Knowledge Assessment

Embedded Electronic Systems

Version	Date	Created/ Modified by	Comment
01	2022-05-13	Caio Cordeiro Eloi Filho	
02	2022-05-16	Eloi Filho	Question 2 state machine definition updated.

WRITTEN BY	INFORMATION SECURITY CLASS	DATE	VERSION
Eloi Filho	Open information	2024-03-05	02

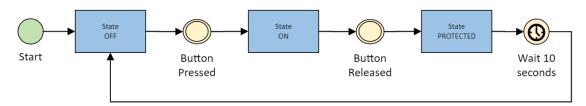| APPROVED BY	SUBJECT		PAGE
Eloi Filho	Knowledge Assessment		1 (4)

Contents

1. Question 1

Imagine a situation where you have a circuit able to activate and deactivate a purely resistive load with a microcontroller. It is required to control the load power over time depending on the situation and the power cannot be instantly changed from 0% to 100%. Explain in simple words a method to achieve such effect, looking at both hardware and firmware aspects of the system. If an inductive load is used instead of a resistive load, can the same triggering method be used or do special precautions need to be taken?

2. Question 2

Imagine a situation where you have an electronic board with a microcontroller and a button. You are required to check if the button is pressed (state ON) or not (state OFF) and for how long the button is kept pressed. If the button gets pressed, you also need to have a clear state definition ranging from ON, PROTECTED and OFF; as per the state diagram below. The protected state is defined as an intermediate stage between the transition from the ON state to the OFF state, so that each time the button is released, the button state changes to PROTECTED for 10 seconds before going to the OFF state. Implement this logic in a generic way using the C language and share this code through a public repository on GitHub. Low-level methods can be abstracted. If you prefer, use frameworks like Arduino, ESP32 or even other market platforms in this low-level code abstraction.

3. Question 3

Implement a function able to calculate and return the average, maximum and minimum value of an array with "n" positions. This function must also return a copy of the input array containing only the even numbers of the original array, as well as the new array size. Implement this logic using the C language and share this code through a public repository on GitHub.

4. Question 4

Imagine that you have a microcontroller that communicates to a generic system that may consist of several other boards via UART. How do you ensure that each message is properly sent and that its content is correct?

Now imagine that you receive a message and an interruption is triggered every time a new information is received. Inside every message you have a different command, and for every command you have different payloads, receiving positive and negative values.

Implement the interrupt and the parser functions in a generic way using the C language and share this code through a public repository on GitHub. Low-level methods can be abstracted. If you prefer, use frameworks like Arduino, ESP32 or even other market platforms in this low-level code abstraction.