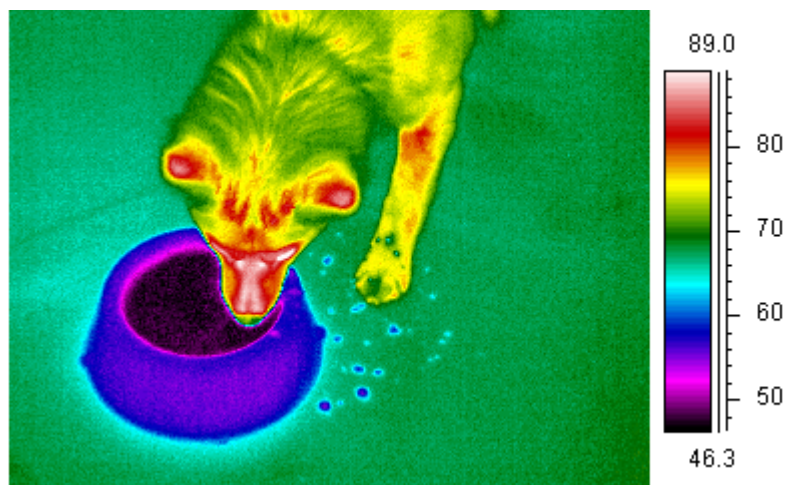


BERTRAND Mathieu
CEPPI Solange
GABIN Manon

Olympiades de la physique 2008/2009

Réalisation d'une caméra thermique



Encadrement : M.BOTTOS
Lycée : C.Baudelaire, Cran-Gevrier (74)

SOMMAIRE

	Pages
Introduction	3
Problématique	3
I. Recherches théoriques.....	3
II. Réalisation du montage	3
1°/ Avec quoi capter la température ?	3
2°/ Schéma fonctionnel pour un capteur (thermomètre)	4
3°/ La matrice	4
4°/ Fabrication de la matrice	5
5°/ Idées pour le traitement des données	6
6°/ Utilisation d'un montage différentiel	6
7°/ Montage amplificateur actuel.....	8
8°/ Montage général actuel.....	10
III. Réalisation de l'image	12
1°/ Nos mesures	12
2°/ A l'aide du PIC.....	12
Conclusion et remerciements.....	14

Introduction :

Nous sommes trois élèves actuellement en terminale scientifique au lycée Charles Baudelaire (Cran-Gevrier, Haute-Savoie). Dans le cadre des Olympiades de la physique, nous nous sommes retrouvés régulièrement depuis la fin de notre année de seconde, encadrés par M.Bottos, professeur de physique-chimie dans notre lycée. Nous avons ainsi élaboré notre projet.

Pour notre projet, nous souhaitons mêler physique et sciences naturelles (plus précisément écologie). Nous voulions également réaliser nous-mêmes un objet nécessitant une partie électronique et une autre informatique. Nous nous sommes donc tournés vers l'étude des rayonnements infrarouges (directement liés à la température) et l'élaboration d'une caméra thermique.

Problématique :

Notre objectif est de réaliser une caméra thermique. Tout d'abord, nous avons étudié le lien entre rayonnement infrarouge et température pour ensuite en déduire le montage de la caméra. Enfin, nous avons réfléchi sur l'exploitation par informatique des données pour obtenir une représentation de la température sur un écran.

II. Recherches théoriques :

Nous avons donc commencé par rechercher les lois qui relient la température d'un corps et son rayonnement.

La loi de Wien nous a permis de relier la température et la longueur d'onde correspondante par la relation : $\lambda_{\max}=C/T$ avec $C=2800 \mu.K^{-1}$. Les longueurs d'onde correspondantes à toutes les températures que nous avons la possibilité d'étudier se situent dans les infrarouges (supérieures à 800 nm).

Nous nous sommes aussi intéressés à un modèle théorique et parfait de corps : le corps noir. Il serait capable d'absorber intégralement tout rayonnement quelle que soit sa fréquence sans en réémettre. Il sert à l'élaboration de certaines lois physiques.

Par exemple, la loi de Stephan-Boltzmann qui énonce que la puissance rayonnée par unité de surface M d'un corps noir est proportionnelle à la puissance quatrième de la température absolue : $M=\sigma T^4$ avec M en $W.m^{-2}$. Nous avons étudié cette loi qui finalement ne nous apporte rien.

IV. Réalisation du montage :

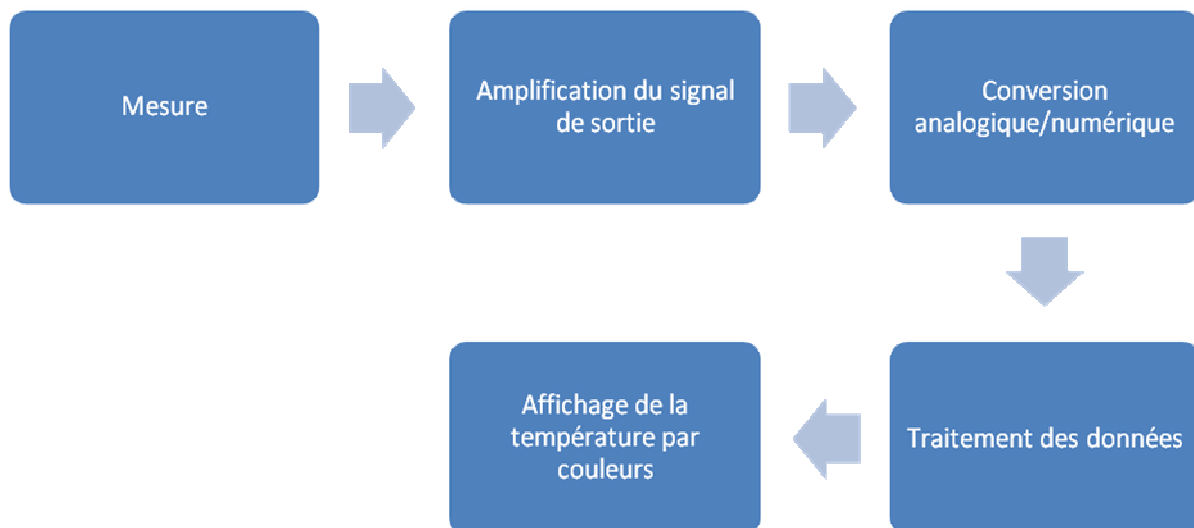
1°/ Avec quoi capter la température ?

Pour les capteurs, nous pensions tout d'abord utiliser des photodiodes, mais nous nous sommes aperçus que la longueur d'onde ne correspondait pas du tout aux

zones d'infrarouges nécessaires pour les températures que nous souhaitons étudier. Nous avons alors étudié les différents types de capteurs disponibles dans les commerces standards et à notre portée financière, comme les cellules CCD ou les thermopiles. Mais M.Bottos nous a plutôt orienté vers les capteurs pyroélectriques, dont la tension de sortie dépend uniquement du rayonnement infrarouge capté, dépendant lui-même de la température (loi de Wien). De plus, ils sont plus accessibles financièrement que les thermopiles. Nous en possédions plusieurs dont les caractéristiques différaient légèrement et nous avons gardé celui qui d'après nos tests correspondait le mieux à ce que nous recherchions. Grâce à la loi de Wien, qui énonce que $\lambda_{\max} = C/T$ où C est une constante égale à $2800 \mu\text{m}\cdot\text{K}^{-1}$ et T la température en Kelvin, nous avons calculé que les températures comprises entre -30 et 130°C correspondaient à des longueurs d'ondes de 7 à $11 \mu\text{m}$. Ceci a donc orienté notre choix vers le capteur pyroélectrique RE 200B, captant les longueurs d'ondes comprises entre 7 et $14 \mu\text{m}$ (≈ -70 à 130°C). Cependant, il s'est récemment trouvé en rupture de stock, ce qui nous a conduits à choisir un équivalent : le capteur IRA-E700STO.

2°/ Schéma fonctionnel pour un capteur (thermomètre) :

Ce capteur est de type capacitif, sa tension varie en présence d'une source de chaleur. Il est néanmoins nécessaire de l'amplifier afin de pouvoir observer des variations (ce qui nous posera le plus de soucis !). Nous avons alors réfléchi aux différentes parties d'un montage avec ce capteur :

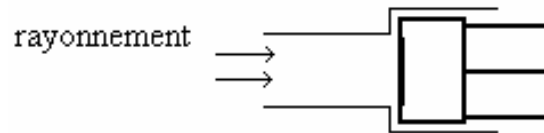


3°/ La matrice :

Après avoir choisi notre capteur, nous avons cherché à savoir comment l'utiliser de façon optimale. Pour cela nous avons démonté un vieux thermomètre infrarouge auriculaire et remarqué que le capteur de ce thermomètre était entouré d'un tube de

métal. Ce montage évite au capteur de relever des rayonnements arrivant de sources non voulues : il empêche les rayonnements parasites.

Schéma du tube du thermomètre en coupe :

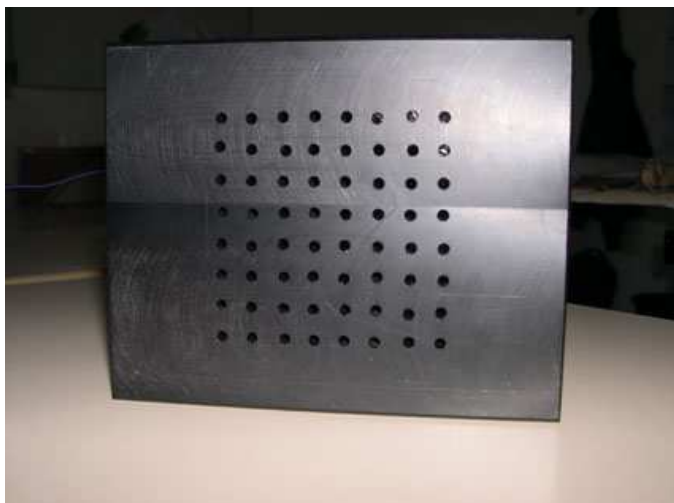


Il comportait également un cache en métal s'ouvrant pour un temps défini et permettant de limiter le temps d'exposition du capteur. Nous avons décidé de nous inspirer de ce dispositif. Après plusieurs tests, nous en avons déduit que le meilleur temps d'exposition serait de deux secondes environ avec une source placée à approximativement vingt centimètres.

Nous voulions réaliser une matrice comportant cent capteurs indépendants pour éviter de devoir réaliser un balayage avec un seul capteur, la partie mécanique étant difficile à réaliser à la fois pour le balayage et pour la réalisation des miroirs spéciaux. En effet, nous avons appris en visitant les laboratoires de physique de l'université Joseph Fourier à Grenoble que ceux-ci devraient être en or ! Cependant nous nous sommes aperçus que la partie électronique à réaliser pour cent capteurs indépendants serait trop importante et trop chère.

Notre professeur a alors pensé à utiliser des multiplexeurs qui permettraient de réduire le problème de l'électronique. En effet, les capteurs seraient regroupés en lignes, reliée chacune à un multiplexeur. Celui-ci fonctionne comme un aiguillage de l'information : il sélectionne successivement chacun des capteurs de la ligne et retransmet le signal à la sortie unique. Nous avons alors préféré choisir des multiplexeurs à huit entrées et, par souci de simplicité, nous avons donc réduit la matrice à soixante-quatre capteurs. En résumé, les huit capteurs sont branchés sur l'entrée du multiplexeur, en sortie de celui-ci se trouve le montage amplificateur. Chaque capteur est amplifié à tour de rôle et le signal envoyé pour le traitement, ceci sur huit lignes.

4°/ Fabrication de la matrice :



L'oncle de Mathieu nous a réalisé le support de la matrice ci-contre.

