

Taller de Sistemas Embebidos – 2025 1C – Parcial

Apellidos: Giamberale Nombres: M. de las Mercedes
Aciertos (suman): 27 Yerros (restan): 5 Nota: 4 (cero)

Favor de indicar en cada punto una única respuesta, la que considere verdadera.

1. Sistema Embebido (SE) es un Sistema Electrónico de Cómputo, compuesto por:

- a. Sólo hardware
- b. Hardware y software
- c. Sólo software
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

2. Durante el Diseño/Desarrollo/Depuración de Software de un SE, nos preocupa:

- a. Sólo el determinismo
- b. Sólo el tiempo de respuesta
- c. Tanto el determinismo como el tiempo de respuesta
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

¿DETERMINISMO?

3. Las funciones de la Unidad Central de Procesamiento (CPU) son:

- a. Aceptar datos, proporciona espacio de memoria temporal de almacenamiento (guarda)
- b. Realizar operaciones aritmético/lógicas de datos
- c. Convertir datos en información
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

4. Unified Modeling Language (UML) es un lenguaje de modelado de propósito general:

- a. Su objetivo es definir una forma estándar de visualizar cómo se ha diseñado un sistema
- b. Es un lenguaje de programación
- c. State Machine Diagrams, no sirven p/representar el estado del sistema en instantes de tiempo
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

5. Para el Diseño/Desarrollo/Depuración de Hardware de SE, se recurre a herramientas:

- a. Con prestaciones de control y análisis, en evolución permanente
- b. Para sistematizar o mejorar la velocidad/precisión de los procesos/subprocesos requeridos
- c. Herramientas Informáticas: CAE/CAD/CAM Computer-Aided Engineering
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

6. Para el Diseño/Desarrollo/Depuración de Software de SE, recurriremos a:

- a. Integrated Development Environment (IDE for STM32): STM32CubeIDE
- b. Version Control System (VCS): Git Bash/GUI & Repository: GitHub
- c. Software Modeling Tool: Itemis CREATE
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

7. ¿Qué es un IDE?

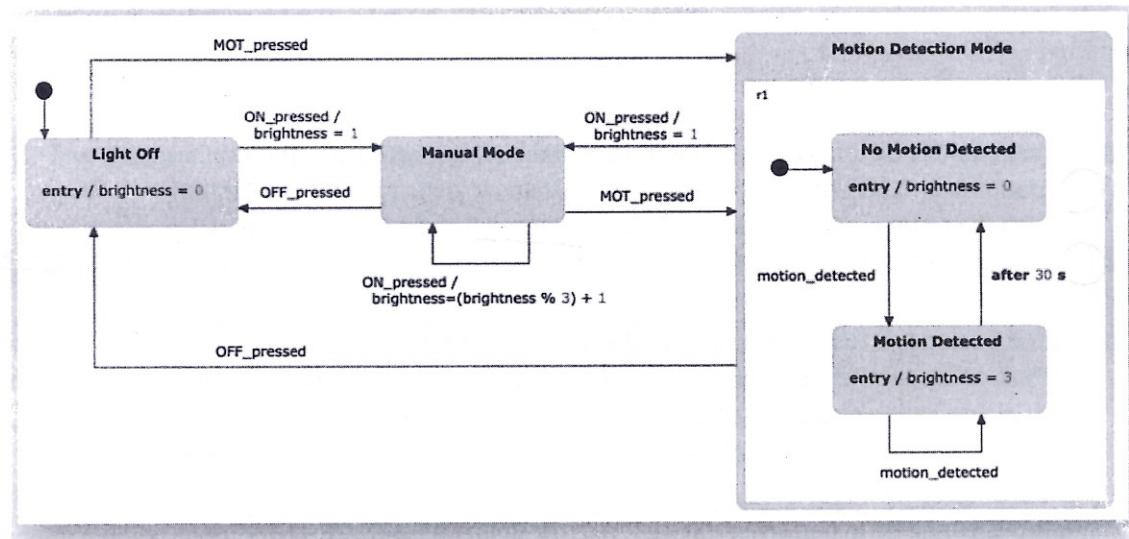
- a. Es un sistema de gestión de código fuente y control de versiones distribuido
- b. Es un servicio de alojamiento (hosting) para repositorios
- c. Es un sistema de desarrollo, que usaremos para desarrollar/depurar software
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

