

7-Segments controller specification

byte offset	word offset	Write action:	Read action:
0	0	Write segments of display 1	Read segments of display 1
4	1	Write segments of display 2	Read segments of display 2
8	2	Write segments of display 3	Read segments of display 3
12	3	Write segments of display 4	Read segments of display 4
16	4	Write hexadecimal to displays	Read segments of display 1
20	5	Write BCD-coded to displays	Read segments of display 2
24	6	Write decimal points	Read segments of display 3
28	7	Write the base Address	Read the base Address

Table 1: Offset mapping of the seven segment display control

The base address of the seven segment controller is `0x50000060`. There are different ways to control the seven segments as shown in table 1. Display 1 is the most-right seven segment display, and display 4 the most left seven segment display. This controller only support word (32-bit) transfers, and does not support burst accesses. If these conditions are not met, the controller will generate a bus-error.

1 Writing the segments

Each of the segments of a seven segment display can be controlled by writing a bit 1 to activate or a bit 0 to deactivate it. The definition of the segments is shown below (based on a `uint32_t`):

31	..	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	dp	g	f	e	d	c	b	a

In the above table a - denotes a don't care, and `dp` the decimal point.

2 Writing a hexadecimal value

A 16-bit hexadecimal value can also be written directly to the display's. Note that in this case the decimal points retain their current value. To write the hexadecimal value the displays are shown below:

31..16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	display4				display3				display2				display1			

3 Writing a BCD-value

A 16-bit Binary Coded Decimal (BCD) value can also be written directly to the display's. Note that in this case the decimal points retain their current value. Writing a nibble that is out of the range 0..9 will result in an empty display. To write the BCD-value the displays are shown below:

31..16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	display4				display3				display2				display1			

4 Writing the decimal points

Each of the decimal points of a seven segment display can be controlled by writing a bit 1 to activate or a bit 0 to deactivate it. The definition of the decimal points is shown below (based on a `uint32_t`):

31	..	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	-	-	-	-	dp4	dp3	dp2	dp1

In the above table a - denotes a don't care, and **dp**x the decimal point of display x.

5 Writing the base address

Important: Do not use this function if you do not know what you are doing, as it could result in funny results.

By default the seven segment controller is at the memory address `0x50000060`, however, it can be moved in the memory space by writing a complete 32-bit memory address to this location. Please note that the bits 4..0 of this address must be 0.