Chaque capteur vient s'encastrent dans un trou. Le support a une épaisseur de quelques centimètres et joue le rôle de tube de métal. La matière utilisée est du delrin particulièrement adaptée car elle ne conduit pas la chaleur. Nous la recouvrons cependant d'un papier d'aluminium car nous ne savons pas

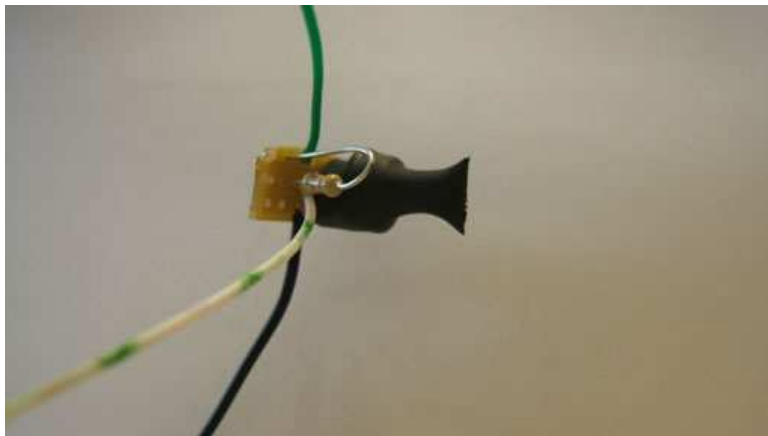
si sa couleur noire serait susceptible d'absorber les rayonnements pour les redistribuer progressivement.

5°/ Idées pour le traitement des données :

Le traitement des données se compose d'une conversion analogique/numérique du signal puis de la réalisation d'un programme permettant l'affichage de la source en couleurs, ces dernières dépendant de la température. Nous avons décidé de réaliser ces deux opérations à l'aide d'un microcontrôleur (PIC), qui semble être la solution la plus simple et la plus actuelle. Celui-ci est capable de recevoir un signal analogique, de le convertir en numérique et de l'exploiter en suivant un programme que nous avons réalisé.

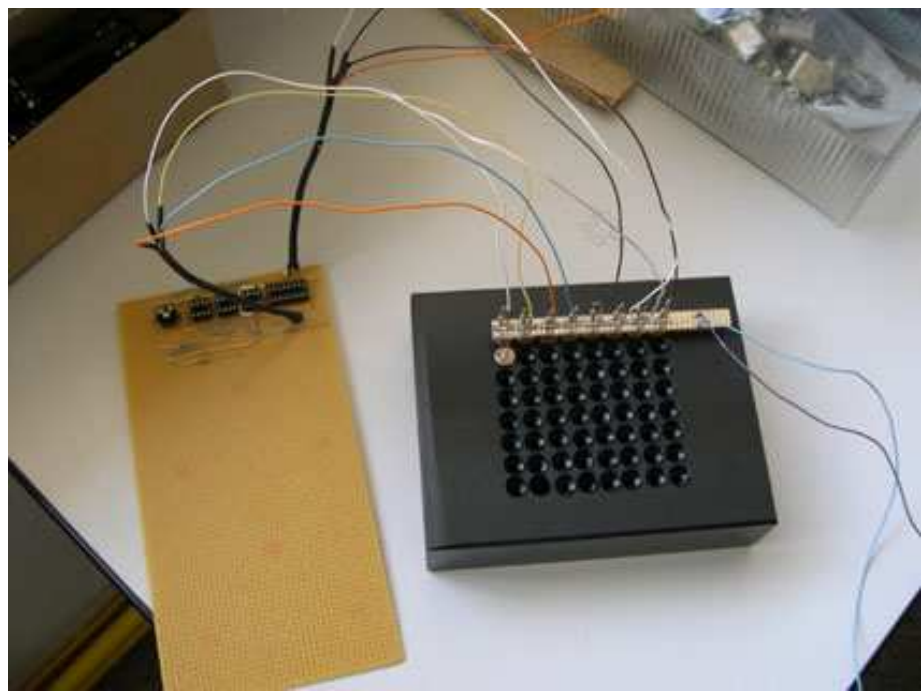
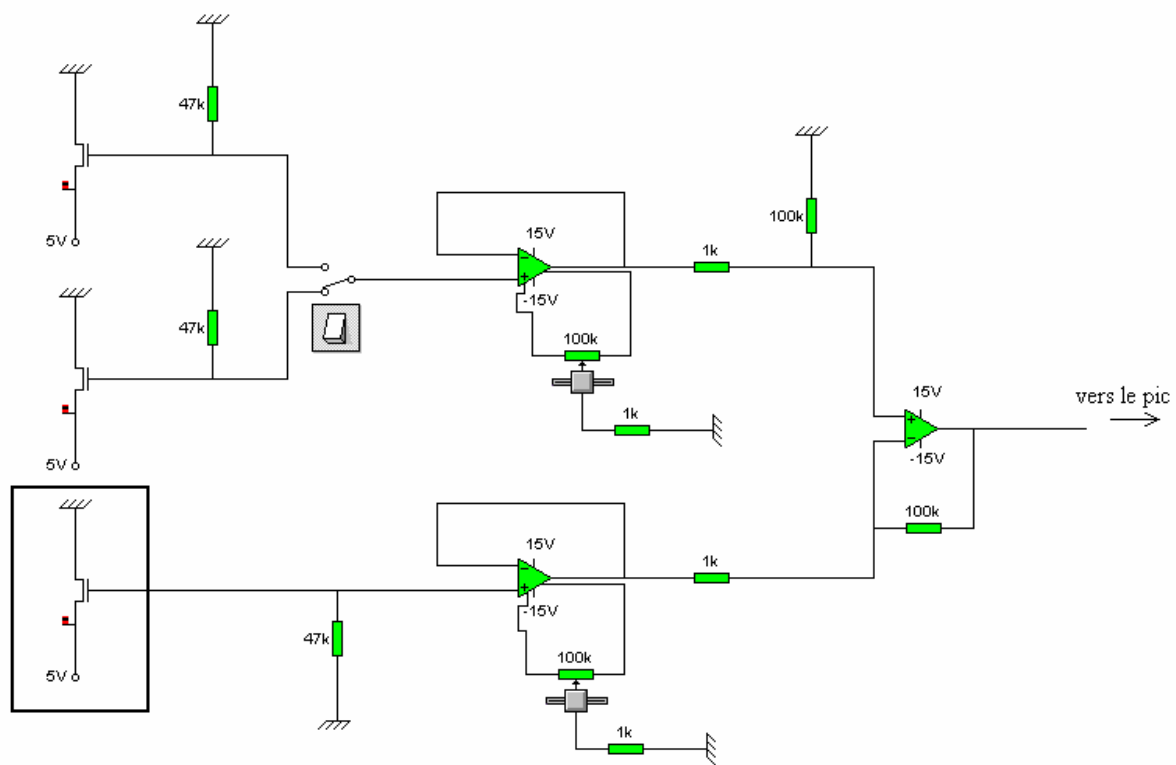
6°/ Utilisation d'un montage différentiel :

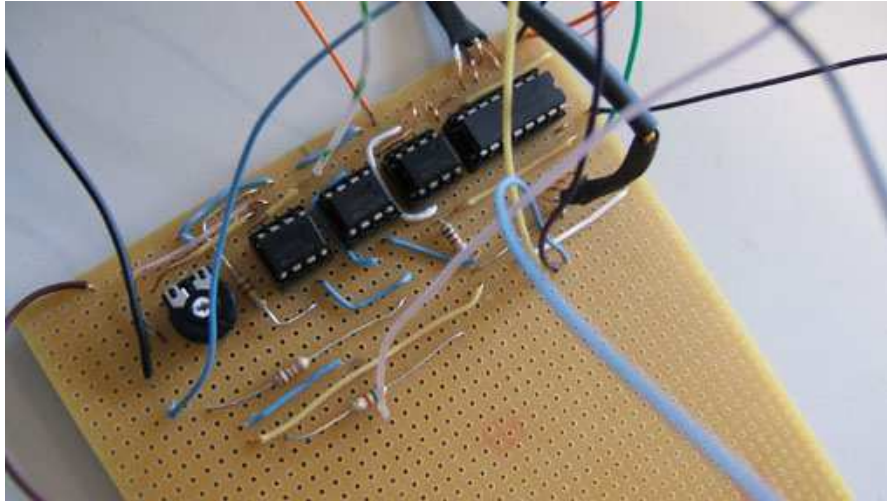
Notre première observation a montré que le signal n'était pas centré sur zéro. Nous avons alors décidé, pour avoir le maximum de variations, de recentrer ce signal sur zéro par l'intermédiaire d'un montage différentiel particulier. De plus, le signal du capteur dépendant à la fois de la température de la source et de la température ambiante, il est difficile de réaliser un programme pour le traitement. M.Bottos nous a alors suggéré d'employer un montage différentiel. Celui-ci est réalisé à l'aide d'un amplificateur opérationnel dont l'une des entrées est reliée à notre capteur et l'autre entrée à un capteur masqué. Le capteur masqué est donc à température ambiante. Le montage différentiel fait la différence entre le signal reçu du capteur et celui du capteur masqué et nous permet de n'obtenir que le signal correspondant à la source. Derrière chacun des deux capteurs nous avons aussi ajouté un montage suiveur (montage particulier d'un amplificateur) qui permet de garder une tension constante quelle que soit l'intensité tirée, c'est-à-dire de ne pas perdre notre signal.



Le capteur masqué (une gaine thermorétractable le cache).

Nous avons alors réalisé le montage pour une ligne de capteurs ci-après. Les transistors simulent les capteurs pyroélectriques, celui qui est encadré est le capteur masqué. Deux capteurs de la ligne sont représentés seulement mais il y en a huit normalement. Les réglages des offsets sont réalisés grâce aux potentiomètres.



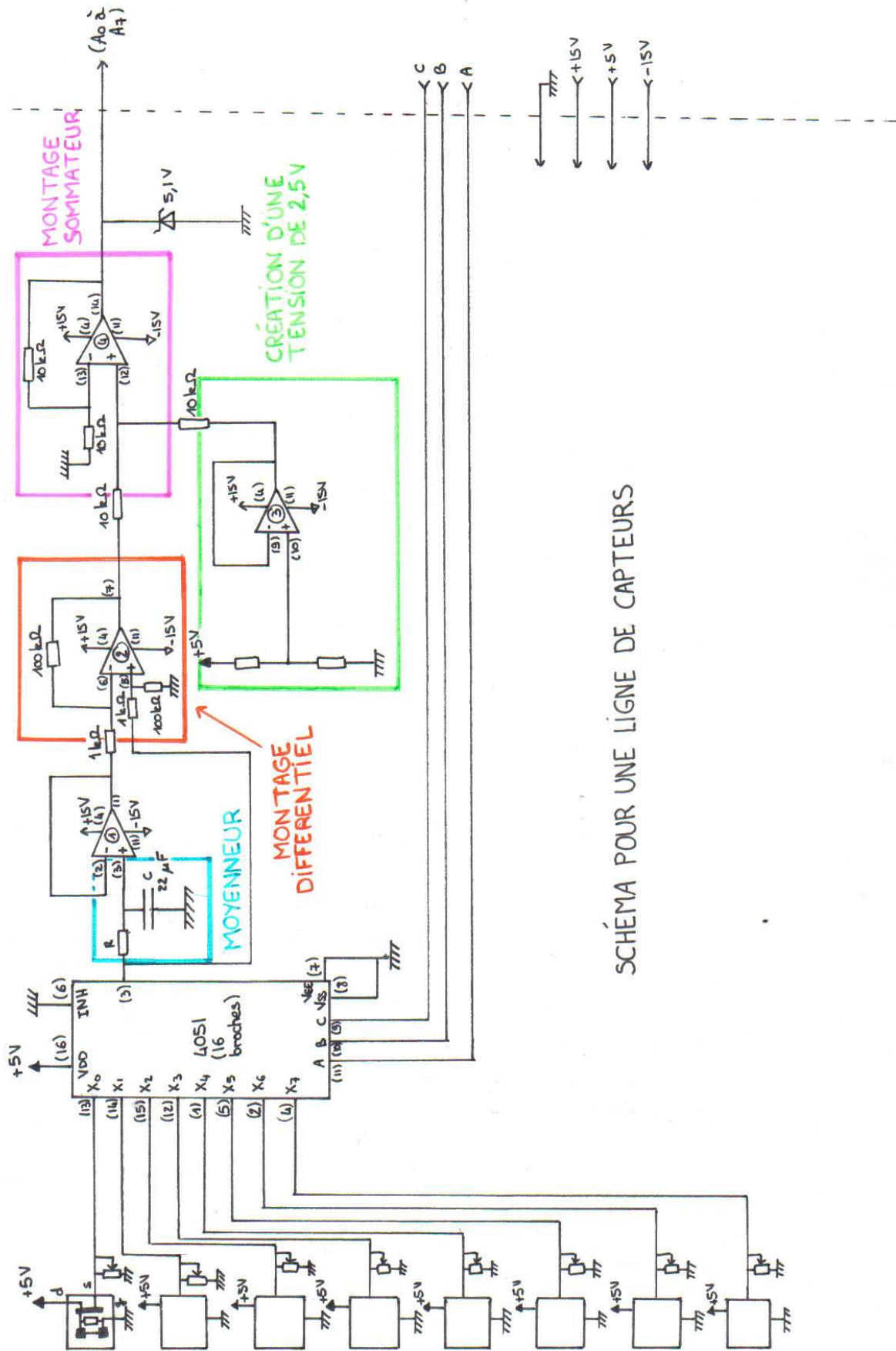


Par la suite, nous avons étudié de plus près les spécifications du constructeur pour l'utilisation du PIC. Nous nous sommes alors rendus compte que ce dernier n'acceptait que des tensions d'entrées comprises entre -0.3V et 5.3V. Or le réglage des offsets et le montage différentiel avaient ramené la tension sur 0V et, lors de l'observation d'une source de chaleur, la tension prenait parfois des valeurs négatives. Nous avons alors enlevé les réglages des offsets et cherché à modifier l'alimentation des amplificateurs car si nous restions en +15V/-15V nous aurions dépassé les tensions requises pour le PIC. Récemment, nous sommes revenus à un montage différentiel légèrement différent suivi d'un montage sommateur.

