Programmation embarquée



Numérique et Sciences de l'Informatique

Programme (1/2)

Contenus	Capacités attendues	Commentaires
Modèle d'architecture séquentielle (von Neumann)	Distinguer les rôles et les caractéristiques des différents constituants d'une machine. Dérouler l'exécution d'une séquence d'instructions simples du type langage machine.	La présentation se limite aux concepts généraux. On distingue les architectures monoprocesseur et les architectures multiprocesseur. Des activités débranchées sont proposées. Les circuits combinatoires réalisent des fonctions booléennes.
Transmission de données dans un réseau Protocoles de communication Architecture d'un réseau	Mettre en évidence l'intérêt du découpage des données en paquets et de leur encapsulation. Dérouler le fonctionnement d'un protocole simple de récupération de perte de paquets (bit alterné). Simuler ou mettre en œuvre un réseau.	Le protocole peut être expliqué et simulé en mode débranché. Le lien est fait avec ce qui a été vu en classe de seconde sur le protocole TCP/IP. Le rôle des différents constituants du réseau local de l'établissement est présenté.

Programme (2/2)

Les différences entre systèmes Identifier les fonctions d'un d'exploitation libres et propriétaires sont évoquées. système d'exploitation. Utiliser les commandes de base Les élèves utilisent un système Systèmes d'exploitation en ligne de commande. d'exploitation libre. Gérer les droits et permissions Il ne s'agit pas d'une étude d'accès aux fichiers. théorique des systèmes d'exploitation. Identifier le rôle des capteurs et Périphériques d'entrée t Les activités peuvent être actionneurs. de sortie développées sur des objets Réaliser par programmation connectés, des systèmes Interface Hommeune IHM répondant à un cahier embarqués ou robots. Machine (IHM) des charges donné.

