$\boxed{\text{codage}}$

● ***Exercice 6:*** Le circuit[3] de la figure 1 permet de générer les tensions v_r, v_v et v_b nécessaires *pour piloter un écran video couleur de type VGA à l'aide d'un circuit numérique. Les trois variables logiques r_2, r_1, r_0 sont utilisées pour générer la tension v_r indiquant le niveau de rouge d'un point de l'écran, les trois variables logiques v_2, v_1, v_0 sont utilisées pour générer la tension v_v indiquant le niveau de vert et les deux variables logiques b_1, b_0 sont utilisées pour générer la tension v_b indiquant le niveau de bleu. Combien y a t-il de niveaux différents de rouge, de vert et de bleu ? Combien y a t-il de couleurs différentes ?*

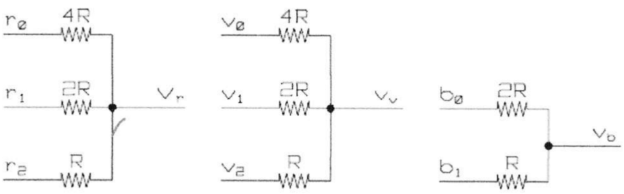


Figure 1: circuit étudié dans l'exercice 6.

[3]Voir *Digilent Basys Board Reference Manual*, `http://www.digilentint.com/basys`

$r_2\, r_1\, r_\phi \rightarrow$ 3 bits $\quad 2^3 = 8$ cas

$v_2\, v_1\, v_\phi \rightarrow$ 3 bits $\quad 2^3 = 8$ cas

$b_1\, b_0 \rightarrow$ 2 bits $\quad 2^2 = 4$ cas

$\rightarrow$ au global 8 bits $\rightarrow 2^8 = 256$ couleurs différentes
<u>par pixel</u>

cf wikipedia VGA mode graphique standard 320 x 200 x 8 ou
(MCGA pour intime...) mode 13h

● ***Exercice 7:*** *Le circuit intégré LM74 de* National Semiconductor[4] *contient un capteur linéaire de température relié à un convertisseur analogique-numérique. Ceci lui permet de fournir à un micro-contrôleur une mesure de température codée en binaire en notation en complément à 2 sur 13 bits. Le pas de quantification est de 0.0625°C. Ainsi, une température de 0°C fournira le nombre 0, tandis qu'une température de 0.5°C fournira le nombre 8.*

1. *Quels sont les codes binaires délivrés par ce capteur pour des températures de +25°C, +150°C, −25°C et −55°C ?*
2. *Quelles sont les températures minimales et maximales qui peuvent être codées de cette façon ? L'étendue de mesure de ce capteur va de −55°C à +150°C. Justifier la nécessité de coder les résultats de mesure sur 13 bits.*

[4]Voir la notice technique "LM74 SPI/Microwire 12-bit plus sign temperature sensor" de *National Semiconductor*, mai 2005.

notation $\bar{2}$ sur 13 bits $\quad 2^{12} = 4096$

$p = 0{,}0625\,°C \longrightarrow z = \frac{\theta}{p} = 16 \times \theta (\text{en } °C)$

θ °C	+25	+150	−25	−55
z	400	2400	−400	−880

$\boxed{+25°C} \longrightarrow z \overset{16\times 25}{=} 400 = 16^2 \times 1 + 16 \times 9 + 0 = (190)_{16} = (\overset{1}{0001}\ \overset{9}{1001}\ \overset{0}{0000})_2$

$\longrightarrow (0\ \underbrace{0001\ 1001\ 0000}_{13 \text{ bits}})_{\bar{2}/13}$

>0

$\boxed{+150°C} \longrightarrow z = 16 \times 150 = 2400 = (960)_{16} = (1001\ 0110\ 0000)_2$

$\longrightarrow (0\ \underbrace{1001\ 0110\ 0000}_{13 \text{ bits}})_{\bar{2}/13}$

