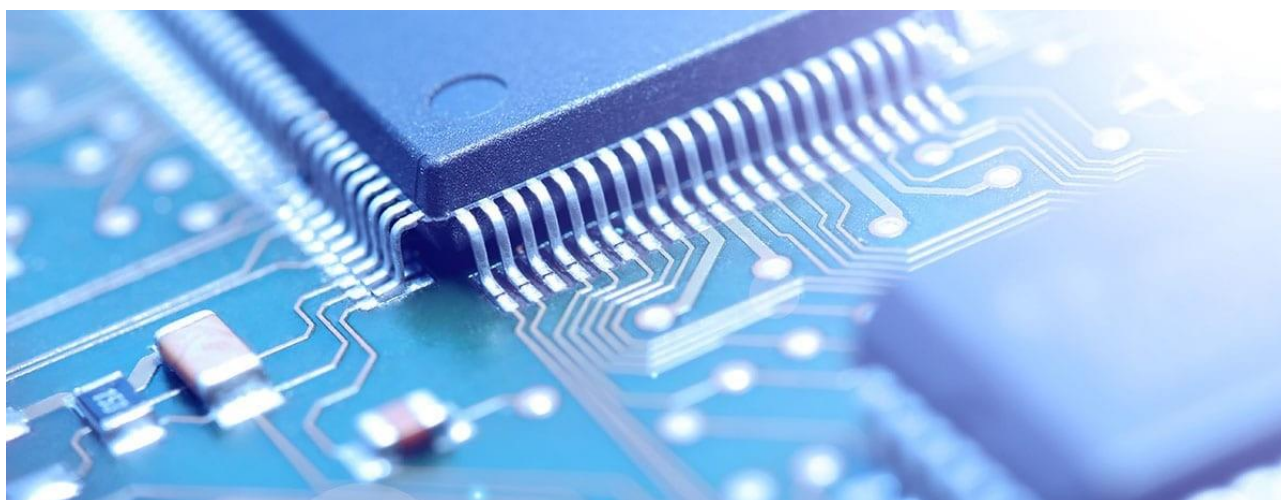

TP MICROCONTROLEUR

TP2 – Communication sur un écran LCD et gestion de la mémoire EEPROM



Par Damien DUBOIS & Marion ESCOUTELOUP

CESI S3E – P1G1 – FIPA26

29 juin 2022

TABLE DES MATIERES

I.	INTRODUCTION	3
II.	Pilotage du clavier et de l'écran LCD	4
1.	Préparation : récupération des adresses	4
2.	Ecriture sur l'écran LCD.....	6
3.	Utilisation du clavier pour écrire sur l'écran LCD	7
III.	Développement mémoire EEPROM	8
1.	Principe et utilisation d'une mémoire EEPROM.....	8
2.	Mémorisation de l'affichage.....	8
IV.	CONCLUSION	9
V.	ANNEXE 1 : Programme final.....	10
1.	Main.c : fichier source du programme globale	10
2.	LCD.c : fichier regroupant les fonctions propres à l'écran LCD	11
3.	Clavier.c : fonctions propres à l'utilisation d'un clavier numérique	12
4.	Eeprom.c : fonctions pour la mémorisation d'élément dans la mémoire EEPROM.....	13

TABLE DES FIGURES

<i>Figure 1 : Capture d'écran -- Architecture du projet</i>	<i>3</i>
<i>Figure 2 : Schéma du branchement du multiplexeur 74HCT138.....</i>	<i>4</i>
<i>Figure 3 : Tableau description de la datasheet du multiplexeur 74HCT138.....</i>	<i>4</i>
<i>Figure 4 : Décryptage des trames pour adressage au clavier et au LCD</i>	<i>5</i>
<i>Figure 5 : Table de conversion ASCII</i>	<i>7</i>

I. INTRODUCTION

L'objectif de ce TP2 est de réussir à communiquer avec la mallette d'essai mis à notre disposition. Lors de ce TP, nous nous sommes attardés sur le développement des éléments nécessaire pour le contrôle du clavier ainsi que l'affichage sur l'écran LCD. Dans une seconde partie, nous attarderons sur la mémorisation d'éléments dans une mémoire EEPROM.

Pour réaliser au mieux ce TP, nous nous configurons sur un microcontrôleur de type PIC18F87K22.

