Datasheet I2C-Master

Arnau Quintana Llorens

8 de juny del 2020

# Introducció

L’I2C és un bus sèrie bidireccional de dos cables que ofereix una manera simple i eficient d’intercanviar dades entre dispositius. És més apropiat per comunicacions de curta distància.

La interfície suporta 5 velocitats:

* Estàndard: 100Kbits/s.
* Ràpida: 400Kbits/s.
* Ràpida +: 1Mbit/s.
* Molt ràpida: 3.4Mbits/s.
* Ultra ràpida: 5Mbits/s (unidireccional).

En el Màster descrit en aquest document només es suporta una velocitat de 400Kbits/s, que és a la que treballen la majoria de sensors I2C.

Característiques:

* Totalment sintetitzable.
* Suporta adreces de 7 bits.
* Suporta l’enviament i rebuda d’un i dos bytes.
* Opera amb una freqüència d’entrada de rellotge de 50MHz.
* Genera les condicions d’’Start’ i ‘Stop’ així com els ’ACK’ i ‘NACK’.
* S’inicialitza de manera externa.
* Treballa amb una entrada de rellotge extern de 50MHz.
* Dissenyat per treballar amb sensors de temperatura del tipus del LM75x, modificable de manera senzilla per treballar amb sensors com ADT7410 o del MCP9808.

# Ports E/S

|  |  |  |  |
| --- | --- | --- | --- |
| Senyal | Num bits | I/O | Descripció |
| Clk\_in | 1 | I | Rellotge del Master |
| Rst | 1 | I | Reset asíncron, actiu per nivell baix |
| Adr | 7 | I | Adreça del slave |
| Data\_in | 8 | I | Dades que es voldran escriure |
| Data\_in2 | 8 | I | Segon byte de dades, en cas d’haver-hi, per escriure |
| Start | 1 | I | Indica un start al sistema |
| Pointer | 8 | I | Adreça del registre del slave |
| Set\_pointer | 1 | I | Si és 1, indica que es vol fixar l’adreça del Slave al que ens volem adreçar. |
| R\_W | 1 | I | Indica si es vol llegir o escriure |
| Ready | 1 | O | Bandera que informa de si la màquina està llesta |
| Error | 1 | O | Bandera que ens indica que hi ha hagut un error |
| Data\_out | 8 | O | Dades que es llegeixen |
| Data\_valid | 1 | O | Bandera que indica que les dades a la sortida són vàlides |
| Sda | 1 | I/O | Línia de dades serial |
| Scl | 1 | O | Rellotge serial |

Tant Sda com Scl estan connectades a una resistència de pull-up.

# Registres

## Llista de registres

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Bits | Accés | Descripció |
| Save\_datain | 8 | RW | Guarda el primer o únic byte dades. |
| Saev\_datain2 | 8 | RW | Guarda el segon, em cas d’haver-hi, byte de dades. |
| Save\_adr | 8 | RW | Guarda l’adreça del Slave juntament amb el bit de WR/RD |
| Save\_pointer | 8 | RW | Guarda el valor del registre al que ens volem dirigir |
| Return | 1 | RW | El valor que guarda indica si s’ha de repetir ‘start’. |

## Descripció dels registres

### 3.2.1 Save\_datain

|  |  |  |  |
| --- | --- | --- | --- |
| Bit # | Accés | Pointer | Descripció |
| 7:0 | RW | 01 | 7:5 → 3 zeros.  4:3 → Fault Queue. Faltes necessàries abans de configurar la sortida O.S. per evitar efectes del soroll.  2 → O.S. Polarity. O.S. és una sortida de drenador obert per totes les condicions.  1 → 0 = Mode comparador. 1 = Mode interrupció.  0 → 1 = Sensor passa al mode shutdown. |
| RW | 10 | Bit 7 és el MSB i el 0 el LSB. |
| RW | 11 | Bit 7 és el MSB i el 0 el LSB. |

