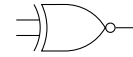


UE S1.2. Les nombres : Comment représenter le réel ?

Module d'Ingénierie



Présentation du module d'ingénierie de l'UE S1.2	iii
Calendrier	vii
Thème 1 - Initiation au traitement numérique des images	3
1 Introduction UE	3
2 SP1 - Images numériques avec Scilab	5
3 PC1 - Différents types d'images numériques.	9
4 PC2 - Retour sur les séances pratiques	13
5 SP2 - Contraste et histogramme d'une image	17
6 PC3 - Point d'étape	21
7 SP3 - Amélioration d'images	23
8 PC4 - Projet : Traitement d'images biologiques	25
Thème 2 - Electronique logique	31
1 PC1 - $1+1=10$	31
3 SP1 - Niveaux logiques en électronique	35
4 PC2 - Chronogrammes logiques	39

Ce document n'est volontairement pas un document de cours complet. Il a pour seul objectif de consigner la trame de de l'enseignement d'Ingénierie à l'attention des élèves. Il est mis à jour régulièrement, disponible sur la page Dokeos de l'UE.

Présentation du module d'ingénierie de l'UE S1.2

Contexte

Cet enseignement d'Ingénierie s'inscrit dans l'unité d'enseignement (UE) **S1.2 : Nombres : comment représenter le réel ?** qui totalise 123h d'enseignement et correspond à un crédit de 10 ECTS.

UE S1.2 Cette UE s'articule autour des nombres, de leur définition et de leur représentation. Dans un premier temps, les mathématiques travailleront autour des nombres et l'informatique introduira l'enseignement de la programmation par l'apprentissage du codage d'un nombre dans la mémoire. Le traitement du signal abordera les notions d'échantillonnage et de quantification afin de montrer comment acquérir des valeurs numériques. Les cours d'électronique logique et d'électricité viendront compléter ces notions d'interface entre grandeurs physiques et représentations numériques. La seconde partie de l'unité d'enseignement abordera les notions d'opérations et de manipulation des nombres par les fonctions mathématiques et les constructions classiques de programmation en informatique.

Ingénierie Les enseignements d'Ingénierie de cette UE s'organisent en deux thèmes :

Thème 1 - Initiation au traitement d'images 16,5 heures

- 5 séances de 1,5h en "Petite Classe" c'est à dire en demi-groupe (**PC**)
- 3 Séances Pratiques de 3h (**SP**) en demi-groupe.

Thème 2 - Electronique logique 12 heures

- 4 séances de 1,5h en "Petite Classe" (**PC**)
- 2 Séances Pratiques de 3h (**SP**)

Objectifs pédagogiques

A l'issue de ces 30 heures d'enseignement, vous serez capables de :

- Afficher une image avec le logiciel **Scilab**
 - Codée en niveaux de gris
 - Couleur
 - En noir et blanc
- Modifier les dimensions d'une image
- Choisir un dispositif d'acquisition d'images en s'appuyant sur la documentation fournie
- Mettre en œuvre et régler un dispositif d'acquisition d'images en s'appuyant sur la documentation fournie
- Représenter sous forme d'image un tableau de données
- Concevoir et mettre en œuvre un circuit électronique logique combinatoire

Vous serez par ailleurs capables de concevoir une carte de connaissances (*Mind map*) avec le logiciel **VUE**.

Contenu

Les différentes notions abordées dans chacune des deux parties sont représentées sur les cartes conceptuelles des figures 1 et 2.

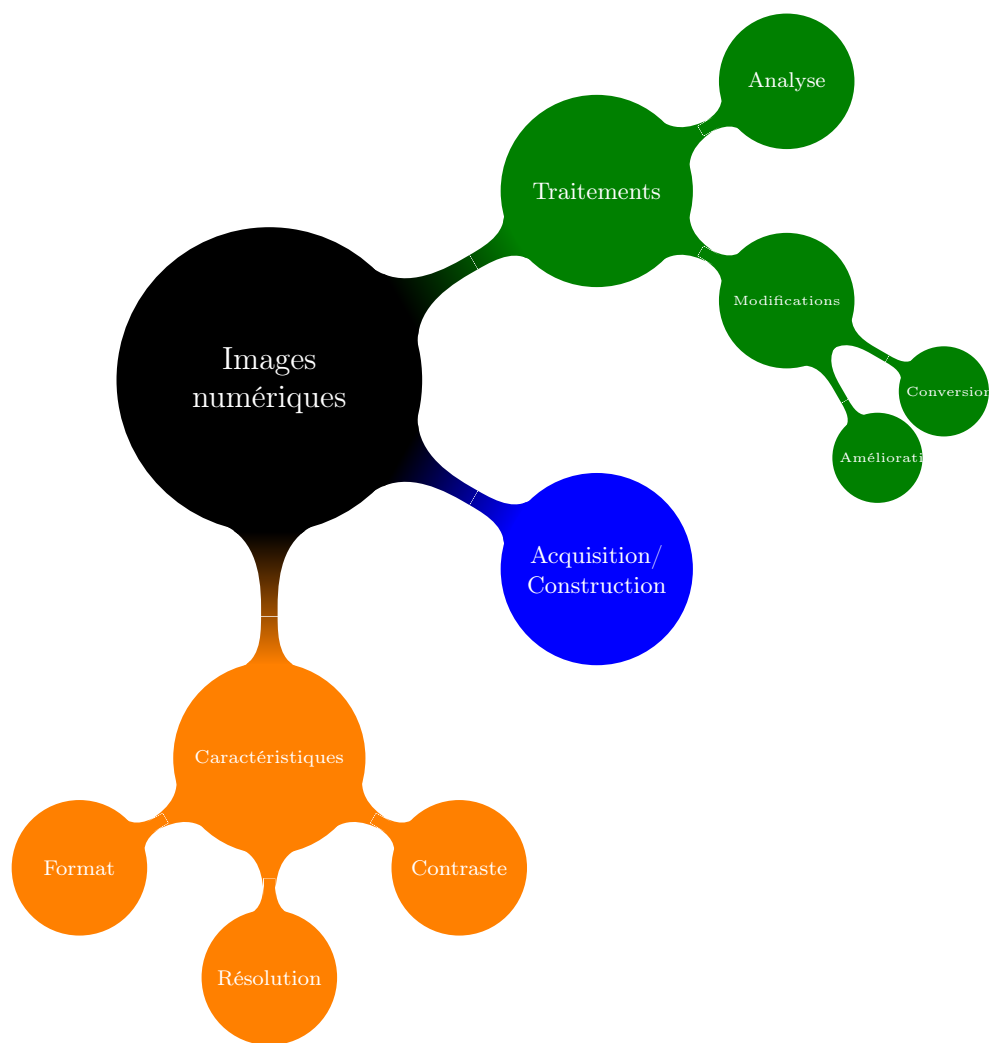


FIGURE 1 – Carte du thème 1 : Initiation au traitement d'images

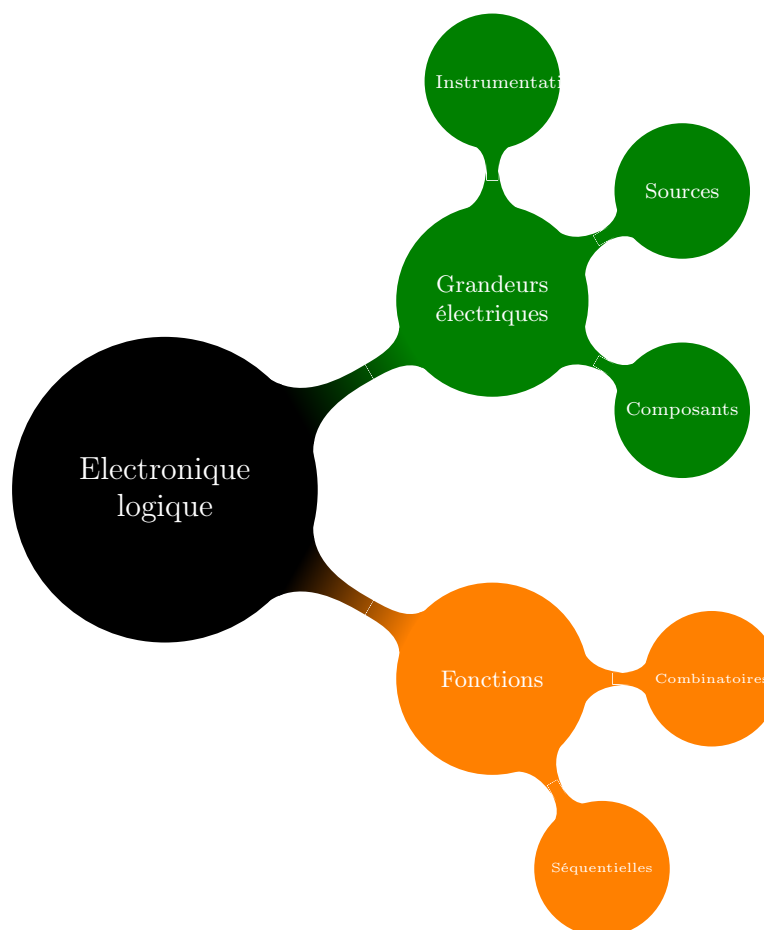


FIGURE 2 – Carte du thème 2 : Electronique logique

Équipe pédagogique

Fabienne BERNARD Professeure agrégée à l'Institut d'Optique graduate school, enseigne l'électronique, le traitement du signal et l'optique. (www.institutoptique.fr)

Tatiana SÉVERIN-FABIANI Ingénieure, Doctorante-enseignante, rattachée à l'Université Paris-Sud, elle effectue sa thèse de Physique, au synchrotron Soleil (www.synchrotron-soleil.fr), sur un sujet de *Imagerie et spectroscopie dans l'ultra-violet*.

