEM4 = malloc(4096);'. A red arrow points to the instruction 'MEM3 = malloc(8192);'.

[illegible]

```
void* mem4 = _malloc(4096 );
```

The screenshot shows the Immunity Debugger interface. The top window is the Disassembly window, displaying assembly code for the 'driver_kell.c' file. The code includes instructions for allocating memory (malloc) and setting up the stack. A red arrow points to the instruction 'mem4 = _malloc(4096);' at address 0x000007F4. The bottom window is the Memory window, showing the memory dump for address 0x20006800. The memory dump shows a series of zeroed-out bytes, with a red arrow pointing to the instruction 'mem4 = _malloc(4096);'.

```
void* mem5 = _malloc(512);
```

The screenshot displays the Keil IDE interface with two main windows open: 'Disassembly' and 'Memory'.

Disassembly Window:

- Register List:** Shows registers R0 through R15, with R0, R1, R2, R3, R4, R5, R6, R7, R8, R9, R10, R11, R12, R13 (SP), R14 (LR), and R15 (PC) listed. The 'Core' register is selected.
- Disassembly Code:**

```

0x000007F8 F7FFD6E BL.W      0x000002D8 _malloc
0x000007FC 4604      MOV      r4,r0
39: mem6 = _malloc( 1024 );
0x000007FE F44F608 MOV      r0,#0x400
0x00000802 F7FFD69 BL.W      0x000002D8 _malloc
0x00000806 4605      MOV      r5,r0
40: mem7 = _malloc( 512 );
41:

```
- Source Code:**

```

33 mem1 = _malloc( 1024 );
34 mem2 = _malloc( 1024 );
35 mem3 = _malloc( 8192 );
36
37 mem4 = _malloc( 4096 );
38 mem5 = _malloc( 512 );
39 mem6 = _malloc( 1024 );
40 mem7 = _malloc( 512 );
41
42 _free( mem6 );
43 _free( mem5 );
44 _free( mem1 );
45 _free( mem7 );
46 _free( mem2 );
47
48 mem8 = _malloc( 4096 );
49
50 _free( mem4 );
51 _free( mem3 );
52 _free( mem8 );

```

Memory Window:

- Address:** 0x20006800
- Memory Content:** A hex dump showing the memory layout. The address 0x20006800 is highlighted.

A red arrow points to the instruction `mem5 = _malloc(512);` in the source code window.

```
void* mem6 = _malloc(1024 );
```

The screenshot displays the Disassembly window of a debugger, showing assembly code and a memory dump.

Registers Panel:

Register	Value
R0	0x20001C00
R1	0x00000028
R2	0x00000028
R3	0x42413938
R4	0x20001800
R5	0x20001C00
R6	0x4A494847
R7	0x20002000
R8	0x20003000
R9	0x20001400
R10	0x20001000
R11	0x00000000
R12	0x56555453
R13 (SP)	0x200057F8
R14 (LR)	0x00000807
R15 (PC)	0x00000808
xPSR	0x61000200

Disassembly Window:

Assembly code is shown with a yellow highlight on the instruction at address 0x00000808:

```

0x000007FC 4604 MOV r4,r0
39: mem6 = _malloc( 1024 );
0x000007FE F44F6080 MOV r0,#0x400
0x00000802 F7FFFD69 BL.W 0x000002D8 _malloc
0x00000806 4605 MOV r5,r0
40: mem7 = _malloc( 512 );
41:
0x00000808 F44F7000 MOV r0,#0x200

```

The source code file is `driver_keil.c`. The code snippet shows memory allocation and deallocation:

```

34 mem1 = _malloc( 1024 );
35 mem2 = _malloc( 1024 );
36 mem3 = _malloc( 8192 );
37 mem4 = _malloc( 4096 );
38 mem5 = _malloc( 512 );
39 mem6 = _malloc( 1024 );
40 mem7 = _malloc( 512 );
41
42 _free( mem6 );
43 _free( mem5 );
44 _free( mem1 );
45 _free( mem7 );
46 _free( mem2 );
47
48 mem8 = _malloc( 4096 );
49
50 _free( mem4 );
51 _free( mem3 );
52 _free( mem8 );
53

```

Memory Window:

The memory dump shows a large block of zeroed-out memory starting at address 0x20006800. The address 0x20006800 is highlighted in the address field.