Architecture



INSTRUCTION SET SUMMARY

	INSTRUCTIONS		*	lia!	T	185	011	11	26	107	ACI	Т	456	,-	Т	-	n.if		-	hq	ь	T.	100	,	17	PAG		T.	495	1	Т	105	-	Τ.	£14"	365	T.	wg-1	iei	Τ,	Pa	u .	7.	ect	sse	. 51	ia?ut							
MARMENE	OPERATION.	6	-						CIP	-			7.		t		-	-	CO	Ė	-	58	Τ.	Τ.	O#	-	Τ.	+-	1.	-	100	-	-	+	T.	Ī.	+	-	Ť.	26	-	-	12	t	5 4	4	7	16	1	NA MENT				
400				12		- 1			65	,		F	+	۳	+	+	4	-		É	÷	75	5	1	75	-	1	10	_	1	1	_	1	-	1	ŀ	10	1	+	128	7	+:	15	×	•	ŧ l	D	2 6	1	405				
AND									25		,	ı	1		i	1	ž		21		į,				100			30						l	ı	1	1	l	ı	ı	1	1	ľ			٠.		2 0		AND				
ASL	C-277 113-0	ľ	Τ.	1					56		,	١.	l,	Ĵ.	Л	1	- 1		•	."	ľ	ľ	1.	l.	1.6		ľ,	1,5			ľ	١.	1	1	1	1		Į	ı	1	1	i	13			٠.		20		451				
900	BRANCHONG . C (2)	ı	1		ľ	١,	1	"	"	١,	1	1	Г	Т	1	1	-					ı	1	ı	1.0	ľ	1	ľ	Ľ	ľ	ı	1	1	90	١.	١,]	l	1	L	1		Ľ				٠.			BCC				
acs	BRANCHONG V 1 121	ı	1	ı	1	1					1			ı	1	1	- 1							ı	1		!	1	!	ı	1			80		,	l	i	i.	ı	1	1	Ľ							8 6 5				
BEG	BRANCH CN 2 . 1 121	۰	۰	٠	+	+	-	-	Н	-	۰	۰	۰	۰	+	+	4	_	Н	Н	-	⊢	t	٠	+	╁	۰	╁	+	+	+	٠	+	10		2	╁	+	+	╀	+	+	÷	÷	<u>:</u>	-	-	-		015				
0 . 1	A s M	l	ı		١,	ı١	J	,	24	,	,			i	ı	1						1	ı	1	1		ı	ı	ı	1	1	ŀ	1	1	١.		ı	1	1	1	1	1	1.		•					8 1 7				
8	65ANCHONN : 1 (2)	ŀ	į.	ı	ľ	1	1		١,			i	ı	ı	ı	1	ł					ı		ı	1			ı	ı	1	1		1	30	1,	. ,	l	l.	1		1	1	1.					: .						
BNE	SFANCHONZ : 9 (2)	ı		ı	ı	1	1				1	1	L	ı	1	i								1	1		1	ļ	ı	ı	1	ı	į		Į,		l	ļ.	1	1	ı	1	1.						.1					
BP:	SRANCH ON N = 0 (2)	ı		1	ı	ı	- 1				!	П	l	ł	1			1				1	i	ı	ı	1		1	1	ı	1	ŀ		10		٠,	l	1	1	ı	!	1	١.							BPL				
B 4 K	BRE #K	t	۰	Ť	t	Ť	7	_	П	Т	Н	t	+	+	t	•	,	,		Т	1	+-	+	t	$^{+}$	+-	۰	t	۰	۰	t	۰	+	۲	۲	۲	۰	۰	۰	+	t	+	†:			,	- 7	-		000				
BVC	STANCHON V 1 0 12-	ı		1	ł	ı	- 1	П						i	1		- 1				i	l	ı	1	l	1	i	l		l	1	1	1	50	,	١,	l	ı		ı	1		ŀ						ŀ	BVC				
8 V S	BRANCH ON V + 1 12-			1	ı	1						ı		П	1		- 1				П		1	1	1		Ĺ	1		ì	1	1	ı	20	1,	1	l	ı	ŀ	1	1		١.							6 V S				
CLE	D - C	1		ı	1	1	- 1					ı		1	ŀ	8	2	1			П			1	ı	i	1	1		l	1		1	ı	1	1	L	ı	1	1	1		1.					- 6	ı.	cic				
CLD	5 - C		L			1						L			ŀ	×.	2	١				L	1	!	ļ.		1	L	1	ł.	ı	1	1	ı		l	ı	ł	1	1	į		ŀ			. :	٠.		٠l	CLD				
6.61	0 -:	Г	Т	Т	Т	T				Г		Г	_	ī	Ţ.	₹.	2	•		Г	П	Г	Г	Т	Т	Т	Т	Т	Т	Т	Т	Т	T	Т	Т	Т	1-	T	T	T	1	T	Ť	-			. c		1	C : 1				
CLV	0 = v	1	1	1	1	1	- 1					1	í			æ.		,			1	l	1		ĺ	-	1	1	Ĺ	ì	1	1	l	1	1	1	Ĺ	ı	1	1	1	ĺ	ŀ	0						CLY				
CMP	A - M				ŀ				C+			ı	ì	ŧ	i	i		J	C1	٠	2	o,	ŀ	2	05		12	oc	į٠	į,	Di	i٠		1	1	1	1	1	i	ı		i	ŀ					2 0	1	CMP				
CP>	x - w				þ		4	3		,	2	1		ŧ	1			Į					1	1	ĺ	ĺ	1	1	1	ĺ	1	1	1	1	1	1	1	ĺ	1	1	٠		1					2 0		CP.				
CPY	Y - M	c:	÷		þ				C.4	3	:	L	Ĺ	i.	1	_	_				L	L	i	Ĺ	L	1	L	L	L	L	L	L	L	L	L	L	L	L	L	L		L	1					2 0	:L	CPY				
DEC	U - 1 - U		i		ļc	•	6	2	Сb	5	2		1		ſ		1					1	Γ	Ī	×	6	2	ce	,	1	Г	Γ	Г	Γ	Ĭ	Γ	Γ	Г	Г	Т	1	T	N					2 .		DIC				
Dix	> - 1 - X	1	1		ı	1	-							1				٠,				[1	1	1		1	1	1	i	1	1		1		1	1	İ	İ	1	1		1					Į.		011				
DE+	Y - 1 - Y		į.	÷	ı	ł	-					1		ŧ	1	6	2	1				ı		ı	1		ı	1	1	1	1	1		1	ı	ı	ı	ı	1	1			1					2 .	-	DE .				
E 0 8	* * O - 7 11.	15	1 2	: :					45			1			П	÷		1	٠	•	,	5.	15	12	35			19	٠	1,	14	1.	1	1	ı	ı	ı	ı	1	1	i		1				٠.	2 .		40=				
1.95	U . 1 -W	╀		+	E	٤	6	2	F	5	2	1	_	Ļ	4	4	_	_	_	_	1	L	1	1	14	4	2	5.6	1	1	4	1	L	L	L	L	L	L	Ĺ	L	i	-	Ŀ					Ζ.	ĿL	* N C				
1 N X	X - 1 - N	1			1	1										ь.		١				1		1	1			1	ì	1	1	ı	1	1		ı	ı	i	i	1	1	ė.	I٠					ζ.						
147	JUMP TO NEW LOC	1			ı	1	. !					1			K	ŧ	2	1				1	1	1			1	1	1	1	1	İ		1		1	ı	ı	i	1	i		!~	٠				, ,		. ~ -				
150	JUMP TO NEW LOC.			ŀ		0	6								ı		- 1							l	1	1	!		ı	١	ı	1	1	ı		ı	64	5	! >	1	1		ŀ	•						,				