8. En una arquitectura de computadora, la conexión interna se realiza a través de Bus/Buses

- a. Sólo el Bus de Direcciones
- b. Sólo el Bus de Direcciones y Bus de Datos
- c. El Bus de Direcciones, el Bus de Datos y el Bus de Control
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

+ 8 - 0

9. ARM's Cortex-M que constituye la base de muchos microcontroladores del mercado actual, es:
- a. Un microprocesador
 - b. Un microcontrolador
 - c. Un microcomputador
 - d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)
- Se agrega nota: Se agúin (y) resuelto) en un MP. Para mí es la familia entera
-
10. Cuando un ingeniero tiene que diseñar una nueva solución SoC, primero genera un prototipo (MPV):
- a. Que permite verificar la cantidad de memoria necesaria para el proyecto
 - b. Que permite verificar el tiempo de respuesta del sistema
 - c. El producto final no dependerá de un SDK
 - d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)
-
11. Para guardar y ejecutar las operaciones, se utilizan registros como fuentes y destinos de datos:
- a. En los procesadores modernos pueden haber 16, 32 o incluso más registros
 - b. Para escribir o leer registros, el acceso se realiza a través de uno o más Buses de Datos internos
 - c. Un registro particular como registro de flags o registro de estado
 - d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)
-
12. Un mejor enfoque de la Unidad de Control, es hacer lo que se llama pipelining, por cada clock:
- a. Se carga una instrucción, se decodifica y se ejecuta, instrucción por instrucción
 - b. Se carga una instrucción, se decodifica y ejecuta la anterior
 - c. Se carga una instrucción, se decodifica la anterior y se ejecuta al anterior a la anterior
 - d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)
- 
-
13. Para transferir instrucciones/datos dentro y fuera del procesador, se requiere de dos o más buses:
- a. von Newman, Instrucciones y Datos comparten el mismo par de Buses de Direcciones y Datos
 - b. Harvard, Instrucciones y Datos comparten el mismo par de Buses de Direcciones y Datos
 - c. von Newman es más potente en términos de eficiencia informática que Harvard
 - d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)
-
14. Un área específica de memoria se utiliza para:
- a. Un heap (heap), en la que el procesador puede hacer PUSH/POP (almacenar/recuperar datos)
 - b. Una pila (stack), en la que el procesador puede hacer PUSH/POP (almacenar/recuperar datos)
 - c. Un pool (pool), en la que el procesador puede hacer PUSH/POP (almacenar/recuperar datos)
 - d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)
- En el heap
no se puede
hacer PUSH / POP
-
15. Los tipos de memorias disponibles según su capacidad de retener o no datos son:
- a. ROM/PROM/EPROM/OTP EPROM volátiles
 - b. EEPROM/FLASH/FRAM/MRAM volátiles
 - c. SRAM/SSRAM volátiles
 - d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)
-
16. Cortex-M0 y Cortex-M0+ ofrecen:
- a. El rendimiento mayor de la familia Cortex-M
 - b. El consumo de energía más bajo de la familia Cortex-M
 - c. El conjunto de instrucciones grande de la familia Cortex-M
 - d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)
-
17. Los diagramas de estado de Harel, introducen un enfoque:
- a. Modular, jerárquico y bien estructurado
 - b. Sin estados compuestos, ni ortogonalidad (sin paralelismo) ✗
 - c. Sin comunicación (sin sincronización)
 - d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)
-