AJOUTER ICI CAPTURE ARCHITECTURE DU PROJECT AVEC FICHIER .h ET .c

Figure 1 : Capture d'écran -- Architecture du projet

Pour rappel, nous connaissons les paramètres suivant :

- far unsigned char CLAVIER @ 0x180000 ;
- far unsigned char LCD_DATA @ 0x1A0002 ;
- far unsigned char LCD_FUNC @ 0x1A0000 ;

II. Pilotage du clavier et de l'écran LCD

1. Préparation : récupération des adresses

Afin de pouvoir communiquer avec les éléments, il nous est important d'être capable de retrouver leurs adresses. Pour ce faire, dans notre cas, nous pouvons le retrouver grâce à deux documents :

- Le schéma électrique de la mallette
- Datasheet constructeur du multiplexeur

Nous avons ainsi les informations suivantes :

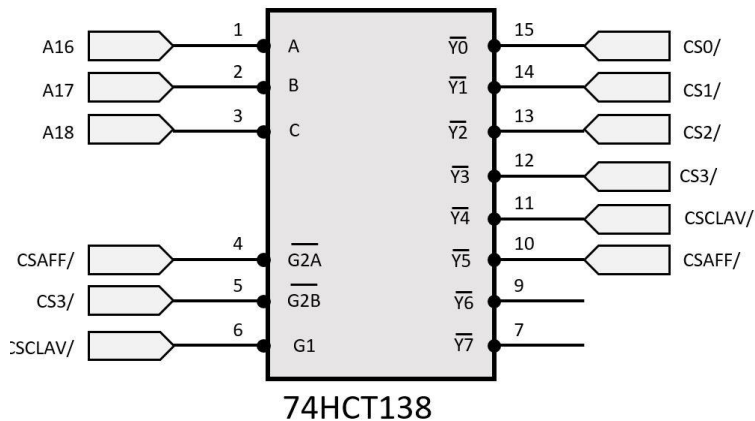


Figure 2 : Schéma du branchement du multiplexeur 74HCT138

Table 2. Pin description

Symbol	Pin	Description
A0, A1, A2	1, 2, 3	address input A0, A1, A2
$\bar{E}1, \bar{E}2$	4, 5	enable input $\bar{E}1, \bar{E}2$ (active LOW)
E3	6	enable input E3 (active HIGH)
$\bar{Y}0, \bar{Y}1, \bar{Y}2, \bar{Y}3, \bar{Y}4, \bar{Y}5, \bar{Y}6, \bar{Y}7$	15, 14, 13, 12, 11, 10, 9, 7	output $\bar{Y}0, \bar{Y}1, \bar{Y}2, \bar{Y}3, \bar{Y}4, \bar{Y}5, \bar{Y}6, \bar{Y}7$ (active LOW)
GND	8	ground (0 V)
V _{CC}	16	positive supply voltage

Table 3. Function table^[4]

Control			Input			Output							
$\bar{E}1$	$\bar{E}2$	E3	A2	A1	A0	$\bar{Y}7$	$\bar{Y}6$	$\bar{Y}5$	$\bar{Y}4$	$\bar{Y}3$	$\bar{Y}2$	$\bar{Y}1$	$\bar{Y}0$
H	X	X	X	X	X	H	H	H	H	H	H	H	H
X	H	X											
X	X	L											
L	L	H	L	L	L	H	H	H	H	H	H	H	L
			L	L	H	H	H	H	H	H	H	L	H
			L	H	L	H	H	H	H	H	L	H	H
			L	H	H	H	H	H	H	L	H	H	H
			H	L	L	H	H	H	L	H	H	H	H
			H	L	H	H	H	L	H	H	H	H	H
			H	H	L	H	L	H	H	H	H	H	H
			H	H	H	L	H	H	H	H	H	H	H

Figure 3 : Tableau description de la datasheet du multiplexeur 74HCT138

TP2 – Communication sur un écran LCD et gestion de la mémoire EEPROM

Nous pouvons donc grâce à ces données retrouver la trame fonctionnelle suivante :

ADRESSAGE CLAVIER

A19	A18	A17	A16	A15	A14	A13	A12	A11	A10	A09	A08	A07	A06	A05	A04	A03	A02	A01	A00
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bus Adresse **Bus de données**