7°/ Montage amplificateur actuel :

Nous avons donc réalisé un montage différentiel à la sortie de chaque multiplexeur et remplacé le capteur caché par un moyennneur. En effet, les capteurs ont tous des tensions au repos un peu différentes ; un moyennneur permet de faire la moyenne des tensions au repos de tous les capteurs de la ligne. Cette méthode est plus juste que celle du capteur caché car celui-ci pouvait avoir une tension très différente de celles des autres capteurs et fausser les mesures. Pour éviter les incertitudes de mesures au maximum, nous avons aussi décidé d'étalonner chaque capteur car leurs tensions de sortie ne sont malheureusement pas toujours identiques. Nous avons donc remplacé la résistance à la sortie des capteurs par des potentiomètres que l'on ajuste dans le but d'uniformiser les tensions des capteurs.

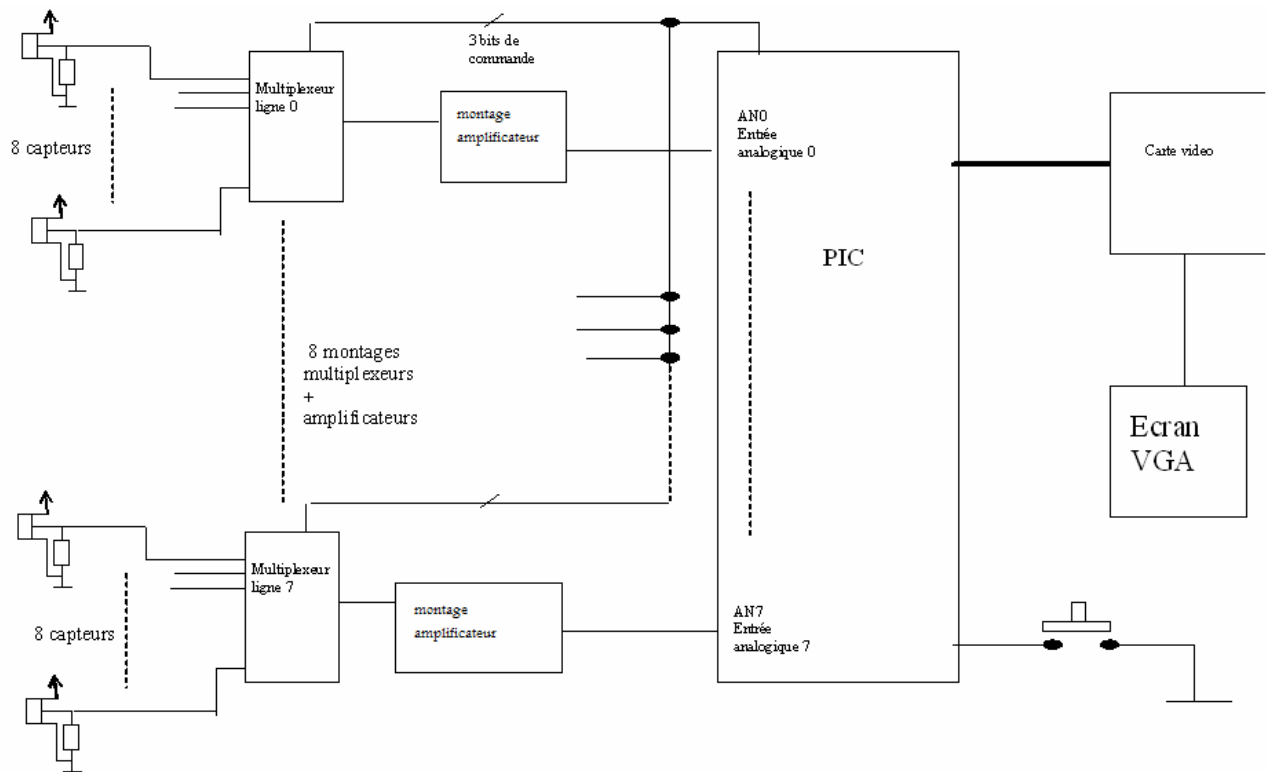
Le montage différentiel est suivi d'un montage sommateur qui additionne 2,5V à la tension de sortie et assure une tension toujours positive. Après, une diode Zener permet une sécurité supplémentaire en empêchant une tension supérieure à 5V d'entrer dans le PIC, cela n'est cependant pas censé arriver.



SCHEMA POUR UNE LIGNE DE CAPTEURS

8°/ Montage général actuel :

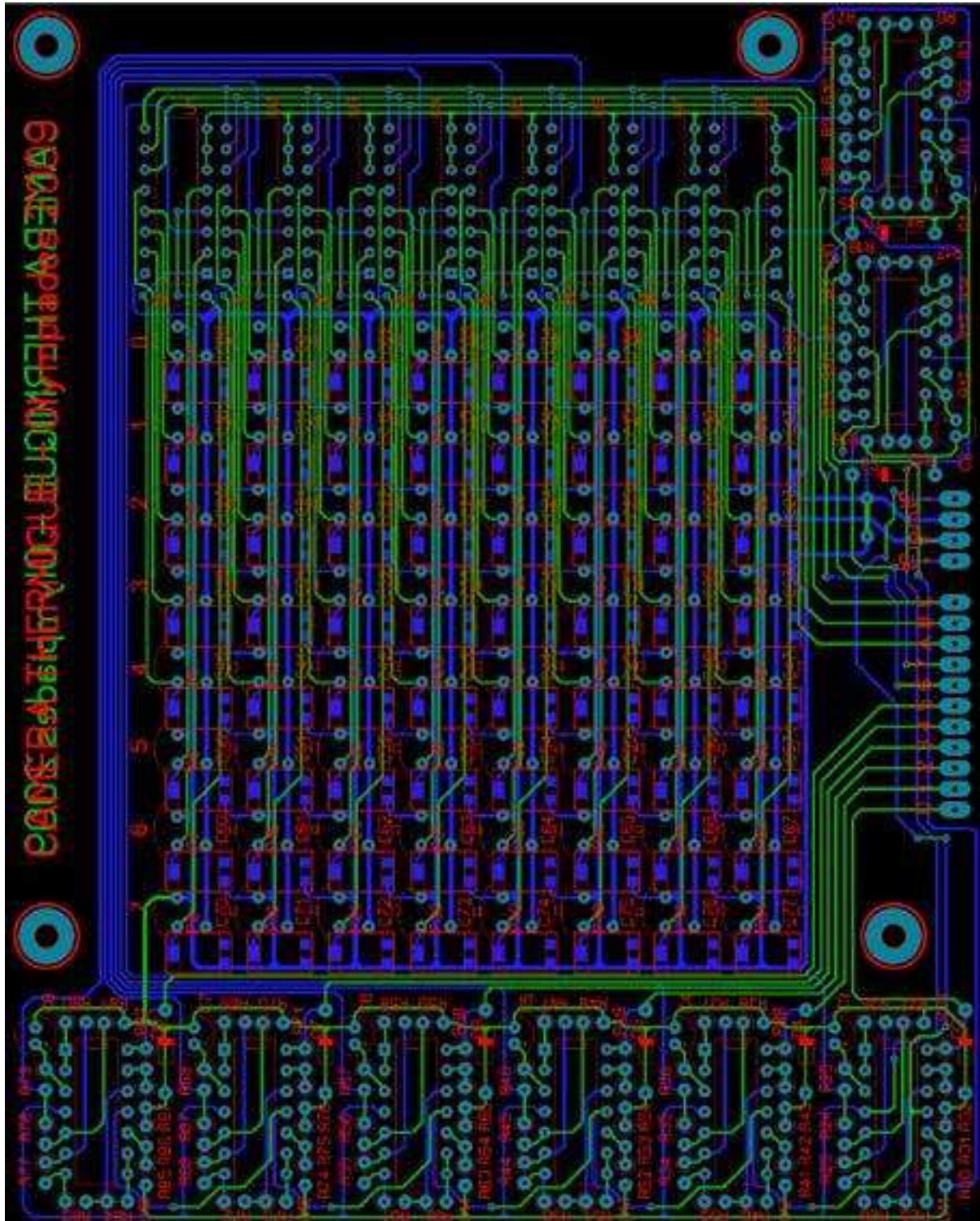
La caméra se composera de huit lignes de huit capteurs, chaque ligne sera reliée à un multiplexeur, lui-même relié à la partie amplificatrice décrite juste au-dessus. Le signal, une fois amplifié, sera traité par le PIC puis affiché grâce à une carte vidéo et un écran. L'interrupteur est le bouton d'acquisition.



Dans l'objectif de réduire la place prise par la partie électronique, de mieux l'organiser et d'éviter les faux contacts comme nous avons pu le remarquer lors de la fabrication du premier prototype. La multiplication des contacts électriques nous a poussé à réaliser un circuit imprimé. Celui-ci devra être un circuit double face car il y a de trop nombreuses connexions pour pouvoir le fabriquer en simple face. En effet, dans le cas contraire, il serait encore trop volumineux. Actuellement, nous avons réalisé (avec le logiciel sprint layout 5.0) le typon et demandé à une entreprise de graver le circuit. Nous ne pouvons pas le faire nous-mêmes car nous ne possédons pas les outils nécessaires à sa fabrication.

Le typon (la nomenclature, les platines vue de dessus et dessous sont jointes à la fin du dossier en annexe) :

Vue générale des connexions et de l'implantation des composants:
Cette vue permet de voir que les connexions sont nombreuses.



V. Réalisation de l'image :

1°/ Nos mesures :

Avant de réaliser l'image sur ordinateur nous avons déjà étudié la tension de sortie des capteurs à l'aide d'une interface et du logiciel correspondant (Logger Pro).

