Lab 3: Pre-Lab Assignment (10 points)

SOLUTION

Note: In your C program, other solutions are possible. For example, using C constants or shifting numbers to the right or left. It is also possible to first SET, and, then, CLEAR bits. All solutions here take into consideration that your first operation is a CLEAR operation.

1. Configure Port A: Pin 6, 7, 8, 9, 10, 15 as Alternative Function Mode

GPIO Mode: Digital Input (00, reset), Digital Output (01), *Alternative Function (10)*, Analog (11)

Register	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MODER	MODER15[1:0]		MODER14[1:0]		MODER13[1:0]		MODER12[1:0]		MODER11[1:0]		MODER10[1:0]		MODER9[1:0]		MODER8[1:0]		MODER7[1:0]		MODER6[1:0]		MODER5[1:0]		MODER4[1:0]		MODER3[1:0]		MODER2[1:0]		MODER1[1:0]		MODER0[1:0]	
MASK (Clear) Alternative Solution	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
MASK (Clear)	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
MASK (Set)	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
DESIRED BIT OUTPUT	1	0	-	-	-	-	-	-	-	-	1	0	1	0	1	0	1	0	1	0	-	-	-	-	-	-	-	-	-	-	-	-

GPIOA Mode Register MASK (Clear) **Alternative Solution** = **0x40155000** (in HEX)
GPIOA Mode Register MASK (Clear) = **0xC03FF000** (in HEX)
GPIOA Mode Register MASK (Set) = **0x802AA000** (in HEX)

Set Alternative Function Port A: Pin 6, 7, 8, 9, 10, and 15 as *AF11 (0b1011 or 0xB)*.

Register	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AFR[0]	AFRL7[3:0]				AFRL6[3:0]				AFRL5[3:0]				AFRL4[3:0]				AFRL3[3:0]				AFRL2[3:0]				AFRL1[3:0]				AFRL0[3:0]			
MASK (Clear) Alternative Solution	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MASK (Clear)	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MASK (Set)	1	0	1	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DESIRED BIT OUTPUT	1	0	1	1	1	0	1	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
AFR[1]	AFRH15[3:0]				AFRH14[3:0]				AFRH13[3:0]				AFRH12[3:0]				AFRH11[3:0]				AFRH10[3:0]				AFRH9[3:0]				AFRH8[3:0]			
MASK (Clear) Alternative Solution	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0
MASK (Clear)	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
MASK (Set)	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	0	1	1	1	0	1	1
DESIRED BIT OUTPUT	1	0	1	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	0	1	1	1	0	1	1	1	0	1	1

GPIOA Alternative Function Register [0] MASK (Clear) **Alternative Solution** = **0x44000000** (in HEX)

GPIOA Alternative Function Register [0] MASK (Clear) = **0xFF000000** (in HEX)

GPIOA Alternative Function Register [0] MASK (Set) = **0xBB000000** (in HEX)

GPIOA Alternative Function Register [1] MASK (Clear) **Alternative Solution** = **0x40000444** (in HEX)

GPIOA Alternative Function Register [1] MASK (Clear) = **0xF0000FFF** (in HEX)

GPIOA Alternative Function Register [1] MASK (Set) = **0xB0000BBB** (in HEX)

2. Configure Port B: Pin 0, 1, 4, 5, 9, 12, 13, 14, and 15 as Alternative Function Mode

GPIO Mode: Digital Input (00, reset), Digital Output (01), *Alternative Function (10)*, Analog (11)

Register	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MODER	MODER15[1:0]		MODER14[1:0]		MODER13[1:0]		MODER12[1:0]		MODER11[1:0]		MODER10[1:0]		MODER9[1:0]		MODER8[1:0]		MODER7[1:0]		MODER6[1:0]		MODER5[1:0]		MODER4[1:0]		MODER3[1:0]		MODER2[1:0]		MODER1[1:0]		MODER0[1:0]	
MASK (Clear) Alternative Solution	0	1	0	1	0	1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
MASK (Clear)	1	1	1	1	1	1	1	1	0	0	0	0	1	1	0	0	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
MASK (Set)	1	0	1	0	1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	1	0
DESIRED BIT OUTPUT	1	0	1	0	1	0	1	0	-	-	-	-	1	0	-	-	-	-	-	-	1	0	1	0	-	-	-	-	1	0	1	0

GPIOB Mode Register MASK (Clear) **Alternative Solution** = **0x55040500** (in HEX)
GPIOB Mode Register MASK (Clear) = **0xFF0C0F0F** (in HEX)
GPIOB Mode Register MASK (Set) = **0xAA080A0A** (in HEX)

Set Alternative Function of Port B: Pin 0, 1, 4, 5, 9, 12, 13, 14, and 15 as *AF11 (0b1011 or 0xB)*.