104			. ,	Ì,		0		,		١.	١.	1			ı		ı				١.,	١,,	١.	١,	83	ı.	12	85	١.	١.	١,	i.	١.	1		!	ı	i	i	1	1		ŀ	•			• •			150				
10x	W - X (1.								46	,	-	۰	-	•	+	-	+	-	A.	*	12	8,	Ľ	+3	85	ŀ	12	185	ŀ	+			13	₽	÷	-	╀	÷	÷	84	-	-	۰	<u>.</u>	-	-		ž.		LDA				
101	M-Y to														ı	ł		1			l	1	1	1	ļ.,	١.	١,	lec	ŀ,	١,	P	Ι.	1	1			1	i		184	٠.	,	Ľ			•				10,				
155	4 -E >- C	Г	Т	ï					26			١.,	٠,		1	l	1				1	1	1	1	C		,				1		1	l				ı	i	ı			10					11		10,				
NOF	NOCPERATION	Ì		!	ľ	1	٠,	1				Г			١,	ċ	, 1					}		1	I٦	١.	1	ľ	1.	11	1		1				ì	i	i	1			1.		:									
284	AVM - A	01	Ċ	١,	ı,	6		,	œ	,		1			ľ		1	1			١,	١.,	l.	١,	١.,	١.	١,	١.٨	٠.	١,	١.,	١.	Ι.			÷	1	1	1	1		į	1:	Ċ.						064				
PHA	4 - Us 5 - 1 - 5	1	-	-	۴	+	7	4	Ė	_	-	t	-	-	t,		3	,	-	٠	۰	Н	۲	۴	۲	۲	ť	۲	۳	+	+	۳	+	+	+	+	+	+	÷	╁	+	+	+:	÷	÷	-	÷	÷		P 14				
P 11 P	# - Ms S - 1 - 5	l			1	1	-1	П				1		ì			> 1				1	l	i	1	1		ı	ı			1	1	1	1	ı		ı	1		1			1.							0 10				
PLA	5 - 1 - 5 MC - A	ı			1	ı	- 1					П		i		4		,			1	1		1	1		1	1		1	1	ı	1	1	1		ı	1		1		1	١,	Ċ				, ,		P1 A				
FLF	5 - 1 - 5 Us - P		į.	i.	1	ı	- !							Ĺ			. !	,						1		Ì	1	1		Ł	1		1		1		1	1		1		1	1		ne e		et D	•		PLP				
HOL	- 51.		i	1	١,		61	,	×		,	١,,		ŧ.	П		- (L					١.	l,	١,	١,	١,	1	1	ì	١.	1	ŀ	ı	1	÷				١.	"				2 0						
ROR		Т	٠	+	10				u			1		+	+	_	_			_	+-	-	-	t	76		12				+	t	٠	+	+	۰	t	t	-	t	•	•	1		÷	-	-	7.5		ROR				
RTI	PERN INT		į	i	1	1	- 1							ŀ		ο:		٠,					i	1			ľ	1			1	ĺ	ı	İ		1	ı	1	ı	ı			1		RES	10	ME D			B T I				
815	STRN SUB		!		1	1	- 1							ı	1	c!	٠	١,		i			Ĺ	1	1	1	İ	1	ı	1	1			1	Ĺ	1	1	1	į	1	:		1.							RTS				
5 8 C		11	12	. 2	k	c	4	3	85	,	2	1		1	1		J		ŧ١	6	2	10	١	2			1	10	4	þ	ŀ	١.	,	1	1	1	1		1		į.		1.	. ,	-	-		2 -3		5 8 C				
SEC	1 - 0			,	1	1	- 1						:	ı		ĸ.						l	Ĺ	1	1	1	1	1	1	i	1	1	1	1	1	Ĺ	1	1	ì	ı	1		1						1	516				
S € 0	1 - 0	L	!	1	1	1	J				_	L	_	i		ě.		1		L	L	L	L	L	L	L	L	1	L	T	L	Ĺ	1	L	L	Ĺ	1	L	l	L	Ĺ	į	ŀ				٠.		1	5 6 0				
SEI	1 *:	Γ	ſ		T	Ĩ	1	П	Г		Г	Γ	_	Т	T	8	2	1		_	Г	Γ	Г	Г	Γ	Т	Г	Т	Г	Т	Т	Т	T	Г	Г	Γ	Г	Т	ī	Т	T	1	1	,	-	-			1	5 6 -				
STA	A - W	1	İ	ı			1	>	e			1	Ė		1		ı		31	6	2	91	e	12	95	1.	2	90	١	þ	91	1	þ		ı	Ĺ	1	ì	1	1	İ	1	1				٠.			5 7 4				
51.	7 - W		1		•		•	3			2		ï		1		ı					l	1		1	1	į.	1	1	1	1	ı			1	1	1	l	ì	w	4	į,	1						1	51.				
577	Y - M	1		ŀ		c	4	3	*	,	,		1		1		1						1		24	1	15	1		İ	!	1		ı		1	1	l	1	1	1	1	1		٠.					511				
14>	A - 1	1	1	1	4	4		Ц	L	_	i-	1	_	4			2	٠		L	L	1	1	L	1	Ĺ	L	L	1	1	L	L	1	L	1	L	L	ļ.	1	L	į.	1	1	4				2 .		144				
147	Ave	1	i		1	1	1									10		t			Г	1	1	ľ	1	1	1	1	Г	1	1	Г	1	Г		ſ	Г	Г	Г	Г	Г	Г	I			-		2 .	T	7 A Y				
154	5	1	1	i	П	1	1				i	1	1			A.		•			Ĺ	ı	1		1	1	1	1	1	1	1	1	1		Ĺ	1	1	i	l		1	i	1	٠.				2 .		151				
1 2 2	1 - A 1 - 5	l		l	1	1	- 1					1	1			4		•				l			1	1	ı	1	İ	1	1	ļ	1		1	ĺ	ŀ		į		1	1	1	٠.				2 .		1 , .				
17.5	Y - 6	1		l	1	í	-					ı					1	1						į	1	1	1	ļ	l	1	1	1	1	1		1	1	ì	1	1	1	1	1.	٠	٠					T × S				
.,,		1	1	÷	1	1	4	ш	L	_	_	ı.	L	4	Ŀ		?		_	L.	L	1	L.	L	1	L	ı_	Ĺ	L	L	L	L	L	L	L	L	L	L.	L	1	L	L	L				•		1	7 Y A				
	1. ADD 1 to N. IF PAGE BOUNDARY IS CROSSED. ADD 1 TO N. IF BRANCH OCCURS TO SAME PAGE.														×			.												ΑŪ						ų.			IOR1															
	ADU 7 10	~		fr.in	AN(H	ÖÇ!	U		č	111	RE	*	64	Gŧ							l		Ĭ																211	MC.	r			۳.				PELES					
	J- CARRY N																					l											res																					
	ACCUMU	4	OR	MU.	511	BE.	ŝ	ò	C	FO	P.26	46	e(۶.,										*																	Siv	f 04	6		•		40	-716						
	ACCUMULATOR MUST BY CHECKED FOR \$400 RESULT											No MEMORY PERSTACK POINTER Y ENCLUSIVE OF																																										

