Embedded Systems: Homework #4

Yousef Alaa Awad

November 2, 2025

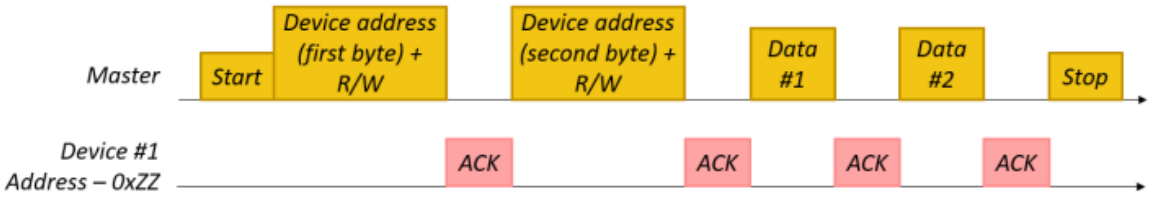
1

A)

When you reserve the 3 most significant bits for special use, then you only have 4 other bits that can be used for addressing. This, therefore, means that you can only have up $2^N = 2^4 = 16$ different/unique addresses. If, for some reason, you wanted more than 16 devices, as well as if the chip select/power-gating is also not available, then total bits should be updated to be 10 bits in length, while keeping the 3 reserved as before. Meaning you now have 7 unique bits for addressing or $2^N = 2^7 = 128$ unique/different addresses/devices that can be supported at a time.

B)

The following diagram (taken from the slides) would be the timing diagram of sending two bytes of data when using 10-bit addressing mode.



C)

Given that we are in fast mode, we know that the frequency is 400KHz. Due to this, using the timing diagram from Part B, we can find that the total clock cycles required for the transmission wanted to be...

- Start (master): 1 cycle
- Address Byte (master): 8 cycles
- ACK (slave):1 cycle
- Address Byte (master): 8 cycles
- ACK (slave):1 cycle
- Address Byte (master): 8 cycles
- ACK (slave):1 cycle
- Address Byte (master): 8 cycles
- ACK (slave):1 cycle
- Stop (master): 1 cycle

This would then take a total of 38 cycles per transmission or, $38 * \frac{1}{400*10^3} \approx 95\mu s$

2

A)

The master would start communicating with device 2 located at physical address 0x30. The communication would specifically be a READ (as the master is requesting data). Device 2 would then send the first data to master, and in response master will send an ACK (acknowledge). After this, device 1 will then send data as usual until the master sends a NACK (no acknowledge) for no more data transmissions and STOP to stop all communications.

B)

Now, the issue with the diagram shown is that the data transfer can only happen between the master and one slave at a time. To fix this, the master has to also realize that device 1 exists and deal with it accordingly after reading from device 2. Specifically, after reading from device 2, the master should send a NACK so that it does not read any of device 1's data stream. After this, the master can then specifically send a start to device 1 so that it can start sending data again and then a NACK to stop it from going to device 2.

3

A)

To find the quantization step size we need to the following:

$$q_{10b} = \frac{3.3V - 0V}{2^{10}} = 3.223mV$$

$$q_{12b} = \frac{3.3V - 0V}{2^{12}} = 0.806mV$$

B)

Now, to find the quantization *error*, we first need to find the maximum possible difference between the analog value and the digital corresponsing value. Since there are discrete steps in digital, the value of digital step in analog will be that error. This, therefore, means that the error is simply half of the step size (as it will round accordingly in the hardware) meaning....

$$q_{error10b} = \frac{3.223}{2}mV = 1.611mV$$

$$q_{error12b} = \frac{0.806}{2}mV = 0.403mV$$

C)

First we need to convert the temperature to voltage range, with 0 being 0V and 100 being 3.3V. Therefore we get...

$$\frac{37}{100} * 3.3 = 1.22V$$

After this, we will now convert using a 10 bit Analog to Digital Converter...

$$a_9 \rightarrow Digital_{Value} = 0.5 * 3.3 = 1.65 > 1.22 \rightarrow 0$$

$$a_8 \rightarrow Digital_{Value} = 0.25 * 3.3 = 1.65 > 1.22 \rightarrow 1$$

$$a_7 \rightarrow Digital_{Value} = 0.125 * 3.3 = 1.65 > 1.22 \rightarrow 0$$

$$...$$

$$a_1 \rightarrow Digital_{Value} = 0.001953125 * 3.3 = 0.00064453125 < 1.22 \rightarrow 0$$

And therefore get the Digital Value with 10 bits to be $(0101111010)_2$ or 378_{10}. If we repeated the same process but with 12-bits, we would instead get $(010111101010)_2$ or 1514_{10}.

D)

Now, to find the actual voltage the ADC would report/output back when re-converted back out we simply need to do the following:

$$ADC_{10} = 378 * 3.223mV = 1.218V$$

$$ADC_{12} = 1514 * 0.806mV = 1.220V$$

As you can see it is not exact but the more precision we have, the more accurate it is.

4

A)

First we find the total capacitance of the given circuit we simply do the following:

$$C_t = 8 + 3.5 + 2.1 + 1.4 = 15$$

After this, we can now find the value of each digital value like so...

$$a_4 \rightarrow V_{comp} = \frac{8 * 3.3}{15} = 1.76 \rightarrow 1.76 < 2.0 \rightarrow a_4 = 1$$

$$a_3 \rightarrow V_{comp} = \frac{3.5 * 3.3}{15} = 2.51 \rightarrow 2.51 > 2.0 \rightarrow a_3 = 0$$

$$a_2 \rightarrow V_{comp} = \frac{2.1 * 3.3}{15} = 2.22 \rightarrow 2.22 > 2.0 \rightarrow a_2 = 0$$

$$a_1 \rightarrow V_{comp} = \frac{1.4 * 3.3}{15} = 2.068 \rightarrow 2.068 > 2.0 \rightarrow a_1 = 0$$

Therefore the non-ideal digital output is 1000 or 8 in decimal.

B)

First we find the total capacitance of the given circuit we simply do the following:

$$C_t = 8 + 4 + 2 + 1 = 15$$

After this, we can now find the value of each digital value like so...

$$a_4 \rightarrow V_{comp} = \frac{8 * 3.3}{15} = 1.76 \rightarrow 1.76 < 2.0 \rightarrow a_4 = 1$$

$$a_3 \rightarrow V_{comp} = \frac{4 * 3.3}{15} = 2.64 \rightarrow 2.64 > 2.0 \rightarrow a_3 = 0$$

$$a_2 \rightarrow V_{comp} = \frac{2 * 3.3}{15} = 2.20 \rightarrow 2.20 > 2.0 \rightarrow a_2 = 0$$

$$a_1 \rightarrow V_{comp} = \frac{1 * 3.3}{15} = 1.98 \rightarrow 1.98 < 2.0 \rightarrow a_1 = 1$$

Therefore the ideal digital output is 1001 or 9 in decimal.

C)

While the non-ideal digital value is 1000 in part a, the ideal digital value is 1001 in part b, showing an error of 1 or $(1 * \frac{3.3}{16} \approx 0.20625V)$