

Consignes

Rédaction du compte-rendu de TP

TP 1 : Les Entrées / Sorties : gestion par scrutation

TP 2 : Les Entrées / Sorties : gestion par interruptions

TP 3 : Utilisation d'un Timer/Counter

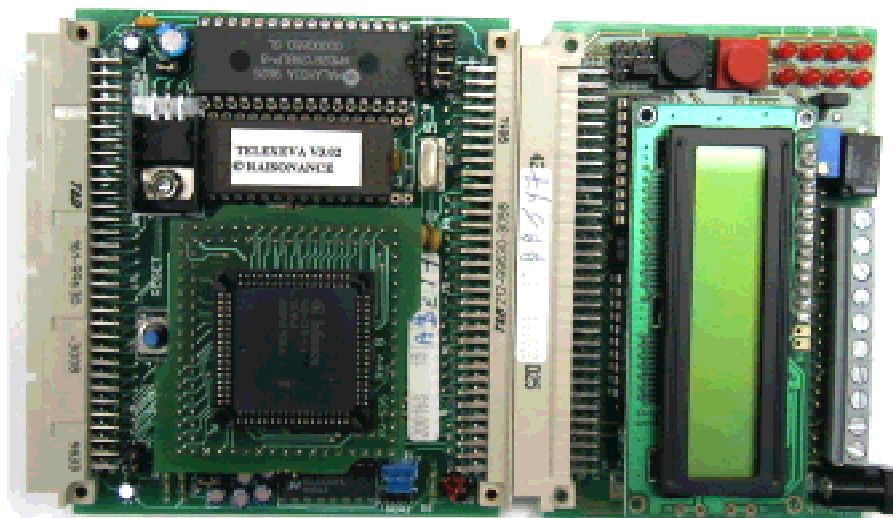
TP 4 : Convertisseur Analogique/Numérique

TP 5 : La liaison Série RS232

TP 6 : Examen de TP

Annexes :

- Carte XEVA
- Carte XEVADEMO
- Carte Feux Routiers
- Carte Multifonctions
- Fichier include « reg515.h »
- Table des codes ASCII



Consignes

Déroulement d'un TP

- Durée : 4 heures (avec une pause de 15mn).
- Programmation de 2 exercices **préalablement préparés** (synoptique, GNS, programme déjà écrit avant le TP...) et corrigés pendant la séance (à l'exception du TP 5).
- Un exercice de synthèse **noté** (rédaction d'un rapport) mettant en pratique les différentes notions acquises au cours du TP. Cette synthèse fait l'objet, lors de la séance suivante, d'une vérification du fonctionnement du programme (ex : le TP_1S est vérifié au début de la séance du TP2).

Matériels nécessaires aux Travaux Pratiques (TP)

- Les étudiants viendront en TP avec l'ensemble de leurs notes de cours et de TD, ainsi que les divers documents distribués (notice d'utilisation du SAB C515 notamment).
- Chaque étudiant est rendu responsable du matériel se trouvant dans la salle lors de sa présence. **Toute détérioration ou disparition sera réprimandée.**

Préparation

Un TP se prépare :

- étudier le sujet pour le comprendre,
- réviser les notions nécessaires à la réalisation du travail demandé,
- préparer les synoptiques et organigrammes relatifs aux différents exercices,
- concevoir le programme des exercices préparés en TD.

Les éléments nécessaires se trouvent soit dans les notes de cours et les exercices de TD, soit dans les documents distribués.

Rapport

Chaque exercice fera l'objet d'un rapport (cf rédaction du compte-rendu de TP). Ce rapport est d'une très grande importance dans la mesure où **il constituera (entre-autres) la base de vos notes à utiliser le jour des tests de Cours, TD et TP.**

Libre Service

La salle est utilisable en libre service, à condition de s'inscrire dans le cahier de présence.

Les horaires sont :

- entre 12h et 14h du lundi au vendredi
- le jeudi après-midi de 12h à 19h
- éventuellement quand la salle est libre

Il faut demander l'ouverture de la salle à un enseignant responsable. **Il faut impérativement signaler lorsque l'on quitte la salle en dernier et demander sa fermeture.**

Rédaction du compte-rendu de TP

Chaque exercice de TP fera l'objet d'un compte-rendu personnel, qui comprendra :

1. Le cahier des charges (CdC)
A partir de l'énoncé de l'exercice plus toute explication complémentaire donnée en TD ou TP
2. Le(s) principe(s) utilisé(s) pour répondre au CdC
Description et justification de la méthode utilisée et des techniques mises en œuvre.
3. Le synoptique de connections des E/S et les blocs fonctions des périphériques
4. Une étude, sous forme de GNS, de l'ensemble des fonctions prévues dans le programme pour répondre au cahier des charges
Note : le GNS de la fonction tempo (cf TP1) n'est pas demandé
5. Une conclusion
analyse des résultats obtenus lors du test du programme et perspective(s) d'évolution
6. Le listing du programme
Listing complet et commenté suivant le modèle fournit de tout le programme principal et de toutes les fonctions

Les rapports correspondant aux exercices corrigés (TP_X1 et TP_X2) ne sont normalement pas ramassés. Chaque étudiant devra être cependant en mesure de le fournir immédiatement au(x) professeur(s) sur simple demande.

La synthèse

L'exercice de synthèse donnera lieu à 2 notes :

- L'une sur le bon fonctionnement de la fonction demandée. L'exercice de synthèse d'un TP sera systématiquement vérifié au début du TP suivant.
- Le compte-rendu (on demandera de n'en rendre qu'un par binôme) correspondant au TP_1S sera ramassé et noté.

Les rapports correspondant aux TP_2S, TP_3S, TP4_S et TP52 devront être fournis le jour du TPtest. **Un ou deux de ces rapports seront tirés au sort, corrigés, notés et rendus.**

Conseil : Rédiger les synthèses au fur et à mesure !! NE PAS ATTENDRE LA VEILLE DU TP TEST...

Les notes correspondant aux TP de synthèse ramassés entreront dans le calcul de la note finale de TP avec celle du TP test.

IMPORTANT : le soin et la qualité de la rédaction entreront dans la notation des rapports.

TP 1 : Le microcontrôleur SAB C515
Les Entrées / Sorties

TP_11 : Les Entrées/Sorties : gestion par scrutation sur niveau

A l'aide du bouton poussoir (**P1.0**) de la carte « Feux Routiers », allumer ou éteindre la diode électroluminescente « Jour » (**P5.4**) selon la table de suivante (*fonction mémoire*) :

Entrée P1.0	Sortie P5.4
1	0
0 (appui)	1
1	1
0 (appui)	0
1	0

La détection se fera par scrutation et sur niveau. Afficher sur l'afficheur LCD de XEVADEMO selon le cas le message « ON » ou « OFF ».

Nota : ce 1^{er} TP sera guidé :

- création du projet TP_11
- options du projet
- compilation
- lancement du programme
- utilisation du debug

TP_12 : Les Entrées/Sorties : gestion par scrutation des fronts

On va maintenant travailler en détection de fronts.

A l'aide du même bouton poussoir, et en effectuant une détection sur front montant (FM), allumer successivement les leds jour/nuit/clign/off/jour/... (fonction *rot_gauche*).

Sur le bouton poussoir (P1.4), détecter un front descendant (FD) et inverser le sens de rotation : clign/nuit/jour/off/clign... (fonction *rot_droite*).

TP_1S : Synthèse : gestion du carrefour avec une base de temps logicielle.

Réaliser un programme gérant le carrefour suivant les 4 points indiqués ci-dessous.

- Sélection du mode de fonctionnement des feux (jour/nuit/clig/off) à l'aide du BP (**P1.0**).
- Au lancement du programme, le mode **Off** est activé.
- Mettre en place la gestion des feux **voitures et piétons pour les modes jour, nuit, clignotant et off** suivant les chronogrammes étudiés en TD.
- On ne demande pas, dans un premier temps, de gérer les boutons d'appel piétons et voitures.

