

FEUILLE D'EXAMEN



Nom :
 Prénom :
 N° CIN :
 Epreuve de : Système temps réel
 Classe : EQI2.1

Signatures des surveillants	N° de la feuille
	Nombre totale des feuilles

Identifiant secret

Ne rien écrire ici

Classe : All3
 Enseignant(s) : Ali Hmidene
 Durée : 1h 30mn
 Nombre totale des pages :pages

Note Attribuée

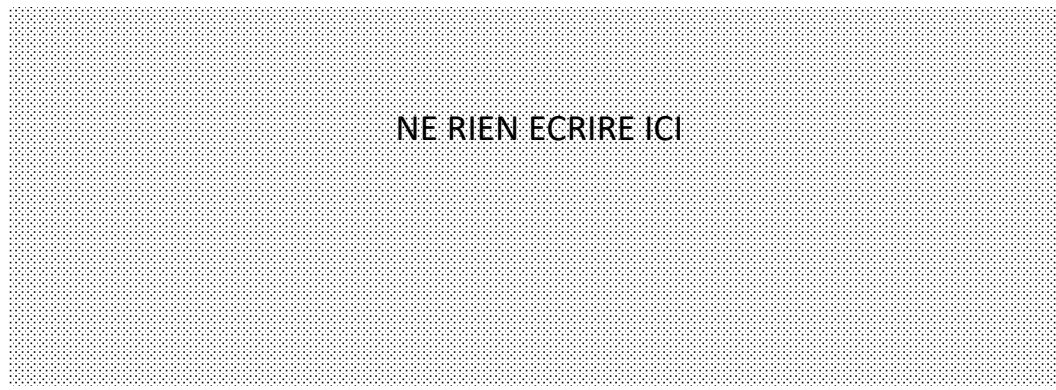
Identifiant secret

Ne rien écrire ici

Problème

Q1. Programme à compléter

```
#include "FreeRTOS.h"
#include "task.h"
#include "semphr.h"
#include "usart.h"
#define PORTCbits.RC1 AV
#define PORTCbits.RC2 AR
SemaphoreHandle_t SemSynAcq;
.....
unsigned char Val_LDR1 = 0;
unsigned char Val_LDR2 = 0;
void main()
{
    Init(); // Initialisation de la carte
    SemSynAcq = xSemaphoreCreateBinary();
    SemSynCmd = xSemaphoreCreateBinary();
```



```
xTaskCreate(Acquisition1, "Acq1", 150, NULL, 1, NULL);
xTaskCreate(Acquisition2, "Acq2", 150, NULL, 1, NULL);
xTaskCreate(Commande, "Cmd", 150, NULL, 1, NULL);
vTaskStartScheduler();
}
void Acquisition1 (void * pdata)
{
    for(;;)
    {
        SetChanADC(ADC_CH0);
        ConvertADC(); // début de conversion
        while(BusyADC());
        Val_LDR1 = ReadADC();
        xSemaphoreGive(SemSynAcq);
        vTaskDelay(1000/portTICK_RATE_MS);
    }
}
void Acquisition2(void* pdata)
{
    for(;;)
    {
        xSemaphoreTake(SemSynAcq, (TickType_t) portMAX_DELAY);
```

```

SetChanADC (ADC_CH1);
ConvertADC();
while(BusyADC());
Val_LDR2 = ReadADC();
xSemaphoreGive(SemSynCmd);
}
}
void Commande (void* pdata)
{ for(;;)
{ xSemaphoreTake(SemSynCmd, (TickType_t) portMAX_DELAY);
if(Val_LDR1 > Val_LDR2)
{
AR = 0;
AV = 1;
}
else { if(Val_LDR1 < Val_LDR2)
{
AV = 0;
AR = 1;
}
else
{ AV = 0;
AR = 0;
}
}
}
}

```

Q2. Oui, 1000ms, synchroniser la tâche 3 (commande) avec les deux tâches (Acquisition1 et Acquisition2).

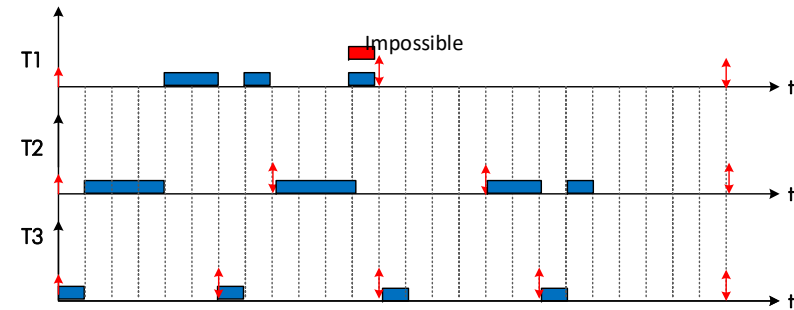
Q3. Oui on peut remplacer Val_LDR2 et Evénement_SynCmd par un BAL.

Exercice

$$1. U = \frac{5}{12} + \frac{3}{8} + \frac{1}{6} = 0,958$$

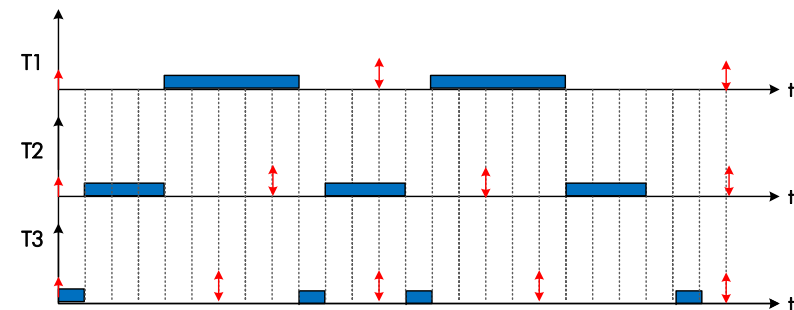
$U > 0,78$ on ne peut rien dire

$$2. PPCM(12,8,6) = 24$$



3. oui $U < 1$, T est ordonnançable EDF

4.



Un seul creux libre N°24