Valérian GIESZ Ingénieur, Doctorant-enseignant, rattaché à l'Université Paris-Sud, il effectue sa thèse de Physique, au Laboratoire de Photonique et Nanostructures (www.lpn.cnrs.fr), sur un sujet de *Téléportation quantique à l'aide de sources semiconductrices*.

Modalités de contrôle des connaissances

Le contrôle des connaissances comprend différents types d'évaluation selon la figure suivante :

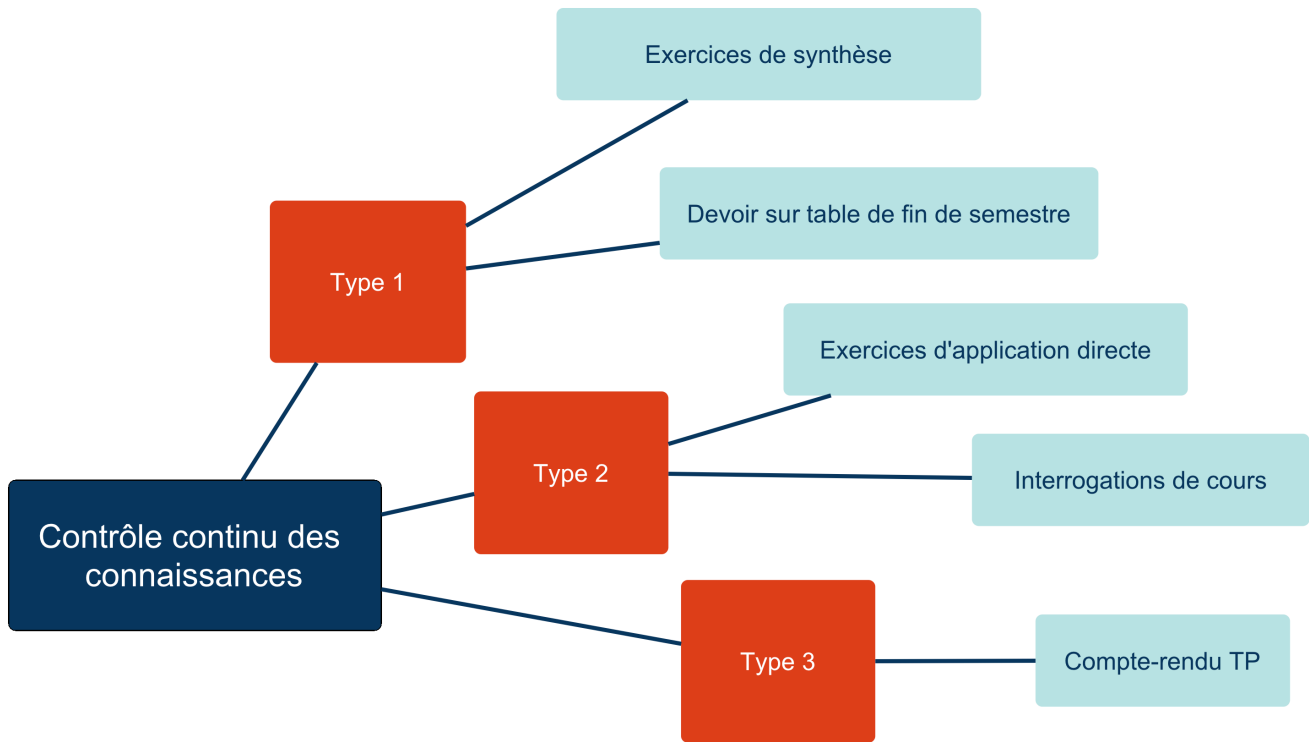


FIGURE 3 – Carte conceptuelle des Modalités de Contrôle des Connaissances, UE S1.2

Calendrier

UES1.2 – Planning des enseignements Ingénierie

<i>S</i>	<i>Date</i>	<i>horaire</i>	<i>durée</i>	<i>Groupe 1</i>	<i>Groupe 2</i>	
38	lundi 22 septembre	10:15	1,5	GC		Images
	jeudi 25 septembre	13:30	3	SP1		
	vendredi 26 septembre	08:30	1,5	IngéIm1		
10:15		1,5		IngéIm1		
39	mercredi 1 octobre	08:30	3		SP1	
	jeudi 2 octobre	13:30	1,5	IngéIm2		
		13:30	1,5		IngéIm2	
40	mercredi 8 octobre	08:30	3	SP2		
	vendredi 10 octobre	08:30	1,5	IngéIm3		
		10:15	1,5		IngéIm3	
41	mercredi 15 octobre	08:30	3		SP2	
	jeudi 16 octobre	13:30	1,5	IngéIm4		
		13:30	1,5		IngéIm4	
42	mercredi 22 octobre	08:30	3	SP3		
	jeudi 23 octobre	13:30	3		SP3	
	vendredi 24 octobre	08:30	1,5	IngéIm5		
		10:15	1,5		IngéIm5	
43	Vacances					
44	jeudi 6 novembre	13:30	1,5	IngéElec1		Electronique logique
		13:30	1,5		IngéElec1	
45	jeudi 13 novembre	08:30	3		IngéElec SP1	
46	mercredi 19 novembre	08:30	3	IngéElec SP1		
	vendredi 21 novembre	08:30	1,5	IngéElec2		
		10:15	1,5		IngéElec2	
47	vendredi 28 novembre	08:30	1,5	IngéElec3		
		10:15	1,5		IngéElec3	
48	jeudi 4 décembre	08:30	1,5	IngéElec4		
		08:30	1,5		IngéElec4	
49	mardi 9 décembre	13:30	3	IngéElec SP2		
	mercredi 10 décembre	08:30	3		IngéElec SP2	

Quelques pistes bibliographiques et crédits

Pour construire ce module, les enseignants se sont aidés de diverses sources, dans lesquelles vous pourrez trouver des informations complémentaires.

Pour le thème de traitement d'image

Cours de traitement d'image, *Richard Alan Peters II*, *Vanderbilt university of engineering*, archive.org/details/Lectures_...
(en anglais)

Cours de Traitement d'images, Introduction, *Anne Vialard*,
dept-info.labri.fr/~vialard/Traitement/cours/cours1.pdf

Cours de Traitement d'images, *Caroline Kulcsár*,

Site de l'INRIA : Interstices interstices.info

Site de vulgarisation scientifique : Minute physics www.youtube.com/user/minutephysics

Cours L'imagerie médicale. MIT OpenCourseWare ocw.mit.edu/HST.582J

Cours "Comment utiliser les histogrammes d'une image numérique", École Nationale des Sciences Géographiques. www.en...

Travail préparatoire d'Imad Amri, élève de l'ENS Cachan au cours de son stage pédagogique.

Pour le thème d'électronique logique

Cours Tout est numérique, *Henri Benisty*,
<http://paristech.institutoptique.fr/site.php?id=89>

Cours Logique des systèmes numériques, *Sylvain Martel*, École polytechnique de Montréal,
<http://wiki.polymtl.ca/nano/fr/index.php/INF1500>

Cours Processeurs et Architectures numériques. *Tarik Graba*, Telecom ParisTech,
<http://perso.telecom-paristech.fr/%7Egraba/>

Thème 1 - Initiation au traitement numérique des images

Séance 1

Introduction UE

L'objectif de cette séance est de présenter et de motiver l'unité d'enseignement dans son ensemble. La première partie, 1 heure environ, est l'occasion de se confronter aux problèmes de codage de l'information, de manipuler et afficher une première image numérique.

1 Consignes de l'activité sémaphore

La moitié d'entre vous va recevoir une liste d'informations à faire passer à l'autre moitié du groupe. Ces informations se présentent sous la forme de triplets de nombres entiers compris entre 1 et 10. Exemple : (4,3,1) (5,6, 2) (10,2,1). Vous ne disposerez pour transmettre ces informations que deux codes gestuels possibles.

Déroulement

Etape 0 Former des binômes. Dans chaque binôme : un membre *A* et un *B*.

Etape 1 Une fois que vous aurez reçu les listes, personne n'aura le droit de parler. Vous n'aurez le droit que d'utiliser deux gestes pour coder l'information à transmettre. Une pause est un geste . . .

Vous avez 15 minutes pour vous mettre d'accord sur les deux gestes, et réfléchir à votre stratégie.

Etape 2 Chaque membre du groupe *A* reçoit sa liste à coder et, sans parler, mime l'information codée à l'aide des deux gestes choisis avec son binôme, suivant la stratégie établie.

Membre du groupe *B*, vous devez décoder et noter l'information reçue.

Etape 3 Exploitation des informations transmises . . .

2 Exploitation des résultats

Les données forment un tableau de 8 lignes et de 10 colonnes. Chaque triplet de nombres correspond (ligne, colonne, valeur). La valeur peut être 0 (noir), 1 (gris) ou 2 (blanc).

Le résultat doit ressembler à l'image suivante :



Séance 2

SP1 - Affichage et manipulation des images numériques avec Scilab

Un compte-rendu de 2 pages max est à remettre au plus tard 7 jours après la séance (dépôt sur Dokéos).

Sommaire

1	Prise en main de l'environnement logiciel	5
2	Lecture et affichage d'une image avec Scilab	5
2.1	Images en niveaux de gris	6
2.2	Images couleurs	6
3	Construction d'images	7

1 Prise en main de l'environnement logiciel

Pour pouvoir faire du traitement d'images (numériques), on doit utiliser un logiciel de calcul. Dans les Séances Pratiques, nous allons utiliser le logiciel **Scilab**. Ce logiciel a l'avantage d'être gratuit et a une syntaxe très proche du logiciel commercial **Matlab**. Vous pourrez donc apprendre à programmer avec ce logiciel chez vous pour approfondir ce que vous avez vu en TP et ainsi mieux comprendre les objectifs de celui-ci. En outre, le logiciel est en français ce qui facilite la compréhension des fonctions notamment lorsqu'on sollicite l'aide.

Pour prendre en main le logiciel, il est conseillé de parcourir le document "Scilab pour vrais débutants" disponible sur le site :

<http://www.scilab.org/fr/resources/documentation/tutorials>.