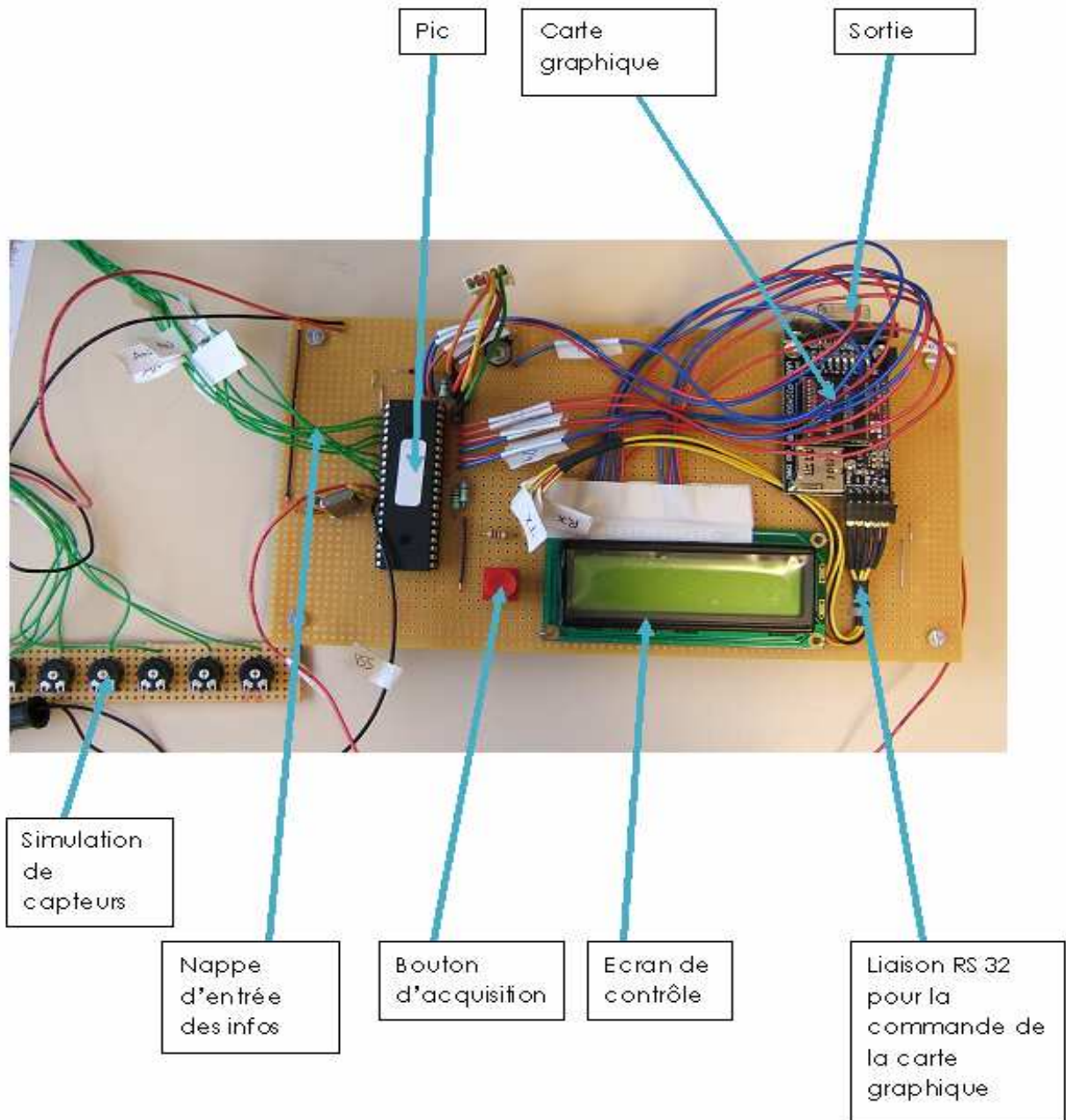
Nous avons commencé par observer les mesures obtenues avec le montage à un seul capteur (thermomètre). Nous avons alors remarqué que la tension variait comme la température : elle augmentait quand la température montait et vice versa. Grâce au réglage de l'offset, nous avons pu obtenir des mesures centrées sur zéro, même s'il fallait plusieurs secondes pour cela après chaque mesure.

Après la réalisation du premier montage différentiel, nous avons observé, sur Logger pro, les variations de tension. Celles-ci ce sont alors révélées inexploitable car l'amplificateur saturait, ce qui se traduisait à l'écran par un bruit très fort et une tension montant toujours au maximum. Nous avons ensuite modifié la place de l'offset mais n'avons pas encore refait de mesures car nous avons changé de montage.

Pressé par le temps, nous n'avons pas réalisé de mesures à l'aide de Logger pro avec notre dernier montage (montage différentiel et sommateur).

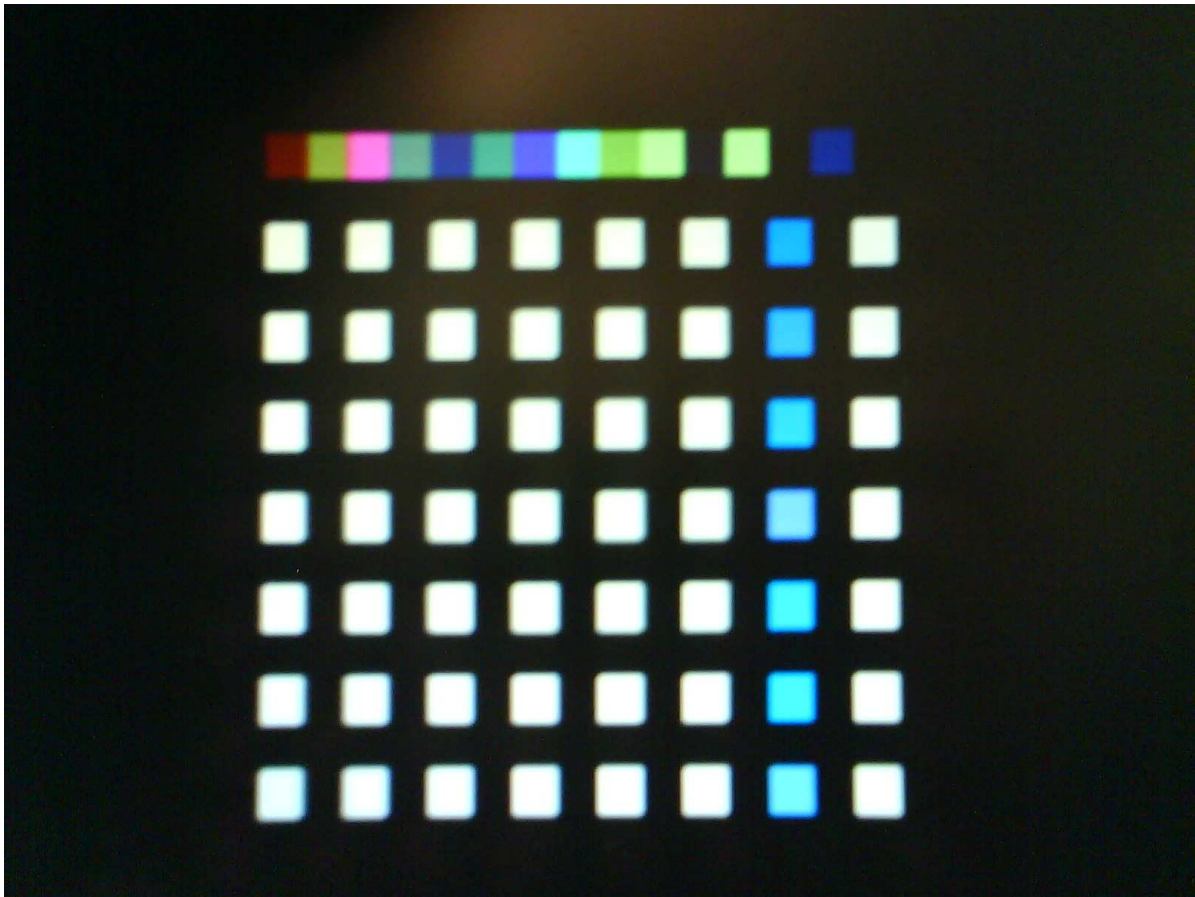
2°/ A l'aide du PIC :

Au début, nous avons simulé notre caméra à l'aide de potentiomètres pour pouvoir comprendre le fonctionnement du PIC et figoler le programme qui le contrôle. Aujourd'hui, nous sommes en train de relier la partie mesure à la partie traitement. Le montage de la partie traitement est celui de la photo ci-dessous. La plaque pour le montage est composée d'un écran, d'une puce programmable (PIC) et d'une carte vidéo. Le PIC nous sert à commander les multiplexeurs, gérer la carte graphique et à afficher les résultats que nous obtenons. Nous avons dû programmer la puce. Pour cela nous avons utilisé un logiciel qui permet d'écrire en basic et ensuite de convertir le basic en hexadécimal, seul langage qu'est capable de lire un PIC. (Le programme en basic est joint a la fin du dossier)



Grâce au programme, le PIC commande la voie à sélectionner pour le multiplexeur, stocke la valeur obtenue, l'affiche sur l'écran de la plaque et le convertit en couleurs et, enfin, l'affiche sur l'écran relié à la carte graphique. Auparavant, l'affichage était de la taille d'un écran d'ordinateur alors que la matrice est beaucoup plus petite. Les pixels de couleurs étaient donc énormes. Nous avons donc tout d'abord modifié ce problème. L'affichage est maintenant de la taille de la matrice. Un pixel correspond à un capteur, il y a donc des « noirs » entre les pixels. Nous avons résolu ce deuxième problème en interpolant les couleurs que se situeraient dans les « noirs » par le calcul de la moyenne des couleurs adjacentes. En ce moment, nous définissons une couleur pour chaque fourchette de tension de sortie.

Aperçu de l'affichage (seule la première ligne a été interpolée) :



Conclusion et remerciements :

Ce projet nous a demandé beaucoup de travail et réflexion dans divers domaines : électronique pour la réalisation du montage, programmation pour le contrôle du pic,... mais nous a apporté à tous. Il est en phase de finalisation et nous espérons ne pas rencontrer de nouvelles difficultés.

Nous tenons à remercier les personnes qui nous ont apporté leur aide à la réalisation de ce projet.

Tout d'abord, M.Bottos, notre professeur, qui nous a soutenus dans nos recherches et nos choix.

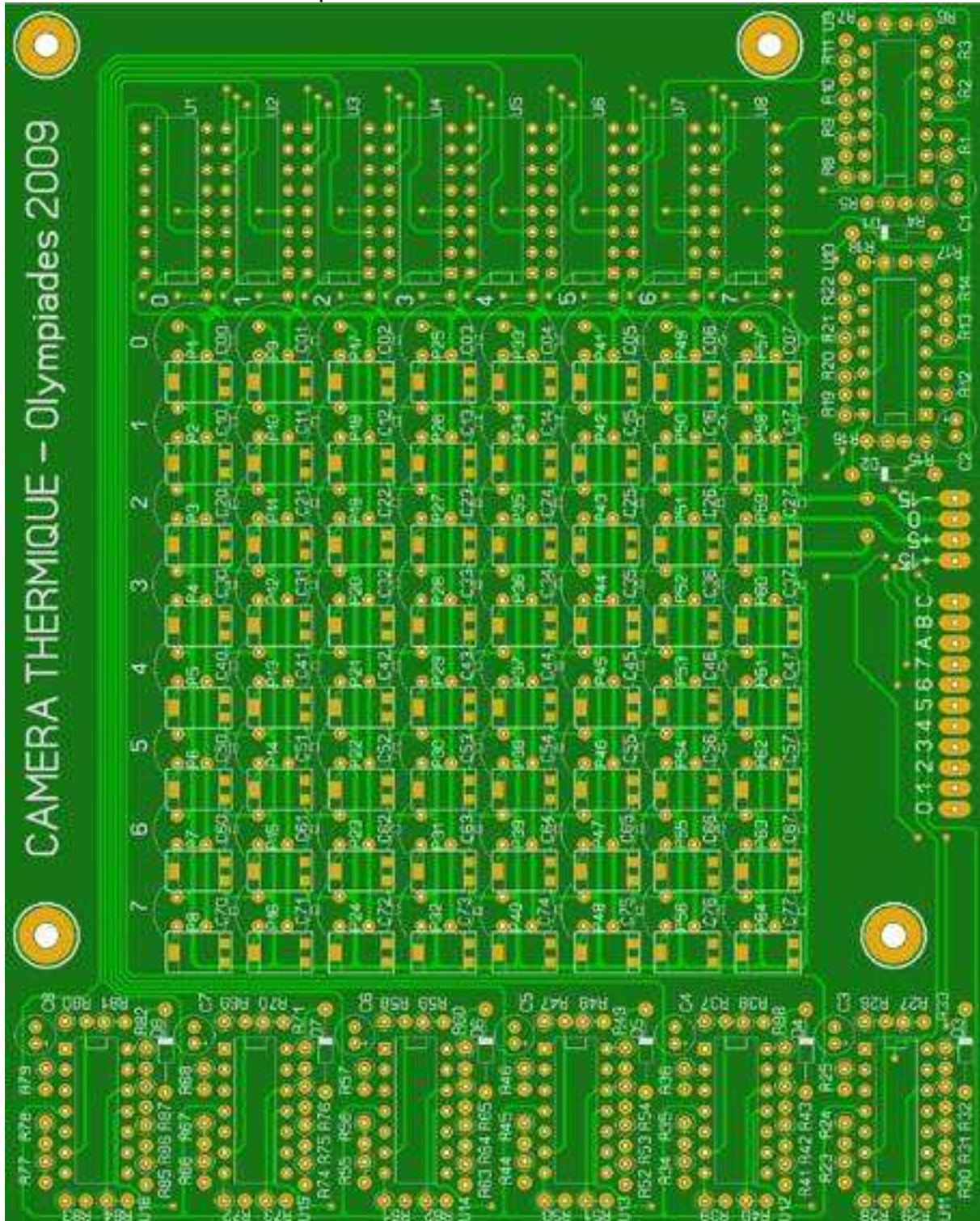
Egalement, M.Telfeyan de la société THESSA technologie qui nous a guidés dans la réalisation de la platine du PIC, du programme et du circuit imprimé et qui nous a prêtés notre module carte vidéo.

