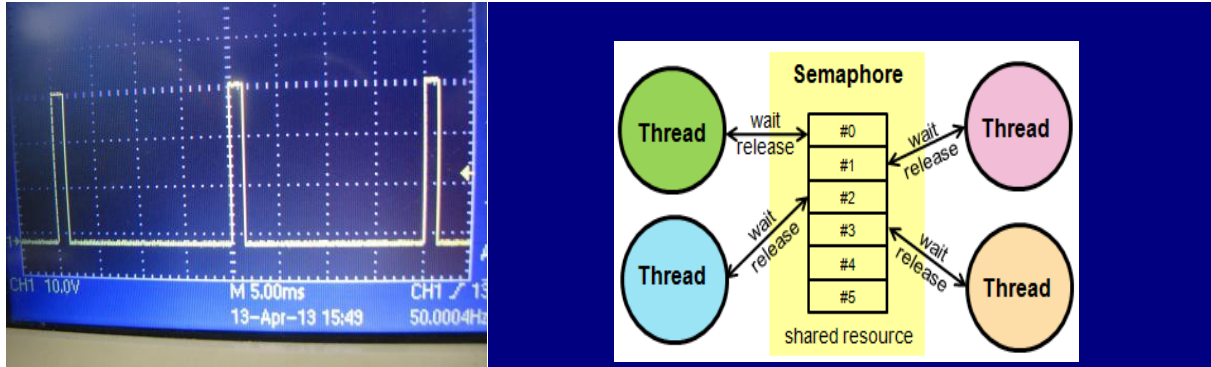


## ARM XMC - experiment No. 8



- E8.1 Name:**

XMC ARM Cortex Mx - RTOS

- E8.2 Overview and purpose:**

The experiment explores the possibilities to use Real Time Operating System - RTX on Infineon XMC platforms. At the end it will own basic information about how to develop applications using DAVE RTX components.

- E8.3 Resources:**

Hardware: Relax kit, Boot kit, XMctoGO,  
Software : Infineon DAVE 4.x with Keil RTX

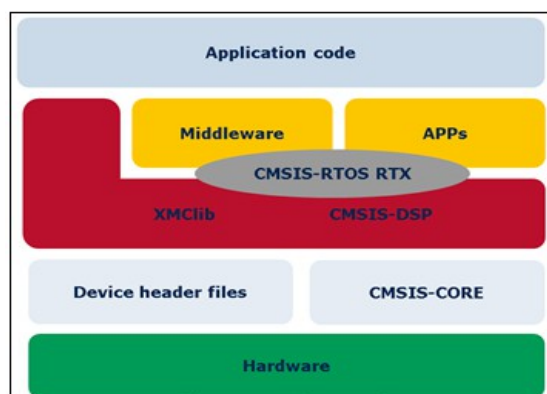


Fig.8.1 RTOS Components



- E8.4 Software example:

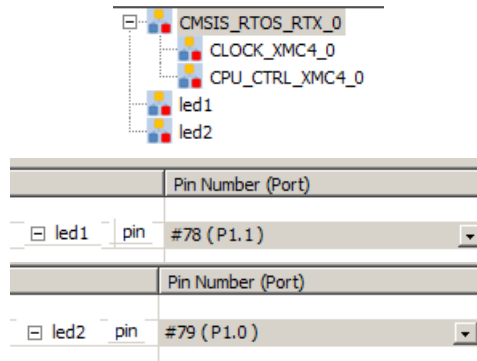


Fig. 8.2 DAVE Bloks and I/O

```
/*Using DAVE RTX
  Create 2 tasks named
  leda and ledb

*/

#include <DAVE.h>

// define tasks function here

void leda()
{
    while(1)
    {
        DIGITAL_IO_ToggleOutput (&led1);

        osDelay(500);
    }
}

void ledb()
{
    while(1)
    {
        DIGITAL_IO_ToggleOutput (&led2);

        osDelay(50);
    }
}
}
```



```
osThreadDef (leda, osPriorityNormal, 1, 256);
osThreadDef (ledb, osPriorityNormal, 1, 1024); // Thread

int main(void)
{
    DAVE_STATUS_t status;

    status = DAVE_Init(); // CMSIS_RTOS_RTX_Init API is called
    // during initialization of DAVE APPS which
    // initialize RTOS kernel
    if (DAVE_STATUS_SUCCESS == status)
    {
        // Enter user code which creates tasks
        //
        osThreadCreate(osThread(leda), NULL); // task1
        osThreadCreate(osThread(ledb), NULL); // task2
        osKernelStart(); // start kernel with user
        // defined task execution
    }
    while(1)
    {
    }

    return (1);
}
```

#### • E8.5 Method of running experiment:

- Analyze TRTX existing components in the environment DAVE;
- Analyze how to define thread type components;
- Analyze how to create task in RTX environment;

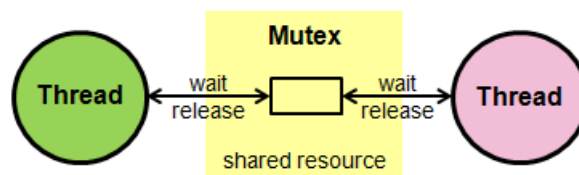


Fig. 8.3 Mutex acces

- Run the example;
- Solve the proposed problems.