### 3.2.2 Save\_datain2

|  |  |  |  |
| --- | --- | --- | --- |
| Bit # | Accés | Pointer | Descripció |
| 7:0 | RW | 10 | No definit |
| RW | 11 | No definit |

### 3.2.3 Save\_adr

|  |  |  |
| --- | --- | --- |
| Bit # | Accés | Descripció |
| 7:4 | RW | Pel tipus de sensor que està pensat han de valer 1001. |
| 3:1 | RW | Adreça del Slave. |
| 0 | RW | Bit de WR/RD. Serà 0 si es vol escriure i 1 si es vol llegir. |

### 3.2.4 Save\_pointer

|  |  |  |
| --- | --- | --- |
| Bit # | Accés | Descripció |
| 7:2 | RW | Són zeros pel tipus de sensor pel que està pensat |
| 1:0 | RW | 00: Registre que guarda les dades de temperatura (només lectura)  01: Registre de configuració (escriptura i lectura)  10: THYST (escriptura i lectura)  11: TOS (escriptura i lectura) |

Aquesta taula està feta pensant amb el sensor LM75x, que té 2 bits d’adreça de registre. En cas d’un altre que tingués més bits d’adreça, s’hauria de modificar la condició del codi per tal d’informar al sistema del comportament segons l’adreça que s’indiqui.

### 3.2.5 Return

|  |  |  |
| --- | --- | --- |
| Bit # | Accés | Descripció |
| 1 | RW | Si = 1, la màquina ha de repetir la condició d’start’. |

# Funcionament

## Configuració del sistema

Les dades es transfereixen entre el Master i l’Slave bit a bit. Cada seqüència de dades té una llargària de 8 bits. Hi ha un pols de Scl per cada bit sent el bit més significatiu el primer en enviar-se. Al byte de dades sempre el segueix un bit de ‘acknowledge’, excepte quan el master treballa com a receptor; en aquest cas, després de rebre l’últim bit, el Master envia un ‘non-acknowledge’. Els bits de dades són estables mentre Scl es troba a nivell alt. Quan Scl és 0 és quan Sda varia el seu valor, per tant, les dades no són vàlides durant aquests instants. Un canvi a Sda quan Scl es troba a nivell alt s’interpreta com una instrucció d’’Start’ o ‘Stop’.

## 4.2 Protocol I2C

Normalment, un comunicació I2C consisteix de 6 parts:

1. Condició d’’Start’.
2. Adreça del Slave.
3. ACK de l’adreça.
4. Byte de dades.
5. ACK del byte de dades.
6. Condició d’Stop.

En el Master que es presenta, es disposa de diversos camins diferents:

1. Condició d’’Start’.
2. Adreça del Slave.
3. ACK de l’adreça.

Si s’escriu:

1. Byte de Pointer.
2. ACK del Pointer.
3. Byte de dades.
4. ACK del byte de dades.

(Segon byte de dades)

(ACK del segon byte de dades)

1. Condició d’Stop.

Si es llegeix:

1. Byte de dades.
2. ACK del byte de dades.

(Segon byte de dades)

(NACK del segon byte de dades)

1. Condició d’’Stop’.

En cas que es llegeixi un sol byte, el pas 5 es convertiria en un NACK del byte de dades.