Ensuite, l'oncle de Mathieu, M.Bertrand, qui nous a aidés à réaliser le support de notre matrice.

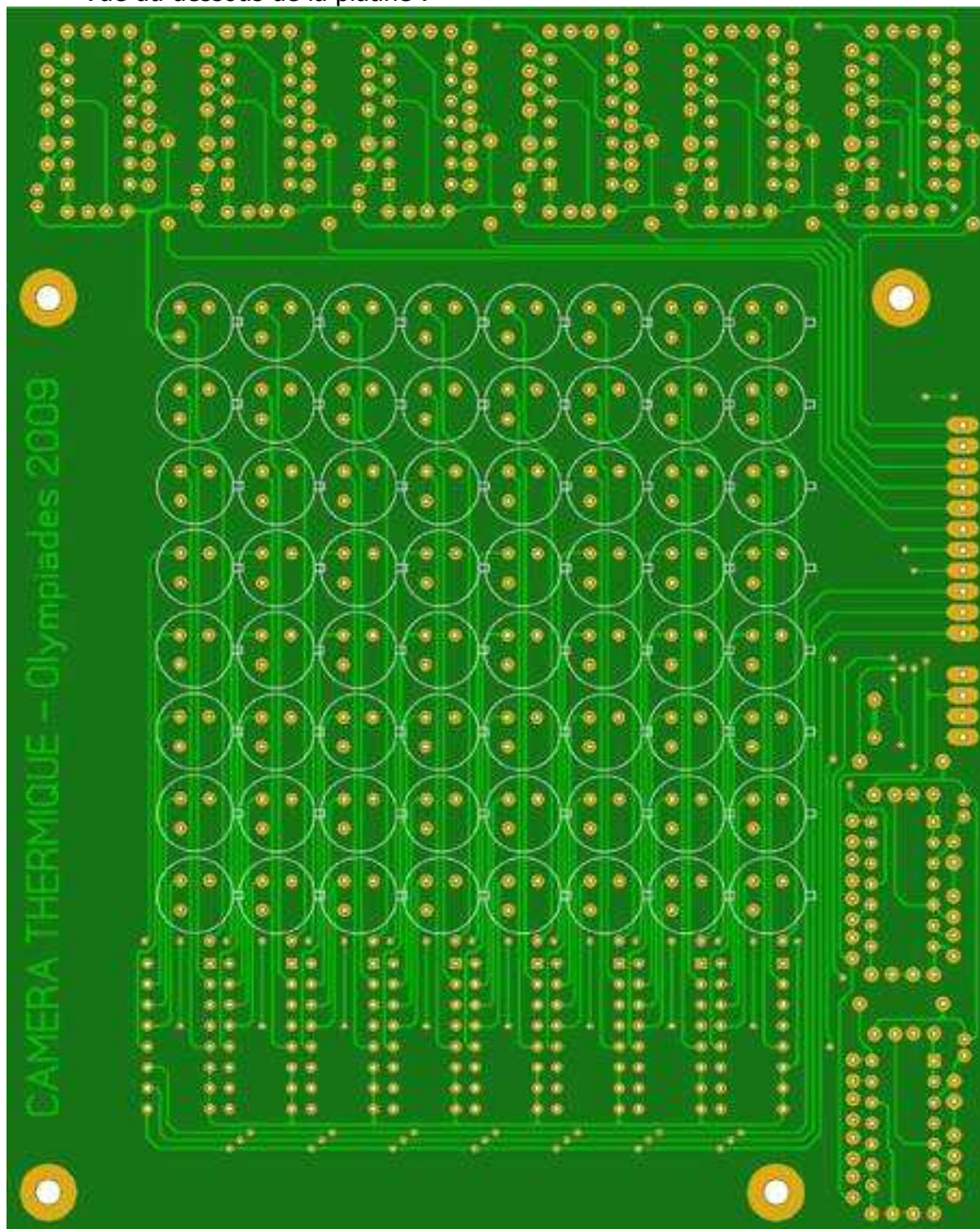
Enfin, les deux préparateurs de notre lycée, Richard et Yann, qui nous ont fournis du matériel et une aide précieuse.

ANNEXES

Vue du dessus de la platine :



Vue du dessous de la platine :



Nomenclature

U1	CD4051	R1	1M	C1	22μ/25V	
U2	CD4051	R2	1k	C2	22μ/25V	
U3	CD4051	R3	100k	C3	22μ/25V	
U4	CD4051	R4	1k	C4	22μ/25V	
U5	CD4051	R5	100k	C5	22μ/25V	
U6	CD4051	R6	10k	C6	22μ/25V	
U7	CD4051	R7	10k	C7	22μ/25V	
U8	CD4051	R8	10k	C8	22μ/25V	
U9	TLO84	R9	10k			
U10	TLO84	R10	10k	D1	Zener 5,1 V	
U11	TLO84	R11	10k	D2	Zener 5,1 V	
U12	TLO84	R12	1M	D3	Zener 5,1 V	
U13	TLO84	R13	1k	D4	Zener 5,1 V	
U14	TLO84	R14	100k	D5	Zener 5,1 V	
U15	TLO84	R15	1k	D6	Zener 5,1 V	
U16	TLO84	R16	100k	D7	Zener 5,1 V	
		R17	10k	D8	Zener 5,1 V	
C00	IRA-E700	R18	10k			
C01	IRA-E700	R19	10k	P1	Ajustable CMS 100k	
C02	IRA-E700	R20	10k	P2	Ajustable CMS 100k	
C03	IRA-E700	R21	10k	P3	Ajustable CMS 100k	
C04	IRA-E700	R22	10k	P4	Ajustable CMS 100k	
C05	IRA-E700	R23	100k	P5	Ajustable CMS 100k	
C06	IRA-E700	R24	1k	P6	Ajustable CMS 100k	
C07	IRA-E700	R25	1M	P7	Ajustable CMS 100k	
C10	IRA-E700	R26	1k	P8	Ajustable CMS 100k	
C11	IRA-E700	R27	100k	P9	Ajustable CMS 100k	
C12	IRA-E700	R28	10k	P10	Ajustable CMS 100k	
C13	IRA-E700	R29	10k	P11	Ajustable CMS 100k	
C14	IRA-E700	R30	10k	P12	Ajustable CMS 100k	
C15	IRA-E700	R31	10k	P13	Ajustable CMS 100k	
C16	IRA-E700	R32	10k	P14	Ajustable CMS 100k	
C17	IRA-E700	R33	10k	P15	Ajustable CMS 100k	
C20	IRA-E700	R34	100k	P16	Ajustable CMS 100k	
C21	IRA-E700	R35	1k	P17	Ajustable CMS 100k	
C22	IRA-E700	R36	1M	P18	Ajustable CMS 100k	
C23	IRA-E700	R37	1k	P19	Ajustable CMS 100k	
C24	IRA-E700	R38	100k	P20	Ajustable CMS 100k	
C25	IRA-E700	R39	10k	P21	Ajustable CMS 100k	
C26	IRA-E700	R40	10k	P22	Ajustable CMS 100k	
C27	IRA-E700	R41	10k	P23	Ajustable CMS 100k	
C30	IRA-E700	R42	10k	P24	Ajustable CMS 100k	
C31	IRA-E700	R43	10k	P25	Ajustable CMS 100k	
C32	IRA-E700	R44	100k	P26	Ajustable CMS 100k	
C33	IRA-E700	R45	1k	P27	Ajustable CMS 100k	
C34	IRA-E700	R46	1M	P28	Ajustable CMS 100k	
C35	IRA-E700	R47	1k	P29	Ajustable CMS 100k	
C36	IRA-E700	R48	100k	P30	Ajustable CMS 100k	
C37	IRA-E700	R49	10k	P31	Ajustable CMS 100k	
C40	IRA-E700	R50	10k	P32	Ajustable CMS 100k	
C41	IRA-E700	R51	10k	P33	Ajustable CMS 100k	
C42	IRA-E700	R52	10k	P34	Ajustable CMS 100k	
C43	IRA-E700	R53	10k	P35	Ajustable CMS 100k	
C44	IRA-E700	R54	10k	P36	Ajustable CMS 100k	
C45	IRA-E700	R55	100k	P37	Ajustable CMS 100k	

C46	IRA-E700	R56	1k	P38	Ajustable CMS 100k
C47	IRA-E700	R57	1M	P39	Ajustable CMS 100k
C50	IRA-E700	R58	1k	P40	Ajustable CMS 100k
C51	IRA-E700	R59	100k	P41	Ajustable CMS 100k
C52	IRA-E700	R60	10k	P42	Ajustable CMS 100k
C53	IRA-E700	R61	10k	P43	Ajustable CMS 100k
C54	IRA-E700	R62	10k	P44	Ajustable CMS 100k
C55	IRA-E700	R63	10k	P45	Ajustable CMS 100k
C56	IRA-E700	R64	10k	P46	Ajustable CMS 100k
C57	IRA-E700	R65	10k	P47	Ajustable CMS 100k
C60	IRA-E700	R66	100k	P48	Ajustable CMS 100k
C61	IRA-E700	R67	1k	P49	Ajustable CMS 100k
C62	IRA-E700	R68	1M	P50	Ajustable CMS 100k
C63	IRA-E700	R69	1k	P51	Ajustable CMS 100k
C64	IRA-E700	R70	100k	P52	Ajustable CMS 100k
C65	IRA-E700	R71	10k	P53	Ajustable CMS 100k
C66	IRA-E700	R72	10k	P54	Ajustable CMS 100k
C67	IRA-E700	R73	10k	P55	Ajustable CMS 100k
C70	IRA-E700	R74	10k	P56	Ajustable CMS 100k
C71	IRA-E700	R75	10k	P57	Ajustable CMS 100k
C72	IRA-E700	R76	10k	P58	Ajustable CMS 100k
C73	IRA-E700	R77	100k	P59	Ajustable CMS 100k
C74	IRA-E700	R78	1k	P60	Ajustable CMS 100k
C75	IRA-E700	R79	1M	P61	Ajustable CMS 100k
C76	IRA-E700	R80	1k	P62	Ajustable CMS 100k
C77	IRA-E700	R81	100k	P63	Ajustable CMS 100k
		R82	10k	P64	Ajustable CMS 100k
		R83	10k		
		R84	10k		
		R85	10k		
		R86	10k		
		R87	10k		
		R88	10k		

