# Programación de Memorias EPROM

Programación de Memorias EPROM:

Sitio:tu aula virtualCurso:LABORATORIOS L3 / Microprocesadores y Microcontroladores / Diseño de Sistemas DigitalesLibro:Programación de Memorias EPROM

Imprimido por: Rafael Antonio Márquez Ramírez

Fecha: miércoles, 25 de septiembre de 2013, 11:41

## Tabla de contenidos

1 Selección del Dispositivo

2 Edición del Buffer / Captura de Datos

3 Guardar Archivo (Buffer)

4 Programación

#### **1** Selección del Dispositivo

Se comienza seleccionando el modelo del dispositivo que se necesita programar.

En esta etapa del proceso se deben realizar las siguientes observaciones:

- Los dispositivos normalmente tienen impreso el Nombre o Logo del Fabricante y el modelo específico, por lo que se debe buscar en el listado disponible.
- En el caso en el que no aparezca en el listado, se hace necesario selecionar un modelo "Genérico"
- Desafortunadamente existe una variedad de voltajes de programación, y en algunas ocasiones el fabricante lo indica en la misma impresión del dispositivo; pero algunas veces no lo hace.
- Por lo tanto, si nuestro dispositivo no apareció en el listado y se tuvo que seleccionar un modelo genérico, y además no tiene impreso el voltaje de programación, tenemos 2 opciones:
  - 1. Buscar el manual del dispositivo para consultar el voltaje de programación específico para ése modelo.
  - 2. Sino se puede consultar el manual y por tanto no se conoce el voltaje de programación, entonces deberemos seleccionar el dispositivo genérico de menor voltaje de programación.

Si con ése voltaje no se pudo programar, entonces podremos seleccionar el siguiente voltaje mayor para volver a intentar programarlo adecuadamente.

💕 • 🔓   🌑 • 🍩   🖉	Select device			
Load Save Load prj Save prj View/Edit IS	All Only selected	type   Only selected manufacturer		
Programmer activity log	_			
0048:   Automatic YES!: "Disable"	Manufacturer	Name	Adapter/module or note	
0049:   Response time: "Standard"	Generic-EPROM	2716 (25V/5V/STD)	Note: none adapter required	
,0050:   ,0051:   Device removal hold off time (j	n Canalo EFREM	2716A (21/VP/VCTD)	Note: none adapter required	
0052:   Device insertion complete time	Generic-EPROM	2716L (12.5V/6V/Intel)	Note: none adapter required	
0053:   <del></del>	AND A	Am2716	Note: none adapter required	
0054:   Suspend on error: "Enable"	AMD	Am2716B	Note: none adapter required	
0055:	AMD	Am2716B [LCC32]	DIL32/PLCC32 ZIF-CS	
0057:  < Automatic YES!	Electronic Arrays	EA2716	Note: none adapter required	
0058:	Fairchild	F2716	Note: none adapter required	
0059: < End of options list	Fujitsu	MBM2716	Note: none adapter required	
0060: 0061: Scleated device: Lattice CM 22V10	Fujitsu	MBM2716H	Note: none adapter required	
0062:	Hitachi	HN462716	Note: none adapter required	
0063: Buffer "Checksum operation", time a	la Hitachi	HN462716G	Note: none adapter required	
0064: Buffer checksum in range of [Oh1]	103. Intel	2716	Note: none adapter required	
0065:	Intel	B2716	Note: none adapter required	
0066: >> 04.09.2012, 14:42:59 0067- Buffer checksum type is set to "Byt	e Intel	C2716	Note: none adapter required	
0068: Buffer block(s) excluded from check	su Intel	D2716	Note: none adapter required	
0069:	Intel	62716	Note: none adapter required	
0070: >> Log file created at 04.09.2012 ]	.4:	12710	Note: none adapter required	
0071: Log file name: C:\Documents and Set		12710	Note: Note adapter required	
ou/2: Log life mode: Append		102/16	Note: none adapter required	
(II a )	Intel	LD2716	Note: none adapter required	
Addresses (hex)	Search: 2716	0.546		
Device v1 1784 8 178	2 Search (2718	1		
Buffer v1 1704 0 170	Use sear     O     Talazante	ch pattern exactly as typed 50 found 112 found		
File x8		search and x characters replacement 112 round		
CheckSum: 0002DD2Fh [0h.1703h]		ОК	Cancel	
Serialization: None	Note: See also Dev	ace into <1.10++1.51		
Split None	To customize devic	e use menu View/Edit UES <alt+s></alt+s>		

En el ejemplo de la imagen se está seleccionando una Memoria EPROM Genérica 2716 con el voltaje de programación más bájo (12.5V) compatible con los dispositivos Intel.