Architecture

- Exemple de l'**Apple II**, un des premiers ordinateurs personnels :
 - Processeur MOS 6502 8 bit à 1Mhz
 - RAM: 4ko
 - ROM: 8ko

INSTRUCTION SET SUMMARY

	INSTRUCTIONS		44:4	٩Ţ	1850	01 H1	1	2686	P1G	1	ACI	98	I	MPL:	fÐ	113	1	٠.	(190	b, 7	1	PA	LE 1	1	105	1	M	NS 1	Ţ	M;	ATH	1	40	NC.		2 .	441	1 10	3063	-	Star	w\$			
MNEMENK	OPERATION.	or.	-1		90	T	:	10	-		7		0	1.	١.	04	.1			-	. 0	p! a	Τ.	10			OP!	.1		90	T.		T	T	٠,	per .	٦.	17	63	5 4	3	7.	Ę	Į.	
400	A . W . C . A . (4-12)	69	2 1	_	60		1	101	1		+	+	Ť	۲	Ľ		L		-	5	-	-	1.	70	_	-	29	1	÷	7	+	٣	+	+	+	7	+	-	¥ :		C.,	17	5	L	-
AND					20				3			ì	ĺ	1	į	21				1				30					,	- 1	-1		1	1		1	1	1.							
A 5 L	C-277 223-0	ľ	-1.		ex								1	1	1	r		T	7	Т							"	1	1	-1	- 1		1	1		-1	i	1.	٠.		-	٠.			
800	BRANCHONG . C 121	Н	- 1	- 1	٠,	١.	1	9	,1,	T	•	1	1				- 1	- 1	- 1	-	-1"	١,	1	ľ	ľ	12	П	- }	- 1.	!	J.	. 1		1	1	1	-	1"		٠.		. z	¢		
acs	BRANCHONE : 0 121	П	- 1	- 1	- 1		-1	ì	- 1	1			1	1				-	- 1	- 1	1	1	1	1	1	1		1			2		1	1	-	i		1				٠,			
		Н	+	+	+	+	+	+	÷	+	+	+	+	+-	-	Н	-	+	+	4	+	4	+	╀	↓_	┞-	1.	-			2		+	+	4	4	4	+			-		•		L
810	BRANCHONZ : 1 IZI	1	- 1		,c	- 1	. İ	J	١,	Л	1	1		1	1			1	1	-	ì	П	i.	1				- 1	ľ	16	3 .	2	1	1	1	1	-1	1	٠,	•		٠.	٠		
		1	i	ľ	125	4	1	•	2	1	1			1	i l			-		- 1	1	1	i.	1		1		- 1	- 1	- 1		1	1	1	-	- 1	- 1	1"	- M, -			- 2			
BMI	65ANCHONN : 1 (2)	П	- 1	- 1	ı	- 1	1	1	-1	1	1	1	ı	1		н	i	-	- 1	-	1	1		1		1		- 1			2 .		1		-	-	-1	1.		• •	-			ı	
	6FANCH (NZ = 0 12)	н	i	- 1	1	- 1	1	- 1	1	П	1	-	1			н	1	-		- 1	П	1		1		1		- 1	- 1:	∞ŀ	٤:	2	ł	1	1	- 1	- 1	1.		٠.				ı	
BP:	BRANCH ON N = 0 (2)	ш	4	4	4		4	4	4	4	4	1	1	_		_	_	1	1	4	4	4.	1	L	_	L	L		1	10	7	4	l	1	_	i	1	Ŀ				٠,		L	
B 4 x	BREAK		1	- }		ı	1	-	1		ì	1	0	,	1	П	- 1	1	1	- 1	1	1	i							- [i	1	Ţ		1	٦		T						Т	ſ
BAC	SRANCH ON V + 0 12-	١.	- 1	1	- 1	- 1	1	ı	1	-	1	i	1		1	П	- 1	1	-	-1	1	1	1	1		1	1	- 1			2		1		1	- 1		1.						1	
BVS	BRANCH ON V + 1 +2-			1	1	- 1	1	- [1	-	1		1		П	П	- 1	1	-	- 1	1	1	1	1			1	- 1	- 1	201	21	:]			1			1.						1	
CIE	0 - C		-	1	- 1	1	1	1		1	÷	1		2	1	П	- 1	1			1	i	1	1		1		- 1	- 1	- 1	- 1	1		1	1	- 1	1	1.					0	1	
CLD	5 - E						1	1		1			D	6 2	1			1			1		1	1	١.	١.		- !	- 1	- 1	ı	1	+	1	1	- 1		1.			•		٠,	Н	
5 1 1	0 -:		Т	7	Т	Т	7	Т	Т	Т	7	T	1	. 2	٠.	П	Т	T	Т	Т	T	T	T	T	_	-	П	7	7	1	7	1	T	1	T	*	_	1.			-	. 2	,	t	۰
CLV	0 - v	ш	- [1	1	-1	1	ł		1	í					Н	- 1	1	- 1	- 1	1	1		ł	i	ì	1	- 1	- 1	- 1	1	1	1	1	1	1	- 1	1.					i.	ı	
CMP	A - M	C9		, 1	lo:	4	d	a.	, :	,	i	1	Г	η.		c+		, .	01	,	, lo	، ا،	١,	lor		١,	pel		,	-1	- 1	П	1	1	1		i	I.		. :				ı	
CP>			,						, ;		i	i						T			ľ	T	Τ,	1	Ľ	ľ	1		1	- 1	- (1	1	1	ı			T.		: :	-	. ;		ı	
CPY	Y - M			3			3		i		1	1				ш	- 1	-1	- 1	- 1	1	1	1	1			Н	ı	1	- 1	- (1	1	i	-			1.			-	٠.,		١	
DIE	U . 1 - U	-	-			6			5	2	+	+	+	-		Η.	_	+	+	+	+	6 6	:†:	CE	,	,	Н	+	+	+	+	+	Ť	+	+	-	-	+	-	-	-	. 2		t	
DEX	2 - 1 - X	- 1		ľ	-1	1	1		1	1	÷	i	6	. 2	ı, l			1	- 1	-	ľ	1	Τ'	1	Ľ	ľ	П	- [-1	-1	-1	1	1	1	1	ı		1.				٠,			
DE -	Y - 1 - Y	Н	1	1	- [1	- 1	ı	1	1		1	6					1		-	1		1	1	1	1	1	- 1	-1	-1	i		1	1	1	1		1.	٠.			٠ ٢		l	
608			,	. J.		J.	, 1	J	j.	. 1		1	ľ	:	1			,[.			J.	J.	١.	1	١.	١.	اسا	J	J	ı	-	1	ĺ	١	1	ŀ		1.						l	
1 3 5	W . 1 - W	"	•									1		:	1		*:	T	1	1	. 1,		11	59	١;		70	٠,	1	- 1	÷	1	1	1	1	1		1.			•	. z		l	
1 10 2	X - 1 - X	-	-	ť	+	+	+	+	-	+	-	÷	+	7	1	+	-	+	+	+	-[-	4	+	+	Ļ	ť,	Н	-	4	+	+	+	+	+	+	+	-	+			-	· 2	_	1	
	7-1-7	:		- 1	-	i	1	i	1	1		1			Ш			1	-	-	1	1		1	i	i	1	- 1	- 1	- 1	- 1	1	1	i	1	÷	4	1.	٠.			٠ ٢		ı	
JMP	JUMP TO NEW LOC			-1	•c	,	J	1	i	1			ľ	, ,	۱'۱	Ш	i	1	-	- 1	ı	i	1	1	1		ш	- 1	-1	- 1	-1	1	.1	i	. 1	į		1*	٠.			٠,		l	
150	JUMPTONEWLOC		1			3		į	1	1					П	Н		1	-	- 1	1		1	1	1	Ĺ	П	- 1	-1	- 1	-1	16		1	1	1		1.				٠.	•	1	
15.	N - A II-		1			1	1	J		J					i I	IJ	. !			.l	. I.	.i		l.		١.	Li	- 1	J		1	1	Í	i	1	1		T.	٠.			• •	٠	١	
10x		42			40	:	#	4	*	4	+	-	+	-	+	-	*	4	-1	3	2 8	4.	- 2	80	ŀ	3	891		4			+	+	÷	+	_		+^		-	<u></u>	- 2	<u>.</u>	+	
101													ı	1	1		i	1	1		1	1		ĺ.	!	ì	Ðŧ	٠,	1			1	1	1	8	36	•	.1.				. 2		l	
		*	2										1	ı		П	i	1	1	1	1							- 1	-1			1	1	i	1			1"				٠,		l	
155	6-EE		- [ľ	46	6	1	ε.	٠.	1	•		1			п		1	-1	1	,	4		14		,		J	ı			1	i	1	1			10					c	t	
NOF	NODPERATION		- 1	Ш		ı							ĮŁ.	. 2	1	Ш	, į	1	- 1	ı.	.1	1		İ				-]	J	- 1	÷	1	Ĺ	1	1			j.	٠.			٠.	٠	ı	
DRA	AVM - A	69	2	7	30:	4	1	Υ.	3:	4	_	4	+		•	01	6	2	**	5	2 1 "	5 0	42	10		3	16	•	1	4	_	1	1	1	1	-	-	١,				· 2		L	
PHA	4 - Us 5 - 1 - 5		i.	1		- [1			1		į	14			П	ı	1	- [- [1		1	1			П	ſ	1	1		T	ľ	T	Т	Ţ		1.					•	Τ	
P 11 P	# - Ms S - 1 - S			-	- 1	- 1	1			1		i		ļ,		ш	-	1	- 1	- 1	1	i	1	1			1	- 1	- 1	- 1		1	1	1	1			1.						Ł	
FLA	S = 1 = S = MC = A		- 1	ı	-	-	1			1	ŧ	!		1		Ш	-	-	÷	-	1	1		1				- 1	- }	- 1		1	1		1		÷	l,						١	
FLF	5 - 1 - 5 U5 - P		- [1	- [i	1			1		1	2		١٠		- 1	1	- 1	-	1	1	i	1		1	1	- 1	-1	-1	•	1	1		1		÷	1	,a	4E S1	roeg	C.		ı	
HOL			_ i_	1	2E	6	1	ж.	6	2 2		:!	1	1		Ш			. i		10		1 2	h	,	,	ш	ı	-1	-1	- 1		1	:	1			1.					c	Ì	
ROH			-	Ţ	52	0	3	и	٠,	7	Α,	7.1	Т			1		T	-	Т	1	6	1 2	71	?	3	П		7	7	-;	T	T	_	T	_		1.			•	. 7		t	
RTI	PERN INT	H	i	1	-1	-1	-1			1		1		100			- !	1	- 1	-	1	1		1			Ιİ		- 1	-1	- [1	1	1	i			1			180			ı	
815	RTRN SUB	!		J		1	1			1		1	6		1	H		-		- 1	1	1		1		1			- }	- }	- 1	1	1	į	1	:		1.						ı	
5 8 C		f 9	2.3	2 1	10	4	3	15	١.	2		1	1	1		€1	6	2		>	۶,	٠,	1	1:		3	19	4	,	-1	1	1	1	1	1	į.		1.				- 2	٠,	1	
S E C	1-0			1	-	-	1			1	÷	1	b	: 2	1.	Н		- 1	- 1	ı	1	1	í	1	1	1	11		- 1	- !	1	1	1	1	1	1		1.						П	
\$ f 0	1 - 0	L!	_	J		1	1			ł		1	1	2	,	ш		1		1	1	1	1	1	İ				- 1	- }	ı	1	1	1	1	ı		1.			,			l	
s t ·	1 = :		_	1	1	T	7	-	1	T	-	+	7	1 - 7	1		-	1	+	7	1	†	+	t	1	•	П	1	+	+	+	+	+	+	+	+	1	+	-	-			-	t	
STA	A - W	Н	1	-	80	4	М	n.	οį.	,	÷		1			31		2	٠l.		, ۱,	۰,	٠,	90	١,	1	99	,	١,	-1	-1	1	ì	1	1	1	1	1						ı	
51.	7-4	П		J	er l	4	١,	16	١iد	,		1	1			Ш		ı	- 1	1	-	1	1	1	ľ	ĺ	П	П	1	- 1	- 1	1	1	1			4	٠.						ı	
514	Y - M	П	1								1	1	1		1	П	- 1	- 1	-1	1	١.	J.	d,	.	1	1	ıŀ		- 1	- 1	-1	1	1	1	ľ	-	11	1		. :		. :	Ċ	l	
141	A - 2		- 1	- [-	1	1	1	1		i			J,	١.	ш	- 1	- 1	ı	1	1	1	Τ,	1	1	1	1	П	- 1		- 1	1	1	1	1		1	1.						١	
* A Y	A = Y	Н	+	+	+	+	+	+	+	+	+	+			1	H	+	+	+	+	+	+	+	+	+	+	Н	+	+	+	+	+	+	+	+	+	÷	+:	-		<u></u>		-		
154	5->	ш		Ī	- 1	- [1			1	i				1.	П		1	- 1	-	1	1	1	П		1	1	١	١	- 1	-1	1	1			- 1	1	1.				٠ ،		1	
111	1 -4		- 1		-	1	-1		- 1	1	1		ļ,		П			- 1	-		1	1	1	1	1	1	П		- 1		1	1	1	ļ	-	١	- 1	1."						l	
115	X - 5	ш	-	J	- 1	1	- 1		- 1	1	1		1.		10		ij	-1		-	1	1	1	1		1	1		-]		1	1	1	1	-	- 1		1"				. 3		۱	
	Y - 4	П		Ţ	ı	1	-	1		1	1		ľ		i i	ш	- 1	١	-	- 1	1	1	1	ļ			ı		- 1	-)	- [1	Ĺ	1	1	- [-	1.		٠.	•	٠.		۱	
.,.		ш	-	_1	_	1	4	4	_	_	1	_	T ₀	1.	L	1	_	4	_		1	j.	1_	1	_	L	ш	Ш	_	_			L	1	1	_1	_	<u>٠</u>					٠.	L	
	- ADU 11																	- 1		,			e s										-		DC.				- 1	¥		(40			
	ADU 2 10	7	* fr#	AN	CHS	oce oce	ᆙ	10	DI	el e	EN		Sŧ.					- 1		•			12.0												181	RA:				Μ.		(uo			
	: CARRY NO	77 .	807	mo	w													-							1OR							•			·D					•		0 01			s
	ACCUMUL					AG.	50		. 12									- 1		•							C.FIVE			155				0			v£ O		•	•	Pr C	0 81	TE!	5	

