Introduction

Memory Map, Topic 2, Topic 3

Exercise class 1

Presenter:
Jürgen Mattheis

In cooperation with:
Pascal Walter

Based on the lecture of:
Marco Zimmerling

November 8, 2022

University of Freiburg, Chair for Embedded Systems

Gliederung

About this exercise class

About this exercise class

- based on the new lecture by Prof. Zimmerling
- the exercise class gets recorded and streamed online and is also hold in presence
- in rotation with the other tutor Pascal
- feedback for me: `https://forms.gle/f3YN8EFrZ1vsfPoC6`
- today a little preview of the exercises that await you
- solving the exercises not directly necessary for the Studienleistung, important for exam $\Rightarrow$ by passing the exam you're also going to get your Studienleistung

Task 1 - Memory map

Task 1 - Memory Map

Task 1.1

Solution a:

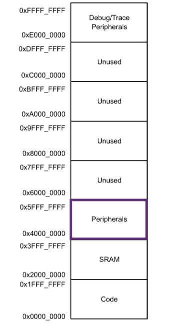


Figure 6-1. Device Memory Zones

- $0x5FFF\_FFFF - 0x4000\_0000 + 1 = 0x2000\_0000$
- $0x2000\_0000 = 2 \cdot 16^7 = 2 \cdot (2^4)^7 = 2 \cdot 2^{4 \cdot 7} = 2^{1+28} = 2^{29}$

Task 1 - Memory Map

Task 1.1

Solution b:

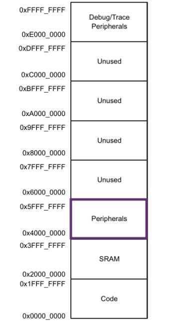


Figure 6-1. Device Memory Zones

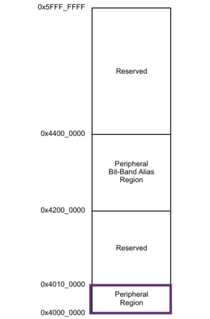


Figure 6-4. Peripheral Zone Memory Map

6.3.3.1 Peripheral Region

The 1MB region from 0x4000_0000 to 0x400F_FFFF is dedicated to the system and application control peripherals of the device. On the MSP432P401x MCUs, a total of 128KB of this region is dedicated for peripherals, while the rest is reserved. Table 6-1 lists the peripheral allocation within this 128-KB space. Note that all peripherals may not be available in all devices of the family (details in the REMARKS column). If a peripheral is listed as N/A for a particular device, treat the corresponding address space as reserved.

ADDRESS RANGE	PERIPHERAL	TABLE	REMARKS
0x4000_2800 to 0x4000_2BFF	eUSCI_B2	Table 6-12	16-bit peripheral
0x4000_2C00 to 0x4000_2FFF	eUSCI_B3	Table 6-13	16-bit peripheral
0x4000_3000 to 0x4000_33FF	REF_A	Table 6-14	16-bit peripheral
0x4000_3400 to 0x4000_37FF	COMP_E0	Table 6-15	16-bit peripheral
0x4000_3800 to 0x4000_3BFF	COMP_E1	Table 6-16	16-bit peripheral
0x4000_3C00 to 0x4000_3FFF	AES256	Table 6-17	16-bit peripheral
0x4000_4000 to 0x4000_43FF	CRC32	Table 6-18	16-bit peripheral
0x4000_4400 to 0x4000_47FF	RTC_C	Table 6-19	16-bit peripheral
0x4000_4800 to 0x4000_4BFF	WDT_A	Table 6-20	16-bit peripheral
0x4000_4C00 to 0x4000_4FFF	Port Module	Table 6-21	16-bit peripheral
0x4000_5000 to 0x4000_53FF	Port Mapping Controller	Table 6-22	16-bit peripheral

- $0x4000\_4FFF - 0x4000\_4C00 + 1 = 0x0400$
- $0x0400 = 4 \cdot 16^2 = 2^2 \cdot (2^4)^2 = 2^2 \cdot 2^{(4 \cdot 2)} = 2^{2+8} = 2^{10}$