Y una vez seleccionado el dispositivo seleccionamos el botón [OK]

• En el caso de que no se pudiera programar se debería intentar con el siguiente voltaje de programación mayor (21V).

El dispositivo seleccionado deberá aparecer en la parte inferior de la interfaz resaltado en color amarillo.



# 2 Edición del Buffer / Captura de Datos

Para capturar los datos que necesitamos programar en la Memoria EPROM tenemos que editar el buffer, esto lo podemos hacer seleccionando el botón [View/Edit] en la barra de herramientas de la interfaz.

al control program for BK PRECISION programmers.							
tions Help	No. or or or						
	W View/Edit Bu						
e pi View/Edit 5e	Buffer size: 16384.	0 kBytes Current address: 0000000000 h Mode: Viet Edit 8 bit 16 bit					
	2	i i i i i i i i i i i i i i i i i i i					
	Address-hex	0 1 2 3 4 5 6 7 8 9 A B C D E F 0 1 2 3 4 5 6 7 8 9 A B C D E F					
	0000000000	FF					
VCCnom. +/-10%	0000000010	FF					
operation optio	0000000020	FF					
	0000000030	FF					
vice options -	0000000040	FF					
ıV]: "5000"	0000000050	FF					
) mA]: "350"	0000000060	FF					
mV]: "6000"	0000000070	FF					
50 mA]: "350"	08000000080	FF					
mV]: "12500"	0000000090	FF					
) mA]: "200"	000000000000000000000000000000000000000	FF					
	000000000000000000000000000000000000000	FF					
Witched on [U.	00000000000	FF					
witched off [0	000000000000000000000000000000000000000	FF					
witched off [0.	0000000000000	FF					
wice ontions -	01000000000						
	0000000100						
)tions list	0000000110						
	0000000120						
ric-EPROM 2716	0000000130						
inge of [Oh7F]	0000000140						
	0000000150						
	0000000170						
er was opened.	0000000180	FF					
	0000000190						
	0000000120	FF					
t End	0000000180	FF					
0 7FF	0000000100	FF					
0 7FF	0000000100	FF					
<u> </u>	00000001E0	FF	_				
FFh]	00000001F0	FF					
	Note: <u>See</u>	also Device into <util+f1>I a device use menu Edit device options </util+f1>					
	TO CUSTOMIZ						

Una vez que se despliega la ventana del Buffer se debe seleccionar el botón del modo [Edit] para comenzar la edición.

En la ventana del Buffer después de seleccionar el modo Edit (1), podemos reiniciar cualquier contenido que pudiera existir, en caso de ser necesario podemos reiniciarlo completamente con valores por default (borrado) FF.

Para poder hacer esto debemos seleccionar el botón [Erase Buffer] (2), el cual despliega la ventana (Erase Buffer Block).



En esta ventana debemos especificar el rango de direcciones que deseamos reiniciar en Hexadecimal. (3)

En la caja de texto (Erase Value) capturamos el estado con el que queremos reiniciar (4).

Para completar este proceso se necesita seleccionar el botón [OK] (5).

Para comenzar la captura de los datos en el buffer, se debe seleccionar el botón de la barra de herramientas [Fill buffer block], que desplegará la ventana (*Fill buffer block*).

ŀ	View/Edit Bu	fer - G:\programador\EPROM\eprom2.HEX	۱×
	Buffer size: 163844	kBytes Current address: 00000000000000000000000000000000000	Бit
	2	) 🔁 🚔 😹 🔎 燥 🏹 🔁 📔 👔	?
Î	Address-hex	0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF	
	0000000000	<b>FF</b> FF	
	0000000010	FF	
	0000000020	FF	
	0000000030	FF	
	0000000040	FF Fill buffer block 🛛 🕺 · · · · · · ·	
	0000000050	FF Address rates	
	0000000060	FF Obstation I and the second se	
	0000000070	FF Block start. U n (18) History Y	
	08000000080	FF Black and ID I I I I I I I I I I I I I I I I I I	
	0000000090	FF BOOCK end. (B P(X8) History T	
	00000000000000	FF Note: Addresses are over-onented.	
	0000000080	FF	
	000000000000000	FF Fill with	
	000000000000000	FF H A6305BDFC17A8390006A	
	UUUUUUUUEU	FF	
	0000000000000	FF Ascii: [O[BAz]]	
	0000000100		
	0000000110	TF OK Cancel	
	0000000120		
	0000000130	T Allow address history logging	
	0000000140	TT I Maintain last incertad values	
	0000000160		
	0000000100		

En esta ventana se debe indicar el rango de direcciones que se desea llenar llenar con los datos, este rango de direcciones **también se indica en Hexadecimal**.

Finalmente en la caja de texto Hex: de la sección Fill with, se pueden capturar los datos que se necesiten.