Address: 0x20006800

The memory dump shows a large block of zeroed-out memory (0x00) across multiple lines.

```
void* mem7 = _malloc(512 );
```

The screenshot displays the Keil IDE interface during a simulation. The top window is the Disassembly view, showing assembly code for a driver. A red arrow points to the instruction `mem7 = _malloc(512);` at address 0x20006800. The bottom window is the Memory view, showing a memory dump starting at address 0x20006800. A red arrow points to the first byte of the dump, which is 0x00.

```
_free(mem6);
```

The screenshot shows the Immunity Debugger interface. The top window is the Disassembly window, displaying assembly code for a function named 'driver_kell.c'. The code includes memory allocation and deallocation instructions. A red arrow points to the instruction 'free(mem6);' at address 0x00000819. The bottom window is the CPU registers window, showing the state of various registers. The Program Counter (PC) is highlighted in blue, indicating the current instruction being executed.

```

_free(mem5);

```

The screenshot displays the Keil IDE interface during a simulation. The **Disassembly** window is active, showing assembly code for a function. A red arrow points to the instruction `free(mem6);` at address `0x0000081E`. The **Memory** window is also visible, showing a memory dump for the address `0x20006800`. A red arrow points to the value `0x00000000` at address `0x20006800`.


```
_free(mem1);
```

[illegible]

```
_free(mem7);
```

The screenshot shows the Immunity Debugger interface. The Disassembly window is active, displaying assembly code for the 'driver_kell.c' file. The code includes memory allocation and deallocation functions. A red arrow points to the instruction 'free(mem7);' at address 0x00000826. The Memory window is also visible, showing the memory dump at address 0x20006800. A red arrow points to the memory location 0x20006800.

```
_free(mem2);
```

The screenshot displays the Keil uVision IDE interface, specifically the Disassembly and Memory windows.

Disassembly Window:

- Registers:** The left pane shows the register list. R0 is selected, with a value of 0x20006840.
- Disassembly:** The main pane shows assembly code for the file 'driver_keil.c'. The code includes:


```

37 mem4 = _malloc( 4096 );
38 mem5 = _malloc( 512 );
39 mem6 = _malloc( 1024 );
40 mem7 = _malloc( 512 );
41
42 _free( mem6 );
43 _free( mem5 );
44 _free( mem1 );
45 _free( mem7 );
46 _free( mem2 );
47
48 mem8 = _malloc( 4096 );
49
50 _free( mem4 );
51 _free( mem3 );
52 _free( mem8 );
53
54 return 0;
55 }
56
```

 A red arrow points to line 46, which is highlighted in yellow.

Memory Window:

- Address:** The address 0x20006800 is entered in the address field.
- Memory Dump:** The main pane shows a memory dump. The first few lines of the dump are:


```

0x20006800: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006810: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006820: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006830: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006840: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006850: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006860: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006870: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006880: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006890: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x200068A0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x200068B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x200068C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x200068D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x200068E0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x200068F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006900: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006910: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006920: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006930: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006940: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006950: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006960: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006970: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006980: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006990: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x200069A0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x200069B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x200069C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x200069D0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x200069E0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x200069F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006A00: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006A10: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006A20: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006A30: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006A40: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006A50: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006A60: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006A70: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006A80: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006A90: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006AA0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006AB0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006AC0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006AD0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006AE0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006AF0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006B00: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006B10: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006B20: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006B30: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006B40: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006B50: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006B60: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006B70: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006B80: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006B90: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006BA0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006BB0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x20006BC0: 00 00 00
```

```
void* mem8 = _malloc(4096 );
```

The screenshot displays a debugger's Disassembly window. On the left, the 'Registers' pane lists various registers, with 'R15 (PC)' selected and showing a value of '0x0000083A'. The main window shows assembly code for 'driver_kell.c'. Line 48, 'mem8 = _malloc(4096);', is highlighted in yellow, and a red arrow points to it. The 'Memory' pane at the bottom shows the address '0x20006800'.