La base de temps sera obtenue à partir d'une fonction calibrée en temps *tempo(unsigned int t)* dont la base de temps (BT) vaut 1 ms) – cette fonction n'est pas à développer (en principe, présente dans le fichier).

TP 2 : Les Entrées / Sorties
Gestion par interruptions

TP_21 : Gestion des lignes d'entrées par interruption

Cet exercice est une adaptation du TP_12 au fonctionnement par interruption.

- Initialiser l'entrée du bouton poussoir (**P1.0**) pour qu'elle génère une interruption à partir d'un front montant (FM). Chaque interruption détectée permettra d'allumer successivement les leds jour/nuit/clign/off/jour/nuit... dans la fonction d'interruption notée : **IRQ_P10()**.
- Initialiser l'entrée du bouton poussoir (**P1.4**) pour qu'elle génère une interruption à partir d'un front descendant (FD). Chaque interruption détectée permettra d'inverser le sens de rotation : droite ou gauche. Cette fonction d'interruption sera notée : **IRQ_P14()**
- Afficher sur l'afficheur LCD de XEVADEMO selon le cas le message :

TP_21	ou	TP_21
ROTATION DROITE		ROTATION GAUCHE

TP_22 : Gestion des priorités entre interruptions

On désire réaliser un « chenillard » (allumage successif de 4 diodes électroluminescentes (DEL) : **P5.4** à **P5.7** (jour/nuit/clign/off/jour/...) selon le cahier des charges suivant :

FM sur P1.0	: rotation à droite, BT = 1s
FM sur P1.1	: rotation à gauche, BT = 1s
FD sur P1.4	: MARCHE (rotation droite par défaut)
FD sur P1.5	: ARRET INTERRUPTION PRIORITAIRE !

- 1^{er} démarrage sur la diode **P5.7** : Off.
- Les redémarrages se font à partir de l'état courant : à partir du dernier arrêt.
- La base de temps (BT) sera logicielle (issue de la fonction *tempo(unsigned int t)* calibrée à 1 ms).

Afficher également le message décrivant l'état du système :

TP_22 ON	(ou OFF selon le cas)
ROTATION DROITE	(ou GAUCHE selon le cas)

TP_2S : Synthèse : gestion d'un carrefour

Etudier et écrire un programme permettant de gérer, par interruptions, les différents modes de fonctionnement des feux routiers d'un carrefour : Jour, Nuit, Clignotant, Off.

- Sélection des modes à la suite d'un FM sur l'entrée **P1.0**.
- **P1.0 est la plus prioritaire des interruptions.**
- Gestion des appels piétons sur la voie principale et voitures sur la voie secondaire pour le mode nuit sur les entrées **P1.4** et **P1.1**, respectivement.
- Utilisation de la base de temps logicielle.

TP 3 : Utilisation d'un Timer / Counter

TP_31 : Utilisation du Timer T2 en tant que base de temps matérielle : mode Timer.

Reprendre le sujet du TP_22 (chenillard avec BT de 1s). La base de temps se fera maintenant à partir du Timer T2 interne au μ C C515. Elle sera donc matérielle et non logicielle.

- Initialiser le Timer/counter T2 en **mode Timer : auto-reload** avec **une base de temps compatible avec la durée d'une seconde**.
- Ce timer devra provoquer cycliquement (toutes les secondes) une interruption du process. La gestion du temps écoulé se fera dans cette routine **irq_T2()**.
- Les interruptions du Timer T2 et du Bouton Poussoir (**P1.5**) seront prioritaires sur toutes les autres.

TP_32 : Utilisation du Timer / Counter T2 en tant que compteur : mode Counter.

On désire commuter une ligne (**P5.0**) après comptage de **12** événements sur le capteur connecté à l'entrée **P1.7**. En d'autres termes :

- la sortie **P5.0** sera activée (allumée) après la détection de 12 FD sur l'entrée **P1.7** (**qui est couplée à P1.0**). Cette entrée permet de faire progresser la valeur du compteur T2 sur chaque front descendant de P1.7. Bien évidemment **cette entrée ne doit pas provoquer de rebonds**, ce qui nécessite obligatoirement un filtrage matériel : intégration par RC suivie d'une remise en forme par porte à trigger de Schmitt (voir annexe3).
- Afin de pouvoir relancer le comptage, un acquittement de la ligne de sortie **P5.0** est nécessaire. Il sera effectué à la suite d'un FD sur **P1.5**, également gérée par interruption. **Pendant la phase de comptage, cet acquittement ne devra pas agir.**

TP_3S : Gestion d'un Départ de Rallye et Chronométrage.

On réalisera un compte à rebours **sur 3 secondes** avec affichage sur un feu.

- Le départ du compte à rebours est donné par un FD sur **P1.0**. On a alors la séquence :

Temps Restant	Led allumée
3s	rouge
2s	rouge
1s	oranges
0s	vert

- A la fin du compte à rebours un chronomètre, précis à 10ms, réalisé à l'aide du **Timer T2** démarre. Ce chronomètre obéit au cahier des charges :
 - DEMARRAGE à la fin du compte à rebours
 - ARRET après FD sur **P1.4** et affichage du temps en ms
 - RESET du chronomètre après FM sur **P1.1**. **Le Reset n'est autorisé que si le chronomètre est à l'arrêt.**
- Faux départs : un FD sur **P1.5** simule un faux départ avec arrêt et remise à 3s du compte à rebours. **La gestion des faux départs est, bien entendu, désactivée si le chronomètre a démarré !**
- Afficher le décompte du compte à rebours, puis l'état du chronomètre et le temps chronométré en **ms** sur l'afficheur LCD.

TP 4 : Convertisseur Analogique/Numérique

TP_41 : Barre-Graphe à affichage digital 10 niveaux

A l'aide du convertisseur analogique/numérique du μC , réaliser un affichage à barre-graphe :

- le capteur analogique (simulé par un potentiomètre) est connecté à l'entrée **P6.2** du μC . La conversion analogique se fera en **mode continu**.
- l'affichage se fait sur la barrette de 10 diodes (LED) selon le principe :

$V < 0.5 \text{ V}$	0 led
$0.5 \leq V < 1 \text{ V}$	1 led allumée
$1 \leq V < 1.5 \text{ V}$	2 leds
$1.5 \leq V < 2 \text{ V}$	3 leds
$2 \leq V < 2.5 \text{ V}$	4 leds
$2.5 \leq V < 3 \text{ V}$	5 leds
$3 \leq V < 3.5 \text{ V}$	6 leds
$3.5 \leq V < 4 \text{ V}$	7 leds
$4 \leq V < 4.5 \text{ V}$	8 leds
$4.5 \leq V < 5 \text{ V}$	9 leds
$5 \leq V$	10 leds

- Les afficheurs sont connectés à deux registres 8 bits dont les adresses sont :
 - 0x8000** LED N°1 (lsb) à LED N°8
 - 0x9000** LED N°9 et LED N°10 (msb)
- DEMARRAGE du convertisseur à l'aide du BP **P1.0** sur **FD**, géré par **interruption**.
- ARRET à l'aide du BP **P1.1** sur **FM**, géré par **interruption**.

TP_42 : Voltmètre à affichage digital

On reprend l'exercice précédent, mais en réalisant maintenant un voltmètre à affichage numérique.

- Le résultat de la conversion analogique/numérique de l'entrée **P6.3** sera visualisé sur les 4 afficheurs 7 segments de la carte Multifonctions.