Para completar el proceso se selecciona el botón [OK].

Para el caso en que se necesite capturar más de 16 datos, será necesario actualizar la dirección de inicio para continuar con la captura hasta terminar en la dirección que se especificó como fin de bloque.

R	🐶 View/Edit Bu	iffer	- G:\programador\EPROM\eprom2.HEX		
	Buffer size: 16384.0	D kBy	utes Current address: 00000000000000000000000000000000000	Edit	8 bit 16 bit
	2	]	🗎 🛼 😼 🍃 🔎 🔊 🦉 🖻		?
	Address-hex 000000000000000000000000000000000000	0 10 FF FF FF FF FF FF FF FF FF FF FF	1       2       3       4       5       6       7       8       9       A       B       C       D       E       F       0       1       2       3       44       55       66       77       88       9       A       B       C       D       E       F       0       1       2       3       44       55       66       77       88       9       A       B       C       D       D       E       F       .       "       3       D       U       m       3       D       U       T       3       D       U       T       3       D       U       T       3       D       U       T       3       D       U       T       T       3       D       U       T       T       3       D       U       T <td< th=""><th>7 8 9 A E 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7</th><th>C D E F</th></td<>	7 8 9 A E 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7	C D E F
	0000000130 0000000140 0000000150	FF FF FF	Allow address history logging     Maintain last inserted values	 	
	0000000170	FF	FF		

Para terminar el proceso de captura se selecciona el botón [OK].

Por último, podemos comprobar que nuestros datos fueron introducidos (capturados correctamente) revisando el contenido del buffer y podemos cerrar la ventana.

💓 View/Edit B	uffer - G	:\prog	ramad	or\EP	ROM	\epro	m2.	HE≻	<														1		
Buffer size: 16384	.0 kBytes	Cu	rrent ad	dress:	0000	000000	)0	h				М	ode		Vi	ew		E	Edit			Г	3 Ы	t 1	6 bit
	h D			6	3	0			T		1		Σ											1	?
Address-hex	0 1	2 3	4	5 6	7	89	A	8	C	D	F	F	7	1	2.3	4	5 f	i 7	8	9.	4 F	3.0	D	F	F
0000000000	00 11	22 33	44 5	5 66	77 8	8 99	AA	BB	cc	DD	EE	FF	f		" 3	D	U 1	Εw	•	123	• >	ì	Ý	î	
0000000010	10 11	12 13	14 1	5 16	17 1	8 19	1A	1B	FF	FF	FF	FF	Г			•									
0000000020	FF FF	FF FF	FFF	FFF	FF F	FFF	FF	FF	FF	FF	FF	FF		•3			•							•	2
0000000030	FF FF	FF FI	FFF	FFF	FF F	FFF	FF	FF	FF	FF	FF	FF	F.	•		•	•		•				•	•	-
0000000040	FF FF	FF FI	FFF	FFF	FF F	FFF	FF	FF	FF	FF	FF	FF					•			•					
0000000050	FF FF	FF FI	FFF	FFF	FF F	FFF	FF	FF	FF	FF	FF	FF		•		•	•						•	•	
0000000060	FF FF	FF FF	FFF	FFF	FFF	FFF	FF	FF	FF	FF	FF	FF		•			•		•					•	
0000000070	FF FF	FF FI	FFF	FFF	FF F	FFF	FF	FF	FF	FF	FF	FF		•			•	•		•				•	
0000000080	FF FF	FF FI	FFF	FFF	FF F	FFF	FF	FF	FF	FF	FF	FF		•			•				• •			•	
0000000090	FF FF	FF FF	FFF	FFF	FF F	FFF	FF	FF	FF	FF	FF	FF		•		•	•						•	•	
0000000000	FF FF	FF FF	FFF	FFF	FF F	FFF	FF	FF	FF	FF	FF	FF													
0000000080	FF FF	FF FF	FFF	FFF	FF F	FFF	FF	FF	FF	FF	FF	FF		•		•	•							•	
0000000000	FF FF	FF FF	FFF	FFF	FF F	FFF	FF	FF	FF	FF	FF	FF		•			• •	•	•	•			•	•	

## 3 Guardar Archivo (Buffer)

Como lo que se necesita es generar un archivo para poder programar la Memoria EPROM, se debe guardar el contenido del buffer que se acaba de editar en un archivo.