↔ Avec l'aide de ce document, réaliser les opérations suivantes :

- Enregistrer la valeur $\sqrt{\pi}$ dans une variable appelée **a**.
- Construire un vecteur (c'est à dire un tableau d'une ligne), noté **T** comprenant les nombres entiers entre -10 et 10 .
- Construire un vecteur, noté **I** comprenant les carrés des nombres entiers entre -10 et 10 .
- Afficher la courbe d'évolution de la fonction $f(x) = x^2$ pour x variant entre -10 et 10 .
- Donner un titre à la figure et la copier dans un fichier de type **MS/Word** ou **LibreOffice/Writer**.
- Construire un tableau **Tab** de 3 lignes et 3 colonnes comprenant les entiers de 1 à 9 .

2 Lecture et affichage d'une image avec Scilab

Pour manipuler des images on utilise un module supplémentaire, nommé **SIVP** (pour *Signal Image and Video Processing*).

SIVP - Quelques commandes utiles

`imread` : convertit un fichier image en un tableau de valeurs numériques.

Exemple : `I=imread('Fichier.bmp');`

`imwrite` construit un fichier image à partir d'un tableau de valeurs numériques. L'extension (`.bmp` ou `.jpg` ou ...) définit le format utilisé (bitmap, jpeg, etc.).

Exemple : `imwrite(I,'Fichier.bmp');`

`imshow` affiche l'image dans une fenêtre SIVP (ce n'est pas une figure Scilab).

Exemple : `imshow(I)`

2.1 Images en niveaux de gris

↪ Convertir l'image `SpaceInvader.bmp` en un tableau de valeurs numériques, noté `TabSI`. Afficher les valeurs du tableau ainsi que l'image (un zoom est le bienvenu...).

Q1 A quelles valeurs numériques correspondent les couleurs blanche, grise et noire ?

↪ Construire un tableau `TabSI2` par division des valeurs du tableau `TabSI` par 2 et afficher l'image correspondante.

↪ Construire un fichier `SpaceInvader2.bmp` à partir du tableau `TabSI2`.

↪ Construire un tableau `TabSI3`, correspondant à l'image "inversée" de la précédente, c'est à dire noir→blanc, blanc→noir.

Scilab - Quelques commandes utiles

`ones` : construit un tableau dont tous les éléments sont égaux à 1.

Exemple : `DesUns=ones(3,4);` est un tableau de 3 lignes et 4 colonnes.

`size` : fournit la dimension d'un tableau, c'est à dire le nombre de lignes et le nombre de colonnes.

Exemple : `[nl,nc]=size(A);`

`:` sélectionner une partie de tableau.

Exemples : `A=T(10:13,20:24);` ou `T(10:13,20:24)=ones(4,5);` ou `L3=T(3,:);`

↪ Convertir en tableau (`TabV`) et afficher l'image `Vache.bmp`.

Q2 Quelle est la dimension de cette image ? Comparer avec l'image `SpaceInvader.bmp`.

↪ Proposer et mettre en œuvre une méthode permettant d'incruster l'image `SpaceInvader.bmp` dans l'image `Vache.bmp`. Sauvegarder cette nouvelle image.

2.2 Images couleurs

↪ Convertir en tableau (`TabT`) et afficher l'image `'Tuyaux.jpg'` et déterminer les dimensions de `TabT`.

↪ Appliquer les lignes de commande suivantes : `TabT_R=TabT(:, :, 1);` `TabT_V=TabT(:, :, 2);` `TabT_B=TabT(:, :, 3);`.

Q3 Quels sont les dimensions des tableaux `TabT_R`, `TabT_V` et `TabT_B` ?

↪ Afficher les trois images correspondantes. Afin d'afficher les trois éléments sur la même figure, on pourra construire un tableau `TabT_RVB` avec la commande suivante : `TabT_RVB=[TabT_R,TabT_V,TabT_B];` et afficher l'image correspondante.

Q4 Repérer en comparant l'image initiale et les 3 sous-images comment les différentes couleurs sont codées.

Une image couleur RGB est la superposition de 3 images en niveaux de gris, elle est représentée dans `scilab` par une *hypermatrice* c'est à dire un ensemble de 3 tableaux (ou matrices) de même taille représentant chaque couche. Les *hypermatrices* se manipulent comme des tableaux mais avec un indice de plus (le troisième) qui désigne la couche R = 1, G = 2, B = 3 .

↪ En prenant une image de votre choix (en couleur), la convertir en niveaux de gris. On pourra utiliser pour cela la fonction `rgb2gray`. L'enregistrer ensuite sous le format `bmp`.

Q5 Donner l'espace de stockage que prend cette image et comparez à celle en couleur.

Q6 Comment fonctionne la fonction `rgb2gray`? (Voir les rubriques d'aide du module `SIVP`)

3 Construction d'images

↪ Afficher l'image correspondant au tableau `Tab` construit au début de la séance.

↪ Comparer les résultats obtenus par les trois commandes suivantes :

- `imshow(Tab);`
- `imshow(uint8(Tab));`
- `imshow(uint8(Tab*255/10));`

Q7 Que fait la commande `uint8`? A quoi sert la multiplication par le coefficient `255/10` dans la troisième commande?

↪ Construire des matrices 3×3 et 10×10 avec des valeurs arbitraires (comprises entre 0 et 255. Puis les afficher avec `imshow`.

↪ Construire une image de 9 couleurs : Rouge, Vert, Bleu, Noir, Blanc, Jaune, Gris, Marron, Violet. (On pourra s'aider de la palette de couleurs du logiciel Microsoft Paint ou de celle de Microsoft Word.)

Séance 3

PC1 - Différents types d'images numériques.

Objectifs de la séance À l'issue de cette séance, vous serez capables de définir les caractéristiques des images numériques et d'utiliser le logiciel VUE pour réaliser une carte de connaissances.

Sommaire

1	Classification	9
2	Utilisation du logiciel VUE	9
3	Mise en commun	10
4	Pour la prochaine fois	10
5	Crédits	10

1 Travail en groupe et individuel - Classification

Première étape Trouver les mots-clés

1. Individuel - 5mn - Qu'est ce qu'une image ? Proposer 10 mots clés correspondant à la définition.
2. Groupe de 4 - 10mn - Se mettre d'accord sur 5 mots-clés.
3. Classe - 5mn - Restitution

Deuxième étape Organiser les concepts

1. Par groupe de 4 élèves, choisir des mots-clés (entre 5 et 10)
2. Individuellement : les relier par un verbe (c'est à dire construire la carte conceptuelle proprement dite).

Quelques propositions de verbes (liste de Rémi Bachelet dans son cours de cartes conceptuelles¹) :

- | | |
|-----------------------------|----------------------------------|
| ... affecte ... | ... est une condition pour ... |
| ... cause ... | ... est une conséquence de ... |
| ... contredit ... | ... est une manifestation de ... |
| ... dépend de ... | ... est une partie de ... |
| ... est associé avec ... | ... est une propriété de ... |
| ... est ... | ... inclus ... |
| ... est composé de ... | ... influence ... |
| ... est un phénomène de ... | ... produit |
| ... est un symptôme de ... | |

2 Travail individuel : Utilisation du logiciel VUE pour représenter la classification obtenue

Le logiciel VUE permet de réaliser des cartes de connaissances (ou *Mind map*) et de les afficher. Il est très facile de le mettre en œuvre pour des cartes simples. Des cartes plus complexes peuvent aussi être réalisées avec ce

1. rb.ec-lille.fr/1/CarteConceptuelle/cours-cartes_conceptuelles.pdf

logiciel, des fonctionnalités de construction de carte et des outils d'affichage sophistiqués sont de plus disponibles mais nous n'en verrons pas ici les fonctionnalités.

Travail demandé : réaliser individuellement la carte conceptuelle précédente avec le logiciel VUE. Il vous est demandé de rajouter des images pour illustrer votre carte. Des images sont disponibles sur Dokéos, fichier PC1.zip (toutes ces images sont sous licence Creative Commons). Vous pouvez ajouter des images de votre choix.

3 Mise en commun


Comparaison des cartes dans chaque groupe. Puis répondre par écrit aux questions suivantes :

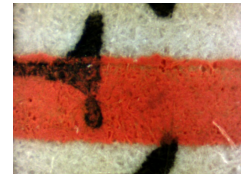
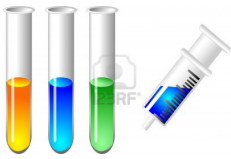
Qu'avez-vous appris au cours de cette séance? (Noter individuellement trois choses que vous avez apprises.)
Quand pensez-vous utiliser ces cartes?

4 Pour la prochaine fois

Mettre au propre la carte et la déposer sur le site d'ici une semaine.

5 Crédits

Les images utilisées dans cette séance sont sous licence  **creative commons**, disponibles pour la plupart sur le site [flickr](#) :



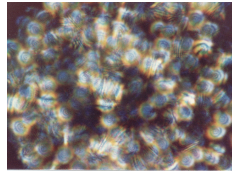
Mind the goat

Mind the goat

Mind the goat

Mind the goat

Microscope image of telegraph code books, hc gilje



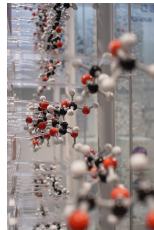
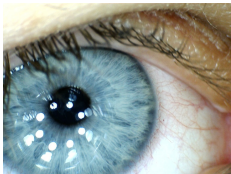
Mind the goat

Iris petal under microscope, hc gilje

John Pathfinder Lester

Science city center, Yaisog Bonegnasher

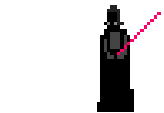
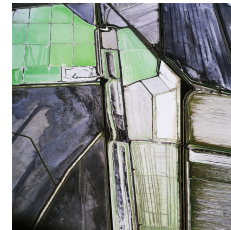
Dava Bay Antineutrino Detector, Lawrence Berkeley Nat'l Lab - Roy Kaltschmidt, photographer.



