

# Written Exercise Sheet 3

(20 Punkte)

**Hints:** These assignments will be discussed at E23, from 10:15 am - 11:45 am on 01, Dec., 2015. You are not obligated to turn in the solutions.

## 1 Hardware in a loop (2 Punkte)

Explain the concept “hardware in a loop”. In the era of Internet of Things (IoT), for some cases, we are replacing the sensors with *human beings*. What are the potential issues when we use human beings to collect the information?

## 2 Internet of Things (IoT) (2 Punkte)

IoT's have become a very hot topic recently. How will you define IoT? How will IoT change the (focus of) the concept of “hardware in a loop”? If a company is very good at controlling and designing actuators, what are the challenges that the company may face if IoT become mature?

## 3 Aliasing (2 Punkte)

Do we lose information due to sampling? What is the problem due to *Aliasing*? Suppose that the incoming signal has a frequency 8000Hz, e.g., human voices. What is the minimum sampling rate we need to avoid aliasing?

## 4 Conversion (2 Punkte)

Suppose our voltage range  $V_{min}..V_{max}$  is [0 V .. 4 V] and  $V_{ref}$  is 4 V. Digital values are supposed to be encoded as unsigned numbers in 4 bits. Input voltages are  $V_{in} = 2.3$  V; 3.7 V and 1.8 V. Conversion is assumed to be done in cycles by successive approximation. Specify for each input voltage and for each step of the approximation cycles:

- the voltage used for comparison,
- the intermediate binary representation.

## 5 Pipeline Analog-Digital-Wandler (3 Punkte)

Eine weitere Möglichkeit analoge Signale zu digitalisieren stellt der Pipeline Analog-Digital-Wandler.

- Skizzieren/Beschreiben Sie die generelle Funktionsweise eines Pipeline Analog-Digital-Wandlers.
- Nennen Sie Vor- und Nachteile eines Pipeline Analog-Digital-Wandlers.
- Vergleichen Sie den Pipeline Analog-Digital-Wandler in mit den bekannten Flash und Sukzessive Approximation Wandlern.

**Hinweis:** Verwenden Sie für die Recherche auch das Internet.

**Hint:** Pipeline-Umsetzer sind mehrstufige Analog-Digital-Umsetzer mit mehreren selbständigen Stufen, die in Pipeline-Architektur aufgebaut sind. Ihre Stufen bestehen in der Regel aus Flash-Umsetzern über wenige Bits.

In jeder Pipelinestufe wird eine grobe Quantisierung vorgenommen, dieser Wert wieder mit einem DAU in ein analoges Signal umgesetzt und vom zwischengespeicherten Eingangssignal abgezogen. Der Restwert wird verstärkt der nächsten Stufe zugeführt. Der Vorteil liegt in der stark verminderten Anzahl an Komparatoren, z. B. 30 für einen zweistufigen Acht-Bit-Umsetzer. Ferner kann eine höhere Auflösung erreicht werden. Die Mehrstufigkeit erhöht die Latenzzeit, aber vermindert die Abtastrate nicht wesentlich. Die Pipeline-Umsetzer haben die echten Parallelumsetzer außer bei extrem zeitkritischen Anwendungen ersetzt. Diese mehrstufigen Umsetzer erreichen Datenraten von 250 MSPS (Mega-Samples Per Second) bei einer Auflösung von 12 Bit (MAX1215, AD9480) oder eine Auflösung von 16 Bit bei 200 MSPS (ADS5485). (wikipedia)

Vorteil: (ziemlich) Schnell, weniger Komponenten -> höhere Auflösung Nachteile: Pipeline-Struktur -> Latenz

Einordnung

Flash | Pipeline | Succ Geschwindigkeit: ++ | + | - Kosten: - | + | +

## 6 Energy and Power (1 Punkt)

Energy and power consumption are both important in embedded systems. Give concrete reasons for minimizing power consumption. Give concrete reasons for minimizing energy consumption.

## 7 DVFS (2 Punkte)

Which equation describes the (dynamic) power consumption of CMOS? Let's assume that the static power consumption is 0 and the execution time is proportional to  $1/f$  (not true in practice), where  $f$  is the CPU frequency. Suppose that the CPU frequency is proportional to the supply voltage. Draw the power consumption with respect to the CPU frequency. To finish the same workload by reducing the CPU frequency from 1 GHz to 0.5 GHz, how much energy can we save? If we can use  $M \geq 2$  processors to finish the same workload at the same time span, draw the energy consumption with respect to  $M$ .

### 8 Saturating arithmetic (3 Punkte)

Suppose that the numbers in the following table are given. Please compute the result of the indicated operation using wrap-around and saturating arithmetic!

| a          |                    | b          |                    | op | a op b      |            |
|------------|--------------------|------------|--------------------|----|-------------|------------|
| (un)signed | bitvector          | (un)signed | bitvector          |    | wrap-around | saturating |
| signed     | 00110 <sub>2</sub> | signed     | 00110 <sub>2</sub> | +  |             |            |
| signed     | 01110 <sub>2</sub> | signed     | 01110 <sub>2</sub> | +  |             |            |
| signed     | 11110 <sub>2</sub> | signed     | 11110 <sub>2</sub> | +  |             |            |
| signed     | 01010 <sub>2</sub> | signed     | 01010 <sub>2</sub> | *  |             |            |
| unsigned   | 10110 <sub>2</sub> | unsigned   | 10110 <sub>2</sub> | +  |             |            |

### 9 Fixed point arithmetic (3 Punkte)

Suppose that the fixed-point numbers in the following table are given. Numbers are assumed to be signed. Please compute the result of the indicated operation!

| a          |                    | b          |                    | op | a op b     |           |
|------------|--------------------|------------|--------------------|----|------------|-----------|
| (iwl, fwl) | bitvector          | (iwl, fwl) | bitvector          |    | (iwl, fwl) | bitvector |
| (3,2)      | 00110 <sub>2</sub> | (3,2)      | 00110 <sub>2</sub> | +  | (3,2)      |           |
| (3,2)      | 00110 <sub>2</sub> | (3,2)      | 00110 <sub>2</sub> | +  | (4,1)      |           |
| (3,2)      | 00110 <sub>2</sub> | (4,1)      | 00110 <sub>2</sub> | +  | (4,1)      |           |
| (3,2)      | 00010 <sub>2</sub> | (4,1)      | 00010 <sub>2</sub> | +  | (2,1)      |           |
| (3,2)      | 00110 <sub>2</sub> | (4,1)      | 00110 <sub>2</sub> | *  | (8,2)      |           |