Se selecciona el botón [Save] de la barra de herramientas, que despliega una ventana (Save buffer).

er Device Programmer Uptions Heip	
Save Load prj Save prj View/Edit Select/def. Select Blank	E Carlo Carl
	Save buffer
mer activity log	
00h 11h 22h 33h 44h 55h 66h 77h 88h 99h AAh BBh CCh DDh	Dir. history: G:\programador\EPRUM\
Elapsed time: 172 ms	Guardar en: 🗁 EPROM 🔹 🕜 🌮 🎛 🗸
Buffer "Fill operation", time elapsed: 16 ms	
>> 04.09.2012, 15:06:11	
Buffer block 10h to 1Bh is filled with string:	
10h 11h 12h 13h 14h 15h 16h 17h 18h 19h 1Ah 1Bh	
Elapsed time: 172 ms	
Buffer "Fill operation", time elapsed: 15 ms	
>> 04.09.2012, 15:07:26	
Buffer block 10h to 1Bh is filled with string:	
10h 11h 12h 13h 14h 15h 16h 17h 18h 19h 1Ah 1Bh	
Elapsed time: 172 ms	Nombre: datos eprom HEX Guardar
>> 04.09.2012, 15:08:25	Tipo: IntelHEX (*.HEX) Cancelar
Dialog View/Edit buffer was closed.	
Buffer "Checksum operation", time elapsed: 0 ms	Advanced save options Additional operation
NN 04 09 2012 15·08·25	Buffer Start: 000000000000 h
Buffer checksum in range of [Oh7FFh]: 0007E516h - Byt	e Bulfer End: 00000007FF h

En la lista desplegable *"Tipo"* se selecciona el formato Hexadecimal de Intel "<u>IntelHEX (\*.HEX)</u>" y acontinuación se le asigna un nombre al archivo con extensión "<u>nombre.HEX</u>".

Finalmente seleccionamos el botón [Guardar] para generar el archivo.

### 4 Programación

Aunque el procedimiento de Programación es muy similar al de programación de dispositivo GAL, se tienen algunas diferencias en cuanto a las opciones de programación.

Se incicia normalmente seleccionando el dispositivo.

Se carga el archivo con extensión .HEX (<u>asegúrese de que esté configurado el formato Hexadecimal de Intel</u>) Se verifica que el dispositivo esté limpio para proceder a programarlo.

• En caso de que no esté limpio (borrado), será necesario suspender el procedimiento para colocar la Memoria en la lámpara de luz Ultravioleta por un periodo de 20 a 30 minutos para que sea borrada y se pueda programar.

Una vez que el dispositivo ha sido verificado de que está limpio, se selecciona el botón [Program] que desplegará la ventana (*Program* ?).

Image: Constraint of the second of the s	f. Select Blank Read Verify Program Frase	
0       Blank check before programming:       Disable \$\rightarrow\$       Delay after Vcc switched on [0.1000 ms];       100         0       Blank check before programming:       Disable \$\rightarrow\$       Delay after Vcc switched on [0.1000 ms];       100         0       Delay after Vcc switched on [0.1000 ms];       100       100       100         0       Blank check before programming:       Disable \$\rightarrow\$       100       100         0       Delay after Vcc switched on [0.1000 ms];       100       100       100         0       Delay after Vcc switched on [0.1000 ms];       100       100       100	d Program? Generic-EPPOM 21100 (12.57/07/intol)	Edit device ontions
0     Command execution       0     Blank check before programming: Disable •       0     Delay after Vcc switched on [01000 ms]; 100       0     Verify after reading: Enable •       0     Delay after Vcc switched of [01000 ms]; 200       0     Delay after Vpg switched on [01000 ms]; 100	Device operations (view cells)         Addresses         Device state         Device end:         Split:         Insertion test and/or ID check         Insertion test         Enable	Edit device options (view only)           Vcc [2000.7000 mV]           Ivoc max [10.350 mA]           350           Vccp [2000.7000 mV]           6000           Ivccp max [10.350 mA]           350           Vpc [2000.25000 mV]           Ipp max [10.320 mA]           200
Verify options: VCCnom. +/-10%  Delay after Vpp switched off [01000 ms] 50	Command execution Blank check before programming: Disable Verify after reading: Enable Verify: Twice Verify options: VCCnom. +/-10%	Delay after Vcc switched on [01000 ms] 100 Delay after Vcc switched off [01000 ms] 200 Delay after Vpp switched on [01000 ms] 10 Delay after Vpp switched off [01000 ms] 50

En esta ventana la única opción que se necesita modificar es (*Device end*) también en Hexadecimal, en la que se debe escribir la última dirección que será programada, (*la misma que se capturó al momento de editar el buffer*).

Para hacer esto se debe seleccionar el botón de la parte superior con letras en azul [Device operation options] el cual desplegará la correspondiente ventana de opciones (similar a la que aparece cuando se programa la GAL) y en la que podremos escribir la dirección final.

A continuación se selecciona el botón [Yes] para que aparezca la correspondiente ventana con la barra de avance del proceso de programación.