

IE-0423 Microprocesadores

Quiz

Timna Belinda Brown Ramírez B61254
timna.brown@ucr.ac.cr

III-2020

Tabla de contenidos

1. Comunicación serial sincrónica	3
1.1. Principio de la operación de la comunicación sincrónica	3
1.2. La interfaz Serial Port Interface (SPI)	3
1.2.1. Interfaz de hardware SPI	3
1.2.2. La interfaz SPI del 9S12	4
1.2.3. Configuración de la Interfaz SPI del S12	6
1.2.4. Configuración en Modo Bidireccional	7
1.2.5. El LTC1661	8
1.2.6. Generador diente de sierra	10
1.3. La Interfaz IIC	12
1.3.1. Interfaz de hardware de Inter Bus	12
1.3.2. Protocolo de Inter Bus	13
1.3.3. Configuración de la Interfaz IIC del 9S12	15
1.3.4. El RTC DS1207	16
2. Cuatro productos o aplicaciones comerciales que utilicen este tipo de comunicación	22

Índice de figuras

1.	Interfaz de hardware SPI. Imagen capturada del video de clase [1]	4
2.	La interfaz SPI del 9S12. Imagen capturada del video de clase [1]	5
3.	La interfaz SPI del 9S12 - Registros. Imagen capturada del video de clase [1]	6
4.	Configuración en Modo Bidireccional - Funcionamiento definido por BIDI-ROE. Imagen capturada del video de clase [1]	8
5.	El LTC1661 - Funcionamiento diagrama de bloques.Imagen capturada del video de clase [1]	8
6.	El LTC1661 - Configuración bits. Imagen capturada del video de clase [1] . .	9
7.	El LTC1661 - Estatus Registro, fuente de energía y comentarios. Imagen capturada del video de clase [1]	9
8.	El LTC1661 - Waveform. Imagen capturada del video de clase [1]	10
9.	Generador diente de sierra - Esquema de construcción.Imagen capturada del video de clase [1]	11
10.	Generador diente de sierra - Registros. Imagen capturada del video de clase [1]	11
11.	Generador diente de sierra - Diagrama de flujo solución ejercicio. Imagen capturada del video de clase [1]	12
12.	Interfaz de hardware de Inter Bus - Esquema de construcción. Imagen capturada del video de clase. [1]	13
13.	La Interfaz IIC - Más de un master. Imagen capturada del video de clase. [1]	13
14.	Protocolo de Inter Bus - 4 partes. Imagen capturada del video de clase. [1] .	14
15.	Protocolo de Inter Bus - Start. Imagen capturada del video de clase. [1] . . .	14
16.	Protocolo de Inter Bus - Slave Address Transmision. Imagen capturada del video de clase. [1]	14
17.	Protocolo de Inter Bus - Data Transfer. Imagen capturada del video de clase. [1]	14
18.	Protocolo de Inter Bus - Stop Signal. Imagen capturada del video de clase. [1]	15
19.	Configuración de la Interfaz IIC del 9S12 - Diagrama. Imagen capturada del video de clase. [1]	15
20.	Configuración de la Interfaz IIC del 9S12 - Registro del modulo. Imagen capturada del video de clase. [1]	16
21.	El RTC DS1207 - Hardware. Imagen capturada del video de clase. [1]	17
22.	El RTC DS1207 - Topología. Imagen capturada del video de clase. [1]	17
23.	El RTC DS1207 - Parámetros valor mínimo, máximo y típico. Imagen capturada del video de clase. [1]	18
24.	El RTC DS1207 - Registros tiempos. Imagen capturada del video de clase. [1]	18
25.	El RTC DS1207 - Data write. Imagen capturada del video de clase. [1] . . .	19
26.	El RTC DS1207 - Data read. Imagen capturada del video de clase. [1]	19
27.	El RTC DS1207 - Slave operando, leyendo data. Imagen capturada del video de clase. [1]	19
28.	El RTC DS1207 - Diagrama CALL_DS1207. [1]	20
29.	El RTC DS1207 - Diagrama INC_ISR. [1]	20
30.	El RTC DS1207 - Diagrama READ_RTC. [1]	21

1. Comunicación serial sincrónica

A continuación se presenta una explicación de los temas que abarca el capítulo 13, para esto se utiliza como fuente principal de información los videos vistos en clase [1].

1.1. Principio de la operación de la comunicación sincrónica

En primer lugar, se considera que una de sus las principales características es que la transferencia entre datos y nodos se a un mismo reloj. Se considera que esta comunicación es libre de errores de paridad, framing, etc. Así mismo, se toma en cuenta que no se implementan protocolos de capa 1 (bit de inicio o de parada).

Además de lo mencionado, este tipo de comunicación utiliza registros desplazantes uno el transmisor y otro para el receptor. Su tendencia es operar en transferencia de datos a alto bit. Esta comunicación fue creada principalmente para elementos ON-BOARD.

1.2. La interfaz Serial Port Interface (SPI)

Para esta sección se considera lo siguiente:

1.2.1. Interfaz de hardware SPI

En primer lugar, es importante conocer el significado de SPI, este se define como *Serial Port Interface* o bien Interfaz de Puerto Serie. Ahora bien, lo que es la comunicación serial sincrónica de Interfaz de Puerto Serie se puede dar de diferentes formas:

- Enlace punto a punto.
- Enlace punto a multipunto.

Esto requiere de utilizar un protocolo estándar de capa física, en el cual se definen nodo *Master* y nodo *Slave*. Cuando se da la comunicación si un slave no es necesario para lo que son las actividades del bus SPI entonces este no interfiere.

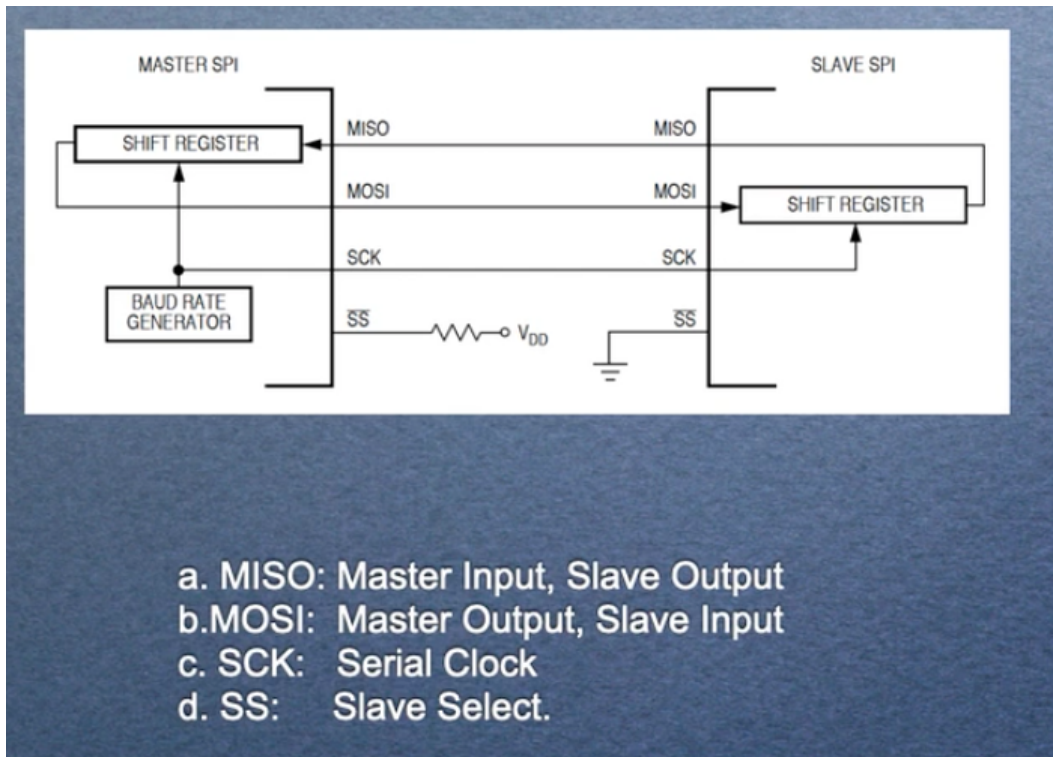


Figura 1: Interfaz de hardware SPI. Imagen capturada del video de clase [1]

Es importante considerar que en una transmisión SPI los datos que son transferidos se desplazan hacia afuera pero se reciben en dirección contraria.