>0

$\boxed{-25°C} \longrightarrow z = -400 < 0$

on utilise la même representation que celle de

$-400 + 4096 = +(3696)_{10} = (E70)_{16}$

$= (1110\ 0111\ 0000)_2$

$\longrightarrow (1\ 1110\ 0111\ 0000)_{\bar{2}/13}$

<0

on alors méthode $\bar{2}$ + 1

400 → 0 0001 1001 0000

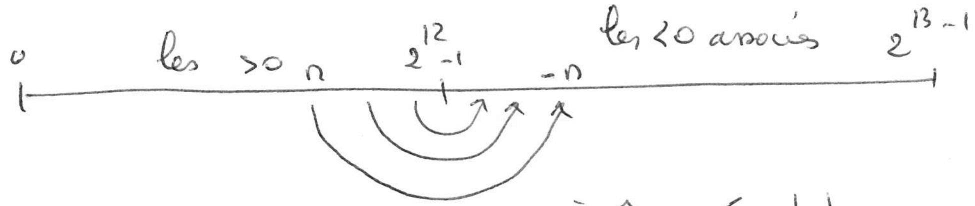
$\bar{2}$ 1 1110 0110 1111

+1 1

idem. $(1\ 1110\ 0111\ 0000)_{\bar{2}/13}$

sur 13 bits 2^{12} valeurs > 0

2^{12} —— < 0



-n à m̂ représentation que

$(2^{12}-1) - n = 4096 - n$

$\boxed{-55°C}$ → $z = -16 \times 55 = -880 < 0$

$(880)_{10} = (370)_{16} = (\underbrace{0011}\ \underbrace{0111}\ \underbrace{0000})_2$

$\left\{ \begin{array}{l} \frac{880}{16} = 55 \times 16 + 0 \\ \frac{55}{16} = 3,\ldots \quad 55 = 3 \times 16 + 7 \end{array} \right.$

$\bar{2}$ 1100 1000 1111

+1

$(1110010010000)_{\bar{2}/13}$

< 0

$-880 + 4096 = (3216)_{10} = (C\,9\,\emptyset)_{16} = 12 \times 16^2 + 9 \times 16 + 0$

$(\underset{\text{signe} < 0}{1}\ \underbrace{1100}_{C}\ \underbrace{1001}_{9}\ \underbrace{0000}_{\emptyset})_{\bar{2}/13}$

● ***Exercice 8:*** *Donner les codes DCB des nombres $(617)_{10}$ et $(9241)_{10}$. Compter le nombre de bits nécessaires pour le codage de ces nombres et comparer ces valeurs aux nombres de bits de leurs représentations binaires.*

$$(617)_{10} = (269)_{16} = (\overbrace{0010}^{2}\,\overbrace{0110}^{6}\,\overbrace{1001}^{9})_2 \longrightarrow \text{10 bits indispensables}$$

$$= (\overbrace{0110}^{6}\,\overbrace{0001}^{1}\,\overbrace{0111}^{7})_{DCB} \longrightarrow \text{11 bits indispensables mais 12 pour le format 3 chiffres}$$

$$(9241)_{10} = (2419)_{16} = (\overbrace{0010}^{2}\,\overbrace{0100}^{4}\,\overbrace{0001}^{1}\,\overbrace{1001}^{9})_2 \longrightarrow \text{14 chiffres}$$

$$= (\overbrace{1001}^{9}\,\overbrace{0010}^{2}\,\overbrace{0100}^{4}\,\overbrace{0001}^{1})_{DCB} \longrightarrow \text{16 chiffres}$$

pour coder $(617)_{10}$, il faut $E\left[\frac{\log(617)}{\log(2)}\right] + 1 = 10$ bits $< 3 \times 4 = 12$

$(9241)_{10}$ $\quad E\left[\frac{\log 9241}{\log 2}\right] + 1 = 14$ bits $< 4 \times 4 = 16$

le code decimal codé binaire DCB occupe plus de place codage

● ***Exercice 9:*** *Compléter toutes les cases du tableau de la figure 2.*

représentation (ou code)			
décimale	binaire	hexadécimale	DCB
15	1111 (2^4-1)	F	0001 0101
22	10110	$(16)_{16}$ 16+6	0010 0010
26 16+10	16 8 2 11010 D	1A	0010 0110
29	16 8 4 1 11101	1 D 16+13	0010 1001

Figure 2: Tableau de comparaison des différentes représentations de nombres entiers naturels.