On rappelle que ces afficheurs doivent être programmés par multiplexage : P1.4 : AFF. N°1 (+/-1) ; P1.5 : AFF. N°2 ; P1.6 : AFF. N°3 ; P1.7 : AFF. N°4 (least significant digit). Le port 5 (P5.0..P5.7) étant utilisé pour l'information de transcodage (relié aux 8 segments des afficheurs) : P5.0 : a ; P5.1 : b ; P5.2 : c ; P5.3 : d ; P5.4 : e ; P5.5 : f ; P5.6 : g ; P5.7 : dp.

- La conversion se fera en « coup par coup » (pas en continu) : lancement logiciel sur le canal 3 (P6.3).
- DEPART de la mesure donné par un **FD** sur la ligne **P1.0**.
- ARRET du process à la suite d'un **FM** sur la ligne **P1.1**.

TP_4S : Capteur différentiel

On utilise le canal 2 (**P6.2**) et le canal 3 (**P6.3**) du convertisseur pour mesurer la différence de potentiel entre les deux voltmètres (**canal 2-canal3**).

- Afficher la différence sur les afficheurs 7 segments.
- Si celle-ci est supérieure à 1V ou inférieure à -1V, il y a alerte : faire clignoter les 10 leds du barre-graphe (ensemble) avec une base de temps de 1s.
- DEPART donné par un **FD** sur la ligne **P1.0**. ARRET donné par un **FD** sur la ligne **P1.1**.

TP 5 : La liaison série : RS232

TP_51 : Commande d'un chenillard via une liaison série

A l'aide de la liaison série RS232 du μ C SAB C515, on désire commander un chenillard à partir d'un terminal (émulé sur le même PC à l'aide du logiciel HyperTerminal). Au lancement du programme le menu ci-dessous sera transmis au terminal :

```
MENU DU CHENILLARD
1 : Arrêt
2 : Rotation à Droite
3 : Rotation à Gauche
Votre choix ? : 1
```

- Le choix est transmis à partir du clavier du PC au μ C. Par défaut, **au lancement du programme, le chenillard est à l'arrêt.**
- Seuls les caractères ASCII '1', '2', '3' seront pris en compte.
- Les rotations droite et gauche, **d'une période de 1s**, s'effectueront sur les 6 leds (vert, orange, rouge) à l'adresse 0x9000.
- Le **Timer T2** sera utilisé pour fournir la base de temps.

Pour la liaison série, on écrira les fonctions suivantes :

- `send_char_UART(char val)` : émission d'un caractère
- `send_string_UART(char *chaine)` : émission d'une chaîne de caractères
- `irq_UART(void)` : gestion de l'interruption de l'UART (réception et émission)

Paramètres de la liaison série : 9600 bauds, 8 bits, pas de parité, un bit de stop.

TP_52 : Voltmètre à affichage déporté

On désire maintenant communiquer entre deux cibles μ C par une liaison série RS232 point à point (utilisation d'un câble croisé). L'une des deux cibles sera utilisée comme « MAITRE » et l'autre comme « ESCLAVE ». Le rôle du Maître est d'effectuer une mesure de tension (sur **P6.2**) dont l'affichage du résultat se fera sur la cible Esclave. Au lancement du programme, s'assurer de la présence de la cible par des émissions successives de codes d'arrêt.

Protocole : Le Maître transmet à l'Esclave la trame : **:Start, Code1, Code2, Code3, Checksum**

Avec : Start	: 0xFF octet de début de trame
Code1	: code ASCII de la mesure : partie entière
Code2	: code ASCII de la mesure : dixièmes de volt
Code3	: code ASCII de la mesure : centièmes de volt
Checksum	: complément à 1 de la somme modulo 8 des 4 octets précédents.

Le rôle de l'esclave sera l'affichage de la mesure (cf TP 42) ainsi que l'acquittement du message reçu, par l'envoi au Maître du caractère ASCII '!' ou du caractère ASCII '?' en cas de problème de transmission.

Le BP connecté à la ligne **P1.0** sert à effectuer une conversion (mode monocoup) sur **P6.2**. Le BP **P1.1** sert à envoyer le résultat à l'esclave qui l'affiche alors sur les afficheurs 7 segments.

TP 6 : Examen de TP

Déroulement

- le TP Test aura une durée de 2 heures.
- Le but étant de valider les connaissances de chacun, vous ne serez plus par binôme !!
- Tous les documents (Cours, TD, TP) **propres à chaque étudiant** sont autorisés.

Le TP Test est basé sur le même principe que l'ensemble des TPs effectués. En fonction du cahier des charges (énoncé) distribué, il faudra :

1. Effectuer l'étude théorique en justifiant les choix et les techniques mises en œuvre (stratégie, synoptique, GNS...).
2. Ecrire et tester le programme sur la cible.
3. Rédiger un compte-rendu.

Notation

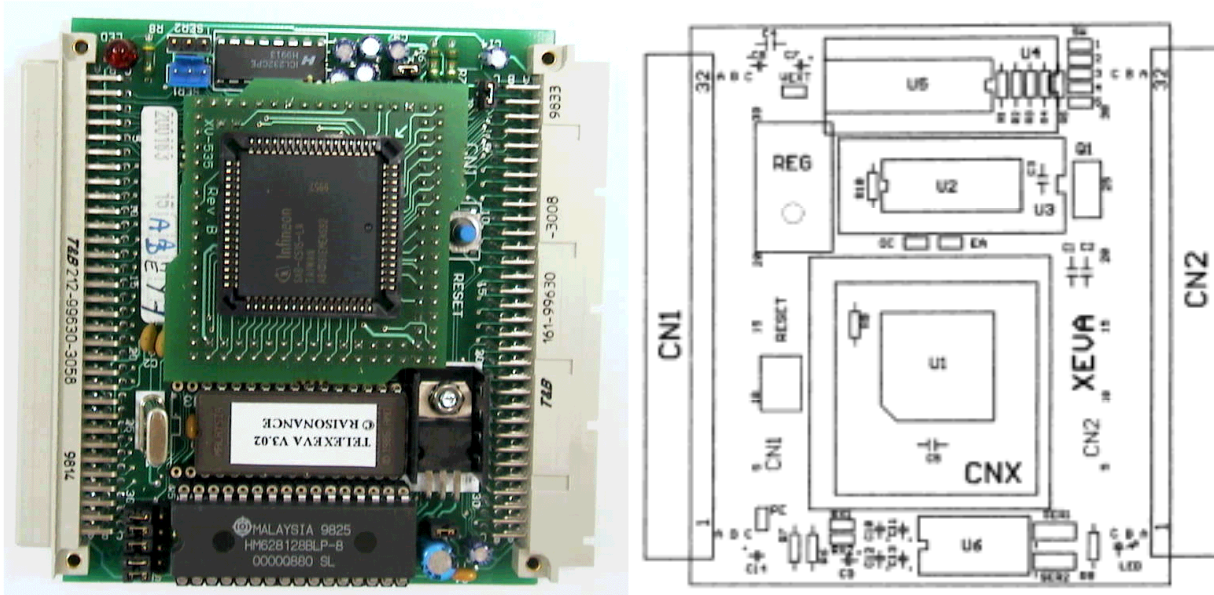
Comme pour les TPs de synthèse, la note sera scindée en 2 parties :

- Le fonctionnement du programme par rapport au cahier des charges,
- La pertinence du compte-rendu.

On demandera notamment de procéder par étapes lors de l'écriture du programme et surtout de faire vérifier le fonctionnement à chaque étape (initialisation, gestion des BP...). Ainsi, l'enseignant sera en mesure de fournir une note relative aux étapes validées (**si aucune étape n'est validée, aucun point n'est attribué**).

ANNEXE 1 :

CARTE XEVA



U4 : Espace RAM : 32 ou 128 ko, permet le téléchargement du programme exécutable pour mise au point.

U3 : Espace ROM : 128 ko, 64 ko ou 32 ko, accessible en lecture seulement (contient le Moniteur de mise au point, pourra contenir à terme le programme utilisateur exécutable).