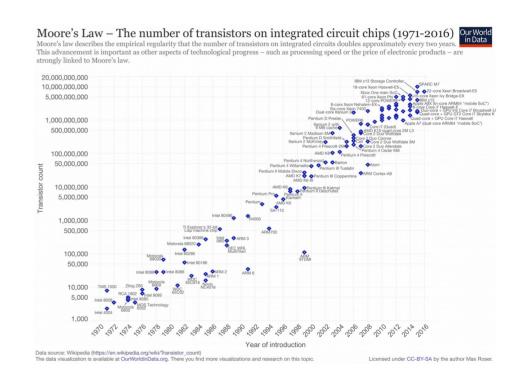
1979 vs 2019

Processeur	8bit 1Mhz	64bit 2.9Ghz
RAM	4Kio	16Gio
ROM	8Ko	1To
Coeurs	1	8
Affichage	6 couleurs 280 x 192	16 millions de couleurs 1920 x 1080
Réseau	-	Ethernet GigaBit

Loi de Moore

« The complexity for minimum component costs has increased at a rate of roughly a factor of two per year »

→ Ne présage évidement pas de l'amélioration de l'expérience utilisateur



Loi de Wirth / May

Cette croissance à créé un fossé entre les développeurs et le matériel

« Software is getting slower more rapidly than hardware becomes faster. »

« Software efficiency halves every 18 months, compensating Moore's law. »



Le logiciel pas toujours à la hauteur ?



I have a Python program I run every day, it takes 1.5 seconds.

I spent six hours re-writing it in rust, now it takes 0.06 seconds.