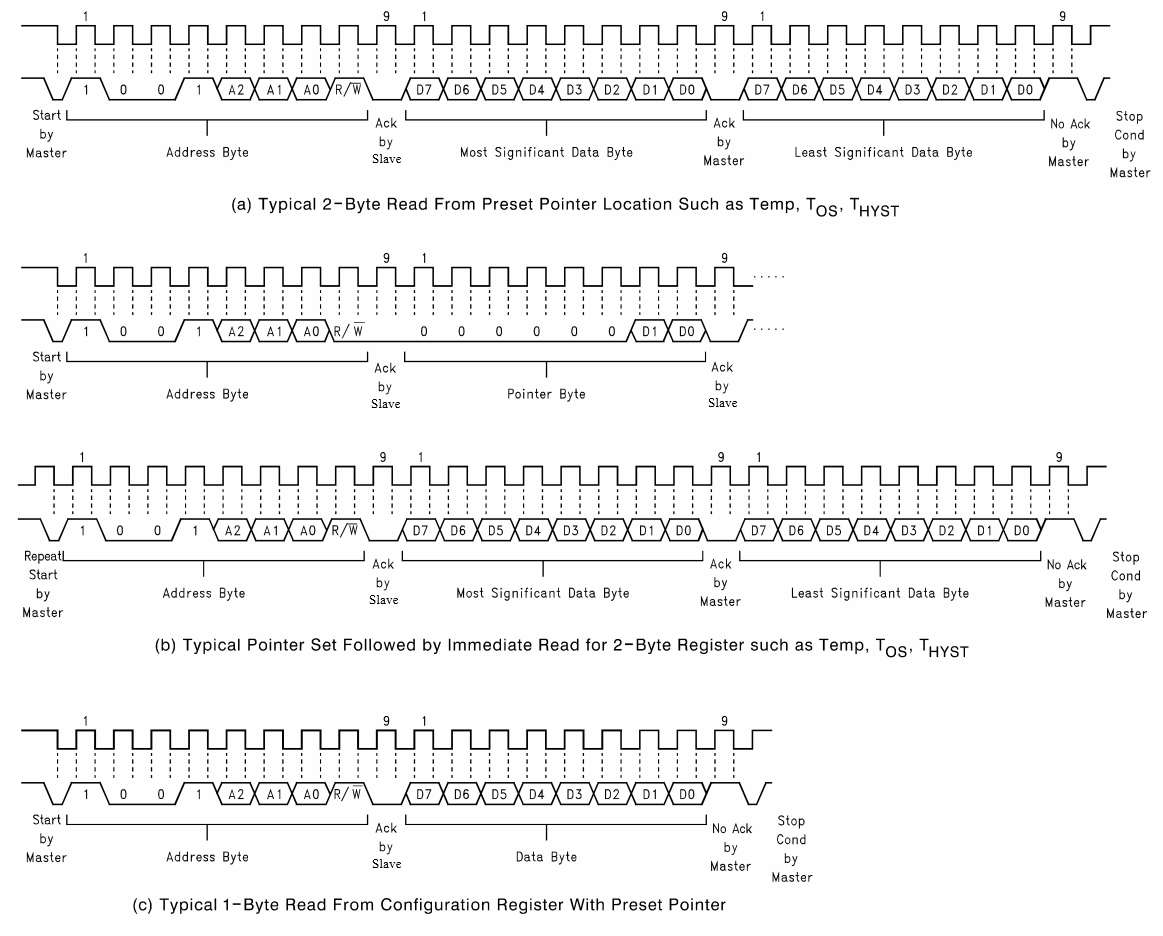
En cas de voler fixar un registre al que llegir el camí seria el següent:

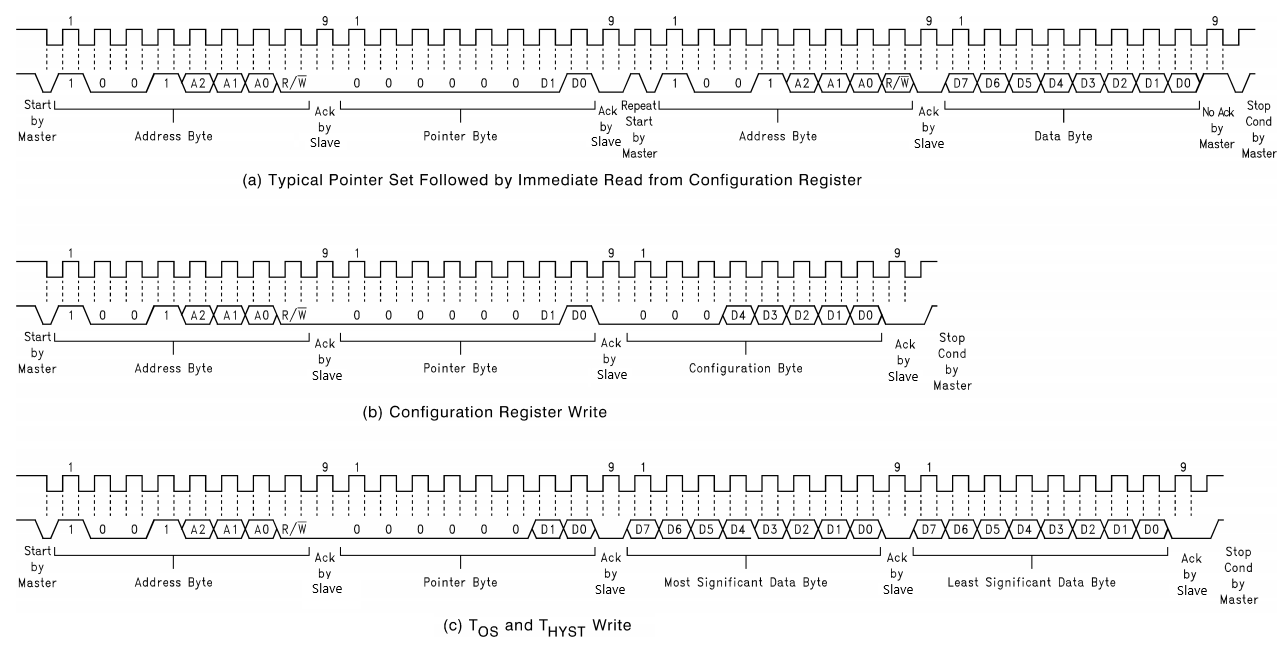
1. Condició d’’Start’.
2. Adreça del Slave.
3. ACK de l’adreça.
4. Byte de Pointer.
5. ACK del Pointer.
6. Repetició de la condició d’’Start’.
7. Adreça del Slave.
8. ACK de l’adreça.
9. Byte de dades
10. ACK del byte de dades. (NACK si es llegeix un registre d’un sol byte)

(Segon byte de dades)

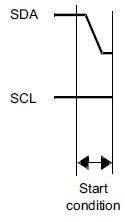
(NACK del segon byte de dades)

1. Condició d’’Stop’.

En cas que es vulgui llegir, però no hi hagi cap registre fixat al que dirigir-se, per defecte s’enviaran les dades del registre 00 del Slave (és de 2 bytes).



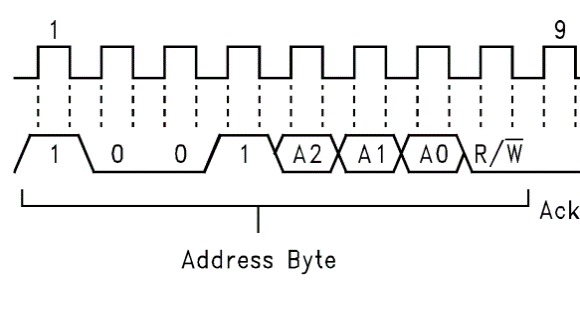
### 4.2.1 Start

El valor d’Sda només varia quan Scl és 0. Quan Sda varii mentre Scl sigui 1 s’interpreta com un Start o un Stop. La condició d’Start, tal com veiem a la figura, es tradueix en un flanc de baixada d’Sda mentre Scl és 1.

### 4.2.2 Adreça + bit de WR/RD

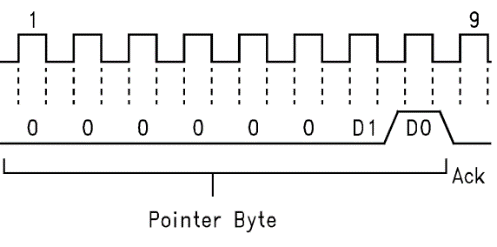
Si es vol comunicar amb un sensor, l’adreça d’aquest es defineix per hardware ja que depèn del valor que es fiqui a les entrades destinades a tal missió. Aquests 2 o 3 bits, segons el sensor, són els bits menys significatius del byte de dades de l’adreça; els primers quatre o cinc tenen el valor que ens digui el datasheet del propi sensor.