ADRESSAGE LCD

A19	A18	A17	A16	A15	A14	A13	A12	A11	A10	A09	A08	A07	A06	A05	A04	A03	A02	A01	A00
1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bus Adresse **Bus de données**

Figure 4 : Décryptage des trames pour adressage au clavier et au LCD

2. Ecriture sur l'écran LCD

Maintenant que nous avons l'ensemble des informations, nous pouvons lancer le développement pour pouvoir afficher sur l'écran LCD ce que l'on écrit via le clavier. Pour cela, il ne faut pas oublier d'utiliser les conversions de caractères ASCII. Sans cela, l'affichage ne sera pas juste. Cette conversion sera intégrée directement dans les fonctions propres à l'utilisation du clavier.

Pour pouvoir utiliser correctement l'écran, nous avons besoin de développer deux fonctions. La première, appelé *init_aff_lcd()* permet d'initialiser l'écran. Il lance toutes les interactions possibles avec cet écran afin de vérifier que l'ensemble fonctionne bien correctement avant l'application de notre besoin.

Dans un second temps, nous avons programmé la fonction *AffCarac()*. Cette fonction, elle, permet de venir directement afficher sur l'écran LCD un ou plusieurs caractères grâce un paramètre d'entrée dans la fonction.

3. Utilisation du clavier pour écrire sur l'écran LCD

ASCII TABLE

Decimal	Hexadecimal	Binary	Octal	Char	Decimal	Hexadecimal	Binary	Octal	Char	Decimal	Hexadecimal	Binary	Octal	Char
0	0	0	0	[NULL]	48	30	110000	60	0	96	60	1100000	140	.
1	1	1	1	[START OF HEADING]	49	31	110001	61	1	97	61	1100001	141	a
2	2	10	2	[START OF TEXT]	50	32	110010	62	2	98	62	1100010	142	b
3	3	11	3	[END OF TEXT]	51	33	110011	63	3	99	63	1100011	143	c
4	4	100	4	[END OF TRANSMISSION]	52	34	110100	64	4	100	64	1100100	144	d
5	5	101	5	[ENQUIRY]	53	35	110101	65	5	101	65	1100101	145	e
6	6	110	6	[ACKNOWLEDGE]	54	36	110110	66	6	102	66	1100110	146	f
7	7	111	7	[BELL]	55	37	110111	67	7	103	67	1100111	147	g
8	8	1000	10	[BACKSPACE]	56	38	111000	70	8	104	68	1101000	150	h
9	9	1001	11	[HORIZONTAL TAB]	57	39	111001	71	9	105	69	1101001	151	i
10	A	1010	12	[LINE FEED]	58	3A	111010	72	:	106	6A	1101010	152	j
11	B	1011	13	[VERTICAL TAB]	59	3B	111011	73	;	107	6B	1101011	153	k
12	C	1100	14	[FORM FEED]	60	3C	111100	74	<	108	6C	1101100	154	l
13	D	1101	15	[CARRIAGE RETURN]	61	3D	111101	75	=	109	6D	1101101	155	m
14	E	1110	16	[SHIFT OUT]	62	3E	111110	76	>	110	6E	1101110	156	n
15	F	1111	17	[SHIFT IN]	63	3F	111111	77	?	111	6F	1101111	157	o
16	10	10000	20	[DATA LINK ESCAPE]	64	40	1000000	100	@	112	70	1110000	160	p
17	11	10001	21	[DEVICE CONTROL 1]	65	41	1000001	101	A	113	71	1110001	161	q
18	12	10010	22	[DEVICE CONTROL 2]	66	42	1000010	102	B	114	72	1110010	162	r
19	13	10011	23	[DEVICE CONTROL 3]	67	43	1000011	103	C	115	73	1110011	163	s
20	14	10100	24	[DEVICE CONTROL 4]	68	44	1000100	104	D	116	74	1110100	164	t
21	15	10101	25	[NEGATIVE ACKNOWLEDGE]	69	45	1000101	105	E	117	75	1110101	165	u
22	16	10110	26	[SYNCHRONOUS IDLE]	70	46	1000110	106	F	118	76	1110110	166	v
23	17	10111	27	[ENG OF TRANS. BLOCK]	71	47	1000111	107	G	119	77	1110111	167	w
24	18	11000	30	[CANCEL]	72	48	1001000	110	H	120	78	1111000	170	x
25	19	11001	31	[END OF MEDIUM]	73	49	1001001	111	I	121	79	1111001	171	y
26	1A	11010	32	[SUBSTITUTE]	74	4A	1001010	112	J	122	7A	1111010	172	z
27	1B	11011	33	[ESCAPE]	75	4B	1001011	113	K	123	7B	1111011	173	{
28	1C	11100	34	[FILE SEPARATOR]	76	4C	1001100	114	L	124	7C	1111100	174	
29	1D	11101	35	[GROUP SEPARATOR]	77	4D	1001101	115	M	125	7D	1111101	175	}
30	1E	11110	36	[RECORD SEPARATOR]	78	4E	1001110	116	N	126	7E	1111110	176	~
31	1F	11111	37	[UNIT SEPARATOR]	79	4F	1001111	117	O	127	7F	1111111	177	[DEL]
32	20	100000	40	[SPACE]	80	50	1010000	120	P					
33	21	100001	41	!	81	51	1010001	121	Q					
34	22	100010	42	"	82	52	1010010	122	R					
35	23	100011	43	#	83	53	1010011	123	S					
36	24	100100	44	\$	84	54	1010100	124	T					
37	25	100101	45	%	85	55	1010101	125	U					
38	26	100110	46	&	86	56	1010110	126	V					
39	27	100111	47	'	87	57	1010111	127	W					
40	28	101000	50	(88	58	1011000	130	X					
41	29	101001	51)	89	59	1011001	131	Y					
42	2A	101010	52	*	90	5A	1011010	132	Z					
43	2B	101011	53	+	91	5B	1011011	133	[
44	2C	101100	54	,	92	5C	1011100	134	\					
45	2D	101101	55	-	93	5D	1011101	135]					
46	2E	101110	56	.	94	5E	1011110	136	^					
47	2F	101111	57	/	95	5F	1011111	137	_					