Programme en basic avant compilation en langage machine :

```
*****
'* Name      : Camerafinal.BAS *
'* Author    : Olympiades 08/09 [M. BOTTOS, S.CEPPI, M.GABIN, M.BERTRAND] *
'* Notice    : *
'*           : Tous droits réservés *
'* Date      : 09/01/2009 *
'* Version   : 7.5 *
'* Notes     : Delay affichage => réduction de 70 à 10 ms (v7.4) *
'*           : Changment variable x => colonne (v7.5) *
*****

Device = 18F452           'déclaration du µControlleur
XTAL = 20                 'déclaration du quartz

***** configuration A/D *****

ADIN_RES      10           'résolution 10 bits
ADIN_TAD      32_FOSC      'osc échantillonnage A/D
ADIN_STIME    50           '50us temps de charge condos
internes convertisseur

***** configuration port RS hardware *****

Declare HSERIAL_BAUD = 9600           ' vitesse du transfert en baud :
9600
Declare HSERIAL_RCSTA = %10010000     ' permet la réception en continue
de données via le port série
Declare HSERIAL_TXSTA = %00100000     ' permet la diffusion et les modes
asynchrones
Declare HSERIAL_CLEAR = On            ' permet de vider le cache avant
la réception

***** configuration LCD *****

Declare LCD_TYPE 0                    'LCD type alphanumérique
Declare LCD_INTERFACE 4                'Interface 4 bits
Declare LCD_LINES 2                    'LCD 2 lignes
Declare LCD_DTPIN PORTB.0              'Port des datas D4-D5-D6-D7
Declare LCD_RSPIN PORTB.4              'Signal RS
Declare LCD_ENPIN PORTB.5              'Signal ENable

Dim L0[8] As Word 'variable valeurs ligne 1 (8 valeurs)
Dim L1[8] As Word 'variable valeurs ligne 1 (8 valeurs)
Dim L2[8] As Word 'variable valeurs ligne 1 (8 valeurs)
Dim L3[8] As Word 'variable valeurs ligne 1 (8 valeurs)
Dim L4[8] As Word 'variable valeurs ligne 1 (8 valeurs)
Dim L5[8] As Word 'variable valeurs ligne 1 (8 valeurs)
Dim L6[8] As Word 'variable valeurs ligne 1 (8 valeurs)
Dim L7[8] As Word 'variable valeurs ligne 1 (8 valeurs)

Dim x As Byte ' variable de boucle

'dim a as bit
```

```

'dim b as byte
'dim c as word
'dim d as dword
'dim s as string * 20

'***** déclaration entrée/sortie des ports *****
Input PORTC.4
Output PORTD
Symbol topmesure=PORTC.4
Symbol colonne = PORTD
Symbol ResetVGA = PORTC.5

'***** déclaration entrée/sortie des ports *****

TRISA = %11111111          'portA en entrée
ADCON1 = %10000000        'portA en analogique et résultat
justifié à droite

DelayMS 1000              'stabilisation µC
Cls
                          '1234567890123456
Print At 1,1, "Camera Thermique"
'low ResetVGA
DelayMS 2000
'high resetVGA
HRSOut $55                'contrôle port RS
DelayMS 1000

HRSOut $59,$03,$01 'passage en VGA de la carte vidéo
DelayMS 2000

HRSOut $55                'commande après le changement de résolution
DelayMS 1000
Cls                        ' nettoyage de l'ecran LCD

'*****
'*                          programme principal                               *
'*****

Loop:
Repeat
Until topmesure =1
DelayMS 100
For colonne=0 To 7
colonne=PORTD              'incréméntation colonne
DelayUS 10

L0[colonne] = ADIn 0       'lecture A/D channel 0 (AN0)
DelayUS 1
L1[colonne] = ADIn 1       'lecture A/D channel 1 (AN1)
DelayUS 1
L2[colonne] = ADIn 2       'lecture A/D channel 2 (AN2)
DelayUS 1
L3[colonne] = ADIn 3       'lecture A/D channel 3 (AN3)
DelayUS 1
L4[colonne] = ADIn 4       'lecture A/D channel 4 (AN4)
DelayUS 1
L5[colonne] = ADIn 5       'lecture A/D channel 5 (AN5)
DelayUS 1
L6[colonne] = ADIn 6       'lecture A/D channel 6 (AN6)

```

```

DelayUS 1
L7[colonne] = ADIn 7           'lecture A/D channel 7 (AN7)
DelayUS 1

```

Next

```

Print At 1,1, DEC4 L0[0], DEC4 L0[1], DEC4 L0[2], DEC4 L0[3]
'visualisation première ligne ecran LCD
Print At 2,1, DEC4 L0[4], DEC4 L0[5], DEC4 L0[6], DEC4 L0[7]
'visualisation deuxième ligne ecran LCD
DelayMS 10

'*****
'*                               programme d'affichage                               *
'*****

'*****
'*                               première ligne                               *
'*****
HRSOut $70,$00 'rend le carre dessiné plein
DelayMS 100
' cmd      |  x1  |  y1  |  x2  |  y2  | valeur a afficher
HRSOut $72, $00,$F0, $00,$86, $00,$F8, $00,$8E, L0[0]
DelayMS 10
HRSOut $72, $00,$F8, $00,$86, $01,$00, $00,$8E, (L0[0]+L0[1])/2
'inter.
DelayMS 10
HRSOut $72, $01,$00, $00,$86, $01,$08, $00,$8E, L0[1]
DelayMS 10
HRSOut $72, $01,$08, $00,$86, $01,$10, $00,$8E, (L0[1]+L0[2])/2
'inter.
DelayMS 10
HRSOut $72, $01,$10, $00,$86, $01,$18, $00,$8E, L0[2]
DelayMS 10
HRSOut $72, $01,$18, $00,$86, $01,$20, $00,$8E, (L0[2]+L0[3])/2
'inter.
DelayMS 10
HRSOut $72, $01,$20, $00,$86, $01,$28, $00,$8E, L0[3]
DelayMS 10
HRSOut $72, $01,$28, $00,$86, $01,$30, $00,$8E, (L0[3]+L0[4])/2
'inter.
DelayMS 10
HRSOut $72, $01,$30, $00,$86, $01,$38, $00,$8E, L0[4]
DelayMS 10
HRSOut $72, $01,$38, $00,$86, $01,$40, $00,$8E, (L0[4]+L0[5])/2
'inter.
DelayMS 10
HRSOut $72, $01,$40, $00,$86, $01,$48, $00,$8E, L0[5]
DelayMS 10
HRSOut $72, $01,$48, $00,$86, $01,$50, $00,$8E, (L0[4]+L0[5])/2
'inter.
DelayMS 10
HRSOut $72, $01,$50, $00,$86, $01,$58, $00,$8E, L0[6]
DelayMS 10
HRSOut $72, $01,$58, $00,$86, $01,$60, $00,$8E, (L0[6]+L0[7])/2
'inter.
DelayMS 10
HRSOut $72, $01,$60, $00,$86, $01,$68, $00,$8E, L0[7]
DelayMS 10

```

```

'*****

```

```

'*          deuxième ligne          *
'*****
HRSOut $70,$00  'rend le carré dessiné plein
DelayMS 100
' cmd      |   x1   |   y1   |   x2   |   y2   | valeur a afficher
HRSOut $72, $00,$F0, $00,$96, $00,$F8, $00,$9E, L1[0]
DelayMS 10
HRSOut $72, $00,$F8, $00,$96, $01,$00, $00,$9E, (L1[0]+L1[1])/2
'inter.
DelayMS 10
HRSOut $72, $01,$00, $00,$96, $01,$08, $00,$9E, L1[1]
DelayMS 10
HRSOut $72, $01,$08, $00,$96, $01,$10, $00,$9E, (L1[1]+L1[2])/2
'inter.
DelayMS 10
HRSOut $72, $01,$10, $00,$96, $01,$18, $00,$9E, L1[2]
DelayMS 10
HRSOut $72, $01,$18, $00,$96, $01,$20, $00,$9E, (L1[2]+L1[3])/2
'inter.
DelayMS 10
HRSOut $72, $01,$20, $00,$96, $01,$28, $00,$9E, L1[3]
DelayMS 10
HRSOut $72, $01,$28, $00,$96, $01,$30, $00,$9E, (L1[3]+L1[4])/2
'inter.
DelayMS 10
HRSOut $72, $01,$30, $00,$96, $01,$38, $00,$9E, L1[4]
DelayMS 10
HRSOut $72, $01,$38, $00,$96, $01,$40, $00,$9E, (L1[4]+L1[5])/2
'inter.
DelayMS 10
HRSOut $72, $01,$40, $00,$96, $01,$48, $00,$9E, L1[5]
DelayMS 10
HRSOut $72, $01,$48, $00,$96, $01,$50, $00,$9E, (L1[5]+L1[6])/2
'inter.
DelayMS 10
HRSOut $72, $01,$50, $00,$96, $01,$58, $00,$9E, L1[6]
DelayMS 10
HRSOut $72, $01,$58, $00,$96, $01,$60, $00,$9E, (L1[6]+L1[7])/2
'inter.
DelayMS 10
HRSOut $72, $01,$60, $00,$96, $01,$68, $00,$9E, L1[7]
DelayMS 10

'*****
'*          troisième ligne          *
'*****
HRSOut $70,$00  'rend le carré dessiné plein
DelayMS 100
'      |   x1   |   y1   |   x2   |   y2   |
HRSOut $72, $00,$F0, $00,$A6, $00,$F8, $00,$AE, L2[0]
DelayMS 10
HRSOut $72, $00,$F8, $00,$A6, $01,$00, $00,$AE, (L2[0]+L2[1])/2
'inter.
DelayMS 10
HRSOut $72, $01,$00, $00,$A6, $01,$08, $00,$AE, L2[1]
DelayMS 10
HRSOut $72, $01,$08, $00,$A6, $01,$10, $00,$AE, (L2[1]+L2[2])/2
'inter.
DelayMS 10
HRSOut $72, $01,$10, $00,$A6, $01,$18, $00,$AE, L2[2]
DelayMS 10

```

```

    HRSOut $72, $01,$18, $00,$A6, $01,$20, $00,$AE, (L2[2]+L2[3])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$20, $00,$A6, $01,$28, $00,$AE, L2[3]
    DelayMS 10
    HRSOut $72, $01,$28, $00,$A6, $01,$30, $00,$AE, (L2[3]+L2[4])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$30, $00,$A6, $01,$38, $00,$AE, L2[4]
    DelayMS 10
    HRSOut $72, $01,$38, $00,$A6, $01,$40, $00,$AE, (L2[4]+L2[5])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$40, $00,$A6, $01,$48, $00,$AE, L2[5]
    DelayMS 10
    HRSOut $72, $01,$48, $00,$A6, $01,$50, $00,$AE, (L2[5]+L2[6])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$50, $00,$A6, $01,$58, $00,$AE, L2[6]
    DelayMS 10
    HRSOut $72, $01,$58, $00,$A6, $01,$60, $00,$AE, (L2[6]+L2[7])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$60, $00,$A6, $01,$68, $00,$AE, L2[7]
    DelayMS 10

'*****
*'          quatrième ligne          *
'*****
HRSOut $70,$00  'rend le carré dessiné plein
DelayMS 100
'          /   x1   /   y1   /   x2   /   y2   /
HRSOut $72, $00,$F0, $00,$B6, $00,$F8, $00,$BE, L3[0]
DelayMS 10
HRSOut $72, $00,$F8, $00,$B6, $01,$00, $00,$BE, (L3[0]+L3[1])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$00, $00,$B6, $01,$08, $00,$BE, L3[1]
    DelayMS 10
    HRSOut $72, $01,$08, $00,$B6, $01,$10, $00,$BE, (L3[1]+L3[2])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$10, $00,$B6, $01,$18, $00,$BE, L3[2]
    DelayMS 10
    HRSOut $72, $01,$18, $00,$B6, $01,$20, $00,$BE, (L3[2]+L3[3])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$20, $00,$B6, $01,$28, $00,$BE, L3[3]
    DelayMS 10
    HRSOut $72, $01,$28, $00,$B6, $01,$30, $00,$BE, (L3[3]+L3[4])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$30, $00,$B6, $01,$38, $00,$BE, L3[4]
    DelayMS 10
    HRSOut $72, $01,$38, $00,$B6, $01,$40, $00,$BE, (L3[4]+L3[5])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$40, $00,$B6, $01,$48, $00,$BE, L3[5]
    DelayMS 10
    HRSOut $72, $01,$48, $00,$B6, $01,$50, $00,$BE, (L3[5]+L3[6])/2
'inter.