U5 : Logique de décodage : circuit GAL (20V8). Décodage paramétrable par cavaliers ou par reprogrammation.

Q1 : Quartz pour le μ C de 12 MHz (interchangeable).

ALIMENTATION : soit par une alimentation externe 5V régulée (on isole alors le régulateur), soit par une alimentation externe redressée et en utilisant le régulateur. LED témoin. Consommation typique avec un 80C31 : 100 mA.

SER1, SER2 : ports séries avec adaptateur RS232-C. SER2 sera utilisé par le Moniteur de mise au point. La connectique est la suivante :

SER1	SER2
1 : RXD1	1 : RXD2
2 : TXD1	2 : TXD2
3 : GND	3 : GND

RESET : bouton reset sur la carte, repris sur les connecteurs CN1 et CN2 pour déport.

CN1 : connecteur mâle coudé 96 points DIN 41612 série C pour fond de panier Europe ou carte d'extension.

CN2 : connecteur femelle coudé 96 points DIN 41612 série R pour connecter une carte d'application (XEVADEMO).

CNX : connecteur d'adaptation des différents μ C de la série 80C51.

SWx, Jx, Rx : cavaliers permettant diverses configurations.

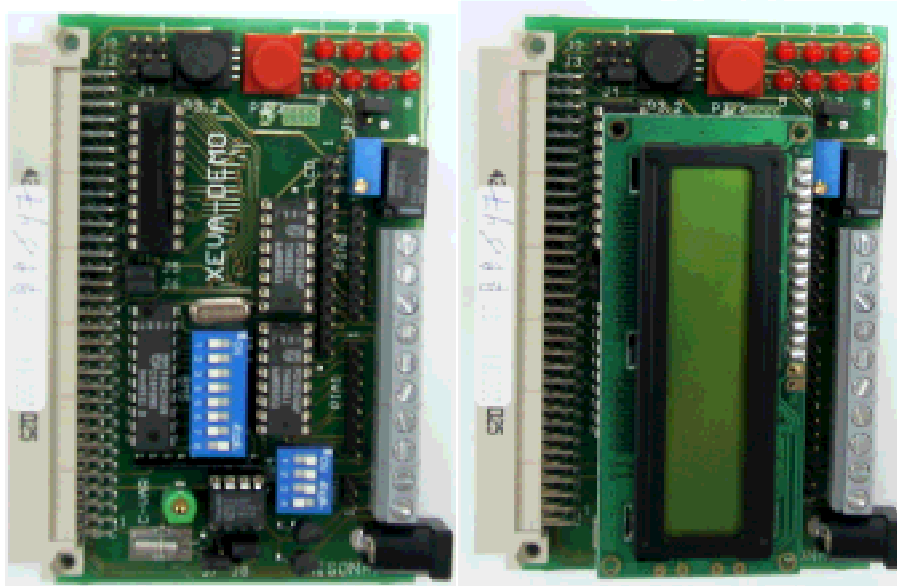


Photo de la carte XEVADEMO (avec et sans afficheur LDC)

La carte XEVADEMO est une carte d'extension pour XEVA. Elle supporte différents périphériques (afficheur LCD, diodes, boutons poussoirs (BP), périphérique I²C...). Elle comprend également une UART externe (SCC2691).

Là encore, on note la présence de divers cavaliers servant à paramétrer la carte (dans le cadre des TP, la carte est réglée une fois pour toute).

CN1 : connecteur 96 points mâle, à connecter sur CN2 de XEVA

UART externe (SCC2691) : Ce périphérique permet de libérer l'UART du microcontrôleur SAB C515 présent sur XEVA. Il permet d'établir la communication entre le PC et le microcontrôleur via la liaison série SER2 dans l'environnement RIDE utilisé en TP. La vitesse de la communication est réglée par les micro rupteurs SW8. On utilise la vitesse de 115 Kbauds.

REL : un relais OMRON G5V-1 est installé. Le relais est activé sur un état bas de P1.0.

Les sorties sont : **CO** : Commun
 NO : Contact normalement ouvert
 NC : Contact normalement fermé

Les caractéristiques du relais sont : Relais pour commutation de signaux
 Tension bobine 5Vcc
 Configuration des contacts : 1 RT
 Courant de commutation max : 1A
 Tension commutable max : 125 Vca / 60 Vcc

Nota : ce relais peut être remplacé par un relais de puissance. Déconnecté pour les TP.

P1.3-P1.7 : 5 lignes du port P1 d'entrées/sorties : P1.3 à P1.7.

GND : masse électrique

VCC : alimentation 5Vcc

LCD : Connecteur 14 points pour un module LCD (compatible HITACHI LM-052L) 2x16 caractères.

LED (1 à 8) : 8 leds accessibles en écriture. L'adresse standard de l'octet décrivant l'état des leds est **0xF000**. Elle dépend des équations du PLD PALCE16V8 (sur U1)

P3.2 : bouton poussoir relié à P3.2 (P3.2 = 0 sur action du BP)

P3.3 : bouton poussoir relié à P3.3 (P3.3 = 0 sur action du BP)

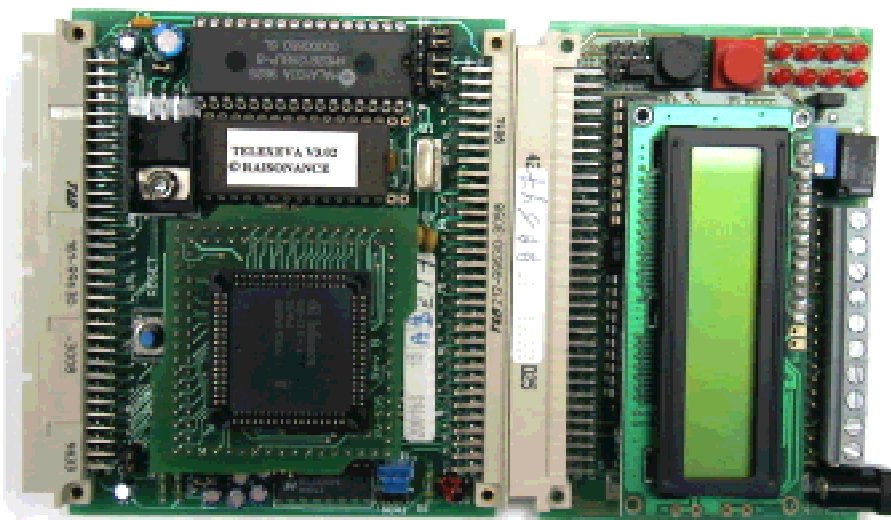
PIA0 : connecteur 8 points. Entrées/Sorties du périphérique I²C PCF8574A (sur U4). Adresse I²C : (A0 = A1 = A2 = 0)

PIA1 : connecteur 8 points. Entrées/Sorties du périphérique I²C PCF8574A (sur U5). Adresse I²C : (A0 = 1 ; A1 = A2 = 0)

AN1 : entrée analogique connectée à P5.1 d'un 80C552

Périphérique I²C DIL 8 : un PCF8583 (RAM/Horloge temps réel) est installé. Adresse I²C : (A0 = 0). La fréquence de l'horloge est réglable à l'aide d'un condensateur variable (C-VAR)

Capteur de température : référence LM 35 (National) relié à l'entrée P5.0 d'un 80C552. Echelle : 10 mV/°C de +2°C à + 120°C.