Finalmente, se toma en cuenta que el SCK (reloj serie) sincroniza los desplazamientos y muestrea en dos líneas seriales, de la cual S permite seleccionar un slave individual.

1.2.2. La interfaz SPI del 9S12

Se toma en cuenta la siguiente microarquitectura:

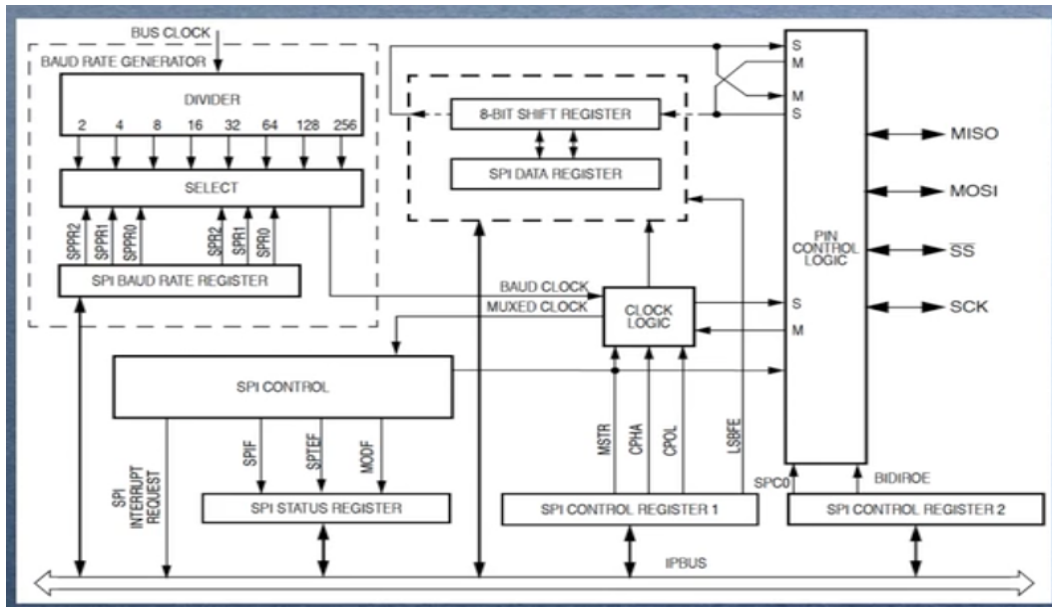


Figura 2: La interfaz SPI del 9S12. Imagen capturada del video de clase [1]

En la parte superior de la información encerrado por líneas punteadas se tiene la configuración del Baud Rate, este se define en el master, a la derecha de este se observa lo que es el registro desplazante. Funciona transmitiendo 8 bits y recibiendo 8 bits, bit a bit.

Sobre el registro desplazante no se escribe, sin embargo, se posee un registro que se llama registro de datos el cual lleva lo que transmite el slave. A.í mismo si observamos la parte inferior se observan dos diferentes registros de control así como el controlador SPI.

A continuación se puede observar como están configurados los diferentes registros con sus puertos (interconexión).

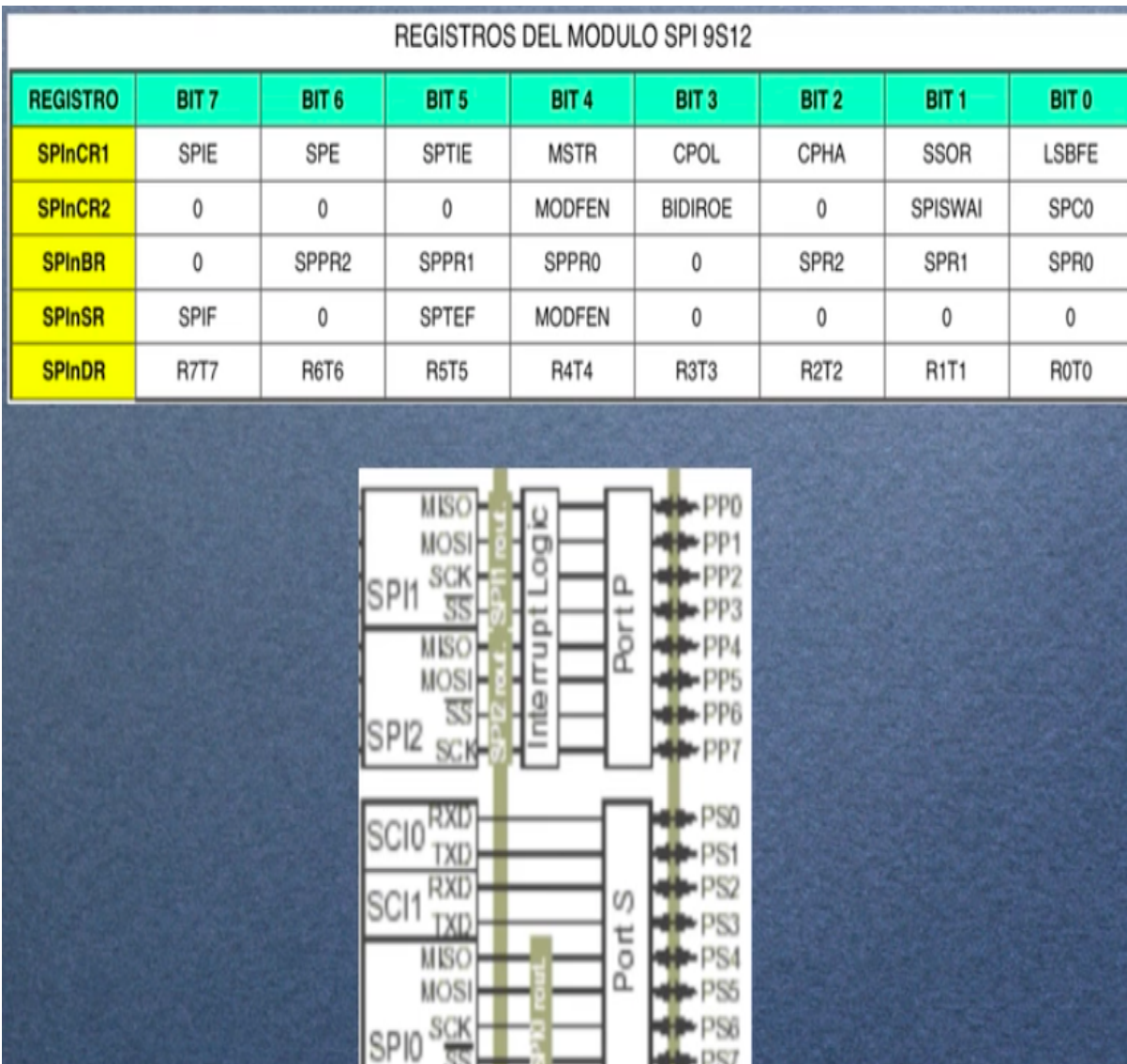


Figura 3: La interfaz SPI del 9S12 - Registros. Imagen capturada del video de clase [1]

1.2.3. Configuración de la Interfaz SPI del S12

La configuración básica se da de la siguiente forma: quien administra es el master (es el SPI) y el slave (es el dispositivo periférico externo) es quien responde a las solicitudes del master. A continuación se enumeran algunos ejemplos de dispositivos periféricos [1]:

1. Sistemas RTC.
2. Convertidores A/D y D/A/
3. Interfaces de bluetooth.
4. Interfaces de enlace TCP/IP ethernet y wi-fi.

Para estos dispositivos es importante considerar que se de aplicar el protocolo de cada dispositivo con el master, esto debe ser considerado por el programador para la capa 2.

Algunas de las funciones de esta comunicación:

- Enviar información de la configuración a los diferentes registros de control del periférico.
- Recibir información del estatus del periférico.
- Enviar datos al periférico.
- Recibir datos del periférico.

Además de esto se consideran los registros de modo que:

1. SPE: habilita el SPI que se va a utilizar.
2. MSTR: indica que el S12 funciona como master.
3. LSBFE: indica el formato el cual es LSB primero/último para transmisión de datos.
4. SSOE: habilita SS como salida (en el master).
5. SPIxBR: contiene lo que es la información del bit rate de las comunicaciones.