Microscope eye, Lilspikey

Bongury

Haraldhobbit

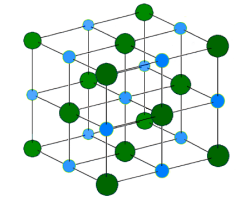
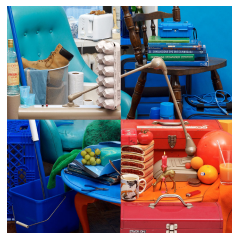


Faig Ahmed

Lac salé David Maisel

Lac salé David Maisel

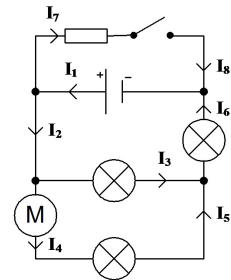
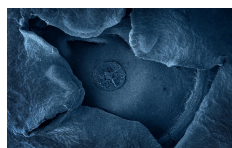
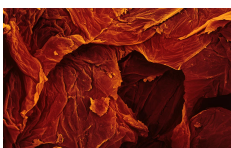
www.flipflopflyin.com/minipops/



Bela Bordosi

Bela Bordosi

Sel x45, Caren Alpert



Tomate séchée, Caren Alpert

Myrtille x 19, Caren Alpert

Bela Bordosi

Séance 4

PC2 - Retour sur les séances pratiques

Objectifs de la séance A l'issue de cette séance, vous aurez mis en évidence les points les plus importants traités en séances pratiques et vous aurez les éléments pour réussir la rédaction d'un compte-rendu.

Sommaire

1	Notions de traitement d'image vues en séances pratiques	13
2	Pour la rédaction d'un compte-rendu	13
2.1	Objectifs et contraintes	13
2.2	Contenu	14
2.3	Structure du document	14
2.4	Mise en forme	14
2.5	Dépôt	15
3	Au sujet du logiciel	15
4	Rédaction du CR SP2	15

1 Notions de traitement d'image vues en séances pratiques (Travail en groupe)

Quelles sont les notions importantes vues au cours de ces deux premières séances pratiques ? Répondre à cette question par groupe de 4.

On rappelle le plan des 2 TPs :

SP1

- 1 Prise en main de l'environnement logiciel
- 2 Lecture et affichage d'une image avec Scilab
 - 2.1 Images en niveaux de gris
 - 2.2 Images couleurs
- 3 Construction d'images
- 4 A vous de jouer

SP2

- 1 Pour bien débiter
- 2 Contraste
- 3 Histogramme
- 4 Quantification - Nombre de niveaux
- 5 Crédits

2 Pour la rédaction d'un compte-rendu

2.1 Objectifs et contraintes

Pour rédiger un compte-rendu, il faut toujours partir des objectifs visés et des contraintes imposées, contraintes sur le format, le contenu et aussi sur la date de remise.

Pour les CR d'Ingénierie (traitement d'images et électronique logique) :

Les objectifs sont de plusieurs ordres :

1. L'objectif principal est, en rédigeant, d'améliorer la compréhension des concepts et de s'appropriier les savoirs-faire (pouvoir dire : ça, je sais faire!)

2. Un deuxième objectif est de s'entraîner à la présentation scientifique, élément qui fait partie à part entière de toute formation scientifique (Pourquoi apprendre si je ne sais pas le faire partager ?)
3. Enfin, ces compte-rendus permettent aux enseignants de suivre la progression dans les apprentissages et d'évaluer par une note le travail entrepris et les connaissances acquises.

Les contraintes sont fixées à deux pages maximum, au format `.pdf`, remis quelques jours après la séance et des éléments de contenu (comme le paragraphe 1 de la SP1) ne sont pas obligatoires.

Premières indications Ce sont les deux premiers objectifs qui doivent être gardés en tête lors de la rédaction, la note qui en découle n'est qu'une conséquence (heureuse!) de leur réussite! Plus précisément :

1. **il faut rédiger le compte-rendu en imaginant que l'on s'adresse non pas à l'enseignant mais à une tierce personne** (un élève à l'autre bout du monde par exemple) qui souhaite faire les mêmes types d'expériences mais avec du matériel différent. C'est à dire pour le traitement d'image, sur d'autres images et avec un logiciel différent.
2. il faut avoir l'ambition d'**expliquer et non pas seulement de décrire**.

La contrainte de 2 pages maximum donne des indications du niveau de détails attendu pour les explications, forcément ici limité. Le compte-rendu est donc plus court que l'ensemble des notes et images prises en séance, les quelques jours de délai donnent le temps de trier, organiser et commenter celles-ci.

2.2 Contenu

Peut figurer dans un compte-rendu tout ce qui peut aider à la réalisation et à la compréhension des phénomènes étudiés : images, relevés de mesure, courbes et surtout commentaires. Ici, ce sont les notions de traitement des images qui doivent être présentées et pas le fonctionnement du logiciel par exemple.

Ne doit pas figurer tout ce qui est inutile : une courbe ou une image sans légende, des lignes de code sans commentaires, l'humeur du binôme, etc.

Toute image ou courbe doit être accompagnée d'une légende et doit être citée dans le corps du texte (s'il n'y a rien à en dire, ce n'est sans doute pas la peine de la faire figurer!).

Les lignes de code scilab ne sont pas obligatoires, elles peuvent être consignées en annexe avec quelques commentaires.

2.3 Structure du document

Le document doit impérativement comporter :

- vos noms,
- la date de la séance,
- le titre,
- une ou deux phrases d'introduction,
- une conclusion.

et il doit être paginé (numéros de page/ Nombre total de pages). Il est conseillé de suivre la structure de l'énoncé dans le compte-rendu (si l'enseignant a choisi de découper la séance en trois parties, c'est peut-être une bonne idée?) mais ce n'est pas obligatoire.

2.4 Mise en forme

La contrainte de format doit être suivie! Suivre les contraintes n'est pas forcément un frein à la créativité, au contraire, cela oblige parfois à explorer d'autres façons de faire, moins évidentes mais plus enrichissantes!

Quelques conseils dans l'utilisation d'un traitement de texte (Microsoft Word, Libre Office Writer ou LaTeX ou ...).

- La taille de la police doit être de 10 ou 12 pts,
- les marges peuvent être réduites par rapport à celles définies par défaut par le logiciel,
- l'organisation sous deux colonnes peut permettre d'avoir un texte plus dense,
- utiliser le correcteur orthographique.

Exporter au format .pdf

Word : Menu Fichier / Enregistrer sous ... et choisir le format .pdf dans la fenêtre suivante.

Writer: Menu Fichier / Exporter au format pdf

2.5 Dépôt

Avant de déposer le fichier sur Dokéos (ne pas l'envoyer par mél, sauf en cas de pb) :

- vérifier que le nombre de pages est bien celui attendu,
- renommer le fichier au format `DupondEtDupontSP4Images.pdf` (...si vous formez le binôme Dupond et Dupont!).

Enfin, merci de le déposer dans le répertoire correspondant! Par exemple : Dokéos/UES1.2 Les Nombres/Travaux/SP2_CR/ pour la séance pratique 2.

3 Au sujet du logiciel

Pour pouvoir travailler efficacement, il faut avoir les bons outils. Le logiciel **Scilab** est l'outil choisi ici pour le traitement des images. Il doit être bien configuré, en particulier le module **SIVP** qui s'installe en tapant : `AtomsInstall SIVP` dans la fenêtre de commande **Scilab** (ce n'est à faire qu'une seule fois). La documentation de ce module peut être trouvée sur la page sivp.sourceforge.net/func-list-0.5.0/whatis.htm.

4 Rédaction du CR SP2 (Travail individuel)

Commencer la rédaction du compte-rendu de la séance pratique 2 sur ordinateur. Pour commencer :

- Ouvrir un nouveau fichier de traitement de texte. Modifier les marges si besoin.
- Noter les noms du binôme dans l'en-tête noter le titre.
- Noter le plan de la séance (quitte à le modifier par la suite si vous le souhaitez)

Séance 5

SP2 - Contraste et histogramme d'une image

Un compte-rendu de 2 pages est à remettre à l'issue de la séance, au plus tard mardi 08 octobre.

Sommaire

1	Pour bien débiter, un petit test	17
2	Contraste	17
3	Histogramme	18
4	Quantification - Nombre de niveaux	19
5	Crédits	20

L'objectif de cette séance est d'aller un peu plus loin dans l'analyse des caractéristiques d'une image numérique en particulier par l'étude de l'histogramme des niveaux de gris présents dans l'image.

1 Pour bien débiter, un petit test

L'objectif de ce test est de faire un bilan à l'issue de la séance sur ce que vous avez appris. Cette fois-ci, vous allez vous prendre en photo avec une webcam. Pour cela un code vous est fourni (Code `essai_webcam.sce`).

↪ Prenez-vous en photo en exécutant le code donné.
Pour cela, taper la commande `exec(essai_webcam.sce)` ;

↪ Sauvegarder la photo sous le format `bmp` en donnant votre nom comme titre de la photo.

Q1 Quel est le nombre de pixels de la webcam ?

↪ Convertir l'image en niveaux de gris et la sauvegarder en `bmp` en lui donnant un nom clair.

↪ Extraire la voie bleue de l'image initiale et la sauvegarder.

↪ Incruster une image dans votre portrait en niveau de gris (un *Space Invader* sur le bout du nez ?) et sauvegarder l'image obtenue.

↪ Incruster une image dans votre portrait couleur et sauvegarder l'image obtenue.

↪ Réunir les 5 images dans une archive `.zip` et la déposer sur le site.

2 Contraste

↪ Télécharger l'archive `SP2.zip` sur la page Dokéos de l'UE.


↪ Créer un répertoire dans lequel décompresser l'archive. Lancer **Scilab** et, dans le navigateur de fichier, choisir le répertoire précédent comme répertoire de travail.

↪ Dans **Scilab**, lancer l'application **Scinotes** (Menu Applications) et créer un nouveau fichier `.sce`, toujours dans le répertoire de travail choisi précédemment.