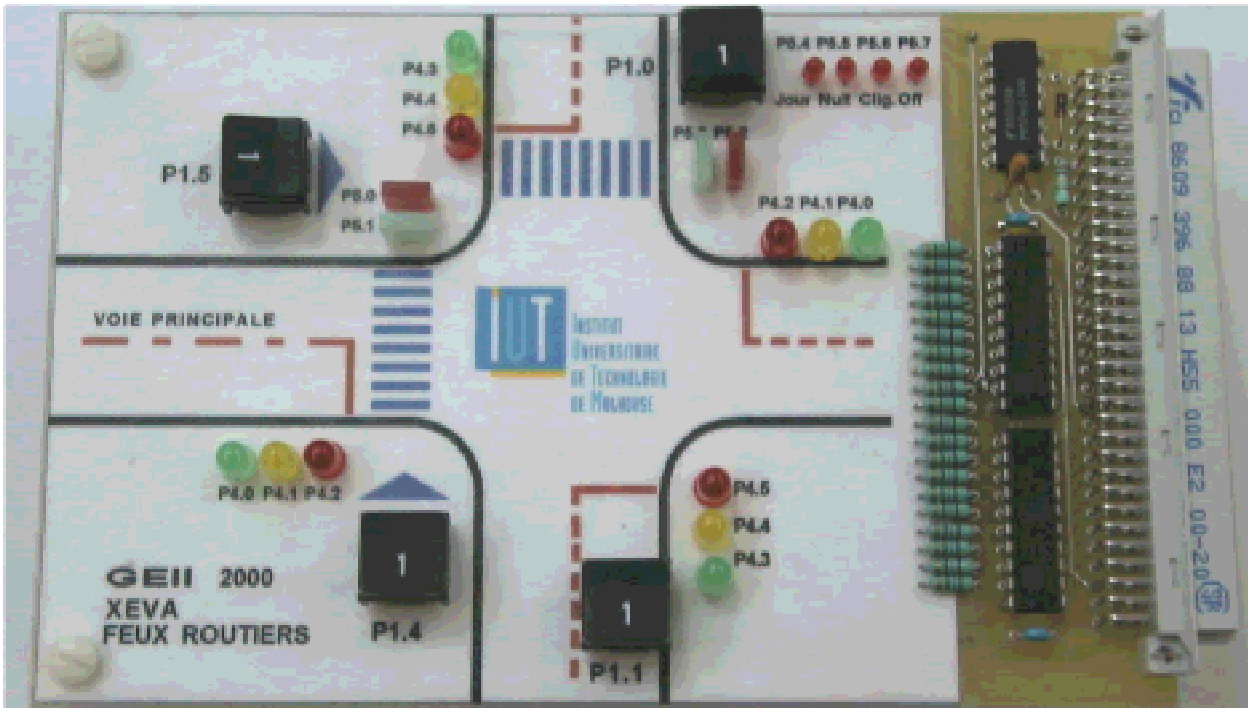


Photographie montrant XEVA et XEVADEMO assemblées.

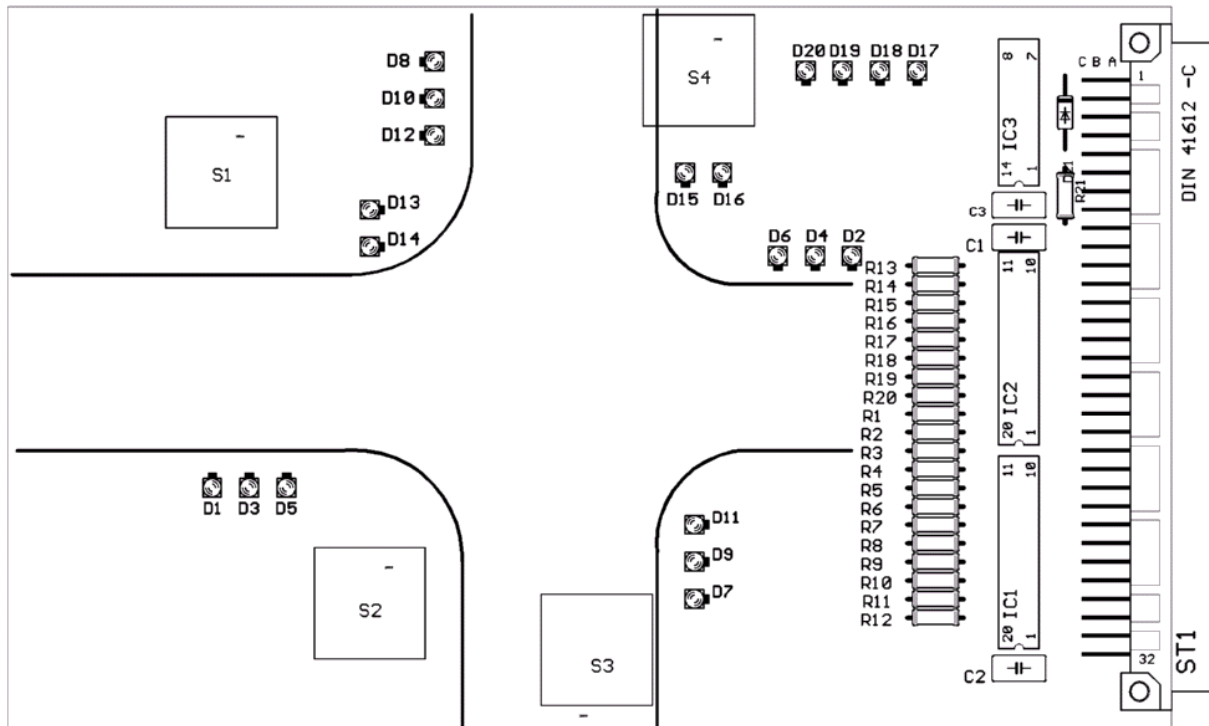
ANNEXE 3:

CARTE FEUX ROUTIERS

Photo :