Extended light switch example as a Harel statechart with composite states

18. Para el diagrama de estado de Harel de la figura, al recibir la siguiente secuencia de entradas:

Indicar por qué **estados** evolucionará el diagrama luego de ser reiniciada:

- Reset => LIGHT OFF / S / _____
- ON_pressed** => MANUAL MODE / S / _____
- ON_pressed** => MANUAL MODE / S / _____
- ON_pressed** => MANUAL MODE / S / _____
- MOT_pressed** => NO MOTION DETECTED / MOTION DETECTION/MOT / S
- motion_detected** => MOTION DETECTED / MOTION DETECTION/MOT / S
- ON_pressed** => MANUAL MODE / S / _____
- OF_pressed** => LIGHT OFF / S / _____
- i. fin

19. Para el diagrama de estado de Harel de la figura, al recibir la siguiente secuencia de entradas:

Indicar por qué valores evolucionará la variable **brightnness** (estado a estado) luego de ser reiniciada:

- Reset => 0 / S / _____
 - ON_pressed** => 1 / S / _____
 - ON_pressed** => 2 / S / $(1 \times 3) + 1 = 2$
 - ON_pressed** => 3 / S / $(2 \times 1) + 1 = 3$
 - MOT_pressed** => 0 / S / _____
 - motion_detected** => 3 / S / _____
 - ON_pressed** => 1 / S / _____
 - OF_pressed** => 0 / S / _____
 - i. fin
- Handwritten notes: $(1 \times 3) + 1 = 2$, $(2 \times 1) + 1 = 3$. A circled '0' is followed by 'cómo que NO?'.

20. La conexión interfaces pasivas a pines del **Microcontrolador**, se validan mediante cálculo aplicando:

- Ley de Coulomb y Faraday-Lenz
- Ley de Ampere y Faraday-Lenz
- Ley de Ohm y Leyes de Kirchhoff**
- Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

21. Conceptos básicos de los puertos de salida digital del **Microcontrolador**:

- Los pines de un puerto de salida pueden tener uno de dos niveles lógicos de salida "0 o 1"
- La tensión máxima de salida "1" depende de la tecnología del uC (puede ser 3.3V o 5V)
- La tensión mínima de salida "0" depende de la tecnología del uC (puede ser 0V)
- Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)**

+ 3/4 - 1/4

22. Conceptos básicos de los **puertos de salida** digital del **Microcontrolador**, conectado a un **LED**:

- a. Usamos un **LED** como indicador en **SE** (niveles lógicos fijos o cambiantes) *actuador*
- b. Un **LED** es un diodo que, polarizado en directa no conduce corriente \times
- c. La corriente que circula es proporcional al brillo (cantidad de luz emitida) del **LED** *osciloscopio*
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

23. Conceptos básicos de los **puertos de salida** digital del **Microcontrolador**, conectado a un **transistor**:

- a. Para componentes que requieren más corriente o tensión que los que pueden manejar el **uC**
- b. El pin de salida digital del **uC** se conecta a la base de un **transistor NPN** (vía $R_{BASE} + R_{PULL-DOWN}$)
- c. El colector del **transistor NPN** se conecta a alimentación (vía carga) y el emisor a tierra
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

24. Conceptos básicos de los **puertos de entrada** digital del **Microcontrolador**, conectado a un **pulsador**:

- a. Los **pulsadores** de contacto no cambian de estado de forma limpia, rebotan (bounce)
- b. El rebote mecánico (vibrar) del **pulsador** de contacto puede durar algunas decenas de **uS**
- c. Podemos filtrar dicho rebote sólo por **software**, con un algoritmo anti-rebote (debounce)
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

25. El diagrama de bloque de **STM32FXXXXY**, usado en la placa **NUCLEO-FXXXXZ (SDK), GPIO**:

- a. El bus **AHB2** les transfiere datos digitales, internamente a una velocidad de hasta 1000 Mbps
- b. El módulo HAL más complejo es **HAL_GPIO**
- c. Hay un único banco de puertos **GPIO** conectados a **AHB2**
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