_free(mem4);

The screenshot shows the Keil uVision IDE with the assembly window open. The assembly code is for the function `_free` in `system_TM4C129.c`. The code is as follows:

```
0x0000083A 4638 MOV r0,r7
0x0000083C F7FFFD54 BL.W 0x000002E8 _free
S1: _free( mem3 );
0x00000840 4640 MOV r0,r8
0x00000842 F7FFFD51 BL.W 0x000002E8 _free
S2: _free( mem8 );
S3:
0x00000846 4620 MOV r0,r4
```

The registers window shows the following values:

Register	Value
R0	0x20006300
R1	0x00000028
R2	0x00000028
R3	0x42413938
R4	0x20001000
R5	0x20001C00
R6	0x20001A00
R7	0x20002000
R8	0x20003000
R9	0x20001400
R10	0x20001000
R11	0x00000000
R12	0x56555453
R13 (SP)	0x200057F8
R14 (LR)	0x00000841
R15 (PC)	0x00000840
xPSR	0x61000200

The memory window shows the memory dump starting at address 0x20006800. The memory is filled with zeros.

_free(mem3);

The screenshot shows the Keil uVision IDE with the assembly window open. The assembly code is for the function `_free` in `system_TM4C129.c`. The code is as follows:

```
0x0000083A 4638 MOV r0,r7
0x0000083C F7FFFD54 BL.W 0x000002E8 _free
S1: _free( mem3 );
0x00000840 4640 MOV r0,r8
0x00000842 F7FFFD51 BL.W 0x000002E8 _free
S2: _free( mem8 );
S3:
0x00000846 4620 MOV r0,r4
```

The registers window shows the following values:

Register	Value
R0	0x20006A00
R1	0x00000028
R2	0x00000028
R3	0x42413938
R4	0x20001000
R5	0x20001C00
R6	0x20001A00
R7	0x20002000
R8	0x20003000
R9	0x20001400
R10	0x20001000
R11	0x00000000
R12	0x56555453
R13 (SP)	0x200057F8
R14 (LR)	0x00000841
R15 (PC)	0x00000846
xPSR	0x61000200

The memory window shows the memory dump starting at address 0x20006800. The memory is filled with zeros.

_free(mem8);

The screenshot displays a debugger interface with three main panels: Registers, Disassembly, and Memory.

Registers Panel: Shows the state of various registers. R0 is highlighted with a value of 0x00000000. R15 (PC) is at 0x0000084C. The xPSR is at 0x61000200.

Disassembly Panel: Shows the assembly code for the function `driver_keil.c`. The current instruction is `MOV r0, #0x00` at address `0x0000084C 2000`. A red arrow points to the instruction `_free(mem8);` at address `0x0000084C 52`.

Memory Panel: Shows a memory dump starting at address `0x20006800`. A red bracket highlights the first 16 bytes of the dump, which are all zeros.

Code Snippets:

```
37 mem4 = _malloc( 4096 );
38 mem5 = _malloc( 512 );
39 mem6 = _malloc( 1024 );
40 mem7 = _malloc( 512 );
41
42 _free( mem6 );
43 _free( mem5 );
44 _free( mem1 );
45 _free( mem7 );
46 _free( mem2 );
47
48 mem8 = _malloc( 4096 );
49
50 _free( mem4 );
51 _free( mem3 );
52 _free( mem8 );
53
54 return 0;
```