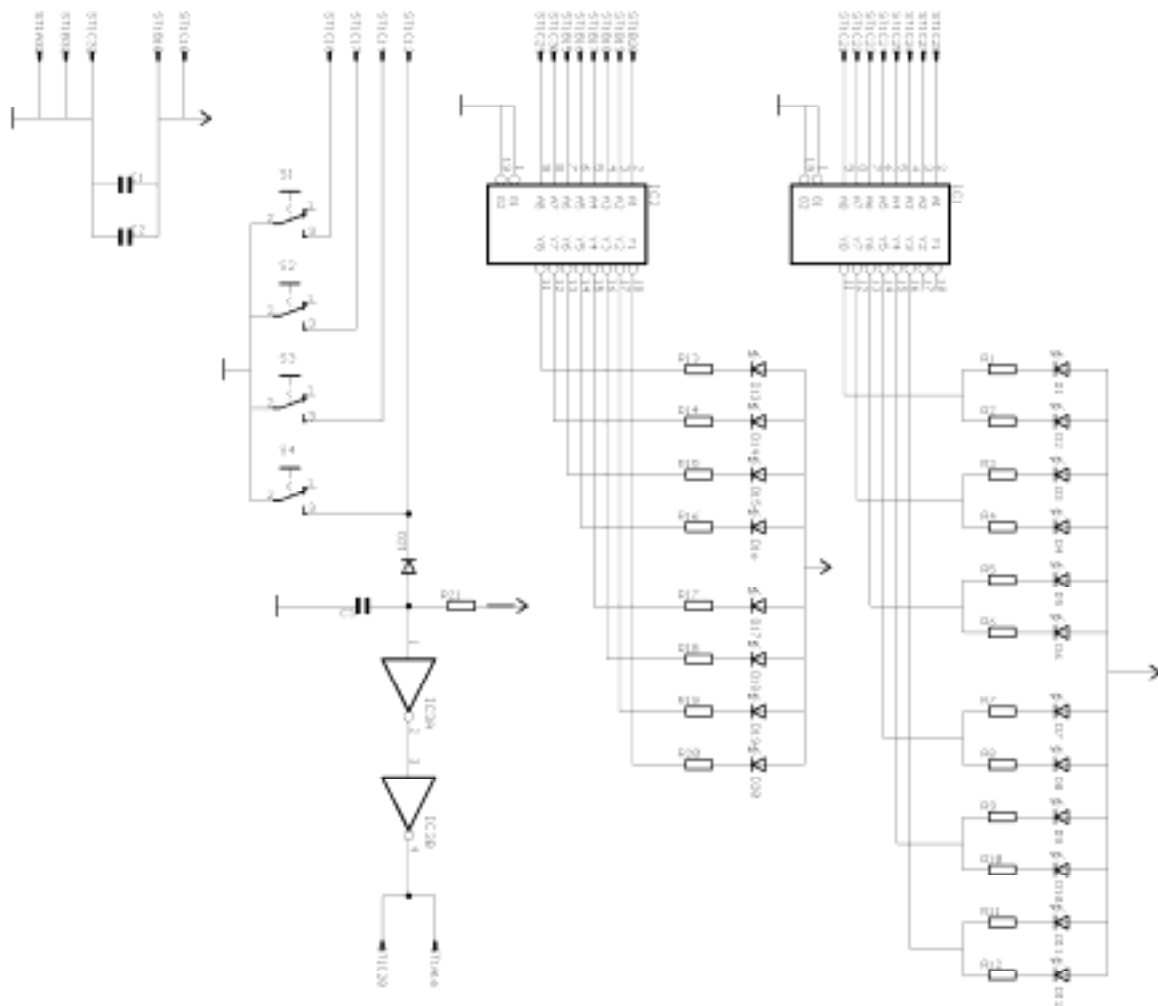


Plan d'implantation :



Carte « Feux Routiers »

schéma développé :



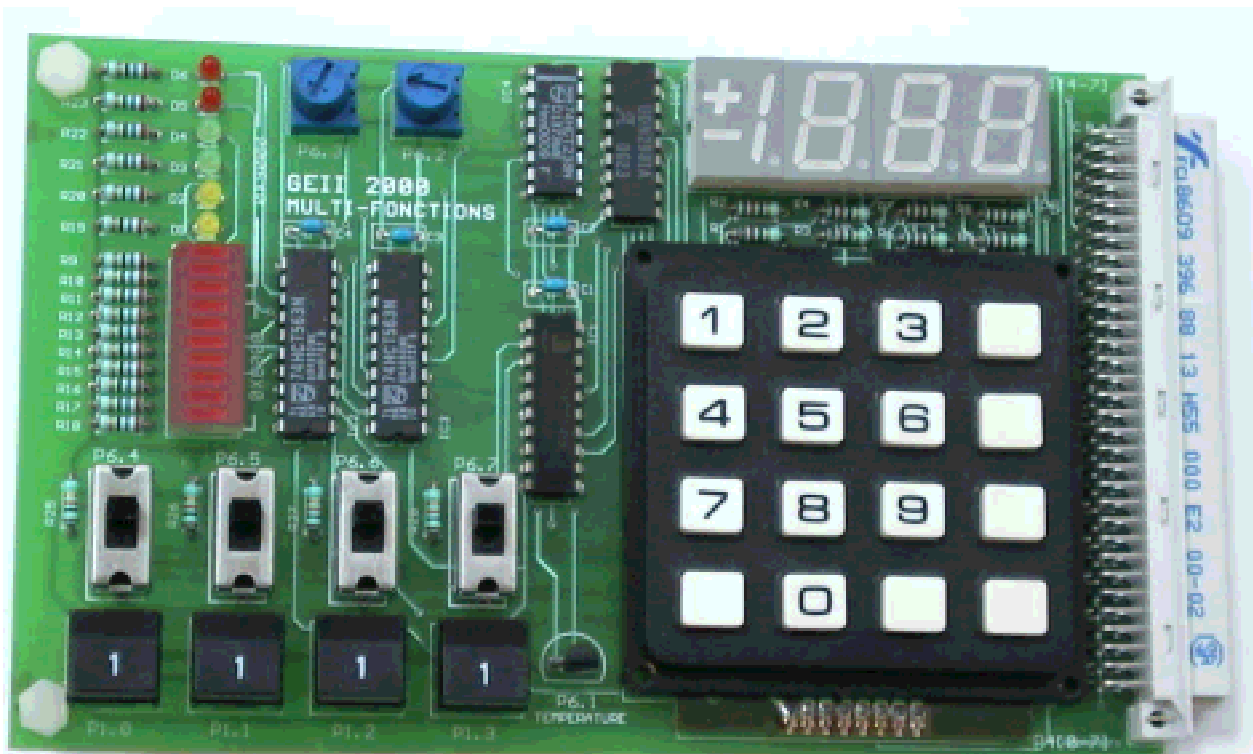
Liste des composants :

Repère	Désignation
ST1	Connecteur DIN 41612- C - 96 points
IC1, IC2	Drivers de bus 74 ALS 540
IC3	Trigger de Schmitt 74 LS 14
R1 à R20	Résistances 330 Ω _ W
R21	Résistance 220 KΩ _ W
C1, C2	Condensateurs de découplage 100 nF
C3	Condensateur Tantal 1 μF
D5, D6, D11, D12	LED rouge 5 mm ; I=10mA
D3, D4, D9, D10	LED orange 5 mm ; I=10mA
D1, D2, D7, D8	LED vertes 5 mm ; I=10mA
D14, D15	LED vertes plates 7 mm ; I=10mA
D13, D16	LED rouges plates 7 mm ; I=10mA
D21	Diode faible signaux 1 N 4148
S1, S2, S3, S4	Boutons - poussoirs à bascule

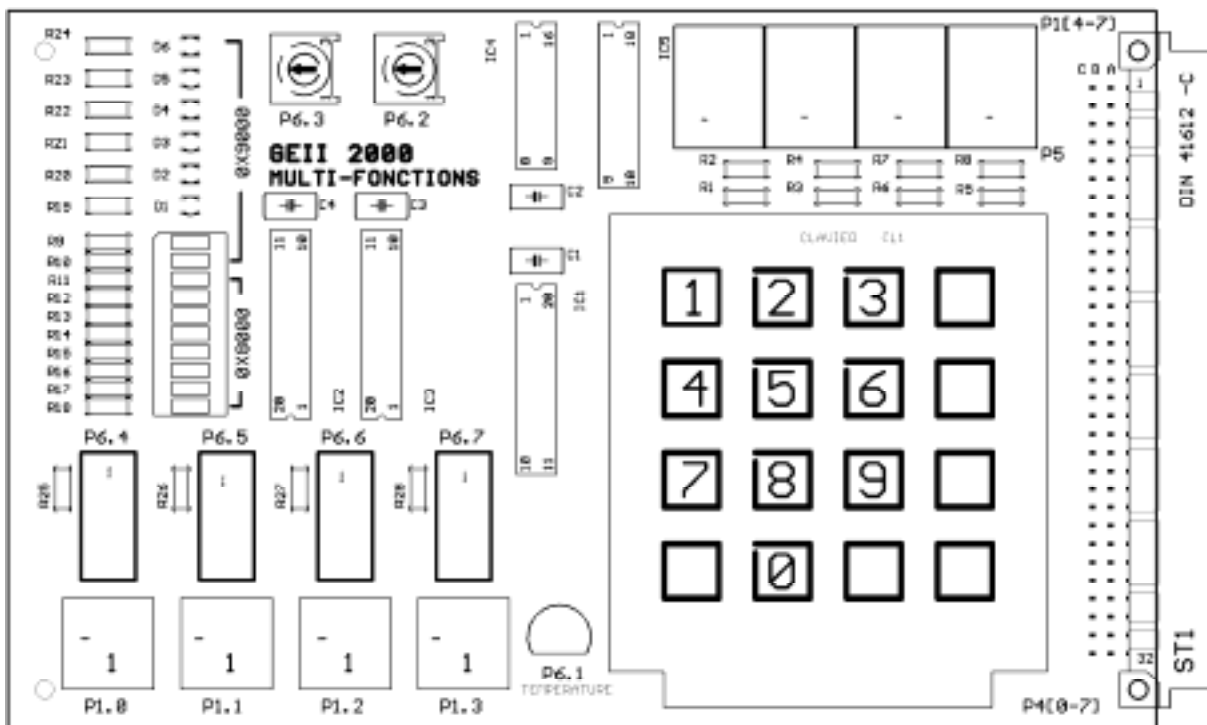
ANNEXE 4 :