Les lignes de codes seront tapées dans ce fichier, exécutées une à une ensuite.

↪ Ecrire un programme qui permet de convertir en tableau numérique, puis d'afficher les images `Mur2.bmp`, `Marches.bmp` et `Marches2.bmp` avec **Scilab** (voir paragraphe suivant pour quelques indications)

Utilisation d'un script Taper les lignes de commande dans le fichier `.sce`. Ensuite, pour exécuter ces lignes de code, deux solutions :

1. Cliquer sur l'icône  **Exécuter**, et alors l'ensemble des lignes écrites dans le fichier sont exécutées.
2. Ou sélectionner une ou plusieurs lignes à la souris, puis clic droit, et dans le menu qui apparaît, choisir **Evaluer la sélection avec écho (Ctrl+E)**

Le contraste est une propriété intrinsèque d'une image qui désigne et quantifie l'apport de niveaux clairs et sombres sur une image. Plus précisément, on définit le **contraste (de Michelson)** par :

$$C = \frac{I_{max} - I_{min}}{I_{max} + I_{min}}$$

où I_{max} désigne le niveau maximal d'intensité de niveaux de gris et I_{min} , le niveau minimal d'intensité.

Q2 Quelle est la valeur maximale possible pour ce paramètre ?

↪ Calculer le contraste de chacune des 3 images précédentes.

Quelques commandes utiles

`double` convertit un nombre entier en un nombre réel. Cette commande est nécessaire dès que l'on souhaite **faire des calculs**.

`uint8` convertit un nombre réel en un nombre entier compris entre 0 et 255 (8 bits). Cette commande est utile lorsque l'on souhaite **afficher** des images.

`min` : donne les valeurs minimales des colonnes d'un tableau de valeurs numériques. Pour obtenir la valeur minimale m du tableau Im dans son ensemble : `m=min(min(Im));`

`max` même utilisation que la commande précédente.

Q3 Peut-on modifier le contraste en multipliant les valeurs par un coefficient ? En ajoutant une constante ?

↪ Faire des essais afin d'obtenir un contraste maximal sur chacune des images.

Q4 Etablir à partir de vos essais une procédure pour maximiser le contraste d'une image.

3 Histogramme

Une analyse plus poussée fait appel au calcul de l'histogramme en niveaux de gris d'une image. Un histogramme est un graphique statistique permettant de représenter la distribution des intensités des pixels d'une image, c'est-à-dire le nombre de pixels pour chaque niveau de gris. Par exemple, on verra si ce sont les niveaux clairs ou noirs qui prédominent sur une image.

↪ Afficher l'histogramme de l'image `SpaceInvader.bmp`

Q5 Combien y-a-t-il de pixels noirs dans cette image? De pixels gris?

↪ Afficher l'histogramme de l'image précédente dont toutes les valeurs ont été divisés par 2.

↪ Afficher l'histogramme de l'image initiale inversée (noir ↔ blanc).

Quelques commandes utiles

`imhist` : permet de calculer l'histogramme d'un image

Exemple : `[N,V]=imhist(img)`; Les différentes valeurs prises par les pixels sont stockés dans un tableau V d'une ligne, et le nombre de pixels correspondant dans un tableau N d'un ligne.

`scf` ouvre une figure.

`title` permet d'ajouter un titre à une figure.

`bar` pour un tracé de courbes de type "barre verticale"

Exemple : `scf(0);bar(V,N);title('Ma figure')`; permet de tracer les valeurs N en fonctions des valeurs de V sur la figure 0.

↪ Visualiser les histogrammes des 3 images étudiées au paragraphe précédent en donnant un titre à chacun de vos histogrammes.

Q6 Comparer et commenter chacun de ces histogrammes. Quel doit être l'allure d'un histogramme pour que l'image soit de bonne qualité visuelle?

Q7 Proposer une solution pour augmenter la qualité de l'image `Terrasse2.bmp`.

↪ La mettre en œuvre et la tester sur des images de votre choix. Attention à convertir toute image couleur en une image en niveaux de gris grâce à la commande `rgb2gray` avant de calculer l'histogramme des niveaux de gris!

4 Quantification - Nombre de niveaux

Un paramètre permettant de déterminer la qualité d'une image est le nombre de nuances de couleur ou de gris utilisées dans l'image Comme on l'a vu jusqu'à présent, les images les plus courantes sont codées sur :

- 256 niveaux de gris pour les images noir et blanc,
- 256 niveaux pour chaque couleur Rouge, Vert, Bleu.

Q8 Combien de bits (de '0' et de '1') sont-ils nécessaires pour coder 256 niveaux différents?

Q9 Combien de couleurs différentes peuvent être codées si chaque voie est codée sur 256 niveaux?

Q10 Combien dans le cas où chaque voie serait codée sur 4 niveaux?

↪ Convertir en tableau numérique `TabPassage` l'image `Passage.bmp` et l'afficher.

Q11 Quelle est la dimension de l'image?

↪ Appliquer la commande `TabPassage2=64*(TabPassage/64)`; et afficher l'image correspondante.

↪ Calculer et afficher l'histogramme de cette image.

Q12 Expliquer le résultat obtenu.

↪ Appliquer le même type de commande sur une image couleur de votre choix.

Q13 Quels sont les effets sur les images de la diminution du nombre de niveaux de codage ?

5 Crédits

Les images utilisées dans cette séance sont sous licence  **creativecommons**, disponibles sur le site [flickr](https://www.flickr.com/photos/) :



Vadim Timoshkin



VT



VT



Charles Henry



Bert Kaufman



Emilio K.



flowcomm



Coconino national forest



Thomas Hawk

Séance 6

PC3 - Point d'étape

Objectifs Cette séance est en deux parties, elle est l'occasion

1. de faire le point sur les travaux remis jusqu'à présent et sur leur mode d'évaluation,
2. de présenter les notions importantes de traitement d'image vues en TP.

Sommaire

1	Au sujet des travaux remis jusqu'à présent	21
1.1	Carte conceptuelle	21
1.2	Compte-rendu des séances pratiques	21
1.3	Auto-évaluation du 04 octobre	22
2	Notions importantes de traitement d'image	22

1 Au sujet des travaux remis jusqu'à présent

1.1 Carte conceptuelle

Notation en 3 parties de C à A.

Pertinence des concepts Les boîtes-concepts ont :

- A toutes un intitulé clair et pertinent,
- B un intitulé pas toujours très clair ou pas toujours en rapport avec le sujet traité,
- C des intitulés obscurs ou bien sont trop nombreuses.

Justesse des liens Les flèches reliant les boîtes sont :

- A toutes légendées avec un verbe donnant du sens, de plus le lien est correct,
- B parfois pas légendées ou parfois avec des erreurs,
- C globalement pas légendées ou légendées incorrectement.

Aspect visuel La représentation graphique :

- A donne du sens par 1) la disposition des différentes boîtes sur la carte, 2) leur couleur ou leur forme, différente selon le type de concept, 3) les illustrations pertinentes qui sont utilisées,
- B manque de clarté, est un peu trop uniforme ou au contraire un peu trop désordonnée,
- C ne permet pas la lecture aisée des liens entre les concepts.

1.2 Compte-rendu des séances pratiques

SP1 Noté ABC, n'est pas prise en compte dans la notation de l'UE.

SP2 Moyenne de 12,5 sur les copies rendues (prise en compte dans le contrôle continu) La consigne de 2 pages max a été respectée (1 seul CR de 3 pages) Mettre des illustrations et expliquer! Un CR sans aucune image ni courbe d'histogramme n'est pas souhaitable!

1.3 Auto-évaluation du 04 octobre

Seuls 19 élèves ont répondu, sans doute parce ce qu'il y avait beaucoup d'auto-évaluations à faire ce jour là ? Il est encore temps pour le faire ! (La note n'est pas prise en compte dans la notation de l'UE)

2 Notions importantes de traitement d'image

voir Diapos.

Séance 7

SP3 - Amélioration d'images

Un compte-rendu de 2 pages est à remettre à l'issue de la séance, au plus tard mardi 15 octobre pour les élèves du groupe 2 et le 22 octobre pour les élèves du groupe 1.

Sommaire

1	Egalisation d'histogramme	23
1.1	Appliquer une transformation d'histogramme	23
1.2	Modifier le programme pour appliquer la transformation à une autre image	23
2	Un autre traitement : seuillage des niveaux	24

Objectifs de la séance Après l'affichage et l'analyse des images, l'objectif ici est de transformer les images afin mieux "voir" certaines informations. L'outil principal des méthodes de transformation est l'histogramme des niveaux de gris, vus à la séance précédente.

1 Egalisation d'histogramme

1.1 Appliquer une transformation d'histogramme

↪ Télécharger sur Dokéos le fichier `SP3.zip` et le décompresser dans votre répertoire de travail.

↪ Editer à l'aide de `Scinotes` le fichier `SP3.sce`. Exécuter le code.

Q1 Comparer l'image et son histogramme avant et après le traitement n°1 ainsi qu'après le traitement n°2. Le traitement n°1 est appelé égalisation d'histogramme. Expliquer ce terme. Quelle est la conséquence de ce traitement sur l'aspect visuel de l'image ?

Q2 Analyse du code : quel nom de variable correspond à l'image initiale ? aux images transformées ? A quoi correspondent les variables `n1` et `nc` ? les variables `sb` et `sh` ?

Q3 Donner un nom au traitement n°2.

↪ Calculer le contraste de l'image avant et après l'égalisation d'histogramme sur différentes images (`Terrasse2.bmp`, `Mur2.bmp`, `Marches2.bmp` par exemple).

Q4 Ce traitement permet-il d'améliorer le contraste de l'image ?

1.2 Modifier le programme pour appliquer la transformation à une autre image

↪ Prendre une photo à l'aide de la webcam, la convertir en une image codée en niveaux de gris et l'enregistrer dans votre dossier de travail.