That efficiency improvement means I'll make my time back in 41 years, 24 days

:-)

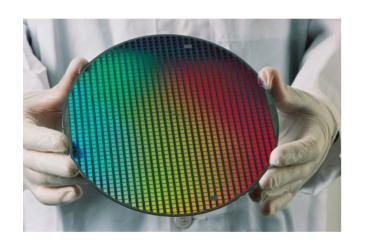
Traduire le Tweet

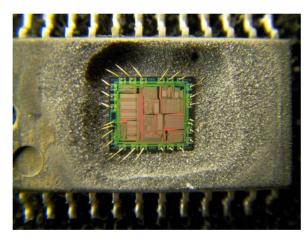
21:07 - 9 sept. 2018

Le logiciel pas toujours à la hauteur ?

- Logiciel lent
 - Windows 10 met **30 minutes** à se mettre à jour ...
 - ... dans le même temps, on aurait réécrit 5 fois le SSD entier
- Logiciel énorme
 - Windows 95 pesait **30 Mo ...**
 - ... l'application clavier de Google (Android) consomme 150 Mo
- Logiciel vite obselète

Du silicium au logiciel







Les types de circuits intégrés

- Sur un circuit, on peut retrouver :
 - Les processeurs (CPU)
 - Les microcontrôleurs (MCU)
 - Les ASIC/ASSP
 - Les DSP
 - Les FPGA

Cartes de développement



PC Portable Windows 3Ghz 50W

~800€



Raspberry Pi Linux 800Mhz 5W ~50€



Arduino Uno Aucun OS 16Mhz 0.2W ~5€

L'outil Arduino

```
- X
sketch_jan22a Arduino 1.6.13
Archivo Editar Programa Herramientas Ayuda
  sketch_jan22a
void setup() {
  // put your setup code here, to run once:
void loop() {
 // put your main code here, to run repeatedly:
                                             Arduino/Genuino Uno en COM5
```

Pas de système d'exploitation

- C'est le cas par exemple des cartes Arduino
- D'après vous, quels sont les avantages ? Les inconvénients ?

Pas de système d'exploitation

- On s'épargne le surcoût mémoire d'un système (raison de l'absence sur Arduino)
 - → Pas de préemption, avantage pour mieux maîtriser ce qui se passe, mais très peu pratique pour développer

Pas de système d'exploitation

- On s'épargne le surcoût mémoire d'un système (raison de l'absence sur Arduino)
 - → Pas de préemption, avantage pour mieux maîtriser ce qui se passe, mais très peu pratique pour développer

Comment ça marche?



Intel core i7 x86 64 bit





ATmega328P AVR 8 bit

Comment ça marche?

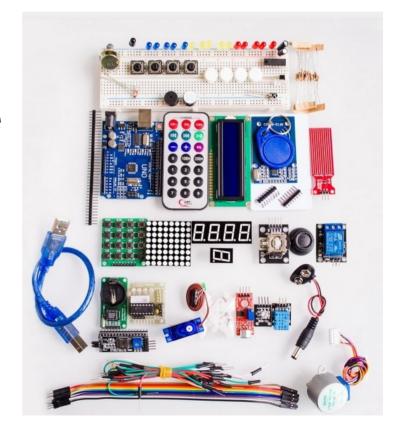
- L'ordinateur compile le programme pour la carte (différente architecture → on parle de cross-compilation)
- Le programme est chargé à bord à l'aide d'un bootloader (programme d'amorçage)

Contrainte :

 l'interpréteur Python est trop lourd pour Arduino, nous programmons donc en C/C++!

Capteurs et actuateurs

 D'un point de vue pratique, on trouve dans le commerce des kits et des capteurs/actuateurs compatibles Arduino



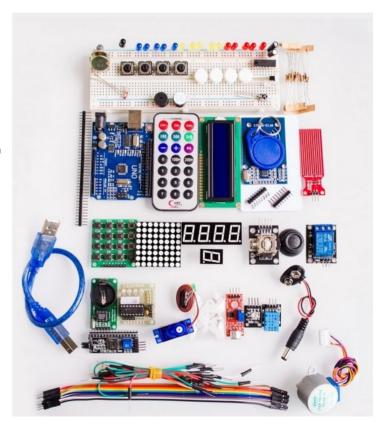
Capteurs et actuateurs

Capteurs

Température, luminosité, distance,
 pression, accélération, vitesse de rotation,
 magnétique (boussole), poids, ...

Actuateur

 Moteurs, servomoteurs, moteurs à pas, moteurs linéaires, électro-aimants ...



Programmer dans Arduino IDE

 Il est possible d'utiliser l'interface série pour afficher des messages :

```
void setup()
{
    // Initialisation de la communication série à
    // 115200bauds
    Serial.begin(115200);
}

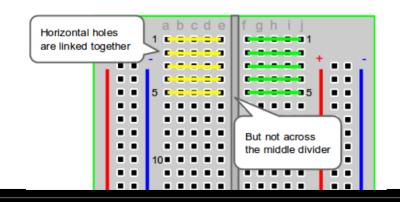
void loop()
{
    // Affichage d'un message
    Serial.println("Bonjour!");
    // On attend 1s
    delay(1000);
}
```

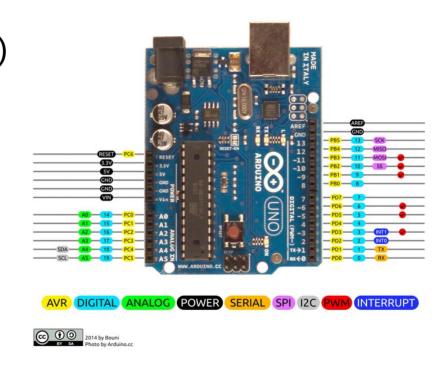
Programmer dans Arduino IDE

- Nous vous proposons un exemple de code commenté ici, qui permet :
- [exemple-complet.ino]

Broches et branchements

- Les cartes de développement sont équipées de broches (*pin* en anglais)
- On trouve aussi des platines
 « breadboard », qui permettent des branchements sans soudure :





Manipulation 1 Entrées/sorties analogiques et digitales

Entrées/sorties générales

- Toutes les broches peuvent être utilisées en entrée/sortie digitale
- On appelle cela GPIO
 - General Purpose Input/Output
- Le niveau « 0 » (bas) et « 1 » (haut) correspondent à un niveau voltage (par exemple 0V et 5V)

Entrées/sorties générales

 Dans Arduino, il faut paramétrer la broche (en sortie OUTPUT ou en entrée INPUT) avec la fonction pinMode, puis on peut contrôler son niveau logique à l'aide de digitalWrite :

```
// La broche 3 est en sortie
pinMode(3, OUTPUT);
// On écrit le niveau "haut" (5V)
digitalWrite(3, HIGH);
```