CARTE MULTIFONCTIONS

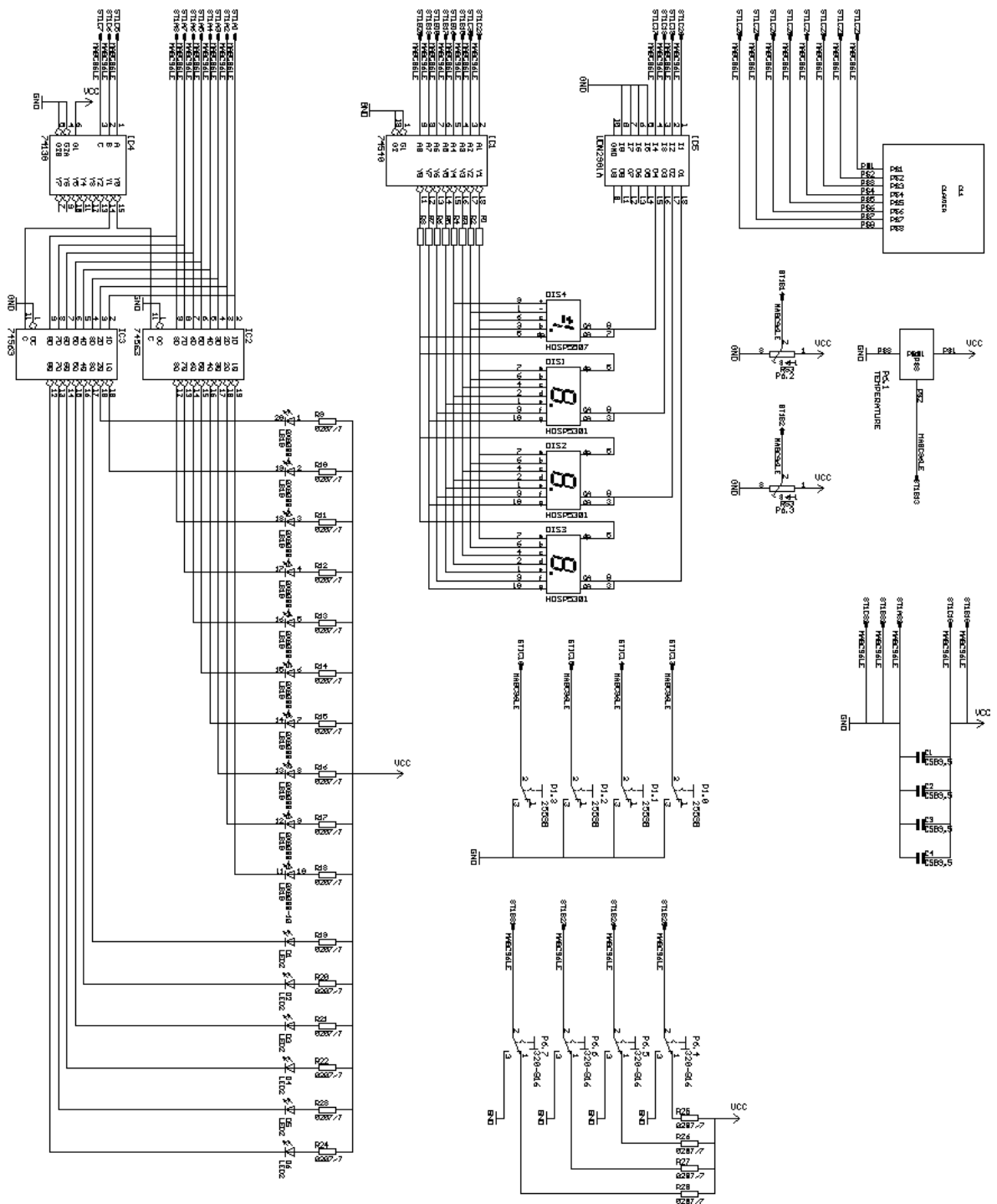
Photo :



Plan d'implantation :



Carte « MultiFonctions » schéma développé :



Repère	Désignation
ST1	Connecteur DIN 41612- C - 96 points
IC1	Drivers de bus 74 HCT 540
IC2, IC3	Registres 8 bits 74 HCT 563
IC4	Décodeur 3 -> 8 74 ACT 238
IC5	Drivers de puissance UDN 2981 A
DIS1, DIS2, DIS3	Afficheurs 7 segments Anodes Communes HDSP-5501
DIS4	Afficheur 7 segments Anodes Communes HDSP-5507
R1 à R8	Résistances 56 Ω _ W
R10 à R24	Résistances 270 Ω _ W
R25 à R28	Résistances 10 K Ω _ W
C1, C2, C3, C4	Condensateurs de découplage 100 nF
P6.1 Température	Capteur de température LM 35 DZ
P6.2, P6.3	Potentiomètres (Trimers) monotour 5 K Ω
P6.4, P6.5, P6.6, P6.7	Commutateurs série 50MP
P1.0, P1.1, P1.2, P1.3	Boutons - poussoirs à bascule
D1, D2	LED vertes 5 mm ; I=10mA
D3, D4	LED orange 5 mm ; I=10mA
D5, D6	LED rouge 5 mm ; I=10mA
0x8000	Barrettes de 10 LED's rouges LTA 1000 HR
Clavier CL1	Clavier 16 touches matriciel

ANNEXE 5 :

Le fichier include « reg515.h »

```

/*-----*/
/* Include file for 80515 SFR Definitions */
/* Copyright Raisonance SA, 1990-1996 */
/*-----*/

```

/* **BYTE Register** */

```

at 0x80 sfr P0 ;
at 0x90 sfr P1 ;
at 0xA0 sfr P2 ;
at 0xB0 sfr P3 ;
at 0xE8 sfr P4 ;
at 0xF8 sfr P5 ;
at 0xD0 sfr PSW ;
at 0xE0 sfr ACC ;
at 0xF0 sfr B ;
at 0x81 sfr SP ;
at 0x82 sfr DPL ;
at 0x83 sfr DPH ;
at 0x87 sfr PCON ;
at 0x88 sfr TCON ;
at 0x89 sfr TMOD ;
at 0x8A sfr TL0 ;
at 0x8B sfr TL1 ;
at 0x8C sfr TH0 ;
at 0x8D sfr TH1 ;
at 0x98 sfr SCON ;
at 0x99 sfr SBUF ;

at 0xA8 sfr IEN0 ;
at 0xB8 sfr IEN1 ;
at 0xA9 sfr IP0 ;
at 0xB9 sfr IP1 ;
at 0xC0 sfr IRCON ;
at 0xC1 sfr CCEN ;
at 0xC2 sfr CCL1 ;
at 0xC3 sfr CCH1 ;
at 0xC4 sfr CCL2 ;
at 0xC5 sfr CCH2 ;
at 0xC6 sfr CCL3 ;
at 0xC7 sfr CCH3 ;
at 0xC8 sfr T2CON ;
at 0xCA sfr CRCL ;
at 0xCB sfr CRCH ;
at 0xCC sfr TL2 ;
at 0xCD sfr TH2 ;
at 0xD8 sfr ADCON ;
at 0xD9 sfr ADDAT ;
at 0xDA sfr DAPR ;