26. El diagrama de bloque de **STM32FXXXXY, HAL_GPIO Module, controlador de entrada** (opciones):

- a. **Alternate function:** Schmitt trigger output interno al **MCU** (no válida)
- b. **Analog:** Bypasses the Schmitt trigger (válida)
- c. **Read:** salida de una de las líneas de registro de lectura asociadas (no válida)
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

27. El diagrama de bloque de **STM32FXXXXY, HAL_GPIO Module, controlador de entrada/salida**:

- a. La **HAL** se basa en la estructura C **GPIO_InitStruct** para configurar cualquier pin de **GPIO**
- b. El método **HAL_GPIO_TogglePin()** fuerza a bajo el nivel de un pin de **GPIO** \times
- c. El método **HAL_GPIO_EXTI_Callback()** conectado al **IRQ EYTI** de un pin de **GPIO** ...
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

28. El diagrama de bloque de **STM32FXXXXY**, usado en la placa **NUCLEO-FXXXXZ (SDK), Interrupciones**:

- a. No las utilizan los canales de **comunicación** de datos
- b. No las utilizan los procesos de acceso directo a la memoria (**DMA**)
- c. Desde una perspectiva de alto nivel, es un proceso que detiene la ejecución de un **programa**
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

29. El diagrama de bloque de **STM32FXXXXY**, usado en la placa **NUCLEO-FXXXXZ (SDK), NVIC**:

- a. El procesador **ARM Cortex-M** no contiene un dispositivo de hardware llamado **NVIC** \times
- b. **NVIC:** Controlador de Interrupción Vectorial Anidadadas (nested vectored interrupt controller)
- c. El propósito del **NVIC** no es gestionar excepciones
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

30. El diagrama de bloque de **STM32FXXXXY**, usado en la placa **NUCLEO-FXXXXZ (SDK), NVIC**:

- a. Puede gestionar hasta **266** canales de interrupción $\rightarrow 256$ (potencia de 2)
- b. **250** canales pueden ser internos/externos *QVE CHOTO QVE SOS*
- c. **16** canales son internos
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

31. El diagrama de bloque de **STM32FXXXXY, NVIC, condiciones previas** a generación de una **IRQ**:

- a. El dispositivo de interrupción debe estar armado (el bit de activación debe estar activado)
- b. La **NVIC** debe estar habilitada
- c. Las **interrupciones globales** deben estar habilitadas
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

32. El diagrama de bloque de STM32FXXXXY, NVIC, eventos posteriores al reconocimiento de IRQ:

- a. Una **interrupción** solicitante se mantiene en pendiente con **interrupciones** no habilitadas
 - b. Una **interrupción** solicitante se mantiene en pendiente con **IRQ** solicitante de prioridad menor
 - c. Cumplidos las condiciones previas, el estado de la **IRQ** solicitante pasa de pendiente a activo
 - d. Ninguna de las anteriores o **Todas las anteriores** (lo que aplique al caso)
-

33. El diagrama de bloque de STM32FXXXXY, usado en la placa NUCLEO-FXXXZZ (SDK), Timers:

- a. Es un **contador** de funcionamiento libre que cuenta los pulsos de una fuente de reloj
 - b. El **reloj maestro** funciona a una velocidad rápida, por lo que debe dividirse mediante hardware
 - c. Pueden contar o acumular conteos hasta que desborden (la cantidad de bits de cuenta)
 - d. Ninguna de las anteriores o **Todas las anteriores** (lo que aplique al caso)
-

34. El diagrama de bloque de STM32FXXXXY, Timers, existen 5 **categorías** de timers STM, Básico:

- a. Timer simple de **16 bits** (sin pines de entrada ni de salida)
 - b. Se utilizan principalmente como "**maestros**" para otros cronómetros
 - c. Se utilizan como fuente de **reloj del convertidor digital a analógico (DAC)**
 - d. Ninguna de las anteriores o **Todas las anteriores** (lo que aplique al caso)
-