Task 1 - Memory Map

Task 1.2

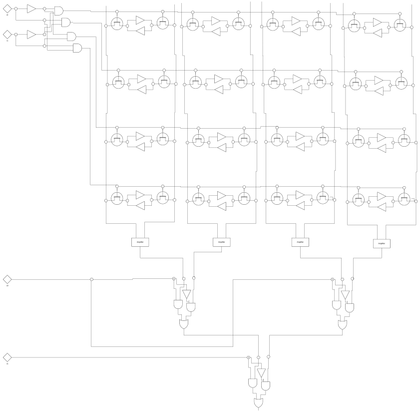


Figure 1: SRAM-cell array

Task 1 - Memory Map

Task 1.2

- u row select bits and w column select bits
- we need:
  - $2^{(u+w)}$ memory cells
  - one u-bit decoder
  - one 2^w-to-1 multiplexer
  - 2^w sense amplifiers

Task 2 - Communication

Task 2 - Communication

Task 2

Solution a:

- *Baudrate of* $115200 \frac{bits}{s}$
- *We require 16 clock periods to sample* $1 bit$
- *Required clock frequency* $= 115200 \frac{bits}{s} \cdot \frac{16}{bits} = 1.8432 MHz$

Task 3 - Interrupt and Polling

Task 3 - Interrupts and Polling

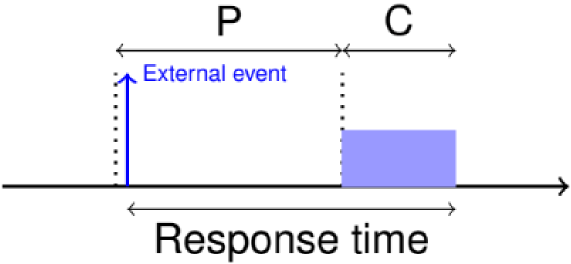
Task 3

Solution a:

- *Computation time C for the event:* $\frac{100}{48 \cdot 10^6 Hz} = 2.0833\mu s$
- *The maximum response time in the worst case is* $P + C$. *This time should not exceed our deadline of* $10\mu s$
- *Therefore, we have* $P + 2.0833\mu s \leq 10\mu s \iff P \leq 7.9167\mu s$
- *Since our polling period may not be shorter than the computation time we obtain* $P \geq 2.0833\mu s$. *Therefore, our range is* $P \in [2.0833\mu s, 7.9167\mu s]$

Solution a:

Task 3



Bonus

Bonus I

Hexadecimal System

- Example: $\underline{beef}_{16} = 11 \cdot 16^3 + 14 \cdot 16^2 + 14 \cdot 16^1 + 15 \cdot 16^0$
  $= 11 \cdot 4096 + 14 \cdot 256 + 14 \cdot 16 + 15$
  $= 48879$
- all Bin and Hex assigned:

```
0    1    2    3    4    5    6    7    8    9
---- ---- ---- ---- ---- ---- ---- ---- ---- ----
0000 0001 0010 0011 0100 0101 0110 0111 1000 1001

A    B    C    D    E    F
---- ---- ---- ---- ---- ----
1010 1011 1100 1101 1110 1111
```

Bonus II

Hexadecimal System

- Hex ⇒ Bin:

```
   D    4    F    6    6    E
1101 0100 1111 0110 0110 1110
```

- Bin ⇒ Hex:

```
1101 0100 1111 0110 0110 1110
   D    4    F    6    6    E
```

Bonus III

Hexadecimal System

- Derivation:

  - $$\begin{aligned} a4_{16} &= 10 \cdot 16^1 + 4 \cdot 16^0 \\ &= 10 \cdot (2^4)^1 + 4 \cdot (2^4)^0 \\ &= 1010_2 \cdot 2^4 + 0100_2 \cdot 1 \\ &= (1000_2 \cdot 2^4 + 10_2 \cdot 2^4) + (100_2 \cdot 2^0) \\ &= (1 \cdot 2^7 + 1 \cdot 2^5) + (1 \cdot 2^2) \\ &= 1010\_0100_2 \end{aligned}$$

  - idea: shifting a number works in hexadecimal system $1a_{16} \cdot 10_{16}^2 = 1a00$ decimal system $17 \cdot 10^2 = 1700$ and binary system $11_2 \cdot 10_2^2 = 1100_2$ quite similar.

Bonus IV

Hexadecimal System

- but beacuse $16 = 2^4$ the hexadecimal and binary system are particulary easy to convert into each other.

Bonus

Quiz question

- How many bits / digits will a hex number with 5 digits have in binary system?
  - ☐ 10
  - ☐ 15
  - ☐ 20
  - ☐ 16

Bonus

Quiz question

- How many bits / digits will a hex number with 5 digits have in binary system?
  - ☐ 10
  - ☐ 15
  - ■ 20
  - ☐ 16
- example: 0xa_aaaa = 0b1010_1010_1010_1010_1010
- the binary number has 4 times as many bits because $16 = 2^4$