↪ Enregistrer une copie du programme `SP3.sce` sous un autre nom et le modifier pour appliquer l'égalisation d'histogrammes à cette photographie.

Q5 Qu'a permis de faire la fonction d'égalisation ? En quoi cette procédure est une "amélioration" d'image ? L'image obtenue est-elle plus une représentation plus fidèle de ce que l'oeil voit ?

↔ Prendre à nouveau une photo comprenant cette fois l'image d'une feuille blanche fortement éclairée par une lampe. La convertir en niveaux de gris.

Q6 Expliquer pourquoi les détails de la scène ne sont pas visibles.

↔ Modifier le programme précédent pour l'appliquer à cette image.

Q7 Expliquer l'aspect visuel de l'image obtenue.

2 Un autre traitement : seuillage des niveaux

Le seuillage permet d'extraire une fenêtre d'intensité.

Avec cette transformation, la nouvelle image ne visualise que les pixels dont le niveau d'intensité appartient à l'intervalle choisi :

- tous les pixels dont la valeur est inférieure à un seuil bas sont mis à zéro,
- tous les pixels dont la valeur est supérieure à un seuil haut sont mis à 255.

↔ Modifier dans le programme précédent les valeurs des paramètres `sb` et `sh` en les fixant à `sb=0` et `sh=128` et appliquer le traitement n°2 à l'image `tableau400.bmp`¹

Q8 Expliquer pourquoi on a ainsi réalisé un seuillage dans l'intervalle `[0 128]`.

Q9 Comment, en général, choisir le seuil pour uniformiser le fond de l'image (et donc mieux voir les écritures...)

↔ Appliquer cette méthode de seuillage à l'image `tableau_flash400.bmp`¹

1. Crédits : Caroline Kulscár

Séance 8

PC4 - Projet : Traitement d'images biologiques

Objectifs pédagogiques du projet A l'issue de ce projet, vous serez capables de transposer les savoirs-faire de traitement d'images vus en séances pratiques à un problème concret d'imagerie biologique.

Sommaire

1	Contexte : expériences d'adhésion cellulaire	25
1.1	Objectifs	25
1.2	Protocole d'acquisition des images	26
1.3	Images obtenues	26
1.4	Informations d'intérêt dans les images	26
2	Travail demandé	27
3	Pour bien débuter	27

1 Contexte : expériences d'adhésion cellulaire sur des échantillons de Titane biocompatibles

Ces expériences sont menées au Centre de Photonique Biomédicale par Guillaume DUPUIS, enseignant-chercheur à l'Université Paris-Sud. Il est l'auteur de ce texte de présentation.

1.1 Objectifs

L'étude porte sur des échantillons de Titane biocompatibles, typiquement destinés à être utilisés comme implant ou comme prothèse osseuse (prothèse totale de hanche par exemple). Chaque échantillon (notés 1, 2 et 3) a été fabriqué dans des conditions de température et de pression différentes, de telle sorte que chacun a une microstructure spécifique : chaque échantillon a une taille moyenne de grain caractéristique.

L'objectif de l'étude est de déterminer l'influence de cette microstructure sur les propriétés d'adhésion cellulaire.



FIGURE 8.1 – Photographie des échantillons de titane

1.2 Protocole d'acquisition des images

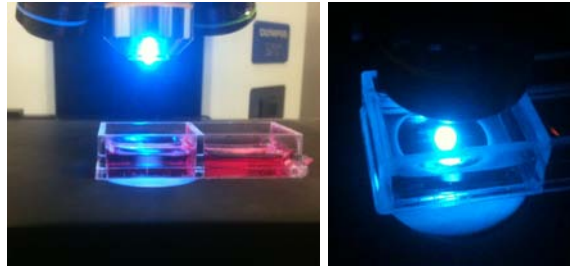


FIGURE 8.2 – Echantillons sous le microscope

A l'instant $t = 0$, les échantillons sont placés dans une cuve contenant des cellules en suspension dans du milieu de culture. Il s'agit de cellules MDCK (Madin-Darby Canine Kidney) dont la tubuline (une protéine structurale) a été marquée avec un fluorophore vert : la GFP (Green Fluorescent Protein). A intervalles de temps réguliers ($t = 2$ h, $t = 4$ h, etc.), les échantillons sont observés grâce à un microscope à épifluorescence : sous une lumière bleue intense, la GFP émet une faible lumière verte. Les images sont enregistrées avec une caméra très sensible.

1.3 Images obtenues

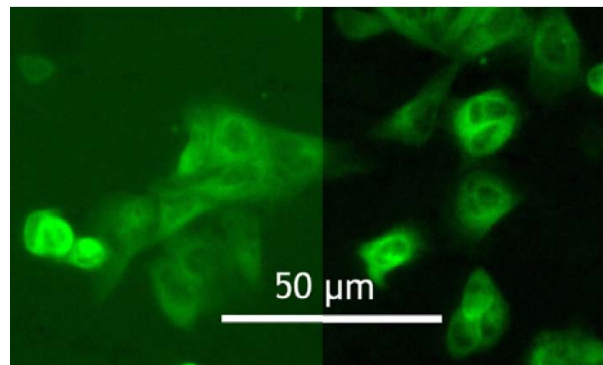


FIGURE 8.3 – Exemple d'images obtenues

On note un important bruit de fond sur les images. Il a plusieurs origines :

- les échantillons de titane sont réfléchissants comme des miroirs ; la lumière incidente (bleue) arrive par le dessus via l'objectif, et la fluorescence (verte) est collectée également par le dessus via l'objectif de microscope ; un filtre très efficace (mais pas parfait) bloque l'essentiel de la lumière bleue réfléchiée par le titane, mais pas l'intégralité.
- le milieu de culture est légèrement trouble et diffusant.

1.4 Informations d'intérêt dans les images

Pour évaluer les propriétés d'adhésion cellulaire des échantillons, il faut s'intéresser à la forme des cellules : les cellules adhérentes ont une fluorescence plutôt moins intense et elles sont de forme polygonale (entourées en orange sur l'image de gauche) tandis que les non-adhérentes sont plutôt plus intenses et circulaires (entourées en rose sur l'image de gauche).

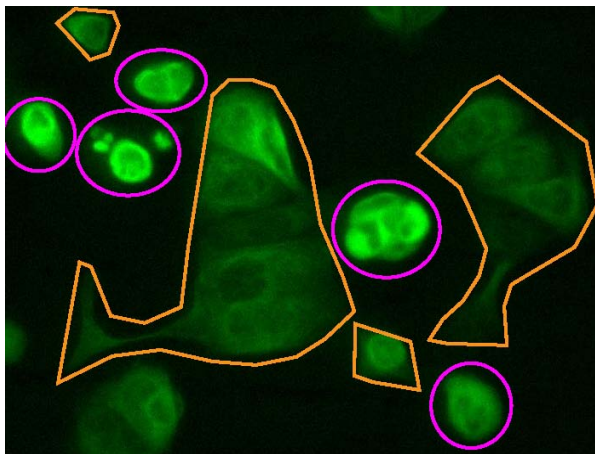


FIGURE 8.4 – Zones d'intérêt de l'image

Idéalement, pour chaque image, il faut retirer le bruit de fond, détecter les cellules, et évaluer le ratio adhérentes/non-adhérentes.

2 Travail demandé

Objectifs techniques du projet Fournir une solution de traitement des images (programme en **Scilab**¹) permettant de répondre en partie à la problématique précédente et ce, à l'aide des notions vues jusqu'à présent : contraste, couleur, histogramme, etc.

Livrables Deux documents sont attendus :

1. le code du programme avec des commentaires (un fichier `.sce`)
2. et un texte d'explication de la procédure suivie, une sorte de manuel d'utilisation à l'usage du chercheur qui souhaite exploiter les images, le format est limité à 4 pages maximum en pdf (vous pouvez faire plus court !). Ce texte sera illustré d'images, résultats des traitements effectués.

Calendrier Les deux documents sont attendus pour le vendredi 25 octobre au plus tard. La séance (GC4) d'Ingénierie du 24 octobre sera consacrée à répondre à vos (dernières) questions.

Images Une image de test est disponible sous Dokéos `ImageBio.jpg`.

Une série plus complète d'images est disponible avec le lien (<https://cloud.institutoptique.fr/public.php?service=files&t=604b9ad59b19ec9f0c7f9f0b8b623c7b>).

3 Pour bien débuter

↪ Convertir l'image `ImageBio.tif` en tableau de valeurs numériques avec **Scilab**.

Q1 Quelle est la taille de l'image ?

↪ Calculer et afficher l'historgramme de l'image convertie en niveaux de gris. Faire de même avec chaque voie RGB.

Q2 Que proposez-vous pour améliorer dans un premier temps le contraste de l'image ?

1. Il vous est possible aussi de proposer du code **Matlab** ou langage **C** si vous vous sentez plus à l'aise dans l'un ou l'autre de ces langages

Thème 2 - Electronique logique

Séance 1

PC1 - 1+1=10

Objectifs de la séance A l'issue de cette séance, vous serez capables d'expliquer comment un ordinateur additionne deux nombres alors qu'il ne manipule que des 0 et des 1. Vous serez en outre capable d'analyser des schémas graphiques utilisant les symboles de portes logiques.

Sommaire

1	Que sait faire un ordinateur ?	31
1.1	Les opérations élémentaires	31
1.2	2 opérations supplémentaires	31
2	Représentation graphique	31
3	Additionner ?	32

1 Que sait faire un ordinateur ?

1.1 Les opérations élémentaires

Opérations les plus simples ET et OU

Les opérations de base que sait réaliser un ordinateur sont les fonctions **ET** et **OU** entre deux bits. On verra lors de la séance **SP1** que ces bits sont codés par des niveaux de tensions électriques.

Q1 Rappeler la table de vérité de ces deux opérations.









1.2 2 opérations supplémentaires

Q2 Voici deux tables de vérité, chacune correspond à une opération logique entre 2 bits, notée **OP1** et **OP2** proposer un nom pour chacune d'elle.