Entrées/sorties générales

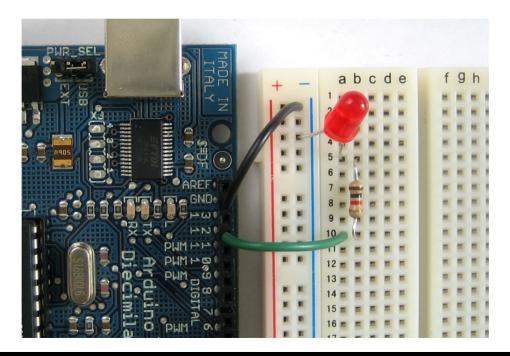
 De la même manière, on peut mettre une broche en entrée et la lire à l'aide de digitalRead :

```
// La broche 3 est en entrée
pinMode(3, INPUT);
// Si on lit 5V, on affiche un message
if (digitalRead(3) == HIGH) {
    Serial.println("Niveau haut!")
}
```

Branchement d'une LED

• Voici à quoi ressemble le branchement d'une LED et de sa

résistance:

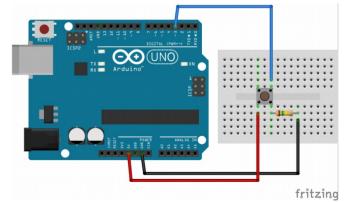


Branchement d'un bouton

- Pour lire un bouton, on peut utiliser un branchement similaire à l'image ci-contre.
- La résistance ici permet de s'assurer que l'on lit un « 0 » lorsque le bouton n'est pas appuyé (résistance de tirage)

Note : une autre solution est d'activer la résistance de tirage interne :

```
pinMode(10, INPUT_PULLUP);
```



Entrées/sorties analogiques

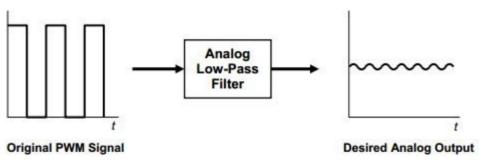
- Les entrées analogiques permettent de mesurer une valeur non-binaire, par exemple entre 0V et 5V. On appelle cela l'échantillonnage par un convertisseur analogique/numérique
- Sur Arduino, cette valeur est sur 10 bits (de 0 à 1023) :

```
// La broche "A1" est configurée en entrée
pinMode(A1, INPUT);

if (analogRead(A1) > 512) {
    Serial.println("Le voltage est plus haut que 2.5V!");
}
```

Entrées/sorties analogiques

- Dans l'autre sens, il n'est pas possible de produire une valeur analogique à proprement parler, car la carte n'est pas équipée de convertisseur numérique/analogique
- Cependant, il est possible de produire un signal carré, dont on règle le rapport cyclique. On nomme ce principe PWM (pulse width modulation):



Entrées/sorties analogiques

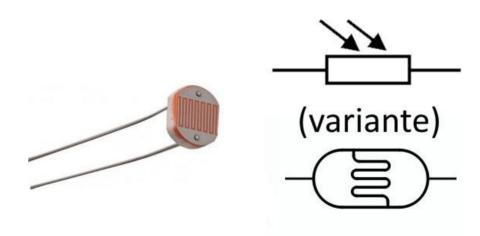
Arduino permet de faire cela à l'aide de analogWrite :

```
// La broche D9 est configurée en sortie
pinMode(9, OUTPUT);

// On règle le rapport cyclique à 200 sur 255
analogWrite(9, 200);
```

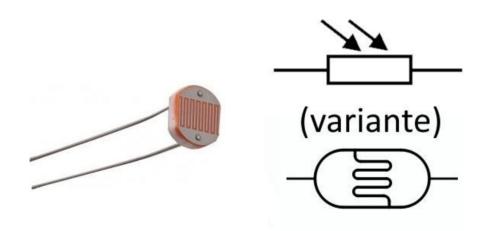
Échantillonnage analogique d'une résistance

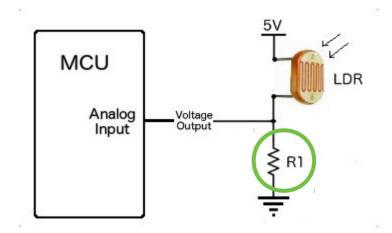
 Certains capteurs sont en fait des résistances qui varient selon la température ou la luminosité. Comment les échantillonner?



Échantillonnage analogique d'une résistance

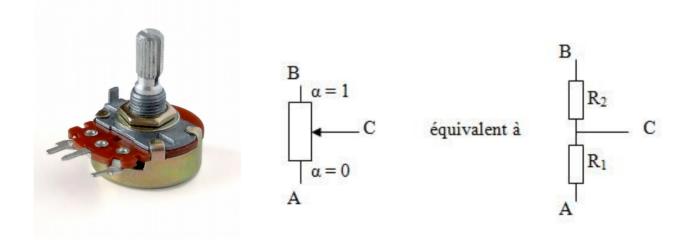
• On utilise simplement un pont diviseur de tension à l'aide d'une résistance fixe que l'on ajoute :





Échantillonnage d'un potentiomètre

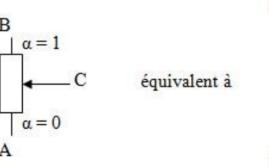
• Le potentiomètre est un type de capteur qui fonctionne en rotation et qui équivaut au circuit suivant :

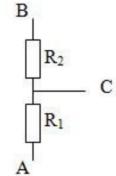


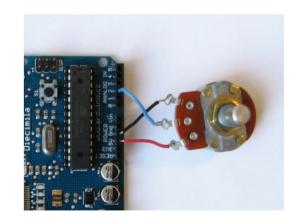
Échantillonnage d'un potentiomètre

 Pour l'échantillonner, on applique donc un voltage aux extrémités (par ex. 5V) et on mesure le voltage au point central, qui variera de 0V à 5V :



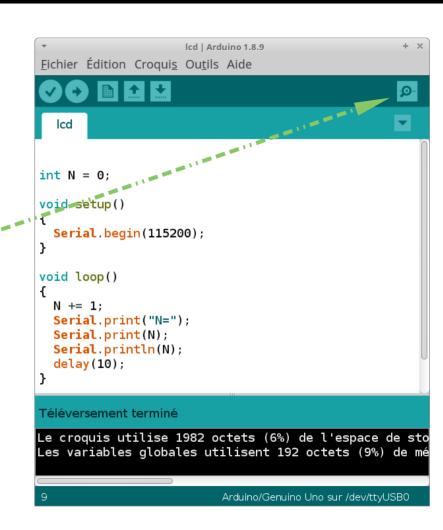






Moniteur série

- La carte peut communiquer pendant qu'elle tourne à l'aide du moniteur série
- Pour cela, il faut utiliser la même fréquence (baudrate) de communication
- [serial-print.ino]



A vous de jouer!