#### • E8.6 Problems proposed:

1. Edit the program to run on the XMC2Go platform;



2. Edit the program to run on the XMC 1100 - Boot kit platform ;
3. Develop a software that will highlight the four tasks execution on the Infineon Boot kit platform ;
4. Make a program that will display both OLED image and activated LED;
5. Make a program that will displays a OLED geometric figure that can be changed by pressing buton1;

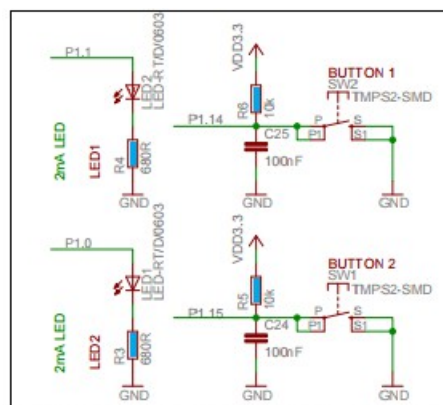


Fig. 8.4 Infineon Relax Kit - kbd&LED

**E8.7 The experiment can be extended to be used for:**

- Making active systems capable of multiple tasks;
- Achieving spatial data display systems;
- Making applications based on concurrent operations;
- Making iObject computer for Internet of Things;

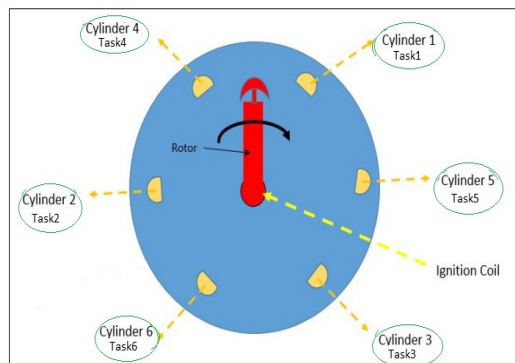


Fig.8.5 Time Sharing principle in internal combustion engine



#### • E8.8 More helpful information:

- [Basics Time Sharing](http://www.britannica.com/technology/time-sharing) - <http://www.britannica.com/technology/time-sharing>
- [RTOS for Infineon](http://www.infineon.com/cms/en/product/microcontroller/32-bit-industrial-microcontroller-based-on-arm-registered-cortex-registered-m/xmc-development-tools-software-tools-and-partner/rtos-and-middleware/channel.html?channel=db3a30433c1a8752013c3d286ba17f06) - <http://www.infineon.com/cms/en/product/microcontroller/32-bit-industrial-microcontroller-based-on-arm-registered-cortex-registered-m/xmc-development-tools-software-tools-and-partner/rtos-and-middleware/channel.html?channel=db3a30433c1a8752013c3d286ba17f06>
- [Basics RTX](http://www.sase.com.ar/2012/files/2012/09/rtos_full_webinar-SMALLer.pdf) - [http://www.sase.com.ar/2012/files/2012/09/rtos\\_full\\_webinar-SMALLer.pdf](http://www.sase.com.ar/2012/files/2012/09/rtos_full_webinar-SMALLer.pdf)
- [MIT RTOS](http://web.mit.edu/16.070/www/year2001/RTOS27.pdf) - <http://web.mit.edu/16.070/www/year2001/RTOS27.pdf>
- [ARM Keil RTX Real-Time Operating System](http://www.keil.com/rl-arm/kernel.asp) - <http://www.keil.com/rl-arm/kernel.asp>
- [Advantage RTX](http://www.keil.com/rl-arm/rtx_benefits.asp) - [http://www.keil.com/rl-arm/rtx\\_benefits.asp](http://www.keil.com/rl-arm/rtx_benefits.asp)
- [Intel RTOS](http://windriver.com/products/vxworks/) - <http://windriver.com/products/vxworks/>
- [RT Linux](http://www.cis.upenn.edu/~lee/06cse480/lec-RTOS_RTlinux.pdf) - [http://www.cis.upenn.edu/~lee/06cse480/lec-RTOS\\_RTlinux.pdf](http://www.cis.upenn.edu/~lee/06cse480/lec-RTOS_RTlinux.pdf)
- [Mbed RTOS](https://developer.mbed.org/handbook/RTOS) - <https://developer.mbed.org/handbook/RTOS>
- [Embeddedcraft RTOS](http://www.embeddedcraft.org/RTOS%20Part1.pdf) - <http://www.embeddedcraft.org/RTOS%20Part1.pdf>
- [Windows Embedded](http://www.microsoft.com/windowseembedded/en-us/windows-embedded.aspx) - <http://www.microsoft.com/windowseembedded/en-us/windows-embedded.aspx>