

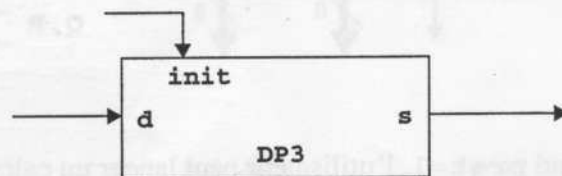
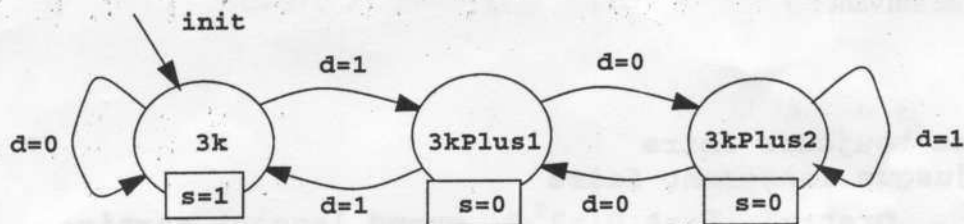
Diplôme d'ingénieurs Informatique et Télécommunication  
Contrôle continu ARC1

Durée de l'épreuve : 2 heures.

Notes de cours, TD et TP autorisées. Le sujet comporte 4 pages.

## Exercice 1 (temps conseillé : 35 mn)

On désire réaliser un circuit séquentiel DP3 :

défini par le diagramme d'états suivant <sup>1</sup> :

## Question

Réaliser ce circuit au moyen de 2 bascules D avec le codage suivant de l'état :

état	q1	q2
3k	1	0
3kPlus1	0	0
3kPlus2	0	1

On pourra se limiter à spécifier le circuit par des formules booléennes :

q1 := ???

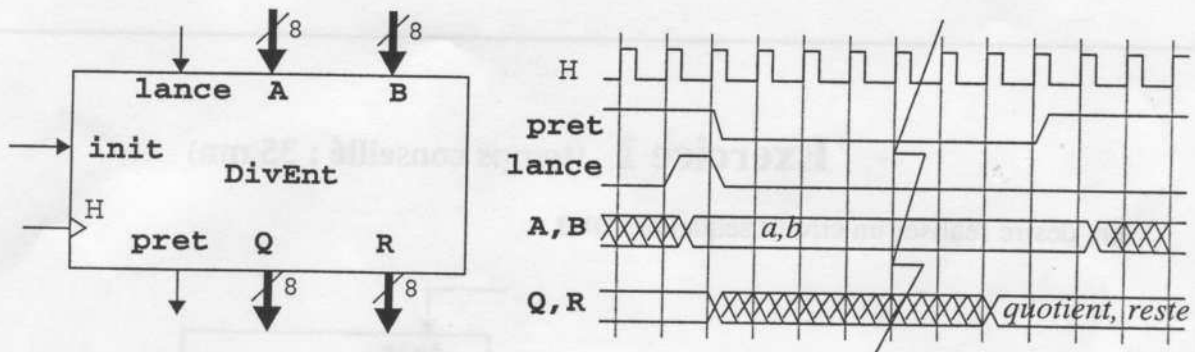
q2 := ???

s = ???

1. La sortie **s** de ce circuit indique si le nombre représenté en binaire par la séquence de bits reçue sur **d** depuis l'initialisation (poids forts en premier) est multiple de 3. Cette remarque n'a aucun impact sur la réponse à la question.

