

1. Flash at 0x0000_0000 and SRAM at 0x2000_0000 can overlap if the flash is large enough.
☐ True ☐ False
2. Why must a pointer to a hardware status register be volatile?
☐ The hardware can change the value, so every access must really read it
☐ It makes the code run faster
☐ Registers cannot be read without it
☐ It stops the register changing
3. With memory-mapped I/O, the CPU uses ordinary load and store instructions to talk to a peripheral.
☐ True ☐ False
4. A timer runs from 84 MHz with prescaler 84 and reload 1000. What is the interrupt rate in Hz?
_____ Hz
5. At 115 200 baud, 8N1, one byte takes
☐ 1 ms
☐ 86.8 μ s (10 bit times)
☐ 8.68 μ s
☐ 69.4 μ s (8 bit times)
6. An 8-bit ADC uses a 3.3 V reference. What is one LSB, in mV? Give 1 decimal place.
_____ mV
7. Bisection searches 512 suspect instructions. How many probes does it need at most?

8. 8 KB of SRAM needs how many address lines?
☐ 8192
☐ 13
☐ 8
☐ 16
9. Each line of assembly corresponds to one machine instruction.

10. A pin stays an input no matter what you write to its data register. The likely cause is

- ☐
- Its direction-register bit was never set to output
-
- ☐
- The clock is too slow
-
- ☐
- volatile is missing
-
- ☐
- The data register is broken

1. **False**

The regions differ in their top address bits, so no address can belong to both; the decoder separates them by those bits.

2. **The hardware can change the value, so every access must really read it**

Without ‘volatile’ the compiler may read once and reuse the value, so a loop waiting for a status bit can spin for ever.

3. **True**

The peripheral’s registers sit at addresses, so a read or write to that address reaches the hardware instead of RAM.

4. **1000 Hz**

$84 \times 10^6 / (84 \times 1000) = 1000$ Hz: a 1 ms tick, with both factors inside 16 bits.

5. **86.8 μ s (10 bit times)**

8N1 is 10 bits per byte, and $10/115\,200 = 86.8$ μ s. The start and stop bits are part of the price.

6. **12.9 mV**

$3.3/2^8 = 3.3/256 = 12.9$ mV. Each extra bit halves the step.

7. **9**

Each probe halves the range: $\log_2 512 = 9$.

8. **13**

8 KB = $8192 = 2^{13}$ bytes, so 13 lines; it spans ‘0x2000_0000’ to ‘0x2000_1FFF’.

9. **True**

That is what makes assembly useful for checking compiler output; one C line often becomes several instructions.

10. **Its direction-register bit was never set to output**

The direction register decides which way the pin works; the data register only carries the level. Set the direction first.