A	B	A OP1 B	A	B	A OP2 B
0	0	1	0	0	0
0	1	1	0	1	1
1	0	1	1	0	1
1	1	0	1	1	0

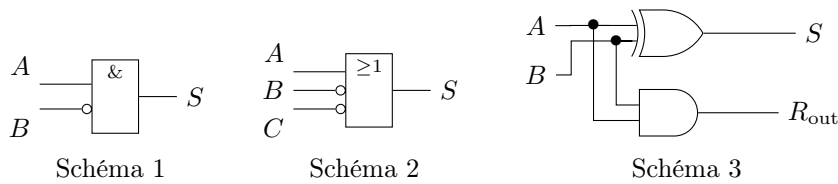
2 Représentation graphique

Combiner ces 4 opérations de base entre elles permet de réaliser des fonctions complexes. Un outil pratique pour les analyser est une représentation graphique. Voici la façon de représenter les fonctions logiques de bases.

Porte	Symbole européen	Gate	Symbole US
ET		AND	
OU		OR	
OU excl		XOR	
NON-ET		NAND	

Q3 Quelle est à votre avis la représentation d'une porte NOR? Quelle est sa table de vérité? Une porte XNOR?

Q4 Construire les tables de vérités des opérations suivantes.



3 Additionner ?

Le schéma de principe d'un additionneur 1 bit est donné figure suivante :

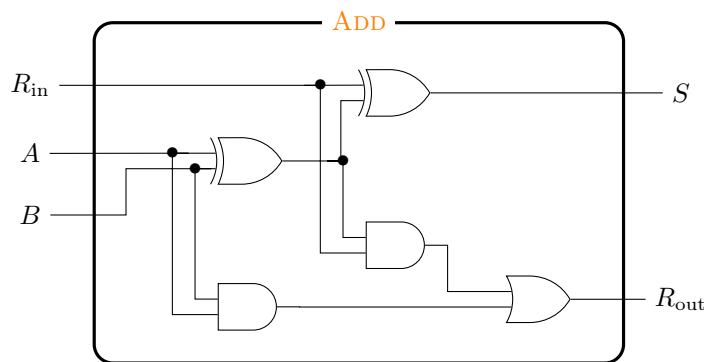


FIGURE 1.1 – Schéma d'un additionneur 1 bit

Q5 Vérifier que la table de vérité de cette opération permet de réaliser une addition de deux bits prenant en compte la retenue.

Pour additionner des nombres codés sur 4 bits, il suffit de cascader 4 modules de ce type...

FIGURE 1.2 – Principe d'un additionneur 4 bits. $S = A + B$

Q6 Quels sont les bits de poids faible? A quoi correspond R_{Out4} ?

Séance 3

SP1 - Niveaux logiques en électronique

Un compte-rendu de 2 pages est à remettre à l'issue de la séance, au plus tard mardi 10 décembre.

Objectifs A l'issue de cette séance, vous serez capables de mettre en œuvre un circuit électronique logique combinatoire à base de portes logiques et d'utiliser des instruments d'électronique de base. Vous serez par ailleurs capables de rechercher des informations dans une documentation de composant électronique.

Sommaire

1	Fonction réalisée par une porte logique électronique	35
2	Niveaux logiques	36
3	Une autre fonction logique	37
4	(Si vous avez le temps.) Régime variable.	37

1 Fonction réalisée par une porte logique électronique

Dans cette partie, vous allez réaliser le circuit de la figure 3.1 où le composant noté IC désigne un composant (une "puce") d'électronique logique.

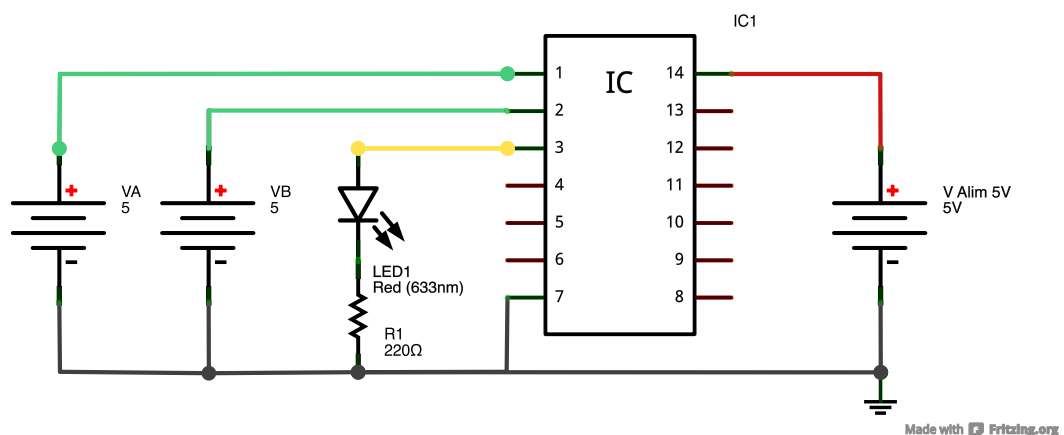


FIGURE 3.1 – Schéma électrique du circuit à réaliser

Votre montage va ressembler à la représentation de la figure 3.2. Cette représentation a été réalisée à l'aide du logiciel fritzing, le fichier correspondant est disponible sur Dokéos.

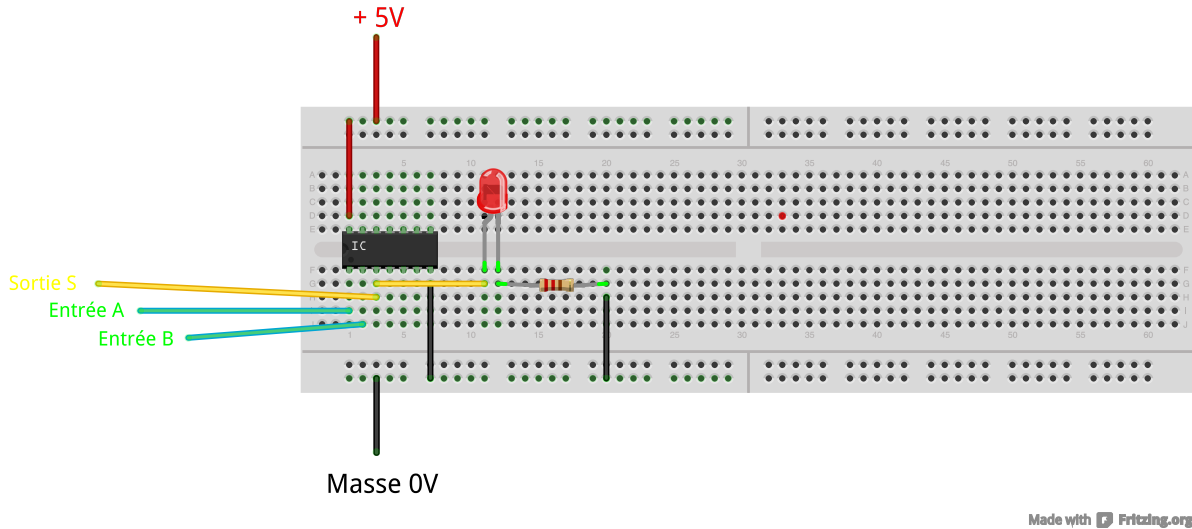


FIGURE 3.2 – Mise en œuvre pratique du circuit à réaliser

↪ Réaliser le câblage décrit sur les figures précédentes **SANS la DEL ni la résistance dans un premier temps** en respectant les couleurs des fils. (Vous pouvez télécharger le fichier `Circuit1.fzz` sur Dokéos si vous le souhaitez.)

↪ Placer sur chacune des entrées une tension de valeur réglable à l'aide de l'alimentation continue. Régler cette tension à 0 Volt dans un premier temps.

Q1 Que faut-il ajouter à ce circuit pour pouvoir mesurer la tension V_S (au niveau de la broche 3 de la "puce") ?

↪ Mesurer la tension V_S . Puis appliquer 5 Volt sur l'entrée A , mesurer la tension V_S . Puis appliquer 5 Volt sur l'entrée B , mesurer à nouveau V_S .

Q2 Remplir le tableau suivant.

V_A (Volt)	V_B (Volt)	V_S (Volt)
0	0	
0	5	
5	0	
5	5	

En déduire la fonction réalisée par ce circuit, si on considère que :

- une tension égale à 5 Volt correspond à un '1' logique,
- et que 0 Volt correspond à un '0' logique.

↪ Le circuit utilisé est un circuit 7400. Rechercher sur internet la documentation de ce circuit. Repérer en particulier dans cette documentation le brochage du composant et sa table de vérité.

Q3 Combien de portes logiques sont présentes dans ce composant. Quels sont les numéros des broches utilisées pour les entrées ? Pour les sorties ? A quoi correspondent les broches 7 et 14 ?

2 Niveaux logiques

↪ Appliquer 5 Volt sur l'entrée A . Faire varier la tension V_B appliquée sur l'entrée B et compléter le tableau suivant :

V_B (Volt)	V_S (Volt)
0	
1	
2	
3	
4	
5	

Q4 Tracer la courbe V_S en fonction de V_B (sur feuille ou à l'aide d'un tableur, ou ...).

Q5 D'après ces mesures, quelles valeurs de tension correspondent à un '1' logique ? à un '0' logique ?

↪ Rechercher sur internet la norme logique "TTL".

Q6 Vérifier que le composant utilisé suit bien cette norme de niveaux logiques.

↪ Rajouter sur le circuit la DEL et la résistance selon les figures 3.2 et 3.1. (Le sens de branchement de la diode peut être trouvé par une recherche sur internet...)

Q7 Dans quel(s) cas la DEL est-elle allumée ?

3 Une autre fonction logique

On souhaite remplacer la porte logique précédente par une porte logique du composant 7402.

↪ Rechercher sur internet la documentation de ce circuit. Repérer dans cette documentation en particulier le brochage du composant et sa table de vérité.