```

```

DelayMS 10
HRSOut $72, $01,$50, $00,$B6, $01,$58, $00,$BE, L3[6]
DelayMS 10
HRSOut $72, $01,$58, $00,$B6, $01,$60, $00,$BE, (L3[6]+L3[7])/2
'inter.
DelayMS 10
HRSOut $72, $01,$60, $00,$B6, $01,$68, $00,$BE, L3[7]
DelayMS 10

'*****
'*                               cinquième ligne                               *
'*****
HRSOut $70,$00  'rend le carré dessiné plein
DelayMS 100
' cmd      /   x1   /   y1   /   x2   /   y2   / valeur a afficher
HRSOut $72, $00,$F0, $00,$C6, $00,$F8, $00,$CE, L4[0]
DelayMS 10
HRSOut $72, $00,$F8, $00,$C6, $01,$00, $00,$CE, (L4[0]+L4[1])/2
'inter.
DelayMS 10
HRSOut $72, $01,$00, $00,$C6, $01,$08, $00,$CE, L4[1]
DelayMS 10
HRSOut $72, $01,$08, $00,$C6, $01,$10, $00,$CE, (L4[1]+L4[2])/2
'inter.
DelayMS 10
HRSOut $72, $01,$10, $00,$C6, $01,$18, $00,$CE, L4[2]
DelayMS 10
HRSOut $72, $01,$18, $00,$C6, $01,$20, $00,$CE, (L4[2]+L4[3])/2
'inter.
DelayMS 10
HRSOut $72, $01,$20, $00,$C6, $01,$28, $00,$CE, L4[3]
DelayMS 10
HRSOut $72, $01,$28, $00,$C6, $01,$30, $00,$CE, (L4[3]+L4[4])/2
'inter.
DelayMS 10
HRSOut $72, $01,$30, $00,$C6, $01,$38, $00,$CE, L4[4]
DelayMS 10
HRSOut $72, $01,$38, $00,$C6, $01,$40, $00,$CE, (L4[4]+L4[5])/2
'inter.
DelayMS 10
HRSOut $72, $01,$40, $00,$C6, $01,$48, $00,$CE, L4[5]
DelayMS 10
HRSOut $72, $01,$48, $00,$C6, $01,$50, $00,$CE, (L4[5]+L4[6])/2
'inter.
DelayMS 10
HRSOut $72, $01,$50, $00,$C6, $01,$58, $00,$CE, L4[6]
DelayMS 10
HRSOut $72, $01,$58, $00,$C6, $01,$60, $00,$CE, (L4[6]+L4[7])/2
'inter.
DelayMS 10
HRSOut $72, $01,$60, $00,$C6, $01,$68, $00,$CE, L4[7]
DelayMS 10

'*****
'*                               sixième ligne                               *
'*****
HRSOut $70,$00  'rend le carré dessiné plein
DelayMS 100
' cmd      /   x1   /   y1   /   x2   /   y2   / valeur a afficher
HRSOut $72, $00,$F0, $00,$D6, $00,$F8, $00,$DE, L5[0]
DelayMS 10

```



```

    HRSOut $72, $00,$F8, $00,$D6, $01,$00, $00,$DE, (L5[0]+L5[1])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$00, $00,$D6, $01,$08, $00,$DE, L5[1]
    DelayMS 10
    HRSOut $72, $01,$08, $00,$D6, $01,$10, $00,$DE, (L5[1]+L5[2])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$10, $00,$D6, $01,$18, $00,$DE, L5[2]
    DelayMS 10
    HRSOut $72, $01,$18, $00,$D6, $01,$20, $00,$DE, (L5[2]+L5[3])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$20, $00,$D6, $01,$28, $00,$DE, L5[3]
    DelayMS 10
    HRSOut $72, $01,$28, $00,$D6, $01,$30, $00,$DE, (L5[3]+L5[4])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$30, $00,$D6, $01,$38, $00,$DE, L5[4]
    DelayMS 10
    HRSOut $72, $01,$38, $00,$D6, $01,$40, $00,$DE, (L5[4]+L5[5])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$40, $00,$D6, $01,$48, $00,$DE, L5[5]
    DelayMS 10
    HRSOut $72, $01,$48, $00,$D6, $01,$50, $00,$DE, (L5[5]+L5[6])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$50, $00,$D6, $01,$58, $00,$DE, L5[6]
    DelayMS 10
    HRSOut $72, $01,$58, $00,$D6, $01,$60, $00,$DE, (L5[6]+L5[7])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$60, $00,$D6, $01,$68, $00,$DE, L5[7]
    DelayMS 10

*****
*                               *
*          septième ligne          *
*****
HRSOut $70,$00  'rend le carré dessiné plein
DelayMS 100
' cmd      |  x1  |  y1  |  x2  |  y2  |  valeur a afficher
HRSOut $72, $00,$F0, $00,$E6, $00,$F8, $00,$EE, L6[0]
DelayMS 10
HRSOut $72, $00,$F8, $00,$E6, $01,$00, $00,$EE, (L6[0]+L6[1])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$00, $00,$E6, $01,$08, $00,$EE, L6[1]
    DelayMS 10
    HRSOut $72, $01,$08, $00,$E6, $01,$10, $00,$EE, (L6[1]+L6[2])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$10, $00,$E6, $01,$18, $00,$EE, L6[2]
    DelayMS 10
    HRSOut $72, $01,$18, $00,$E6, $01,$20, $00,$EE, (L6[2]+L6[3])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$20, $00,$E6, $01,$28, $00,$EE, L6[3]
    DelayMS 10
    HRSOut $72, $01,$28, $00,$E6, $01,$30, $00,$EE, (L6[3]+L6[4])/2
'inter.

```

```

    DelayMS 10
    HRSOut $72, $01,$30, $00,$E6, $01,$38, $00,$EE, L6[4]
    DelayMS 10
    HRSOut $72, $01,$38, $00,$E6, $01,$40, $00,$EE, (L6[4]+L6[5])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$40, $00,$E6, $01,$48, $00,$EE, L6[5]
    DelayMS 10
    HRSOut $72, $01,$48, $00,$E6, $01,$50, $00,$EE, (L6[5]+L6[6])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$50, $00,$E6, $01,$58, $00,$EE, L6[6]
    DelayMS 10
    HRSOut $72, $01,$58, $00,$E6, $01,$60, $00,$EE, (L6[6]+L6[7])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$60, $00,$E6, $01,$68, $00,$EE, L6[7]
    DelayMS 10

'*****
'*          huitième ligne          *
'*****
HRSOut $70,$00 'rend le carré dessiné plein
DelayMS 100
' cmd      |  x1  |  y1  |  x2  |  y2  |  valeur a afficher
HRSOut $72, $00,$F0, $00,$F6, $00,$F8, $00,$FE, L7[0]
DelayMS 10
HRSOut $72, $00,$F8, $00,$F6, $01,$00, $00,$FE, (L7[0]+L7[1])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$00, $00,$F6, $01,$08, $00,$FE, L7[1]
    DelayMS 10
    HRSOut $72, $01,$08, $00,$F6, $01,$10, $00,$FE, (L7[1]+L7[2])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$10, $00,$F6, $01,$18, $00,$FE, L7[2]
    DelayMS 10
    HRSOut $72, $01,$18, $00,$F6, $01,$20, $00,$FE, (L7[2]+L7[3])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$20, $00,$F6, $01,$28, $00,$FE, L7[3]
    DelayMS 10
    HRSOut $72, $01,$28, $00,$F6, $01,$30, $00,$FE, (L7[3]+L7[4])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$30, $00,$F6, $01,$38, $00,$FE, L7[4]
    DelayMS 10
    HRSOut $72, $01,$38, $00,$F6, $01,$40, $00,$FE, (L7[4]+L7[5])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$40, $00,$F6, $01,$48, $00,$FE, L7[5]
    DelayMS 10
    HRSOut $72, $01,$48, $00,$F6, $01,$50, $00,$FE, (L7[5]+L7[6])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$50, $00,$F6, $01,$58, $00,$FE, L7[6]
    DelayMS 10
    HRSOut $72, $01,$58, $00,$F6, $01,$60, $00,$FE, (L7[6]+L7[7])/2
'inter.
    DelayMS 10
    HRSOut $72, $01,$60, $00,$F6, $01,$68, $00,$FE, L7[7]

```

DelayMS 10

```
*****
'*                               programme d'affichage inter-lignes                               *
*****

*****
'*                               L0[0]~L1[7]                               *
*****
HRSOut $70,$00   'rend le carre dessiné plein
DelayMS 100
'           /   x1   /   y1   /   x2   /   y2   /
HRSOut $72, $00,$F0, $00,$8E, $00,$F8, $00,$96, (L0[0]+L1[0])/2
DelayMS 10
HRSOut $72, $00,$F8, $00,$8E, $01,$00, $00,$96,
(L0[0]+L0[1]+L1[0]+L1[1])/4
DelayMS 10
HRSOut $72, $01,$00, $00,$8E, $01,$08, $00,$96, (L0[1]+L1[1])/2
DelayMS 10
HRSOut $72, $01,$08, $00,$8E, $01,$10, $00,$96,
(L0[1]+L0[2]+L1[1]+L1[2])/4
DelayMS 10
HRSOut $72, $01,$10, $00,$8E, $01,$18, $00,$96, (L0[2]+L1[2])/2
DelayMS 10
HRSOut $72, $01,$18, $00,$8E, $01,$20, $00,$96,
(L0[2]+L0[3]+L1[2]+L1[3])/4
DelayMS 10
HRSOut $72, $01,$20, $00,$8E, $01,$28, $00,$96, (L0[3]+L1[3])/2
DelayMS 10
HRSOut $72, $01,$28, $00,$8E, $01,$30, $00,$96,
(L0[3]+L0[4]+L1[3]+L1[4])/4
DelayMS 10
HRSOut $72, $01,$30, $00,$8E, $01,$38, $00,$96, (L0[4]+L1[4])/2
DelayMS 10
HRSOut $72, $01,$38, $00,$8E, $01,$40, $00,$96,
(L0[4]+L0[5]+L1[4]+L1[5])/4
DelayMS 10
HRSOut $72, $01,$40, $00,$8E, $01,$48, $00,$96, (L0[5]+L1[5])/2
DelayMS 10
HRSOut $72, $01,$48, $00,$8E, $01,$50, $00,$96,
(L0[4]+L0[5]+L1[4]+L1[5])/4
DelayMS 10
HRSOut $72, $01,$50, $00,$8E, $01,$58, $00,$96, (L0[6]+L1[6])/2
DelayMS 10
HRSOut $72, $01,$58, $00,$8E, $01,$60, $00,$96,
(L0[6]+L0[7]+L1[6]+L1[7])/4
DelayMS 10
HRSOut $72, $01,$60, $00,$8E, $01,$68, $00,$96, (L0[7]+L1[7])/2
DelayMS 10

*****
'*                               L1[0]~L2[7]                               *
*****
HRSOut $70,$00   'rend le carre dessiné plein
DelayMS 100
'           /   x1   /   y1   /   x2   /   y2   /
HRSOut $72, $00,$F0, $00,$9E, $00,$F8, $00,$A6, (L1[0]+L2[0])/2
DelayMS 10
HRSOut $72, $00,$F8, $00,$9E, $01,$00, $00,$A6,
(L1[0]+L1[1]+L2[0]+L2[1])/4
DelayMS 10
```

```

HRSOut $72, $01,$00, $00,$9E, $01,$08, $00,$A6, (L1[1]+L2[1])/2
DelayMS 10
HRSOut $72, $01,$08, $00,$9E, $01,$10, $00,$A6,
(L1[1]+L1[2]+L2[1]+L2[2])/4
DelayMS 10
HRSOut $72, $01,$10, $00,$9E, $01,$18, $00,$A6, (L1[2]+L2[2])/2
DelayMS 10
HRSOut $72, $01,$18, $00,$9E, $01,$20, $00,$A6,
(L1[2]+L1[3]+L2[2]+L2[3])/4
DelayMS 10
HRSOut $72, $01,$20, $00,$9E, $01,$28, $00,$A6, (L1[3]+L2[3])/2
DelayMS 10
HRSOut $72, $01,$28, $00,$9E, $01,$30, $00,$A6,
(L1[3]+L1[4]+L2[3]+L2[4])/4
DelayMS 10
HRSOut $72, $01,$30, $00,$9E, $01,$38, $00,$A6, (L1[4]+L2[4])/2
DelayMS 10
HRSOut $72, $01,$38, $00,$9E, $01,$40, $00,$A6,
(L1[4]+L1[5]+L2[4]+L2[5])/4
DelayMS 10
HRSOut $72, $01,$40, $00,$9E, $01,$48, $00,$A6, (L1[5]+L2[5])/2
DelayMS 10
HRSOut $72, $01,$48, $00,$9E, $01,$50, $00,$A6,
(L1[4]+L1[5]+L2[4]+L2[5])/4
DelayMS 10
HRSOut $72, $01,$50, $00,$9E, $01,$58, $00,$A6, (L1[6]+L2[6])/2
DelayMS 10
HRSOut $72, $01,$58, $00,$9E, $01,$60, $00,$A6,
(L1[6]+L1[7]+L2[6]+L2[7])/4
DelayMS 10
HRSOut $72, $01,$60, $00,$9E, $01,$68, $00,$A6, (L1[7]+L2[7])/2
DelayMS 10