Figure 5 : Table de conversion ASCII

Notre clavier est un clavier numérique, c'est-à-dire qu'il écrit seulement des chiffres allant de 0 à 9. Ainsi, d'après le tableau ASCII ci-dessus, pour afficher ces chiffres sur notre écran LCD nous avons besoin des lignes 0x30 à 0x39.

Utilisation du clavier nécessite l'utilisation d'interruption.

III. Développement mémoire EEPROM

4. Principe et utilisation d'une mémoire EEPROM

Une mémoire EEPROM est une mémoire dite non-volatile (aucune donnée supprimés lors de la mise hors tension) permettant de stocker des données. Le contenu de cette mémoire peut-être directement changer et modifier de manière quasiment illimité (en réalité modifiable approximativement 1millions de fois au maximum) grâce à un programme. Cette mémoire, interne au microcontrôleur ou externe de part une puce supplémentaire sur un PCB à plusieurs applications :

- Mémorisation de réglages et configurations pour un système
- Mémorisation de données d'étalonnage
- Mémorisation de mesures sur une longue durée (Data Logger)

Il existe de nombreuses autres applications.

Dans notre cas, nous utilisons la mémoire EEPROM directement présente dans notre microcontrôleur PIC18K87K22. Cette mémoire peut contenir jusqu'à maximum 1024 octets (10 bits).

5. Mémorisation de l'affichage

IV. CONCLUSION

Ce projet de TP microcontrôleur nous a permis de développer un programme en langage C pour la gestion d'un clavier numérique et d'un affichage sur écran LCD. Indirectement, cela nous a permis d'introduire plusieurs éléments de développement :

- Création de bibliothèques de fonctions (files .c et header .h)
- Gestion des interruptions d'un microcontrôleur
- Gestion d'une conversion ASCII
- Gestion d'une mémoire EEPROM

Malheureusement, nous n'avons tout de même pas réussi à obtenir un résultat satisfaisant sur le développement mémoire de l'EEPROM. Cette partie ne fonctionne pas encore sur les mallettes d'essais mises à notre disposition. Plusieurs approfondissements sur ce point sont à prévoir de manière personnels.

V. ANNEXE 1 : Programme final

1. Main.c : fichier source du programme globale

2. LCD.c : fichier regroupant les fonctions propres à l'écran LCD

3. Clavier.c : fonctions propres à l'utilisation d'un clavier numérique

4. Eeprom.c : fonctions pour la mémorisation d'élément dans la mémoire EEPROM