35. El diagrama de bloque de STM32FXXXXY, usado en la placa NUCLEO-FXXXZZ (SDK), ADC:

- a. La técnica usada en los **ADC** de STM, es la de Registro de Aproximación Sucesiva (**SAR**)
 - b. Se copia una tensión analógica en un circuito de **TRACK/HOLD** (muestreo/retención) V_{IN}
 - c. La muestra instantánea de la tensión analógica variable se convertirá en un número digital
 - d. Ninguna de las anteriores o **Todas las anteriores** (lo que aplique al caso)
-

36. El diagrama de bloque de STM32FXXXXY, usado en la placa NUCLEO-FXXXZZ (SDK), ADC:

- a. Se puede ver que el proceso **SAR** converge en un valor final
 - b. Un **ADC** de 12 bits requeriría 12 comparaciones
 - c. Las operaciones del módulo **SAR LOGIC** se realizan en software
 - d. Ninguna de las anteriores o **Todas las anteriores** (lo que aplique al caso)
-

37. El diagrama de bloque de STM32FXXXXY, ADC, se puede controlar por uno/más métodos:

- a. Por encuesta (**Polling**) o por interrupción (**Interrupt**) o por Acceso Directo a Memoria (**DMA**)
 - b. Sólo por encuesta (**Polling**) o por interrupción (**Interrupt**)
 - c. Sólo por encuesta (**Polling**)
 - d. Ninguna de las anteriores o **Todas las anteriores** (lo que aplique al caso)
-

38. El diagrama de bloque de STM32FXXXXY, usado en la placa NUCLEO-FXXXZZ (SDK), Timer PWM:

- a. La generación de señal es simple e implica el uso de dos timers en modo de conteo ascendente
 - b. Cuando el conteo alcanza un número, se genera un evento que depende del modo del timer
 - c. Cada timer GP tiene 2 registros de captura de comparación (**CCR_x**), que almacenan un número
 - d. Ninguna de las anteriores o **Todas las anteriores** (lo que aplique al caso)
-

39. El diagrama de bloque de STM32FXXXXY, usado en la placa NUCLEO-FXXXZZ (SDK), SPI:

- a. Es de las interfaces de comunicación del tipo "**Inter System Protocol**"
 - b. Es de las interfaces que no ayuda a reducir la cantidad de **GPIO** digitales del Microcontrolador
 - c. Es de las interfaces de comunicación del tipo "**sincrónica**"
 - d. Ninguna de las anteriores o **Todas las anteriores** (lo que aplique al caso)
-

40. El diagrama de bloque de STM32FXXXXY, usado en la placa NUCLEO-FXXXZZ (SDK), SPI:

- a. Datos transmitidos entre el nodo principal y el subnodo se sincronizan con el **reloj** del principal
 - b. Los dispositivos **SPI** admiten frecuencias de **reloj** mucho más bajas que las interfaces **I2C**
 - c. La conexión de **líneas** entre el nodo principal y el subnodo es: **MOSI => SDO**
 - d. Ninguna de las anteriores o **Todas las anteriores** (lo que aplique al caso)
-

41. El diagrama de bloque de STM32FXXXXY, usado en la placa NUCLEO-FXXXZZ (SDK), I2C:

- a. Es de las interfaces de comunicación del tipo "**Intra System Protocol**"
 - b. Es de las interfaces de comunicación para conectar sólo dos dispositivos diferentes
 - c. Es de las interfaces de comunicación del tipo "**asincrónica**"
 - d. Ninguna de las anteriores o **Todas las anteriores** (lo que aplique al caso)
-

LCDTM, COMO VAS A PREGUNTAR ESO

42. El diagrama de bloque de **STM32FXXYY**, usado en la placa **NUCLEO-FXXXZZ (SDK)**, I2C:

- a. Los bits se registran en los flancos ascendentes de la señal **SCL**
- b. El **protocolo** permite conectar **dispositivos** de menor velocidad al microcontrolador
- c. La señal de **reloj** siempre está controlada por el subnodo
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