Register	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AFR[0]	AFRL7[3:0]				AFRL6[3:0]				AFRL5[3:0]				AFRL4[3:0]				AFRL3[3:0]				AFRL2[3:0]				AFRL1[3:0]				AFRL0[3:0]			
MASK (Clear) Alternative Solution	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0
MASK (Clear)	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
MASK (Set)	0	0	0	0	0	0	0	0	1	0	1	1	1	0	1	1	0	0	0	0	0	0	0	0	1	0	1	1	1	0	1	1
DESIRED BIT OUTPUT	-	-	-	-	-	-	-	-	1	0	1	1	1	0	1	1	-	-	-	-	-	-	-	-	1	0	1	1	1	0	1	1
AFR[1]	AFRH15[3:0]				AFRH14[3:0]				AFRH13[3:0]				AFRH12[3:0]				AFRH11[3:0]				AFRH10[3:0]				AFRH9[3:0]				AFRH8[3:0]			
MASK (Clear) Alternative Solution	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
MASK (Clear)	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
MASK (Set)	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0
DESIRED BIT OUTPUT	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	-	-	-	-	-	-	-	-	1	0	1	1	-	-	-	-

GPIOB Alternative Function Register [0] MASK (Clear) **Alternative Solution** = **0x00440044** (in HEX)
GPIOB Alternative Function Register [0] MASK (Clear) = **0x00FF00FF** (in HEX)
GPIOB Alternative Function Register [0] MASK (Set) = **0x00BB00BB** (in HEX)

GPIOB Alternative Function Register [1] MASK (Clear) **Alternative Solution** = **0x44440040** (in HEX)
GPIOB Alternative Function Register [1] MASK (Clear) = **0xFFFF00F0** (in HEX)
GPIOB Alternative Function Register [1] MASK (Set) = **0xBBBB00B0** (in HEX)

3. Configure Port C: Pin 3, 4, 5, 6, 7, and 8 as Alternative Function Mode

GPIO Mode: Digital Input (00, reset), Digital Output (01), *Alternative Function (10)*, Analog (11)

Register	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MODER	MODER15[1:0]		MODER14[1:0]		MODER13[1:0]		MODER12[1:0]		MODER11[1:0]		MODER10[1:0]		MODER9[1:0]		MODER8[1:0]		MODER7[1:0]		MODER6[1:0]		MODER5[1:0]		MODER4[1:0]		MODER3[1:0]		MODER2[1:0]		MODER1[1:0]		MODER0[1:0]	
MASK (Clear) Alternative Solution	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	0	1	0	0	0	0	0	0
MASK (Clear)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0
MASK (Set)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0
DESIRED BIT OUTPUT	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	0	1	0	1	0	1	0	1	0	1	0	-	-	-	-	-	-

GPIOC Mode Register MASK (Clear) **Alternative Solution** = **0x00015540** (in HEX)
GPIOC Mode Register MASK (Clear) = **0x0003FFC0** (in HEX)
GPIOC Mode Register MASK (Set) = **0x0002AA80** (in HEX)

Set Alternative Function of Port C: Pin 3, 4, 5, 6, 7, and 8 as *AF11 (0b1011 or 0xB)*.