Se considera que el bit rate viene dado por:

$$bit_rate = \frac{Bus_{clk}}{(SPPR + 1) * (2^{SPPR+1})} \quad (1)$$

1.2.4. Configuración en Modo Bidireccional

Esta configuración permite utilizar un medio de comunicación *peer to peer a dos hilos* [1]. Funciona mediante una línea de datos, el bit MSTR define el que se va a utilizar. Esto se da cuando SPC0=1.

A continuación se adjunta una imagen captura del video de clase la cual permite esclarecer el funcionamiento de SPC0 y la decisión de MSTR.

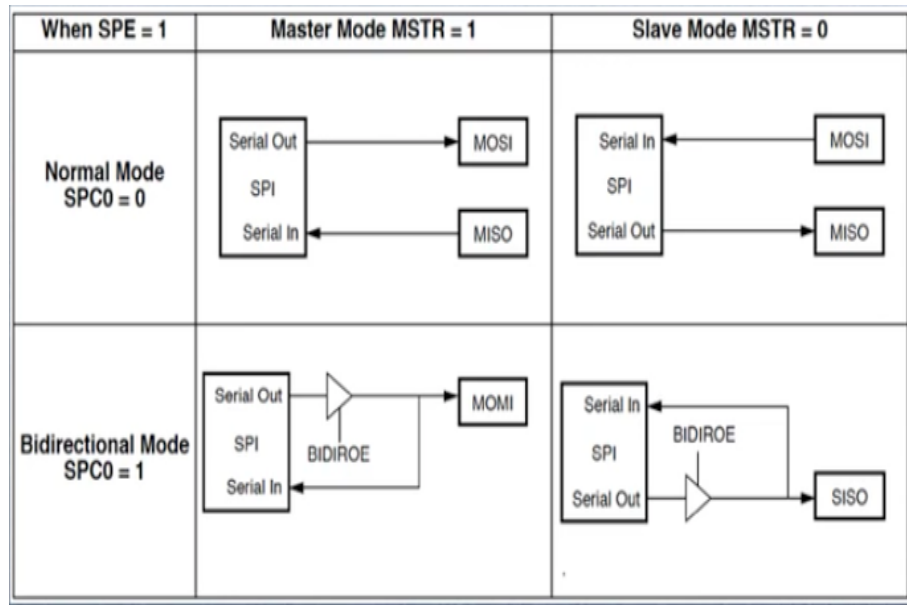


Figura 4: Configuración en Modo Bidireccional - Funcionamiento definido por BIDIROE. Imagen capturada del video de clase [1]

1.2.5. El LTC1661

Es un convertidor digital analógico de doble salida cuyo funcionamiento se explica en el siguiente diagrama de bloques:

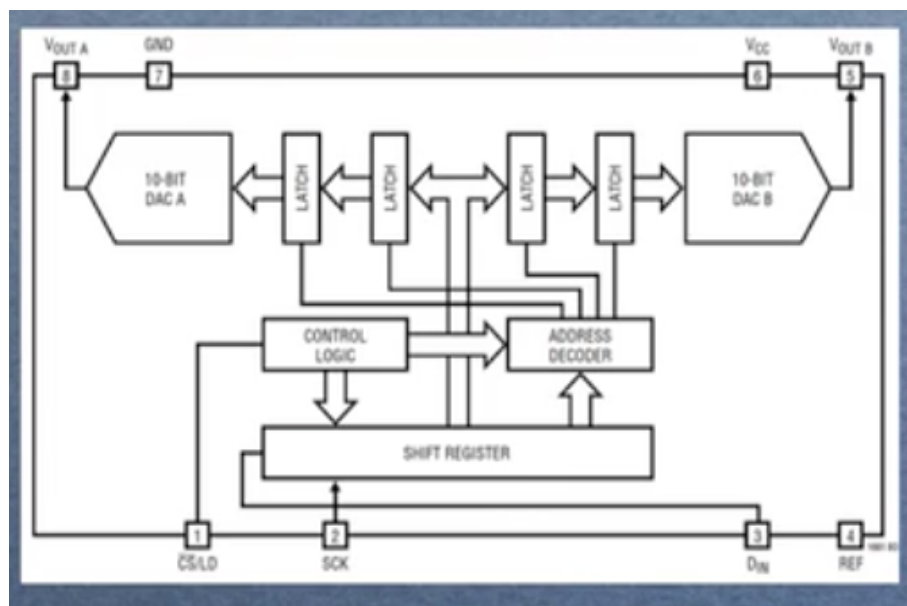


Figura 5: El LTC1661 - Funcionamiento diagrama de bloques. Imagen capturada del video de clase [1]

A continuación se muestra como están configurados los bits:

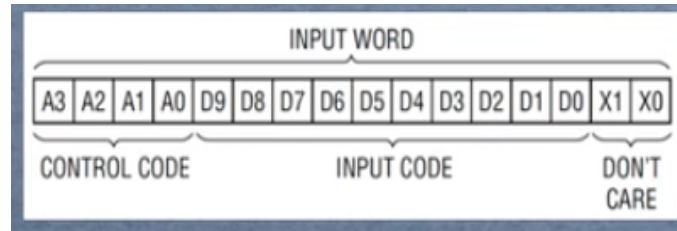


Figura 6: El LTC1661 - Configuración bits. Imagen capturada del video de clase [1]

Del mismo, se adjunta una tabla que muestra el estatus de los registros, el estado de la fuente de energía y los comentarios respectivos. Esta información se encuentra en inglés pero se considera que dado los efectos del curso y el nivel de conocimiento previo necesario se poseen las herramientas para poder entender el contenido en el idioma original de la fuente de información.

CONTROL				INPUT REGISTER STATUS	DAC REGISTER STATUS	POWER-DOWN STATUS (SLEEP/WAKE)	COMMENTS
A3	A2	A1	A0				
0	0	0	0	No Change	No Update	No Change	No Operation. Power-Down Status Unchanged (Part Stays in Wake or Sleep Mode)
0	0	0	1	Load DAC A	No Update	No Change	Load Input Register A with Data. DAC Outputs Unchanged. Power-Down Status Unchanged
0	0	1	0	Load DAC B	No Update	No Change	Load Input Register B with Data. DAC Outputs Unchanged. Power-Down Status Unchanged
0	0	1	1	Reserved			
0	1	0	0	Reserved			
0	1	0	1	Reserved			
0	1	1	0	Reserved			
0	1	1	1	Reserved			
1	0	0	0	No Change	Update Outputs	Wake	Load Both DAC Regs with Existing Contents of Input Regs. Outputs Update. Part Wakes Up
1	0	0	1	Load DAC A	Update Outputs	Wake	Load Input Reg A. Load DAC Regs with New Contents of Input Reg A and Existing Contents of Reg B. Outputs Update. Part Wakes Up
1	0	1	0	Load DAC B	Update Outputs	Wake	Load Input Reg B. Load DAC Regs with Existing Contents of Input Reg A and New Contents of Reg B. Outputs Update. Part Wakes Up
1	0	1	1	Reserved			
1	1	0	0	Reserved			
1	1	0	1	No Change	No Update	Wake	Part Wakes Up. Input and DAC Regs Unchanged. DAC Outputs Reflect Existing Contents of DAC Regs
1	1	1	0	No Change	No Update	Sleep	Part Goes to Sleep. Input and DAC Regs Unchanged. DAC Outputs Set to High Impedance State
1	1	1	1	Load DACs A, B with Same 10-Bit Code	Update Outputs	Wake	Load Both Input Regs. Load Both DAC Regs with New Contents of Input Regs. Outputs Update. Part Wakes Up

Figura 7: El LTC1661 - Estatus Registro, fuente de energía y comentarios. Imagen capturada del video de clase [1]

Así mismo se muestra un esquema de arquitectura y el waveform:

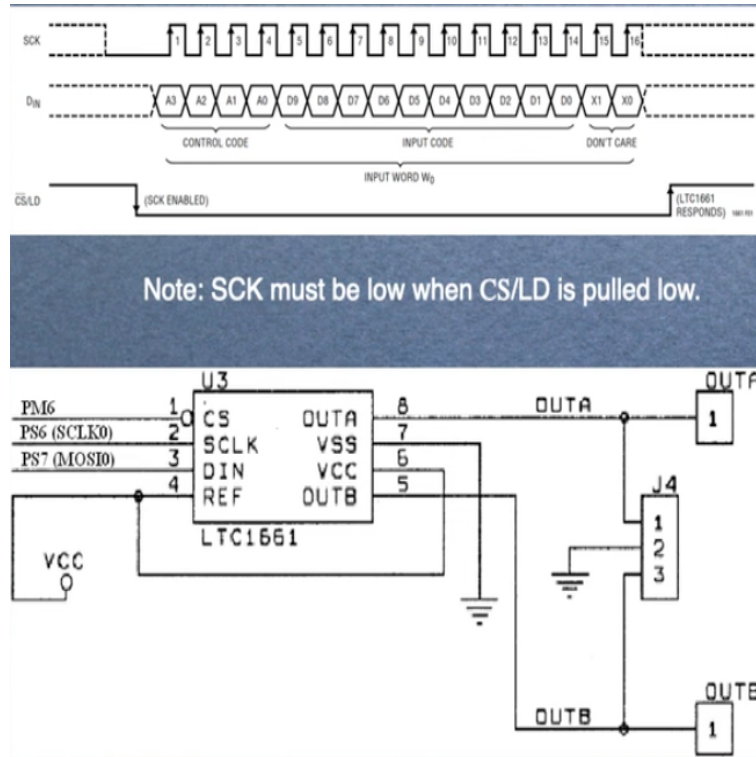


Figura 8: El LTC1661 - Waveform. Imagen capturada del video de clase [1]

1.2.6. Generador diente de sierra

Para esta sección se considera el ejemplo expuesto en los videos de clase. El cual corresponde a la implementación de un generador de diente de sierra con un periodo de onda de 10 s. Se utiliza la interrupción RTI para lograr incrementar el contador de 10 bits (CONT_DA), incrementa cada 10 ms. El valor de CONT_DA se transmite utilizando el canal A. Además, se considera una frecuencia de transmisión de 75 Kbps.

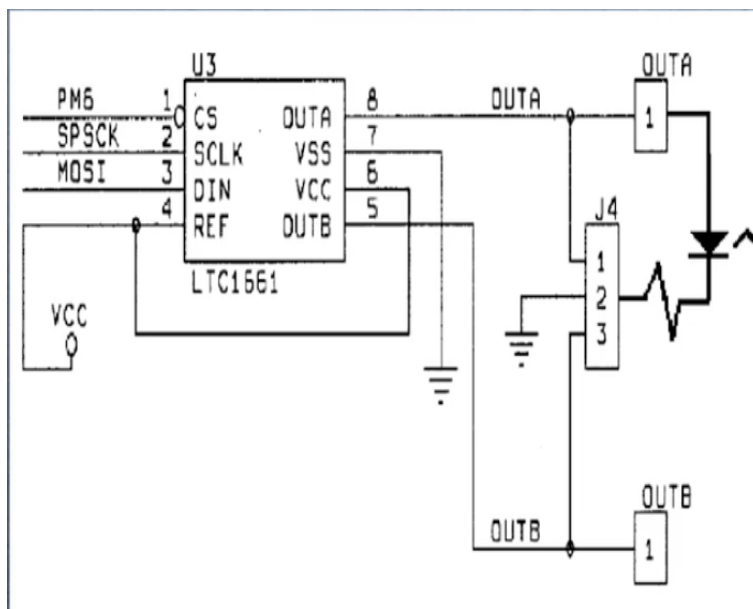


Figura 9: Generador diente de sierra - Esquema de construcción. Imagen capturada del video de clase [1]

Ahora bien, el diseño fue realizado considerando la siguiente tabla de registros:

REGISTROS DEL MODULO SPI9S12								
REGISTRO	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
SPIxCR1	SPIE	SPE	SPTIE	MSTR	CPOL	CPHA	SSOR	LSBFE
SPIxCR2	0	0	0	MODFEN	BIDIROE	0	SPISWAI	SPC0
SPIxBR	0	SPPR2	SPPR1	SPPR0	0	SPR2	SPR1	SPR0
SPIxSR	SPIF	0	SPTEF	MODFEN	0	0	0	0
SPIxDR	R7T7	R6T6	R5T5	R4T4	R3T3	R2T2	R1T1	R0T0

Figura 10: Generador diente de sierra - Registros. Imagen capturada del video de clase [1]

Finalmente, se considera el diagrama de flujo diseñado por el profesor Geovanni Delgado:

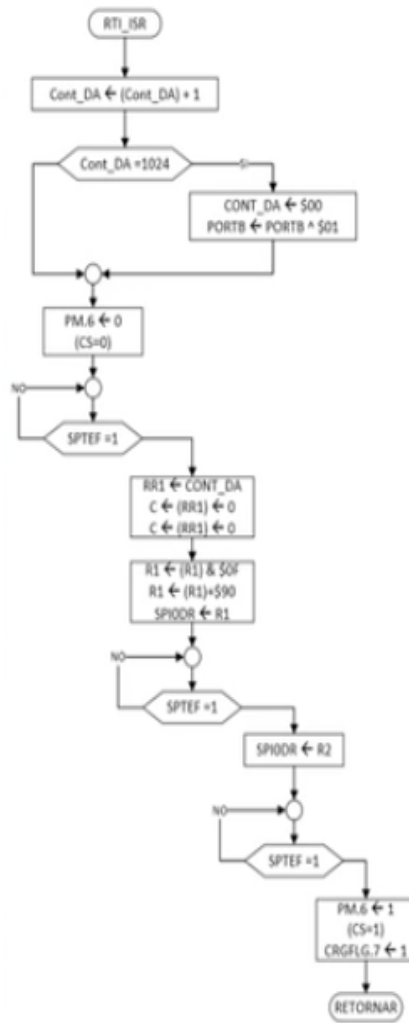


Figura 11: Generador diente de sierra - Diagrama de flujo solución ejercicio. Imagen capturada del video de clase [1]

1.3. La Interfaz IIC

Para esta interfaz es importante considerar el esquema de construcción. Opera con dos hilos, sirve para la comunicación entre circuitos integrados de corta distancia, típicamente componente on board.

1.3.1. Interfaz de hardware de Inter Bus

De la figura 12, se observa el hilo superior SCL y el hilo inferior SDA siendo esto las dos líneas o bien hilos. Utiliza una Y alambrada, en cada nodo se posee un drive para el clock y otro para los datos dado que esto hace una Y alambrada. Si todos los transistores están en bajo esa línea está alto lo que permite leer la entrada. Si un nodo quiere poseer el control este debe saturar el transistor y llevar la línea a cero para que esto pueda ser leído,

asimismo, si se reduce la capacitancia de carga considerando las resistencias de pull up se pueden mejorar los tiempos aumentando las velocidades.

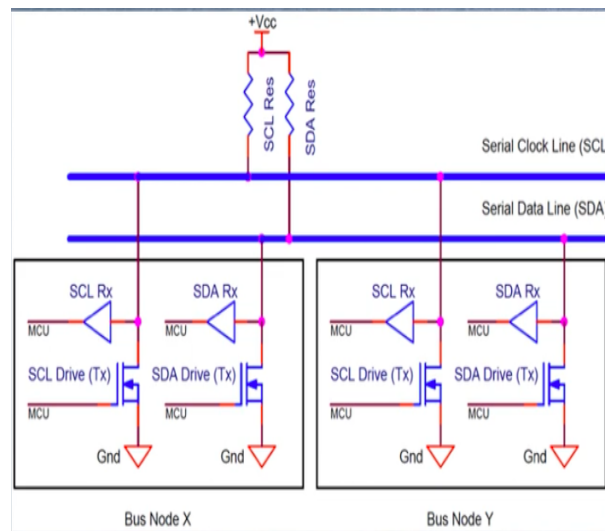


Figura 12: Interfaz de hardware de Inter Bus - Esquema de construcción. Imagen capturada del video de clase. [1]

En una red multidrop puede existir más de un master como se observa en la siguiente imagen:

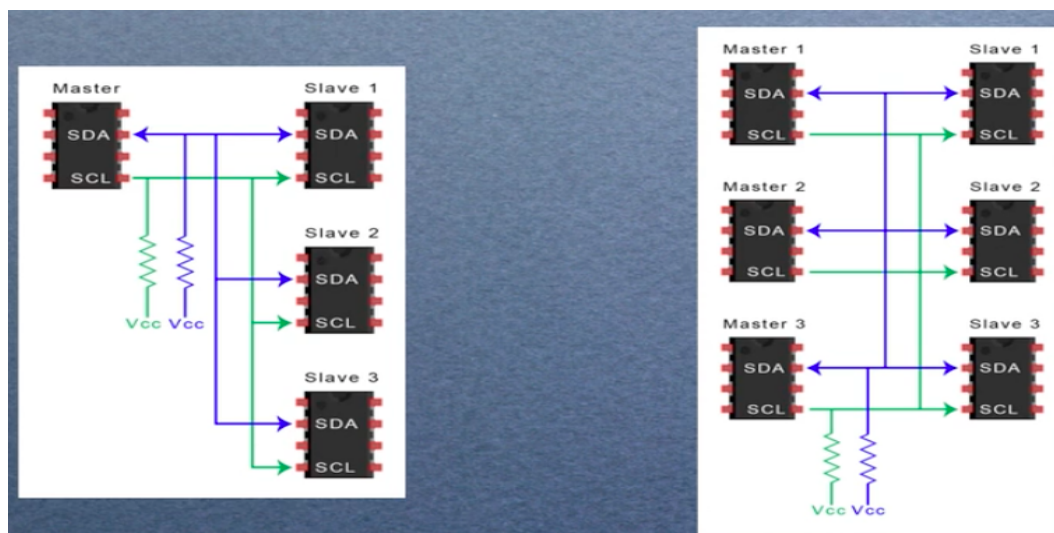


Figura 13: La Interfaz IIC - Más de un master. Imagen capturada del video de clase. [1]

1.3.2. Protocolo de Inter Bus

Las 4 partes que conforman una comunicación estándar IIC son:

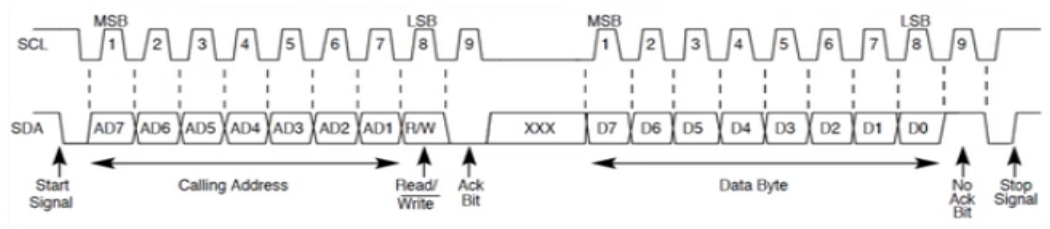


Figura 14: Protocolo de Inter Bus - 4 partes. Imagen capturada del video de clase. [1]

1. Start Signal

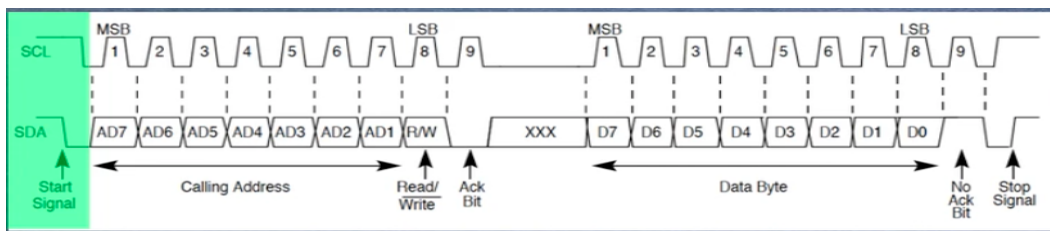


Figura 15: Protocolo de Inter Bus - Start. Imagen capturada del video de clase. [1]

2. Slave Address Transmission

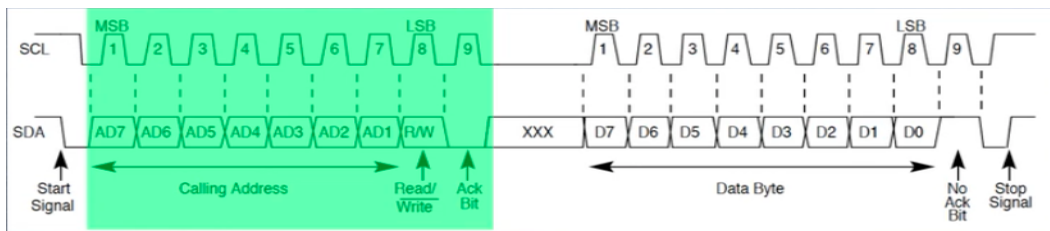


Figura 16: Protocolo de Inter Bus - Slave Address Transmission. Imagen capturada del video de clase. [1]

3. Data Transfer

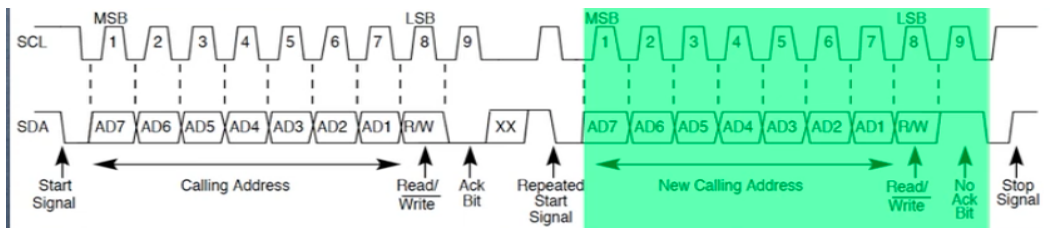


Figura 17: Protocolo de Inter Bus - Data Transfer. Imagen capturada del video de clase. [1]

4. Stop Signal

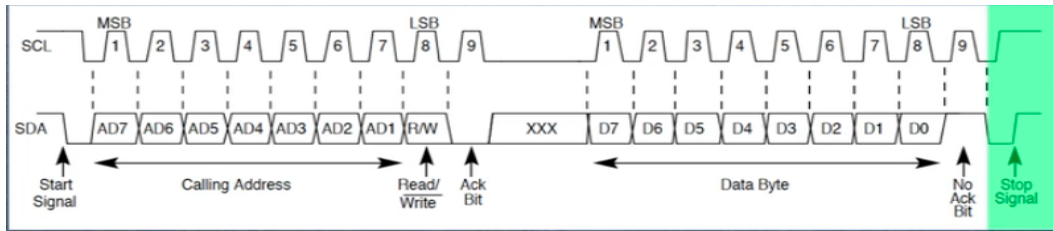


Figura 18: Protocolo de Inter Bus - Stop Signal. Imagen capturada del video de clase. [1]

1.3.3. Configuración de la Interfaz IIC del 9S12

Para esta configuración se sigue el siguiente esquema:

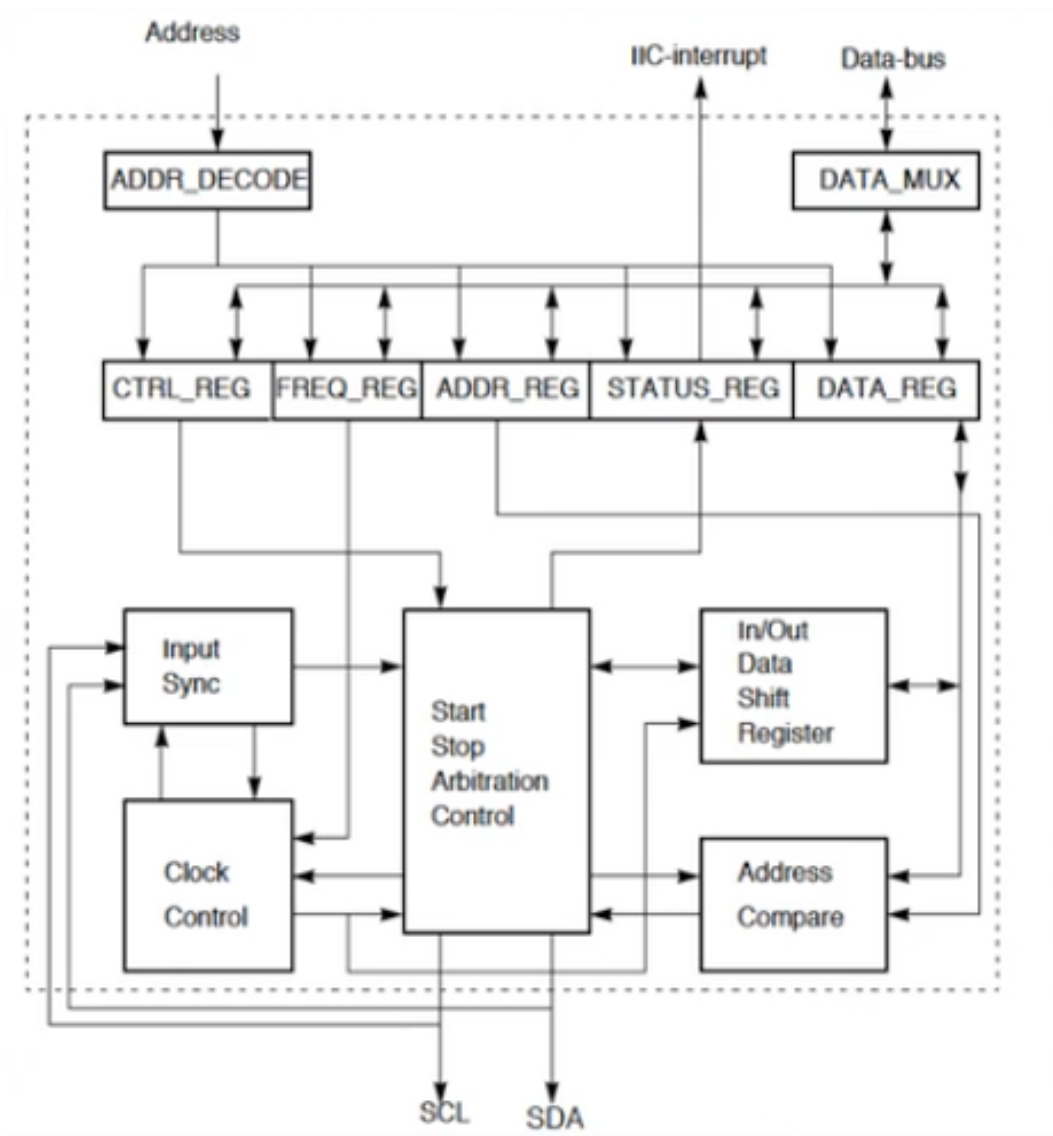


Figura 19: Configuración de la Interfaz IIC del 9S12 - Diagrama. Imagen capturada del video de clase. [1]

Así como la siguiente tabla de registro del modulo:

REGISTROS DEL MODULO IIC 9S12								
REGISTRO	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
IIC-Bus Address Register (IBAD)	ADR7	ADR6	ADR5	ADR4	ADR3	ADR2	ADR1	
IIC-Bus Frequency Divider Register (IBFD)	IBC7	IBC6	IBC5	IBC4	IBC3	IBC2	IBC1	IBC0
IIC-Bus Control Register (IBCR)	IBEN	IBIE	MS/SL	Tx/Rx	TXAK	RSTA	0	IBSWAI
IIC-Bus Status Register (IBSR)	TCF	IAAS	IBB	IBAL	0	SRW	IBIF	RXAK
IIC-Bus Data I/O Register (IBDR)	D7	D6	D5	D4	D3	D2	D1	D0

Figura 20: Configuración de la Interfaz IIC del 9S12 - Registro del modulo. Imagen capturada del video de clase. [1]

Se considera que para SRW (Slave Read/Write) cuando su valor es cero el slave recibe y el master le escribe al slave. Caso contrario, el slave transmite y el master lee del slave.

Para concluir con esta sección se considera la fórmula para calcular el SCL Divider:

$$SCLDivider = \frac{CPU \text{ Bus Frequency}}{IIC \text{ Bus SerialClock Frequency}} \quad (2)$$

1.3.4. El RTC DS1207

En primer lugar, se define RTC como Real Time Calendar con interfaz IIC la cual permite desplegar a información sobre 7 registros de 1 byte.

1. Segundos.
2. Minutos.
3. Horas.
4. Día de la semana.
5. Fecha.
6. Mes.
7. Año.

A continuación se muestra una imagen que se presenta en el video de clase la cual ejemplifica el hardware:

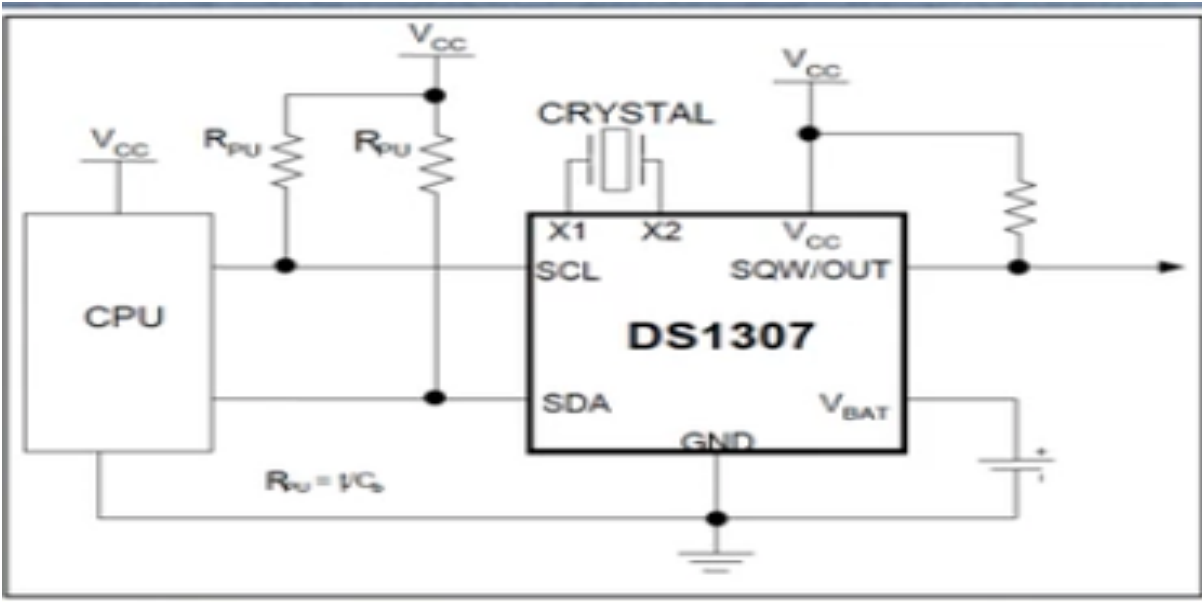


Figura 21: El RTC DS1207 - Hardware. Imagen capturada del video de clase. [1]

También a nivel topológico:

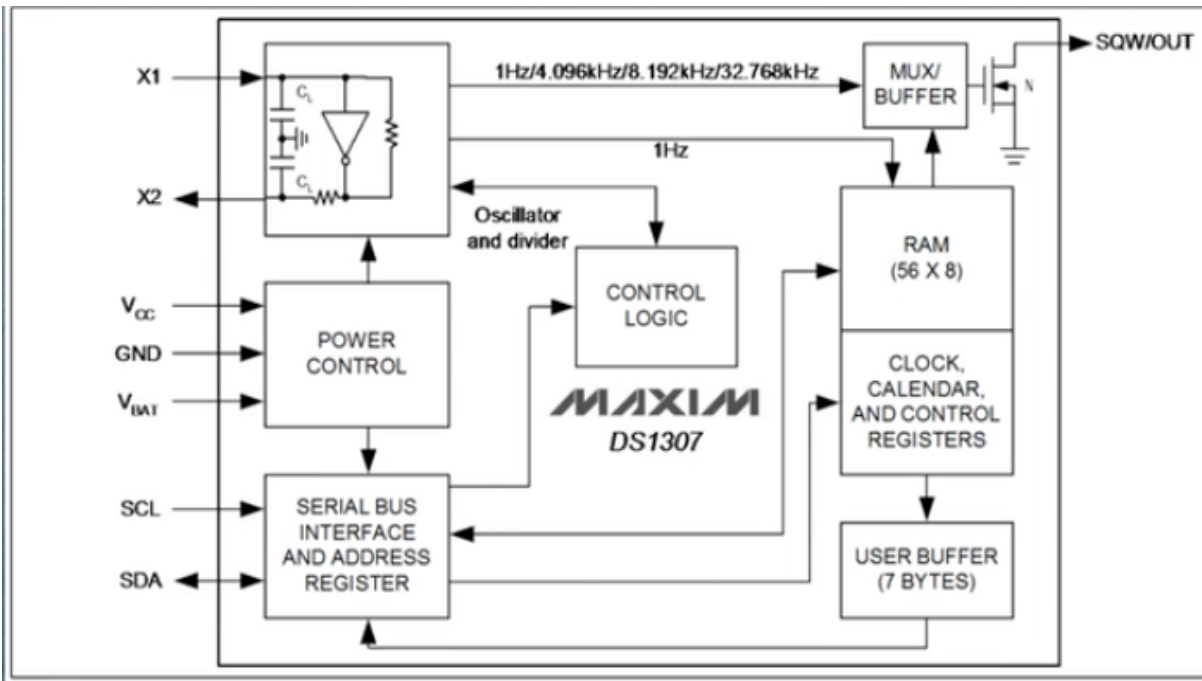


Figura 22: El RTC DS1207 - Topología. Imagen capturada del video de clase. [1]

