

I2C bus specifications

for the CMS tracker 2S & PS modules

V 2.0

21/4/2016

Revision History

Version	Date	Comments
V1.0	22/11/2015	First draft
V2.0	21/4/2016	Device ID extended to 3 bytes. Table 2 revised to reflect the non-programmable CiC address field. Added electrical and timing specifications.

1 INTRODUCTION

The present document discusses the implementation of the I²C serial communication bus in the CMS Outer Tracker PS (Pixel-Strip) & 2S (Strip-Strip) modules. The use of the I²C bus is to monitor and control the operation of the front-end ASICs populating the PS & 2S modules. The aim of the document is to present guidelines for the development of I²C circuitry of the front-end ASICs.

It should be noted that only a subset of features of the I²C Bus standard are necessary to support the functionality of the ASICs of the PS & 2S modules and therefore the design guidelines are limited to these features. Particular attention is taken not to violate the I²C Bus standard for the supported features.

The I²C features for the PS & 2S modules include:

- ☐ Compliance to the Philips Semiconductor I²C specification v2.1
- ☐ Single master bus
- ☐ 7-bit addressing
- ☐ Single-byte & Multi-byte transfers
 - Need to define maximum number of bytes
- ☐ General Call Addressing
 - Use within a chip family (CBC, MPA, SSA, CiC)
 - Set operational modes, test pulse amplitude & delay, latency value in common for all chips
 - CBC2 “*broadcast address scheme*” (1111111+data+data+....) does not conform to the “General Call Addressing” format but can be generated using standard multi-byte Write transfers (see next slide)
- ☐ 100KHz, 400KHz & 1MHz bit rate

1.1 I²C Bus Topology

The I²C Bus Topology for the PS and 2S is shown in Figure 1. The optical link communication ASIC; namely the IpGBT (low power GigaBit Transceiver) is equipped with two independent I²C master interfaces that realize two independent I2C busses. The IpGBT I²C master interface is the only master interface permitted to connect on the bus. All front-end ASICs; namely the MPA, the SSA, the CiC, the CBC, must implement an I²C Slave interface. Generation of clock signals on the I²C-bus is always the responsibility of the master device.

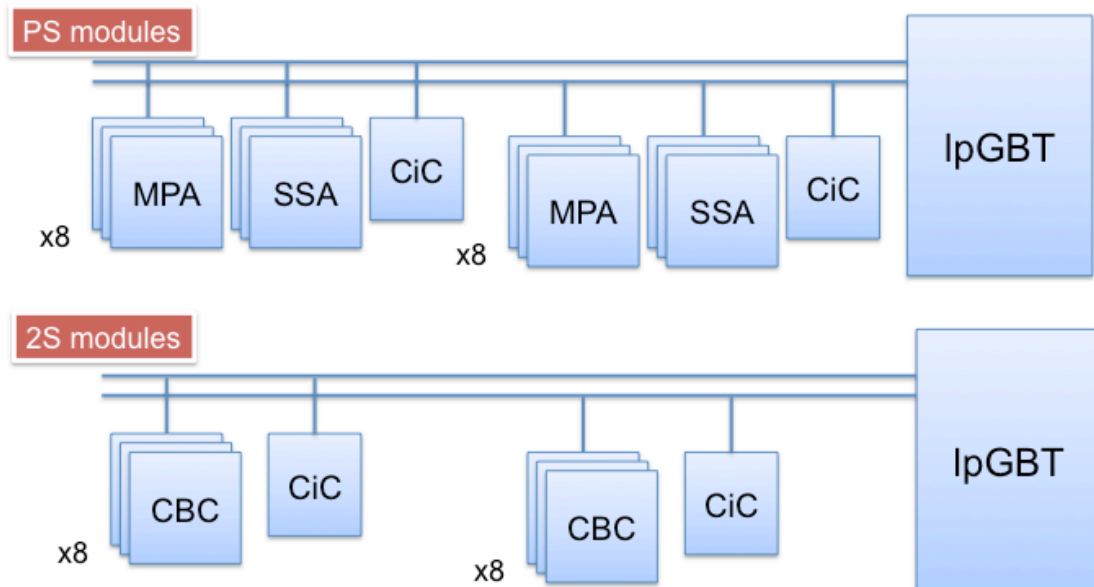


Figure 1 I2C Bus topology for PS and 2S modules.

1.2 SDA & SCL Signals

Both SDA and SCL are bidirectional lines, connected to a positive supply voltage via a current-source or pull-up resistor. When the bus is free, both lines are HIGH. The output stages of devices connected to the bus must have an open-drain to perform the wired-AND function. Data on the I2C-bus can be transferred at rates of up to 100 kbit/s in the Standard-mode, up to 400 kbit/s in the Fast-mode, up to 1 Mbit/s in Fast-mode Plus. The bus capacitance limits the number of interfaces connected to the bus. For a single master application, the master's SCL output can be a push-pull driver design or an open-drain driver design with an external pull-up resistor. There are no slave devices on the bus which would stretch the clock.