43. TP0-05 - 3er Proyecto p/placa **NUCLEO-F103RB (Event Driven System (EDS) -...)**, en la práctica: Un ejecutor cíclico de tareas, con estructuras de configuración (**cfg**) y datos (**dta**), se observó la evolución del campo **WCET**, Worst-Case Execution Time

- a. task_a, **Blocking** Code, se ejecuta de forma sincrónica (se ejecuta línea por línea)
- b. task_b, **Non-Blocking** Code, se ejecuta de forma asincrónica (no se ejecuta línea por línea)
- c. task_c, **Update by Time** Code, se ejecuta por ocurrencia de ticks
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

44. TP0-05 - 3er Proyecto p/placa **NUCLEO-F103RB (Event Driven System (EDS) -...)**, en la práctica:

- a. **WCET**, Worst-Case Execution Time, se mide en nano Segundos
- b. **WCET**, Worst-Case Execution Time, se mide en micro Segundos
- c. **WCET**, Worst-Case Execution Time, se mide en mili Segundos
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

45. TP0-05 - 3er Proyecto p/placa **NUCLEO-F103RB (Event Driven System (EDS) -...)**, en la práctica:

- a. El método **HAL_SYSTICK_Callback**, inicializa **g_task_test_tick_cnt**
- b. El método **HAL_SYSTICK_Callback**, decrementa **g_task_test_tick_cnt**
- c. El método **HAL_SYSTICK_Callback**, incrementa **g_task_test_tick_cnt**
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

46. TP1-01 - **My First Statechart**, en la práctica: Al ingresar por primera vez a **Itemis Create Cloud Editor**

=> se genera automáticamente el **modelo My First Statechart**

- a. Un **statechart** puede tener variables y modificarlas
- b. Un **evento** puede tener una guarda que opera como una proposición "OR"
- c. La **sincronización** entre **statecharts** es mediante el "raise" de un **estado**
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

47. TP1-01 - **My First Statechart**, en la práctica: Al ingresar por primera vez a **Itemis Create Cloud Editor**

- a. Un **statechart** no necesita tener **pseudo estado** inicial
- b. Una **transición** puede tener un **pseudo estado** choose que opera como una proposición "AND"
- c. La **sincronización** entre **statecharts** es mediante el "raise" de un **evento**
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

48. TP1-01 - **My First Statechart**, en la práctica: Al ingresar por primera vez a **Itemis Create Cloud Editor**

- a. Un **statechart** debe tener **pseudo estado final**
- b. Un **estado compuesto** debe tener **pseudo estado** inicial
- c. Un **estado compuesto** debe tener **pseudo estado** final
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

49. TP2-01 - 4to Proyecto p/placa **NUCLEO-F103RB (Sensor Statechart) (1 sensor)**, en la práctica: Al

codificar la tarea modelo del sensor: **Sensor Statechart**

- a. Cada sensor produce un **evento** de la máquina de estado
- b. Cada estado produce un "case" en el "switch" de la máquina de estado
- c. Cada evento produce **desif** en el "case" del estado al que afecta/excita
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)

50. TP2-01 - 4to Proyecto p/placa **NUCLEO-F103RB (Sensor Statechart) (N sensor)**, en la práctica: Al

codificar la tarea modelo del sensor: **Sensor Statechart**

- a. Los arrays de **estructuras** de configuración (**cfg**) y datos (**dta**), son de **N-1** elementos c/u
- b. Los arrays de **estructuras** de configuración (**cfg**) y datos (**dta**), son de **N+1** elementos c/u
- c. Los arrays de **estructuras** de configuración (**cfg**) y datos (**dta**), son de **N** elementos c/u
- d. Ninguna de las anteriores o Todas las anteriores (lo que aplique al caso)