Asimismo, se presenta una tabla con los valor mínimo, máximo y típico de los diferentes parámetros. Es de suma importancia prestarle atención al parámetro marcado en amarillo, esto de acuerdo a lo que menciona el profesor.

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
SCL Clock Frequency	f_{SCL}		0		100	kHz
Bus Free Time Between a STOP and START Condition	t_{BUF}		4.7			μs
Hold Time (Repeated) START Condition	$t_{HD,STA}$	(Note 4)	4.0			μs
LOW Period of SCL Clock	t_{LOW}		4.7			μs
HIGH Period of SCL Clock	t_{HIGH}		4.0			μs
Setup Time for a Repeated START Condition	$t_{SU,STA}$		4.7			μs
Data Hold Time	$t_{HD,DAT}$		0			μs
Data Setup Time	$t_{SU,DAT}$	(Notes 5, 6)	250			ns
Rise Time of Both SDA and SCL Signals	t_R				1000	ns
Fall Time of Both SDA and SCL Signals	t_F				300	ns
Setup Time for STOP Condition	$t_{SU,STO}$		4.7			μs

Figura 23: El RTC DS1207 - Parámetros valor mínimo, máximo y típico. Imagen capturada del video de clase. [1]

Así como los tiempos para los registros:

Table 2. Timekeeper Registers

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	FUNCTION	RANGE
00h	CH	10 Seconds			Seconds				Seconds	00–59
01h	0	10 Minutes			Minutes				Minutes	00–59
02h	0	12	10 Hour	10 Hour	Hours				Hours	1–12 +AM/PM 00–23
		24	PM/ AM							
03h	0	0	0	0	0	DAY			Day	01–07
04h	0	0	10 Date		Date				Date	01–31
05h	0	0	0	10 Month	Month				Month	01–12
06h	10 Year				Year				Year	00–99
07h	OUT	0	0	SQWE	0	0	RS1	RS0	Control	—
08h–3Fh									RAM 56 x 8	00h–FFh

0 = Always reads back as 0.

Figura 24: El RTC DS1207 - Registros tiempos. Imagen capturada del video de clase. [1]

El acomodo de los bit para el caso de escribir la data cuando el slave recibe:

Figure 4. Data Write—Slave Receiver Mode

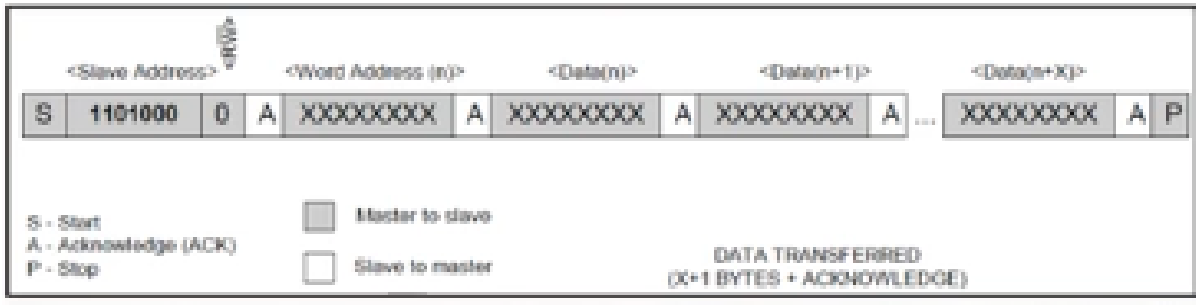


Figura 25: El RTC DS1207 - Data write. Imagen capturada del video de clase. [1]

Para el caso de lectura de la data y el slave transmite:

Figure 5. Data Read—Slave Transmitter Mode

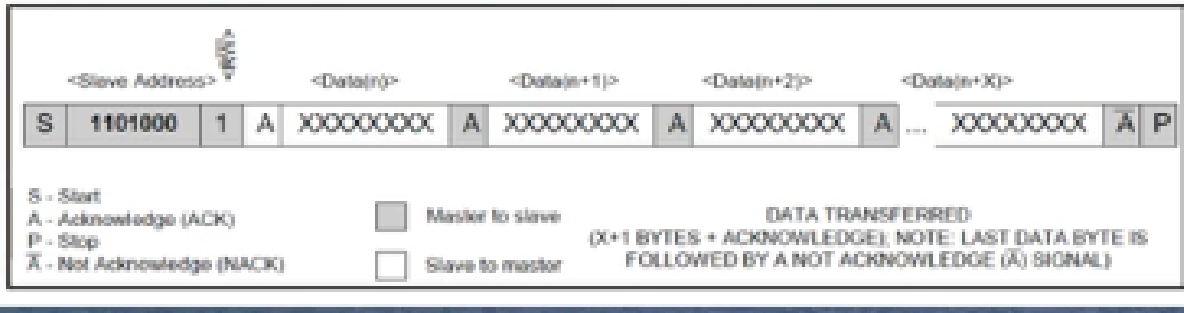


Figura 26: El RTC DS1207 - Data read. Imagen capturada del video de clase. [1]

Ahora bien, cuando se encuentra operando y leyendo la data.

Figure 6. Data Read (Write Pointer, Then Read)—Slave Receive and Transmit

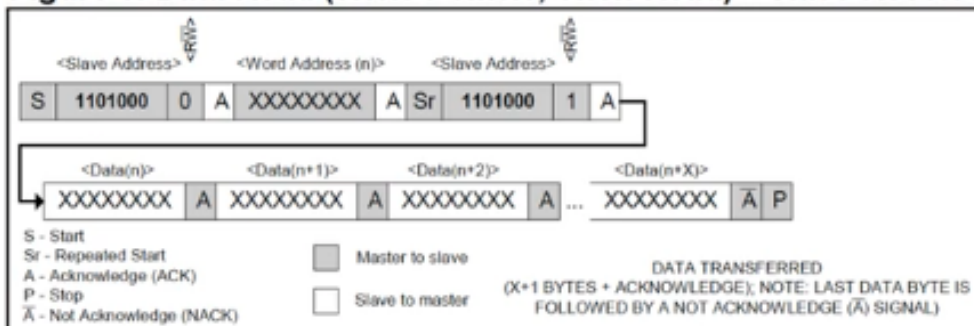


Figura 27: El RTC DS1207 - Slave operando, leyendo data. Imagen capturada del video de clase. [1]

Finalmente se muestran 3 diagramas explicados por el profesor los cuales se dan a entender

con leer la simbología considerando toda la practica reiterativa que se ha tenido a lo largo del curso respecto al diseño de diagramas de flujo.