- Affichez dans le moniteur série les valeurs du potentiomètre
- A l'aide de analogWrite, faites varier l'intensité de la LED en fonction du potentiomètre!

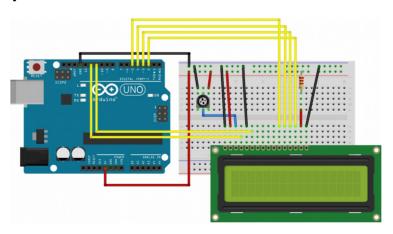
Attention : le potentiomètre est échantillonné entre 0 et 1023 (10 bits) alors que analogWrite attend une valeur entre 0 et 255 (8

bits)

Manipulation 2 Pilotage de moteurs

Bus de communication

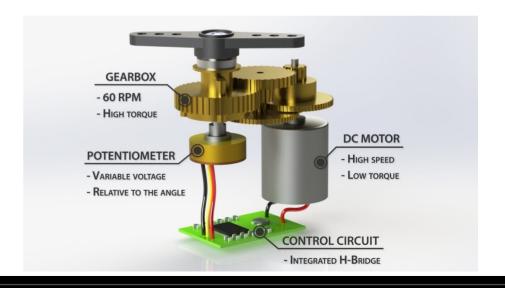
- La carte peut également dialoguer avec des capteurs via des bus de communication (UART, I²C, SPI, ...)
- Il existe de nombreuses bibliothèques, et la possibilité d'en télécharger des supplémentaires sur internet!



Servomoteurs

- Un servomoteur est un actuateur équipé :
 - D'un moteur DC
 - D'un étage de réduction (engrenages / gearbox)
 - D'un potentiomètre



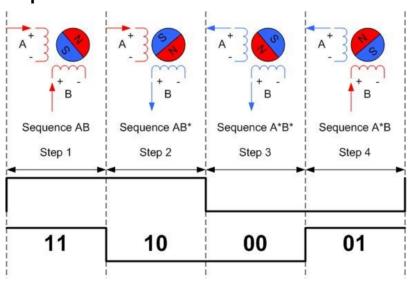


Servomoteurs

• La connexion se fait simplement (2 broches d'alimentation + 1 broche de signal), et la communication peut se faire via la bibliothèque

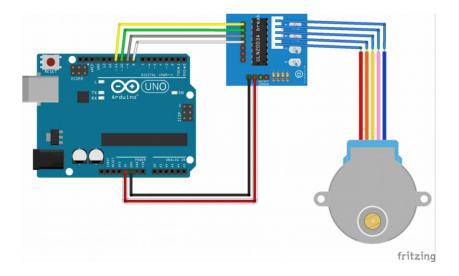
Moteurs à pas

 Les moteurs à pas sont basés sur des pôles, en général deux, qui sont activés avec des polarités différentes selon une certaine séquence, par exemple :



Moteurs à pas

 Nous utiliserons une carte de contrôle pour piloter le moteur, qui se branche comme cela :



A vous de jouer!

- Cette carte est dotée de 4 broches, IN1, IN2, IN3 et IN4
- A partir du tableau ci dessous, faites tourner le moteur dans un sens, puis dans l'autre!

Phase	IN1	IN2	IN3	IN4
1	HIGH	LOW	LOW	LOW
2	LOW	HIGH	LOW	LOW
3	LOW	LOW	HIGH	LOW
4	LOW	LOW	LOW	HIGH

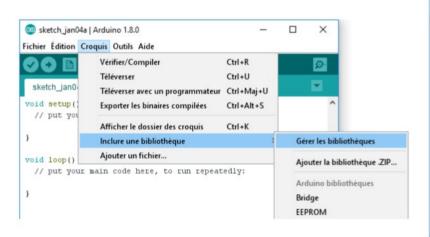
Manipulation 3 Écran LCD

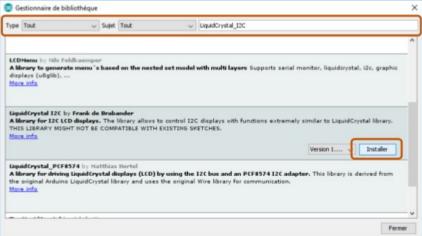
Ecran LCD en l²C

 Le kit fourni contient un écran LCD en l²C, pour le brancher nous allons utiliser le câblage suivant :

Écran LCD en l²C

 Rendez-vous ensuite dans « Gérer les bibliothèques » et installez LiquidCrystal_I2C :





Écran LCD en l²C

- Voici un exemple de programme qui affiche le temps écoulé :
 - 0x27 est l'adresse l²C du périphérique
 - 16 et 2 sont le nombre de lignes et colonnes
 - setCursor permet de se placer sur une ligne et une colonne données
- [lcd.ino]

```
#include <LiquidCrystal I2C.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);
void setup()
  lcd.init();
void loop()
  lcd.backlight();
  lcd.setCursor(0, 0);
  lcd.print("Temps:");
  lcd.setCursor(0, 1);
  lcd.print(millis());
  lcd.print(" ms");
```

A vous de jouer!

- Le kit contient un joystick qui fournit une position X, Y et un bouton appuyable (qui sont des entrées sorties)
- Faites le fonctionner, et faites un programme qui déplace un « X » sur l'écran en fonction du joystick!



Manipulation 4 Communication avec Python

Communication avec Python

- Il est possible de communiquer avec Python à travers l'interface série
- Un premier exemple qui permet d'envoyer des messages du code python vers l'écran LCD :
 - [lcd-receive.ino]
 - [send.py]

Communication avec Python

- Voici un exemple qui fonctionne dans l'autre sens, qui échantillonne le JoyStick et le lis depuis Python :
 - [joystick-send.ino]
 - [joystick.py]

Récupérer des valeurs

- Le programme [read.py] permet de récupérer des valeurs envoyées par Serial.println() et les stocke finalement dans un fichier CSV
- Le programme [plot.py] utilise la bibliothèque matplotlib pour dessiner une courbe des valeurs échantillonnées

A vous de jouer!

• En vous aidant des exemples précédent, ainsi que de [date.py], un programme Python qui affiche la date, ecrivez un programme qui affiche la date sur la première ligne de l'écran LCD, et l'heure sur la deuxième ligne

On devrait voir les secondes bouger!



Il y a encore plein de choses dans le kit que vous n'avez pas utilisé, n'hésitez pas à les découvrir!