Register	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AFR[0]	AFRL7[3:0]				AFRL6[3:0]				AFRL5[3:0]				AFRL4[3:0]				AFRL3[3:0]				AFRL2[3:0]				AFRL1[3:0]				AFRL0[3:0]			
MASK (Clear) Alternative Solution	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MASK (Clear)	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
MASK (Set)	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
DESIRED BIT OUTPUT	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	-	-	-	-	-	-	-	-	-	-	-	-
AFR[1]	AFRH15[3:0]				AFRH14[3:0]				AFRH13[3:0]				AFRH12[3:0]				AFRH11[3:0]				AFRH10[3:0]				AFRH9[3:0]				AFRH8[3:0]			
MASK (Clear) Alternative Solution	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
MASK (Clear)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
MASK (Set)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
DESIRED BIT OUTPUT	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	0	1	1

GPIOC Alternative Function Register [0] MASK (Clear) **Alternative Solution** = **0x44444000** (in HEX)
GPIOC Alternative Function Register [0] MASK (Clear) = **0xFFFFF000** (in HEX)
GPIOC Alternative Function Register [0] MASK (Set) = **0xBBBBB000** (in HEX)

GPIOC Alternative Function Register [1] MASK (Clear) **Alternative Solution** = **0x00000004** (in HEX)

GPIOC Alternative Function Register [1] MASK (Clear) = **0x0000000F** (in HEX)

GPIOC Alternative Function Register [1] MASK (Set) = **0x0000000B** (in HEX)

4. Configure Port D: Pin 8, 9, 10, 11, 12, 13, 14, and 15 as Alternative Function Mode

GPIO Mode: Digital Input (00, reset), Digital Output (01), *Alternative Function (10)*, Analog (11)

Register	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MODER	MODER15[1:0]		MODER14[1:0]		MODER13[1:0]		MODER12[1:0]		MODER11[1:0]		MODER10[1:0]		MODER9[1:0]		MODER8[1:0]		MODER7[1:0]		MODER6[1:0]		MODER5[1:0]		MODER4[1:0]		MODER3[1:0]		MODER2[1:0]		MODER1[1:0]		MODER0[1:0]	
MASK (Clear) Alternative Solution	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MASK (Clear)	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MASK (Set)	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DESIRED BIT OUTPUT	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

GPIOD Mode Register MASK (Clear) **Alternative Solution** = **0x55550000** (in HEX)

GPIOD Mode Register MASK (Clear) = **0xFFFF0000** (in HEX)

GPIOD Mode Register MASK (Set) = **0xAAAA0000** (in HEX)

Set Alternative Function of Port D: Pin 8, 9, 10, 11, 12, 13, 14, and 15 as *AF11 (0b1011 or 0xB)*.

Register	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AFR[0]	AFRL7[3:0]				AFRL6[3:0]				AFRL5[3:0]				AFRL4[3:0]				AFRL3[3:0]				AFRL2[3:0]				AFRL1[3:0]				AFRL0[3:0]			
MASK (Clear)	No mask needed.																															
MASK (Set)	No mask needed.																															
DESIRED BIT OUTPUT	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
AFR[1]	AFRH15[3:0]				AFRH14[3:0]				AFRH13[3:0]				AFRH12[3:0]				AFRH11[3:0]				AFRH10[3:0]				AFRH9[3:0]				AFRH8[3:0]			
MASK (Clear) Alternative Solution	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0
MASK (Clear)	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
MASK (Set)	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1
DESIRED BIT OUTPUT	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1

GPIOD Alternative Function Register [1] MASK (Clear) **Alternative Solutions** = **0x44444444** (in HEX)

GPIOD Alternative Function Register [1] MASK (Clear) = **0xFFFFFFFF** (in HEX)

GPIOD Alternative Function Register [1] MASK (Set) = **0xBBBBBBBB** (in HEX)

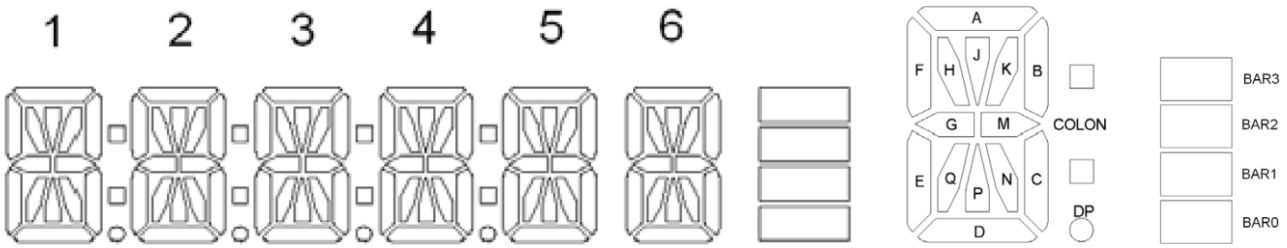
5. Complete the following configuration table for LCD registers.