Al byte d’adreça el segueix un bit que indica si s’escriu o llegeix. A Aquest últim bit el segueix un ACK del Slave si aquesta adreça coincideix amb la del Slave. En cas de no haver cap Slave connectat a la línia amb l’adreça enviada, es rebrà un NACK.

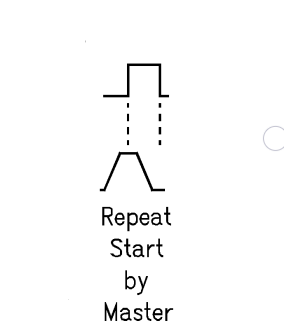


### 4.2.3 Pointer

Byte que indica el registre del Slave al que ens volem adreçar. Per enviar aquest byte, el bit de WR/RD ha de ser un 0, és a dir, s’ha d’escriure.



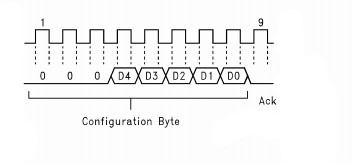
### 4.2.4 Repeat Start

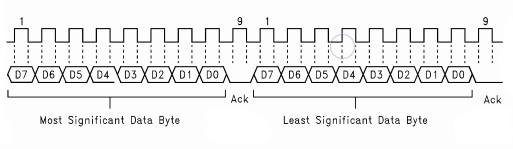
Si es vol llegir fixant primer una adreça(no es vol llegir el registre per defecte), l’entrada Set\_pointer ha de valer 1. Amb aquesta condició, quan s’hagi rebut l’ACK del Slave conforme s’ha rebut la direcció del registre, el Master repetirà la condició d’’Start’ i es tornarà al principi, és a dir, s’enviarà l’adreça de nou, però aquest cop amb el bit de WR/RD en mode lectura sempre i quan s´hagi canviat el valor del bit externament.

### 4.2.5 Byte/s de dades

#### 4.2.5.1 Escriptura

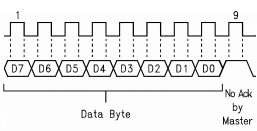
Depenent del registre al que es dirigeixi el Master es poden escriure un o dos bytes. En cas d’escriure un sol byte, després de rebre l’ACK del byte del Pointer, s’envia 8 bits de dades al quals els seguirà un ACK conforme l’’Slave’ els ha rebut. En cas de dos bytes, després del ACK del primer byte de dades, el seguirà un segon amb el corresponent ACK conforme s’ha rebut.

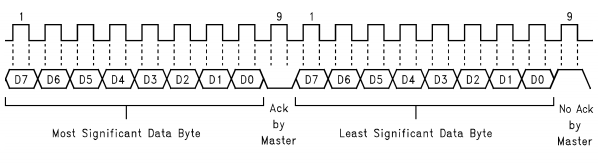




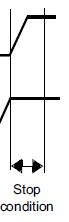
#### 4.2.5.2 Lectura

Depenent del registre al que es dirigeixi el Master es poden llegir un o dos bytes. En cas de llegir un sol byte, després de rebre l’ACK del byte de l’adreça, es reben els 8 bits de dades als quals els seguirà un NACK conforme el Master els ha rebut. En cas de dos bytes, després del ACK del primer byte de dades, el seguirà un segon amb el corresponent NACK conforme s’ha rebut.





### 4.2.6 Stop

Com s’ha comentat anteriorment, Sda només canvia quan Scl és 0, i si Sda canvia quan Scl és 1 s’interpreta com un Start o un Stop. La condició d’Stop coincideix amb un posedge mentre Scl és igual a 1.

# Arquitectura

\*FOTO\*

## Divisor de freqüència

En trobem dos: un que divideix per 25 i l’altre per 5. La sortida del primer és el rellotge del Master (2Mhz) i la sortida del segon és el rellotge d’Scl (400KHz). L’entrada del primer és l’entrada de rellotge extern de 50Mhz. L’entrada del segon és el rellotge de 2MHz.