Q8 Combien de portes logiques sont présentes dans ce composant. Quels sont les numéros des broches utilisées pour les entrées ? Pour les sorties ? A quoi correspondent les broches 7 et 14 ?

↪ Modifier le circuit afin d'ajouter le composant 7402 en parallèle du précédent. Tester son fonctionnement.

4 (Si vous avez le temps.) Régime variable.

Vous allez remplacer l'alimentation continue présente sur l'entrée B par le générateur de fonction. La première étape est de régler le générateur de fonction.

↪ Régler, à l'aide des indications données au paragraphe suivant, le générateur de fonction afin qu'il délivre une tension qui évolue selon la courbe de la figure 3.3.

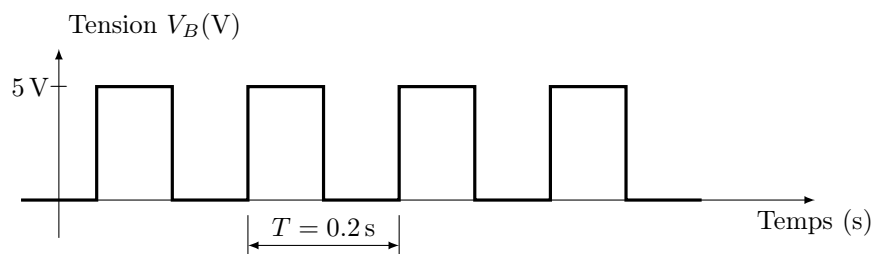


FIGURE 3.3 – Évolution attendue de la tension V_B , la période est notée T

↪ Vérifier le réglage en branchant directement le signal de sortie du générateur sur l'entrée de l'oscilloscope.



Réglage du générateur de fonctions Agilent 3320A

Réglage de la forme du signal Sélectionner la forme rectangulaire en appuyant sur **Square**

Réglage de la période Sélectionner **Period** et régler à l'aide du bouton la période à 0.2s.

Réglage de l'amplitude Sélectionner **LoLevel** (Pour *Low Level* : Niveau bas) et régler à 0V puis sélectionner **HiLevel** (Pour *High Level* : Niveau haut) et régler à 5V.

Observer la tension à l'oscilloscope Réglage de la gamme de temps Menu **Horizontal**, régler la base de temps sur le calibre 100ms/carreau.

Réglage de la gamme en amplitude Menu **Vertical**, régler le calibre vertical sur 2 Volt/carreau.

Récupérer une image de l'écran de l'oscilloscope Voir la notice de description du matériel disponible sur chaque table.

↔ Connecter le générateur de fonctions sur l'entrée *B* du circuit. Régler la tension dans un premier temps à 5V puis à 0V.

Q9 Décrire et expliquer les phénomènes observés.

Séance 4

PC2 - Chronogrammes logiques

Objectifs A l'issue de cette séance, vous serez capables d'utiliser les chronogrammes pour étudier le fonctionnement de circuits logiques. Cette séance est aussi l'occasion d'une introduction aux circuits logiques séquentiels.

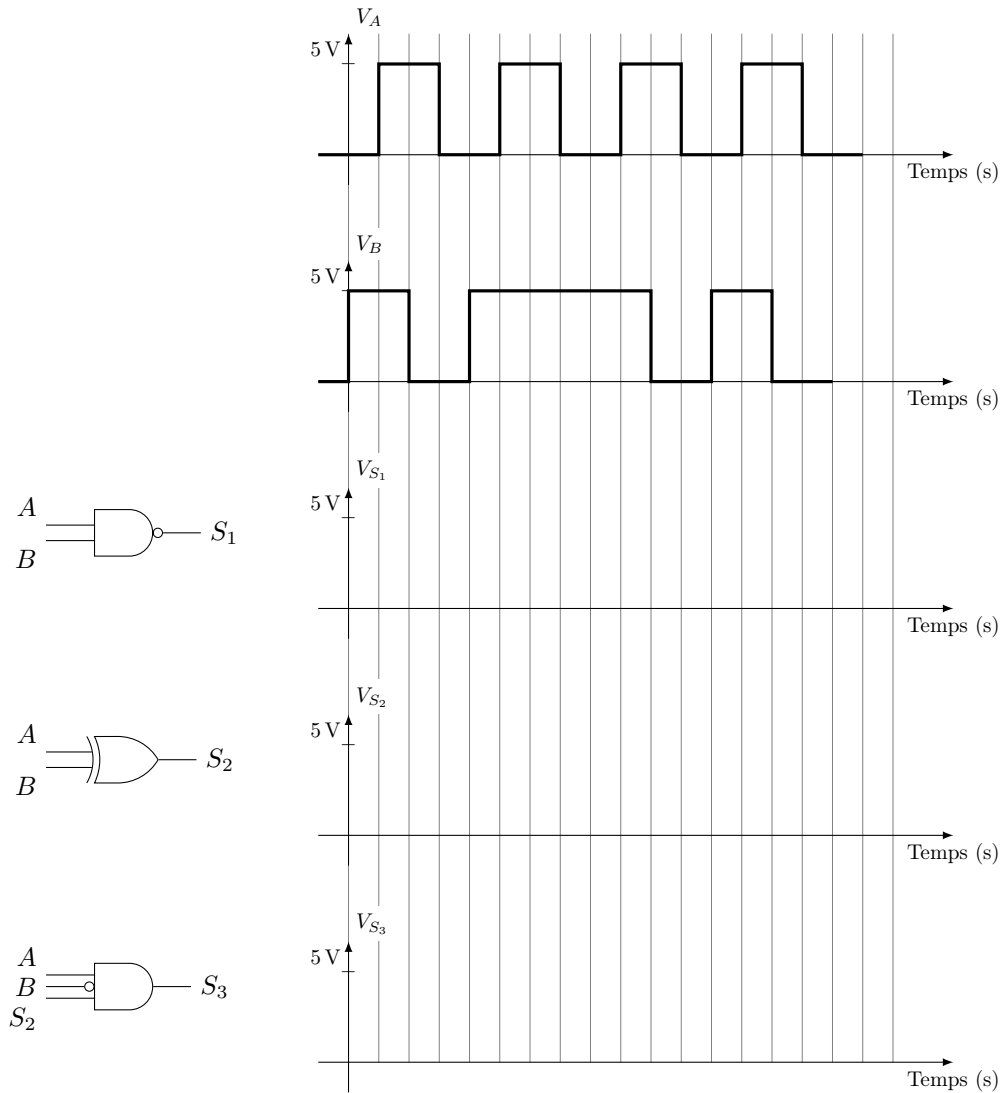
Sommaire

1	Chronogrammes de circuits logiques combinatoires	40
2	Introduction à la logique séquentielle	40
2.1	La bascule D	40
2.2	Chronogramme d'un registre à décalage	41

Le fonctionnement de l'électronique séquentielle ne peut être étudiée qu'en analysant l'évolution des différents signaux logiques au cours du temps. On trace pour cela des chronogrammes. Cet outil d'analyse est d'abord appliqué à des exemples de circuits combinatoires.

1 Chronogrammes de circuits logiques combinatoires

Q1 Compléter les chronogrammes suivants :



2 Introduction à la logique séquentielle

2.1 La bascule D

La bascule D est le composant de base de l'électronique séquentielle. Son symbole est celui de la figure 4.1 :

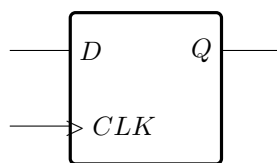


FIGURE 4.1 – Symbole d'une bascule D

Ce composant possède deux entrées dont une entrée un peu particulière : l'entrée d'horloge (*clock*, *CLK*) elle est repérée par un triangle.

La sortie *Q* n'est modifiée qu'aux seuls instants de transition de l'horloge de '0' à '1'. Un chronogramme typique qui décrit son fonctionnement est donné figure 4.2 :

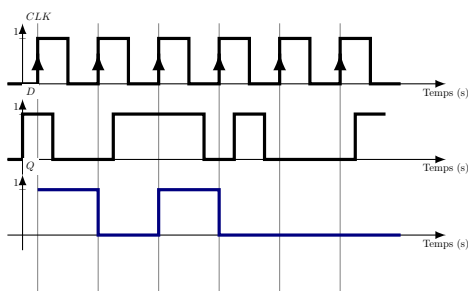


FIGURE 4.2 – Chronogrammes typiques d'une bascule D

On peut décrire simplement le fonctionnement d'une telle bascule : la sortie *Q* recopie l'entrée *D* sur chaque front montant de l'horloge.

Sa table de vérité peut s'écrire :

<i>CLK</i>	<i>D</i>	<i>Q</i>
↑	0	0
↑	1	1
↓	X	Q
1	X	Q
0	X	Q

Cette table de vérité (qui semble un peu compliquée...) signifie en fait que le signal de sortie *Q* ne peut être modifié que sur un front montant d'horloge (symbolisé par ↑).

2.2 Chronogramme d'un registre à décalage

Une première application de la bascule D est...la guirlande de Noël. Voici le schéma d'un registre à décalage 4 bits.

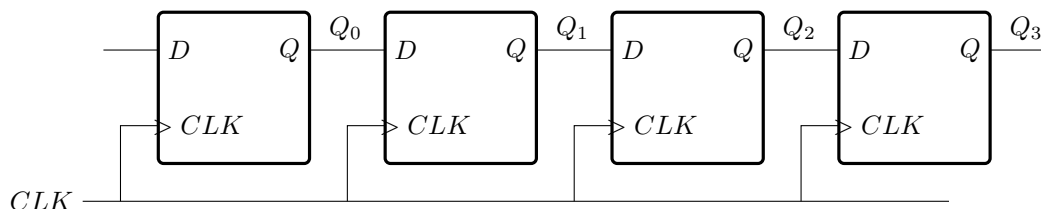


FIGURE 4.3 – Registre à décalage 4 bits

Q2 Compléter le chronogramme suivant :