- Refer to **Figure 17-10** the programming flow chart in Textbook (3rd edition) and **STM32L4 Reference Manual** (available at Online Classroom) to complete the following table.
- For this question, you just need to fill in the missing bits (the spaces without a number or a dash). The desired value for the **LCD_CR** register is given as an example.
- In order to configure the LCD, we have to enable and disable bits in a certain order, and, therefore, it is not possible to use just one or two masks. Please, refer to **Figure 17-10** in textbook for more information.
- The LCD configuration must be implemented in your lab assignment.

Register	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LCD_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUFEN	MUX_SEG	BIAS[1:0]		DUTY [2:0]			VSEL	LCDEN
DESIRED BIT OUTPUT	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	1	0	0	1	1	0	1
LCD_FCR	Res.	Res.	Res.	Res.	Res.	Res.	PS[3:0]				DIV[3:0]				BLINK[1:0]		BLINKF[2:0]			CC [2:0]			DEAD [2:0]			PON [2:0]			UDDIE	Res.	SOFIE	HD
DESIRED BIT OUTPUT	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	1	1	-	-	-	1	1	1	-	-	-	-
LCD_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FCRSF	RDY	UDD	UDR	SOF	ENS
DESIRED BIT OUTPUT	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
LCD_CLR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UDDC	Res.	SOFC	Res.
DESIRED BIT OUTPUT	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

6. Write down your last name and complete the following table.

Make sure you have read at least Sections 1 to 3 from Chapter 17! Specifically, Figure 17.8 shows a similar table for displaying a "2" in the 6th position of the LCD.



Your Last Name: _____ (First Six Characters)

Note: Mark in the table below the LCD segments that you would need to turn on to display your last name. Then, convert these bits to hexadecimal values and complete the masks at the bottom.

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LCD_RAM[0]	4E	4G	3M	3B		6G	5M	5B	1M	1B					6E		3E	3G	2M	2B			6B	6M		2E	2G	1E	1G			
LCD_RAM[1]																													5E	5G	4M	4B
LCD_RAM[2]	4D	4F	3C	3A		6F	5C	5A	1C	1A					6D		3D	3F	2C	2A			6A	6C		2D	2F	1D	1F			
LCD_RAM[3]																													5D	5F	4C	4A
LCD_RAM[4]	4P	4Q	3 Col	3K		6Q	3 Bar	5K	1 Col	1K					6P		3P	3Q	2 Col	2K			6K	1 Bar		2P	2Q	1P	1Q			
LCD_RAM[5]																													5P	5Q	4 Col	4K
LCD_RAM[6]	4N	4H	3 DP	3J		6H	2 Bar	5J	1 DP	1J					6N		3N	3H	2 DP	2J			6J	0 Bar		2N	2H	1N	1H			
LCD_RAM[7]																													5N	5H	4 DP	4J

LCD_RAM is an array of 32-bit unsigned integers. The initial values for all LCD_RAM are 0x00. So, we just need to set the bits corresponding to each segment in the LCD. For example, a number 2 in the 6th position would need to turn on segments **6A, 6B, 6G, 6M, 6E** and **6D**. Thus, we could set the LCD_RAM as follows: **`LCD->RAM[0] |= 0x04020300;`** and **`LCD->RAM[2] |= 0x00020200;`**

```
LCD->RAM[0] (Set mask) = 0x_____ (in Hex)
LCD->RAM[1] (Set mask) = 0x_____ (in Hex)
LCD->RAM[2] (Set mask) = 0x_____ (in Hex)
LCD->RAM[3] (Set mask) = 0x_____ (in Hex)
LCD->RAM[4] (Set mask) = 0x_____ (in Hex)
LCD->RAM[5] (Set mask) = 0x_____ (in Hex)
LCD->RAM[6] (Set mask) = 0x_____ (in Hex)
LCD->RAM[7] (Set mask) = 0x_____ (in Hex)
```