|  |  |
| --- | --- |
| Campus CIudad de México  Escuela de Ingeniería y Ciencias  Departamento de Computación | TE2004B - Diseño de sistemas embebidos avanzados |
| Profesor: Rolando Bautista Montesano/ Alfredo Mantilla |
| Actividad Evaluativa Módulo de Multiprocesamiento |
| Nombres y matrículas |
| Fecha 29 de Noviembre de 2021 |

**Actividad Evaluativa Multiprocesamiento**

Equipos de 3

*Actividades previas*

1. Asegúrate de tener una NUCLEO H745ZI-Q.
2. Consigue el material necesario.

*Equipo y material*

|  |  |
| --- | --- |
| Equipo | Material |
| Computadora con STM32CubeIDE  1 NUCLEO H745ZI-Q  1 Cable micro-USB |  |

*Desarrollo*

1. Desarrollaremos un proyecto para trabajar con ambos núcleos de la NUCLEO STM32H745ZI en paralelo.
   1. Genera un proyecto nuevo.
      1. Habilita dos LEDs. Asigna uno a CM7 y otro a CM4. No olvides configurarlos como pull-up.
      2. Localiza *System Core >> NVIC2* y habilita *HSEM2 global interrupt*.
      3. Genera código.
   2. Código para CM7. Desarrolla código que lea datos de la memoria compartida de ambos procesadores y las envíe por serial a tu computadora.
      1. Genera un *struct* del tipo *shared data*.
         1. Declara una variable de 8 bits sin signo llamada *size*.
         2. Declara un arreglo de 64 elementos de 8 bits sin signo llamada *M4toM7*.
         3. Asígnales una dirección de memoria con la siguiente instrucción:

volatile struct shared\_data \* const xfr\_ptr = (struct shared\_data \*)0x38001000;

* + 1. Dentro del ciclo principal habilita y deshabilita semáforos para transmitir datos por serial.

Localiza dentro del código las siguientes líneas:

#ifndef HSEM\_ID\_0

#define HSEM\_ID\_0 (0U) /\* HW semaphore 0\*/

#endif

Aquí se declara un semáforo nuevo con un ID específico. Localiza dentro del *main* las líneas donde se utiliza y reporta su funcionamiento.

* + - 1. Activa una señal de semáforo usando la instrucción:

HAL\_HSEM\_FastTake(HSEM\_ID\_0);

* + - 1. Haz que parpadee un LED al menos 5 veces durante 1 segundo. Esto ayudará a saber de forma visual que ambos núcleos están comunicándose.
      2. Imprime por serial los datos que se encuentran dentro de *M4toM7*. Ten en cuenta que el número de valores está dentro de *size*.
      3. Libera al semáforo usando la instrucción:

HAL\_HSEM\_Release(HSEM\_ID\_0,0);

* 1. Código para CM4. Desarrolla código que genere datos de la memoria compartida de ambos procesadores.
     1. Implementa el mismo *struct* del punto b para el CM7. Asegúrate de que su apuntador esté en el mismo lugar de memoria.
     2. Después de la inicialización de módulos del CM4 agrega las interrupciones para el semáforo con las siguientes instrucciones:

HAL\_NVIC\_SetPriority(HSEM2\_IRQn, 10, 0);

HAL\_NVIC\_EnableIRQ(HSEM2\_IRQn);

* + 1. Define e implementa la función void HAL\_HSEM\_FreeCallback(uint32\_t SemMask)
       1. Cambia un LED de estado cada que se entre en esta función.
       2. Implementa código tal que cada que el *callback* sea activado se almacenen diferentes valores en *M4toM7*. Ten en cuenta que el número de valores está dentro de *size*.
       3. Notifica a CM7 que se liberó el semáforo con:

HAL\_HSEM\_ActivateNotification(\_\_HAL\_HSEM\_SEMID\_TO\_MASK(HSEM\_ID\_0));

* 1. Para programar ambos núcleos realiza lo siguiente:
     1. Programa el CM7 como siempre lo has hecho.
     2. Para programar el CM4 sigue el procedimiento usual que con el CM7, pero en *port number* dentro de *Debugger* asigna únicamente 612.
     3. Debe haber dos *Run configurations* disponibles cuando uses *Run as*.

Entregables

1-En la sesión del Lunes 29 de Noviembre cada equipo presentará su prototipo con el sistema funcionando y responderá las preguntas relacionadas con el código implementado. Cada equipo tendrá 15 minutos para presentar su actividad. El orden de presentación se les publicará en CANVAS previamente.

Los equipos que presentan en zoom pueden auxiliarse de un video para mostrar los resultados si así lo consideran.

2- Se debe subir a CANVAS previo a la sesión del 29 de Noviembre a las 11:00am un reporte escrito documentando el diseño. De igual forma subir en uno o varios archivos .zip los códigos de los programas.

3- La Evaluación de la actividad incluye los siguientes elementos

* Presentación del sistema funcionando en la tarjeta (50%)

Incluye evidencia de los programas funcionando en ambos cores

* Preguntas formuladas a los integrantes del equipo (25%)
* Reporte escrito subido a CANVAS previo a la sesión (25%)