```

```

'*****
' *                L2[0]~L3[7]                *
'*****
HRSOut $70,$00 'rend le carre dessiné plein
DelayMS 100
' / x1 / y1 / x2 / y2 /
HRSOut $72, $00,$F0, $00,$AE, $00,$F8, $00,$B6, (L2[0]+L3[0])/2
DelayMS 10
HRSOut $72, $00,$F8, $00,$AE, $01,$00, $00,$B6,
(L2[0]+L2[1]+L3[0]+L3[1])/4
DelayMS 10
HRSOut $72, $01,$00, $00,$AE, $01,$08, $00,$B6, (L2[1]+L3[1])/2
DelayMS 10
HRSOut $72, $01,$08, $00,$AE, $01,$10, $00,$B6,
(L2[1]+L2[2]+L3[1]+L3[2])/4
DelayMS 10
HRSOut $72, $01,$10, $00,$AE, $01,$18, $00,$B6, (L2[2]+L3[2])/2
DelayMS 10
HRSOut $72, $01,$18, $00,$AE, $01,$20, $00,$B6,
(L2[2]+L2[3]+L3[2]+L3[3])/4
DelayMS 10
HRSOut $72, $01,$20, $00,$AE, $01,$28, $00,$B6, (L2[3]+L3[3])/2
DelayMS 10
HRSOut $72, $01,$28, $00,$AE, $01,$30, $00,$B6,
(L2[3]+L2[4]+L3[3]+L3[4])/4
DelayMS 10
HRSOut $72, $01,$30, $00,$AE, $01,$38, $00,$B6, (L2[4]+L3[4])/2
DelayMS 10

```

```

    HRSOut $72, $01,$38, $00,$AE, $01,$40, $00,$B6,
(L2[4]+L2[5]+L3[4]+L3[5])/4
    DelayMS 10
    HRSOut $72, $01,$40, $00,$AE, $01,$48, $00,$B6, (L2[5]+L3[5])/2
    DelayMS 10
    HRSOut $72, $01,$48, $00,$AE, $01,$50, $00,$B6,
(L2[4]+L2[5]+L3[4]+L3[5])/4
    DelayMS 10
    HRSOut $72, $01,$50, $00,$AE, $01,$58, $00,$B6, (L2[6]+L3[6])/2
    DelayMS 10
    HRSOut $72, $01,$58, $00,$AE, $01,$60, $00,$B6,
(L2[6]+L2[7]+L3[6]+L3[7])/4
    DelayMS 10
    HRSOut $72, $01,$60, $00,$AE, $01,$68, $00,$B6, (L2[7]+L3[7])/2
    DelayMS 10

```

```

'*****

```

```

' *                L3[0]~L4[7]                *

```

```

'*****

```

```

    HRSOut $70,$00 'rend le carre dessiné plein
    DelayMS 100
    '
    '      /   x1   /   y1   /   x2   /   y2   /
    HRSOut $72, $00,$F0, $00,$BE, $00,$F8, $00,$C6, (L3[0]+L4[0])/2
    DelayMS 10
    HRSOut $72, $00,$F8, $00,$BE, $01,$00, $00,$C6,
(L3[0]+L3[1]+L4[0]+L4[1])/4
    DelayMS 10
    HRSOut $72, $01,$00, $00,$BE, $01,$08, $00,$C6, (L3[1]+L4[1])/2
    DelayMS 10
    HRSOut $72, $01,$08, $00,$BE, $01,$10, $00,$C6,
(L3[1]+L3[2]+L4[1]+L4[2])/4
    DelayMS 10
    HRSOut $72, $01,$10, $00,$BE, $01,$18, $00,$C6, (L3[2]+L4[2])/2
    DelayMS 10
    HRSOut $72, $01,$18, $00,$BE, $01,$20, $00,$C6,
(L3[2]+L3[3]+L4[2]+L4[3])/4
    DelayMS 10
    HRSOut $72, $01,$20, $00,$BE, $01,$28, $00,$C6, (L3[3]+L4[3])/2
    DelayMS 10
    HRSOut $72, $01,$28, $00,$BE, $01,$30, $00,$C6,
(L3[3]+L3[4]+L4[3]+L4[4])/4
    DelayMS 10
    HRSOut $72, $01,$30, $00,$BE, $01,$38, $00,$C6, (L3[4]+L4[4])/2
    DelayMS 10
    HRSOut $72, $01,$38, $00,$BE, $01,$40, $00,$C6,
(L3[4]+L3[5]+L4[4]+L4[5])/4
    DelayMS 10
    HRSOut $72, $01,$40, $00,$BE, $01,$48, $00,$C6, (L3[5]+L4[5])/2
    DelayMS 10
    HRSOut $72, $01,$48, $00,$BE, $01,$50, $00,$C6,
(L3[4]+L3[5]+L4[4]+L4[5])/4
    DelayMS 10
    HRSOut $72, $01,$50, $00,$BE, $01,$58, $00,$C6, (L3[6]+L4[6])/2
    DelayMS 10
    HRSOut $72, $01,$58, $00,$BE, $01,$60, $00,$C6,
(L3[6]+L3[7]+L4[6]+L4[7])/4
    DelayMS 10
    HRSOut $72, $01,$60, $00,$BE, $01,$68, $00,$C6, (L3[7]+L4[7])/2
    DelayMS 10

```

```

'*****

```

```

' *                               L4[0]~L5[7]                               *
'*****
HRSOut $70,$00   'rend le carre dessiné plein
DelayMS 100
'
'   /   x1   /   y1   /   x2   /   y2   /
HRSOut $72, $00,$F0, $00,$CE, $00,$F8, $00,$D6, (L4[0]+L5[0])/2
DelayMS 10
HRSOut $72, $00,$F8, $00,$CE, $01,$00, $00,$D6,
(L4[0]+L4[1]+L5[0]+L5[1])/4
DelayMS 10
HRSOut $72, $01,$00, $00,$CE, $01,$08, $00,$D6, (L4[1]+L5[1])/2
DelayMS 10
HRSOut $72, $01,$08, $00,$CE, $01,$10, $00,$D6,
(L4[1]+L4[2]+L5[1]+L5[2])/4
DelayMS 10
HRSOut $72, $01,$10, $00,$CE, $01,$18, $00,$D6, (L4[2]+L5[2])/2
DelayMS 10
HRSOut $72, $01,$18, $00,$CE, $01,$20, $00,$D6,
(L4[2]+L4[3]+L5[2]+L5[3])/4
DelayMS 10
HRSOut $72, $01,$20, $00,$CE, $01,$28, $00,$D6, (L4[3]+L5[3])/2
DelayMS 10
HRSOut $72, $01,$28, $00,$CE, $01,$30, $00,$D6,
(L4[3]+L4[4]+L5[3]+L5[4])/4
DelayMS 10
HRSOut $72, $01,$30, $00,$CE, $01,$38, $00,$D6, (L4[4]+L5[4])/2
DelayMS 10
HRSOut $72, $01,$38, $00,$CE, $01,$40, $00,$D6,
(L4[4]+L4[5]+L5[4]+L5[5])/4
DelayMS 10
HRSOut $72, $01,$40, $00,$CE, $01,$48, $00,$D6, (L4[5]+L5[5])/2
DelayMS 10
HRSOut $72, $01,$48, $00,$CE, $01,$50, $00,$D6,
(L4[4]+L4[5]+L5[4]+L5[5])/4
DelayMS 10
HRSOut $72, $01,$50, $00,$CE, $01,$58, $00,$D6, (L4[6]+L5[6])/2
DelayMS 10
HRSOut $72, $01,$58, $00,$CE, $01,$60, $00,$D6,
(L4[6]+L4[7]+L5[6]+L5[7])/4
DelayMS 10
HRSOut $72, $01,$60, $00,$CE, $01,$68, $00,$D6, (L4[7]+L5[7])/2
DelayMS 10

'*****
' *                               L5[0]~L6[7]                               *
'*****
HRSOut $70,$00   'rend le carre dessiné plein
DelayMS 100
'
'   /   x1   /   y1   /   x2   /   y2   /
HRSOut $72, $00,$F0, $00,$DE, $00,$F8, $00,$E6, (L5[0]+L6[0])/2
DelayMS 10
HRSOut $72, $00,$F8, $00,$DE, $01,$00, $00,$E6,
(L5[0]+L5[1]+L6[0]+L6[1])/4
DelayMS 10
HRSOut $72, $01,$00, $00,$DE, $01,$08, $00,$E6, (L5[1]+L6[1])/2
DelayMS 10
HRSOut $72, $01,$08, $00,$DE, $01,$10, $00,$E6,
(L5[1]+L5[2]+L6[1]+L6[2])/4
DelayMS 10
HRSOut $72, $01,$10, $00,$DE, $01,$18, $00,$E6, (L5[2]+L6[2])/2
DelayMS 10

```

```

    HRSOut $72, $01,$18, $00,$DE, $01,$20, $00,$E6,
(L5[2]+L5[3]+L6[2]+L6[3])/4
    DelayMS 10
    HRSOut $72, $01,$20, $00,$DE, $01,$28, $00,$E6, (L5[3]+L6[3])/2
    DelayMS 10
    HRSOut $72, $01,$28, $00,$DE, $01,$30, $00,$E6,
(L5[3]+L5[4]+L6[3]+L6[4])/4
    DelayMS 10
    HRSOut $72, $01,$30, $00,$DE, $01,$38, $00,$E6, (L5[4]+L6[4])/2
    DelayMS 10
    HRSOut $72, $01,$38, $00,$DE, $01,$40, $00,$E6,
(L5[4]+L5[5]+L6[4]+L6[5])/4
    DelayMS 10
    HRSOut $72, $01,$40, $00,$DE, $01,$48, $00,$E6, (L5[5]+L6[5])/2
    DelayMS 10
    HRSOut $72, $01,$48, $00,$DE, $01,$50, $00,$E6,
(L5[4]+L5[5]+L6[4]+L6[5])/4
    DelayMS 10
    HRSOut $72, $01,$50, $00,$DE, $01,$58, $00,$E6, (L5[6]+L6[6])/2
    DelayMS 10
    HRSOut $72, $01,$58, $00,$DE, $01,$60, $00,$E6,
(L5[6]+L5[7]+L6[6]+L6[7])/4
    DelayMS 10
    HRSOut $72, $01,$60, $00,$DE, $01,$68, $00,$E6, (L5[7]+L6[7])/2
    DelayMS 10

```

```

'*****
'*          L6[0]~L7[7]          *
'*****
    HRSOut $70,$00  'rend le carre dessiné plein
    DelayMS 100
    '          /   x1   /   y1   /   x2   /   y2   /
    HRSOut $72, $00,$F0, $00,$EE, $00,$F8, $00,$F6, (L6[0]+L7[0])/2
    DelayMS 10
    HRSOut $72, $00,$F8, $00,$EE, $01,$00, $00,$F6,
(L6[0]+L6[1]+L7[0]+L7[1])/4
    DelayMS 10
    HRSOut $72, $01,$00, $00,$EE, $01,$08, $00,$F6, (L6[1]+L7[1])/2
    DelayMS 10
    HRSOut $72, $01,$08, $00,$EE, $01,$10, $00,$F6,
(L6[1]+L6[2]+L7[1]+L7[2])/4
    DelayMS 10
    HRSOut $72, $01,$10, $00,$EE, $01,$18, $00,$F6, (L6[2]+L7[2])/2
    DelayMS 10
    HRSOut $72, $01,$18, $00,$EE, $01,$20, $00,$F6,
(L6[2]+L6[3]+L7[2]+L7[3])/4
    DelayMS 10
    HRSOut $72, $01,$20, $00,$EE, $01,$28, $00,$F6, (L6[3]+L7[3])/2
    DelayMS 10
    HRSOut $72, $01,$28, $00,$EE, $01,$30, $00,$F6,
(L6[3]+L6[4]+L7[3]+L7[4])/4
    DelayMS 10
    HRSOut $72, $01,$30, $00,$EE, $01,$38, $00,$F6, (L6[4]+L7[4])/2
    DelayMS 10
    HRSOut $72, $01,$38, $00,$EE, $01,$40, $00,$F6,
(L6[4]+L6[5]+L7[4]+L7[5])/4
    DelayMS 10
    HRSOut $72, $01,$40, $00,$EE, $01,$48, $00,$F6, (L6[5]+L7[5])/2
    DelayMS 10
    HRSOut $72, $01,$48, $00,$EE, $01,$50, $00,$F6,
(L6[4]+L6[5]+L7[4]+L7[5])/4

```

```
DelayMS 10
HRSOut $72, $01,$50, $00,$EE, $01,$58, $00,$F6, (L6[6]+L7[6])/2
DelayMS 10
HRSOut $72, $01,$58, $00,$EE, $01,$60, $00,$F6,
(L6[6]+L6[7]+L7[6]+L7[7])/4
DelayMS 10
HRSOut $72, $01,$60, $00,$EE, $01,$68, $00,$F6, (L6[7]+L7[7])/2
DelayMS 10
```

GoTo Loop

End