```

/* **BIT Register** */

```

/* PSW */
at 0xD7 sbit CY ;
at 0xD6 sbit AC ;
at 0xD5 sbit F0 ;
at 0xD4 sbit RS1 ;
at 0xD3 sbit RS0 ;
at 0xD2 sbit OV ;
at 0xD1 sbit F1 ;
at 0xD0 sbit P ;

/* TCON */
at 0x8F sbit TF1 ;
at 0x8E sbit TR1 ;
at 0x8D sbit TF0 ;
at 0x8C sbit TR0 ;
at 0x8B sbit IE1 ;
at 0x8A sbit IT1 ;
at 0x89 sbit IE0 ;
at 0x88 sbit IT0 ;

/* IEN0 */
at 0xAF sbit EAL ;
at 0xAE sbit WDT ;
at 0xAD sbit ET2 ;
at 0xAC sbit ES ;
at 0xAB sbit ET1 ;
at 0xAA sbit EX1 ;
at 0xA9 sbit ET0 ;
at 0xA8 sbit EX0 ;

/* IEN1 */
at 0xBF sbit EXEN2 ;
at 0xBE sbit SWDT ;
at 0xBD sbit EX6 ;
at 0xBC sbit EX5 ;
at 0xBB sbit EX4 ;
at 0xBA sbit EX3 ;
at 0xB9 sbit EX2 ;
at 0xB8 sbit EADC ;

```

/* **P3** */

```

at 0xB7 sbit RD ;
at 0xB6 sbit WR ;
at 0xB5 sbit T1 ;
at 0xB4 sbit T0 ;
at 0xB3 sbit INT1 ;
at 0xB2 sbit INTO ;
at 0xB1 sbit TXD ;
at 0xB0 sbit RXD ;

/* SCON */
at 0x9F sbit SM0 ;
at 0x9E sbit SM1 ;
at 0x9D sbit SM2 ;
at 0x9C sbit REN ;
at 0x9B sbit TB8 ;
at 0x9A sbit RB8 ;
at 0x99 sbit TI ;
at 0x98 sbit RI ;

/* T2CON */
at 0xCF sbit T2PS ;
at 0xCE sbit I3FR ;
at 0xCD sbit I2FR ;
at 0xCC sbit T2R1 ;
at 0xCB sbit T2R0 ;
at 0xCA sbit T2CM ;
at 0xC9 sbit T2I1 ;
at 0xC8 sbit T2I0 ;

/* ADCON */
at 0xDF sbit BD ;
at 0xDE sbit CLK ;
at 0xDC sbit BSY ;
at 0xDB sbit ADM ;
at 0xDA sbit MX2 ;
at 0xD9 sbit MX1 ;
at 0xD8 sbit MX0 ;

/* IRCON */
at 0xC7 sbit EXF2 ;
at 0xC6 sbit TF2 ;
at 0xC5 sbit IEX6 ;
at 0xC4 sbit IEX5 ;
at 0xC3 sbit IEX4 ;
at 0xC2 sbit IEX3 ;
at 0xC1 sbit IEX2 ;
at 0xC0 sbit IADC

```

ANNEXE 6 :

Table des codes ASCII standard

Decimal	Octal	Hex	Binary	Value	Decimal	Octal	Hex	Binary	Value
000	000	000	00000000	NUL (Null char.)	064	100	040	01000000	@
001	001	001	00000001	SOH (Start of Header)	065	101	041	01000001	A
002	002	002	00000010	STX (Start of Text)	066	102	042	01000010	B
003	003	003	00000011	ETX (End of Text)	067	103	043	01000011	C
004	004	004	00000100	EOT (End of Transmission)	068	104	044	01000100	D
005	005	005	00000101	ENQ (Enquiry)	069	105	045	01000101	E
006	006	006	00000110	ACK(Acknowledgment)	070	106	046	01000110	F
007	007	007	00000111	BEL (Bell)	071	107	047	01000111	G
008	010	008	00001000	BS (Backspace)	072	110	048	01001000	H
009	011	009	00001001	HT (Horizontal Tab)	073	111	049	01001001	I
010	012	00A	00001010	LF (Line Feed)	074	112	04A	01001010	J
011	013	00B	00001011	VT (Vertical Tab)	075	113	04B	01001011	K
012	014	00C	00001100	FF (Form Feed)	076	114	04C	01001100	L
013	015	00D	00001101	CR (Carriage Return)	077	115	04D	01001101	M
014	016	00E	00001110	SO (Serial In)	078	116	04E	01001110	N
015	017	00F	00001111	SI (Serial Out)	079	117	04F	01001111	O
016	020	010	00010000	DLE (Data Link Escape)	080	120	050	01010000	P
017	021	011	00010001	DC1 (XON) (Device Control 1)	081	121	051	01010001	Q
018	022	012	00010010	DC2 (Device Control 2)	082	122	052	01010010	R
019	023	013	00010011	DC3 (XOFF)(Device Control 3)	083	123	053	01010011	S
020	024	014	00010100	DC4 (Device Control 4)	084	124	054	01010100	T
021	025	015	00010101	NAK (Negative Ack)	085	125	055	01010101	U
022	026	016	00010110	SYN (Synchronous Idle)	086	126	056	01010110	V
023	027	017	00010111	ETB (End of Trans. Block)	087	127	057	01011111	W
024	030	018	00011000	CAN (Cancel)	088	130	058	01011000	X
025	031	019	00011001	EM	089	131	059	01011001	Y
026	032	01A	00011010	SUB	090	132	05A	01011010	Z
027	033	01B	00011011	ESC (Escape)	091	133	05B	01011011	[
028	034	01C	00011100	FS(File Separator)	092	134	05C	01011100	\
029	035	01D	00011101	GS	093	135	05D	01011101]
030	036	01E	00011110	RS (Request to Send)	094	136	05E	01011110	^
031	037	01F	00011111	US	095	137	05F	01011111	_
032	040	020	00100000	SP (Space)	096	140	060	01100000	`
033	041	021	00100001	!	097	141	061	01100001	a
034	042	022	00100010	"	098	142	062	01100010	b
035	043	023	00100011	#	099	143	063	01100011	c
036	044	024	00100100	\$	100	144	064	01100100	d
037	045	025	00100101	%	101	145	065	01100101	e
038	046	026	00100110	&	102	146	066	01100110	f
039	047	027	00100111	'	103	147	067	01100111	g
040	050	028	00101000	(104	150	068	01101000	h
041	051	029	00101001)	105	151	069	01101001	i
042	052	02A	00101010	*	106	152	06A	01101010	j
043	053	02B	00101011	+	107	153	06B	01101011	k
044	054	02C	00101100	,	108	154	06C	01101100	l
045	055	02D	00101101	-	109	155	06D	01101101	m
046	056	02E	00101110	.	110	156	06E	01101110	n
047	057	02F	00101111	/	111	157	06F	01101111	o
048	060	030	00110000	0	112	160	070	01110000	p
049	061	031	00110001	1	113	161	071	01110001	q
050	062	032	00110010	2	114	162	072	01110010	r
051	063	033	00110011	3	115	163	073	01110011	s
052	064	034	00110100	4	116	164	074	01110100	t
053	065	035	00110101	5	117	165	075	01110101	u
054	066	036	00110110	6	118	166	076	01110110	v
055	067	037	00110111	7	119	167	077	01110111	w
056	070	038	00111000	8	120	170	078	01111000	x
057	071	039	00111001	9	121	171	079	01111001	y
058	072	03A	00111010	:	122	172	07A	01111010	z
059	073	03B	00111011	;	123	173	07B	01111011	{
060	074	03C	00111100	<	124	174	07C	01111100	
061	075	03D	00111101	=	125	175	07D	01111101	}
062	076	03E	00111110	>	126	176	07E	01111110	~
063	077	03F	00111111	?	127	177	07F	01111111	DEL