### Senyals d’entrada i sortida del divisor per 25

|  |  |  |  |
| --- | --- | --- | --- |
| Senyal | Bits | I/O | Descripció |
| Clk\_in | 1 | I | Rellotge de 50MHz introduït externament. |
| Clk | 1 | O | Rellotge de 2MHz, controla el Master. |

### 5.1.2 Senyals d’entrada i sortida del divisor per 5

|  |  |  |  |
| --- | --- | --- | --- |
| Senyal | Bits | I/O | Descripció |
| Clk | 1 | I | Rellotge de 2MHz, controla el Master. |
| Clk\_scl | 1 | O | Rellotge de 400kHz, controla la comunicació. |

## Comptador

N’hi ha dos: un que compta fins a 8 i l’altre que compta fins a 5. El primer compta els cicles de rellotge del Scl per saber quan s’ha enviat o rebut el byte de dades. El segon compta els cicles de rellotge del Master, el qual és 5 cops més ràpid que Scl. Aquest últim indica el moment del període en que es troba la màquina en tot moment; roman sempre actiu.

### Senyals d’entrada i sortida del comptador de cicles del Clk

|  |  |  |  |
| --- | --- | --- | --- |
| Senyal | Bits | I/O | Descripció |
| En | 1 | I | Habilita el comptador, actiu per nivell alt |
| Rst | 1 | I | Reset asíncron, actiu per nivell baix |
| Clk | 1 | I | Rellotge que controla el comptador (2MHz) |
| Out | 4 | O | Quan arriba a 5, al següent cicle, treu un 1 |

### 5.2.1 Senyals d’entrada i sortida del comptador de cicles del Scl

|  |  |  |  |
| --- | --- | --- | --- |
| Senyal | Bits | I/O | Descripció |
| En | 1 | I | Habilita el comptador, actiu per nivell alt |
| Rst | 1 | I | Reset asíncron, actiu per nivell baix |
| Clk | 1 | I | Rellotge que controla el comptador (400KHz) |
| Out | 4 | O | Quan arriba a 8, al següent cicle, treu un 1 |

## 5.3 Registre de desplaçament paral·lel a sèrie

Aquest mòdul carrega les dades en paral·lel i les va traient en sèrie. Disposa d’un comptador intern per reiniciar el registre quan toca i carregar l’entrada en el moment oportú.

### 5.3.1 Senyals d’entrada i sortida

|  |  |  |  |
| --- | --- | --- | --- |
| Senyal | Bits | I/O | Descripció |
| In | 8 | I | Entrada del registre de desplaçament |
| Load | 1 | I | Si = 1, carrega dades. Si = 0, desplaça a l’esquerra |
| Rst | 1 | I | Reset asíncron, actiu per nivell baix |
| Clk | 1 | I | Rellotge que controla el registre de desplaçament (2MHz) |
| Out | 1 | O | La primera dada en sortir serà el MSB del byte |