## Exercice 2 (temps conseillé : 85 mn)

On veut réaliser une machine de division entière, **DivEnt**, pour des nombres entiers positifs représentés en binaire sur 8 bits. Le brochage et le protocole d'utilisation de la machine **DivEnt** sont les suivants :



Quand **pret**=1, l'utilisateur peut lancer un calcul. Pour cela il affiche les données **A** et **B** et active le signal **lance**=1 pendant un seul cycle. La machine fait alors passer **pret** à 0. Quand le résultat est prêt sur les sorties **Q** et **R**, la machine le signale en faisant passer **pret** à 1 et le maintient jusqu'au prochain lancement de calcul. On se propose pour cela de mettre en œuvre l'algorithme suivant :

```

jusque toujours faire
  jusque lancement faire
    pret:=1, R:=A, D:=27×B, quand lance=1 sortir
  fait;
  pret:=0, Q:=0, I:=0;
  jusque FinCalcul faire
    cas
      I=7 : sortir
      I≠7 et R≥D : R:=R-D, Q:=2×Q+1, D:=D/2, I:=I+1
      I≠7 et R<D : Q:=2×Q, D:=D/2, I:=I+1
    fcas
  fait;
fait
  
```

Le calcul du quotient et du reste est réalisé par l'algorithme classique de division adapté pour la base 2. En 8 étapes, cet algorithme divise **A** par **B** et calcule le quotient dans **Q** et le reste dans **R**.

L'exemple suivant illustre la division de **A**=49 par **B**=5 ce qui conduit au résultat **Q**=9 et **R**=4.

Q	0		00110001	R
	0	R<D	00000101	D
	00	R<D	00110001	
	000	R<D	00000101	
	0000	R<D	00110001	
	00001	R<D	00000101	
	000010	R<D	00110001	
	0000100	R<D	00000101	
Q	00001001	R≥D	00001001	R

## Conception de l'unité de traitement

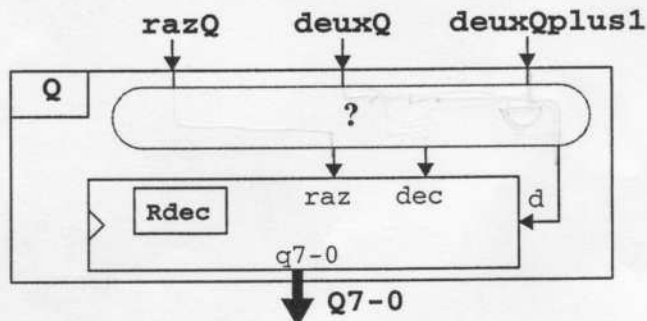
### Question 2.1 (temps conseillé : 15 mn)

On possède un registre à décalage doté d'une commande de mise à zéro **Rdec** dont voici les spécifications :

```

q7-0 := si raz=1 alors 00000000
        sinon si dec=1 alors q6 q5 q4 q3 q2 q1 q0 d
        sinon q7-0
    
```

Montrer comment réaliser très simplement, à l'aide d'un registre **Rdec**, un registre **Q** qui réalise les changements d'état suivants selon les commandes **razQ**, **deuxQ** et **deuxQplus1** :



```

Q7-0 := cas (razQ,deuxQ,deuxQplus1)
        soit 1,0,0 alors 2[0]
        soit 0,1,0 alors 2[2×[Q7-0]2]
        soit 0,0,1 alors 2[2×[Q7-0]2+1]
        soit 0,0,0 alors Q7-0
        (indéfini pour tout autre cas, c'est-à-dire n'importe quoi)
    
```

**Question 2.2 (temps conseillé : 35 mn)**

Donner un schéma de l'unité de traitement de la machine **DivEnt**. On dispose pour **D** d'un registre à décalage 15 bits (8+7) doté d'une commande de chargement, du registre **Q** de la question précédente, de compteurs avec mise à zéro...

On dispose également d'un soustracteur binaire 15 bits (bien qu'un soustracteur 8 bits suffirait comme le suggère l'exemple de calcul).

**Conception de l'unité de contrôle**

**Question 2.3 (temps conseillé : 15 mn)**

Donner le **diagramme d'états** de l'unité de contrôle qui respecte l'algorithme proposé. On fera apparaître clairement les conditions sur chaque flèche et les commandes générées dans chaque état.

**Question 2.4 (temps conseillé : 20 mn)**

Donner un schéma de réalisation de l'unité de contrôle réalisée au moyen d'un compteur chargeable.

