Embedded Systems

Exercise 3

Introduction

The main goal of the exercise is to write an application that uses the 7-segment display available on the KAMELEON-STM32L4 evaluation board as a counter. The application shall be implemented in C using the STM32CubeIDE development environment. Using the STM32 Hardware Abstraction Layer (HAL) library is forbidden. Reading/writing of the microcontroller's GPIO registers should be implemented with pointers.

Tasks

- Analyse the wiring diagrams of the KAmeleon-STM32L4 evaluation board
- Identify the connections between the microcontroller and the 7-segment display
- Understand how to configure the 7-segment display pins to display digits
- Configure all I/O pins required to implement program logic according to their functionality
- Write an application implementing a counter according to the requirements presented below
- Compile, run and debug the application

Initial Questions

Read the schematics of the evaluation board and determine which ports and pins are connected to the 7-digit display. Electrical diagrams showing the connections of individual board components are available on the course website at: https://fiona.dmcs.pl/es/doc_stm/Kameleon_STM32L4-v1-0_Schematics.pdf.

As a preparation for writing the program, please answer the following questions:

1. What is the model of the 7-segment display mounted on the evaluation board? …OPD-Q5621LA-BW…..
2. What is the type of display? (common anode / common cathode) …common cathode……………………
3. What logic state should be set on pins A-G of the display to turn on the segments? …High (logic 1)………
4. What logic state should be set on pins DIG1…4 of the display to select the digit? …Low (logic 0)………..
5. Identify microcontroller ports and pins connected to the display

Display Pin	Microcontroller Port	Microcontroller Pin
A	G	0
B	G	1
C	G	2
D	G	3
E	G	4
F	G	5
G	G	6
DP	G	9
DIG1	B	2
DIG2	B	3
DIG3	B	4
DIG4	B	5

6. What should be the configuration of the microcontroller pins to control the segments (A-G)? (input/output, with/without pull-up resistors etc.) …GPIO_MODE_OUTPUT_PP…. with… GPIO_NOPULL………
7. What should be the configuration of the microcontroller pins to control the digits (DIG1…4)? (input/output, with/without pull-up resistors etc.) ….GPIO_MODE_OUTPUT_PP… with… GPIO_NOPULL

8. What logic state should be set on microcontroller pins to turn on the segments (A-G)? …high (logic 1)…..
9. What logic state should be set on microcontroller pins to select the digits (DIG1…4)? ..high (logic 1).......
10. Identify segments that should be turned on to display the specified digits:

Digit	Segments of the display that should be turned on
0	A, B, C, D, E, F
1	B, C
2	A, B, D, E, G
3	A, B, C, D, G
4	B, C, F, G
5	A, C, D, F, G
6	A, C, D, E, F, G
7	A, B, C
8	A, B, C, D, E, F, G
9	A, B, C, D, F, G

Requirements for the application

The final goal of the exercise is to implement a counter counting up and down in the range of 0-9999.

The application must be delivered in two separately assessed phases:

1. Counting from 0 to 9 (after reaching 9, counting starts from 0).
2. Counting up and down in the range 0-9999 (counting direction controlled with joystick)

Detailed requirements:

1. Phase 1
   - The application uses a single digit of the 7-segment display to implement a counter from 0 to 9.
   - After reaching the maximum value counter starts again from 0.
   - Displayed digits should change approximately every 1 second.
   - The program should enable easy and quick change of the digit used to display the values, according to the teacher's indications.
2. Phase 2
   - The application uses all four digits of the 7-segment display to implement a counter from 0 to 9999.
   - The application should use the multiplexing technique (quick switching on and off of the appropriate digits of the display combined with a change of value).
   - The counting direction should be controlled with the OK button of the joystick. If the button is released, the application counts up, and when the maximum is reached, it returns to 0. If the button is pressed, the application counts down, and when it comes to 0, it returns to the maximum.
   - Displayed value should change approximately every 1 second.
   - It must be ensured that pressing the button does not affect the counting speed.