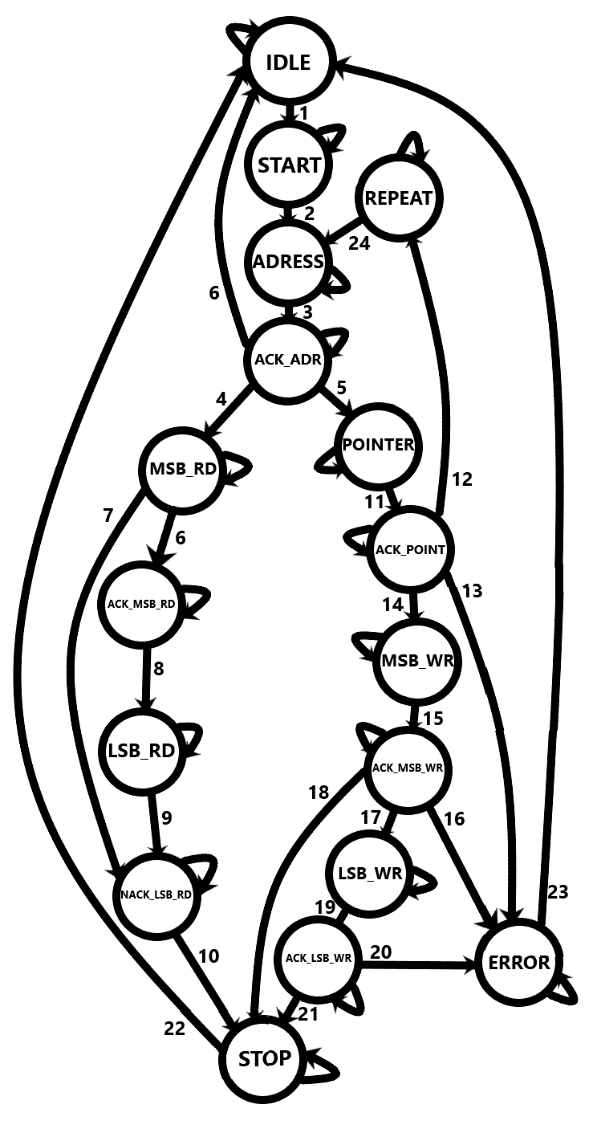
## 5.4 Registre de desplaçament sèrie a paral·lel

Aquest mòdul carrega dades en sèrie alhora que les treu en paral·lel.

### 5.4.1 Senyals d’entrada i sortida

|  |  |  |  |
| --- | --- | --- | --- |
| Senyal | Bits | I/O | Descripció |
| In | 1 | I | Entrada del registre de desplaçament |
| Load | 1 | I | Si = 1, carrega dades. Si = 0, el valor del registre es conserva |
| Rst | 1 | I | Rellotge que controla el comptador |
| Clk | 1 | I | Rellotge que controla el registre de desplaçament (2MHz) |
| Out | 8 | O | Sortida del registre de desplaçament, treu dades alhora que carrega |

## 5.5 Unitat de control



### 5.5.1 Senyals d’entrada i sortida

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Bits | I/O | Descripció |
| Clk | 1 | I | Rellotge de 2MHz (controla el Màster) |
| Clk\_scl | 1 | I | Rellotge de 400kHz (Scl) |
| Rst | 1 | I | Reset asíncron |
| Start | 1 | I | Indica de manera externa un inici de procés |
| R\_W | 1 | I | Si = 1, es llegirà. Si = 0, s’escriurà. |
| Datain\_sda | 1 | I | Porta el valor d’Sda |
| Pointer | 8 | I | Direcció del registre del Slave amb el que comunicar-se |
| Set\_pointer | 1 | I | Indica si es vol fixar el valor del registre de Pointer del Slave |
| Return | 1 | I | Indica que es vol una repetició de la condició d’’Start |
| Repeat | 1 | O | Avisa que s’està al estat de repetició d’’Start’ |
| Out\_cont\_cycle | 4 | I | Sortida del comptador de cicles del Clk |
| Out\_cont\_data | 4 | I | Sortida del comptador de cicles d’Scl |
| En\_cont\_data | 1 | O | Habilita el comptador de cicle d’Scl |
| Load\_shiftPLSR | 1 | O | Habilita el registre de desplaçament paral·lel a sèrie |
| Load\_shiftSRPL | 1 | O | Habilita el registre de desplaçament sèrie a paral·lel |
| Enable\_sda | 2 | O | Tria el valor del ‘enable’ del tri-state connectat a Sda |
| SelectPLSR | 3 | O | Tria l’entrada del registre de desplaçament paral·lel - sèrie |
| Enable\_clk | 2 | O | Escull el valor del ‘enable’ del tri-state connectat a Scl |
| Ready | 1 | O | Indica que la màquina està llesta per un nou procés |
| Data\_valid | 1 | O | Indica si les dades a la sortida són vàlides |
| Error | 1 | O | Indica si hi ha hagut un error en el procés |