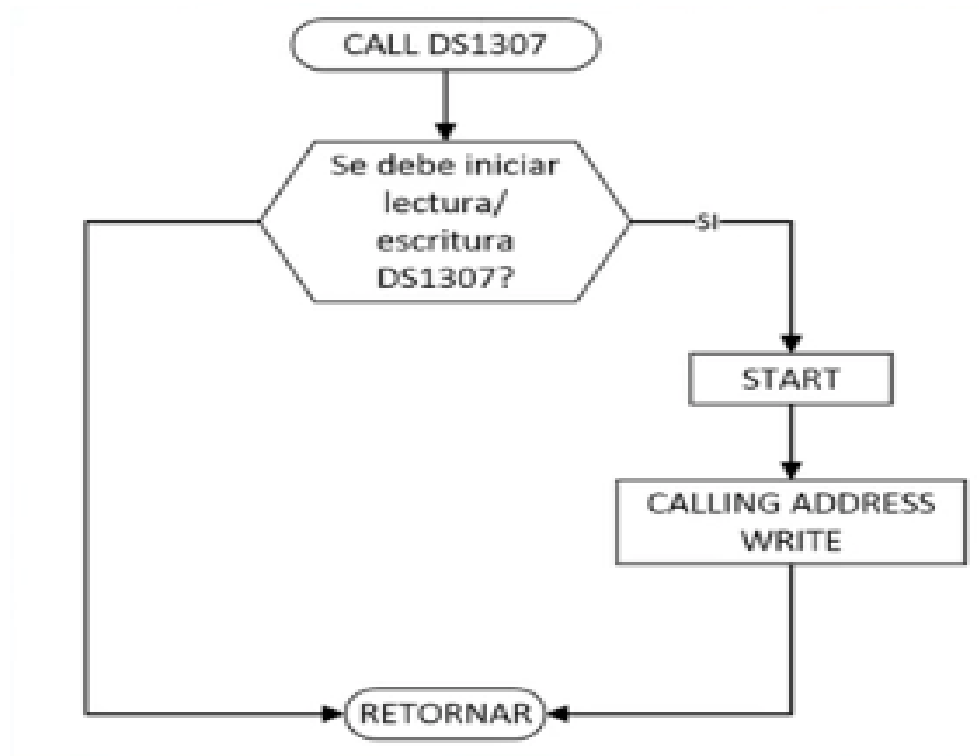


Figura 28: El RTC DS1207 - Diagrama CALL_DS1207. [1]

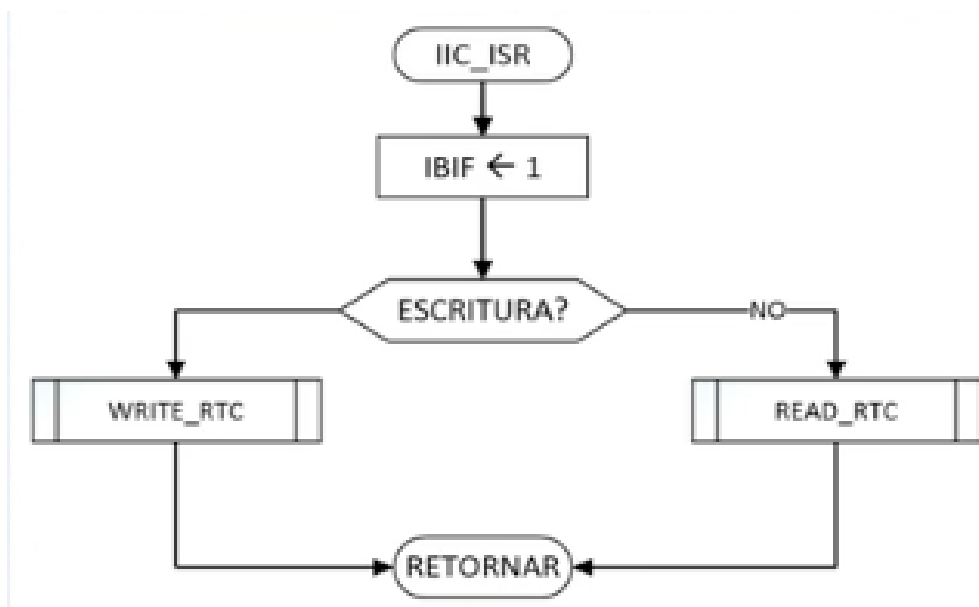


Figura 29: El RTC DS1207 - Diagrama INC_ISR. [1]

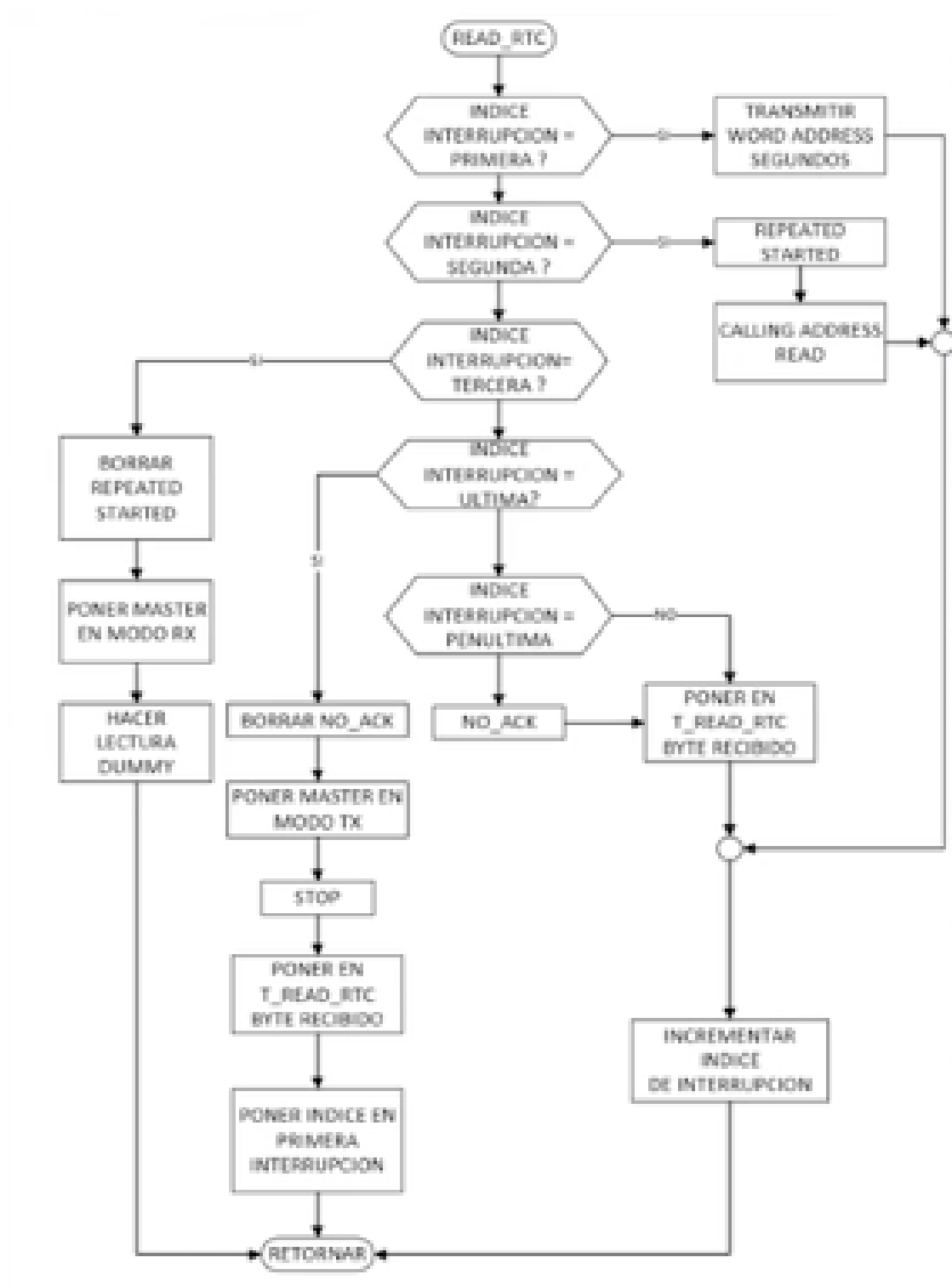


Figura 30: El RTC DS1207 - Diagrama READ_RTC. [1]

2. Cuatro productos o aplicaciones comerciales que utilicen este tipo de comunicación

1. Los ventiladores de laptop.
2. Sensores de temperatura
3. Los giroscopios.
4. SD cards, modo bajo de operación funciona con el protocolo SPI.

Cualquier sensor de bajo ancho de banda, quienes no transmitan mucha información puede funcionar bajo IIC. Los motivos pueden ser porque suele ser barata, no requieren muchos cables. Con los mismos dos cables uno conecta varios dispositivos sobre las mismas líneas por lo que permite ahorrar espacio que se requiere para colocar más cables.

Referencias

- [1] Delgado. G. *Material brindado en el curso.*