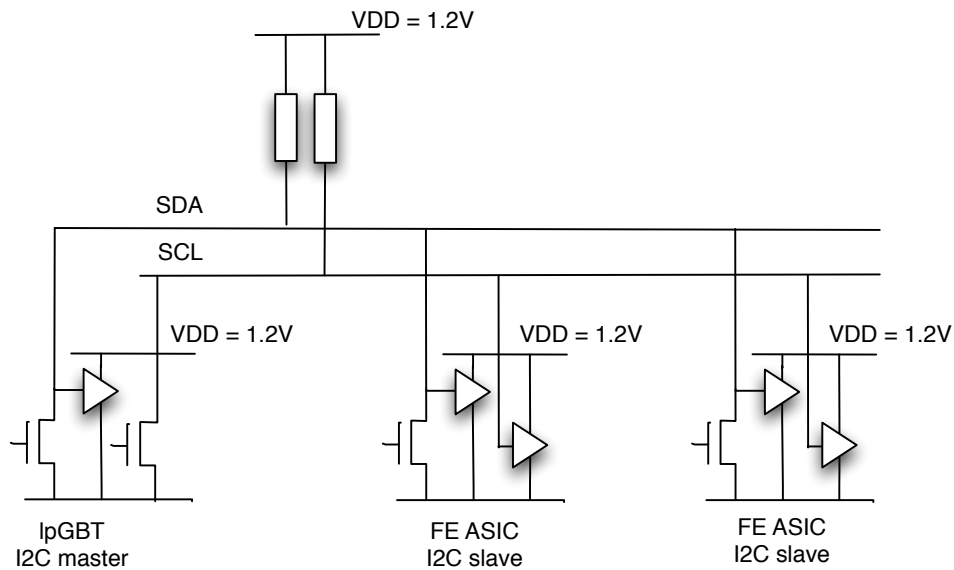


Figure 2 I2C bus connectivity

1.3 Bus clear

In the unlikely event where the clock (SCL) is stuck LOW, the preferential procedure is to reset the bus using the HW reset signal if the I²C devices have HW reset inputs. If the I²C devices do not have HW reset inputs, cycle power to the devices to activate the mandatory internal Power-On Reset (POR) circuit.

If the data line (SDA) is stuck LOW, the master should send nine clock pulses. The device that held the bus LOW should release it sometime within those nine clocks. If not, then use the HW reset or cycle power to clear the bus. The master I²C must be able to generate this “bus clear” sequence.

2 DATA TRANSFER PROTOCOL

The data transfer protocol is according to the I²C standard. The START, REPEATED START and STOP conditions as well as data transfer protocol are specified in the I2C Specification [PHIL01].

Data transfers follow the format shown in Figure 3. After the START condition (S), a slave address is sent. This address is seven bits long followed by an eighth bit which is a data direction bit (R/W) — a 'zero' indicates a transmission (WRITE), a 'one' indicates a request for data (READ). A data transfer is always terminated by a STOP condition (P) generated by the master. However, if a master still wishes to communicate on the bus, it can generate a repeated START condition (Sr) and address another slave without first generating a STOP condition.

- Note 1: Each byte is followed by an acknowledgment bit as indicated by the A or A blocks in the sequence.
- Note 2: I2C-bus compatible devices must reset their bus logic on receipt of a START or repeated START condition such that they all anticipate the sending of a slave address, even if these START conditions are not positioned according to the proper format.
- Note 3: 10-bit addressing is NOT supported

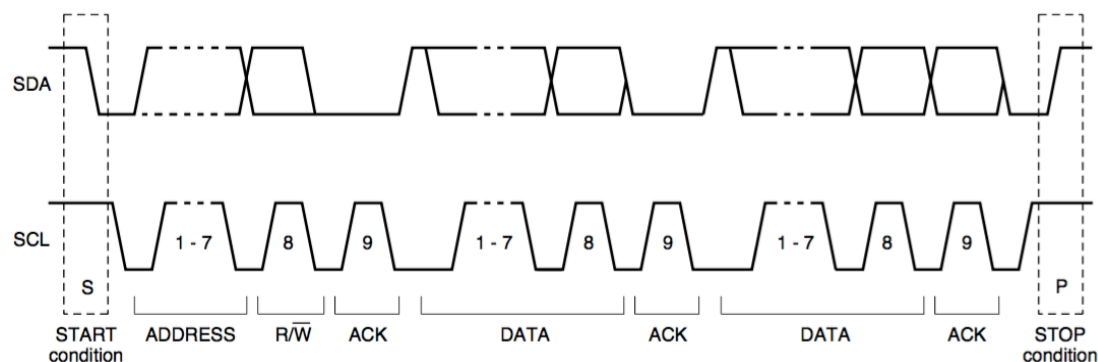


Figure 3 A complete data transfer

Various combinations of read/write formats are then possible within such a transfer.

2.1 Single Read from Random Location

In single read from random location the master does a dummy write operation to desired index, issues a repeated start condition and then addresses the slave again with read operation. After acknowledging its slave address, the slave starts to output data onto SDA line. This is illustrated in Figure 4 . The master terminates the read operation by setting a negative acknowledge and stop condition.

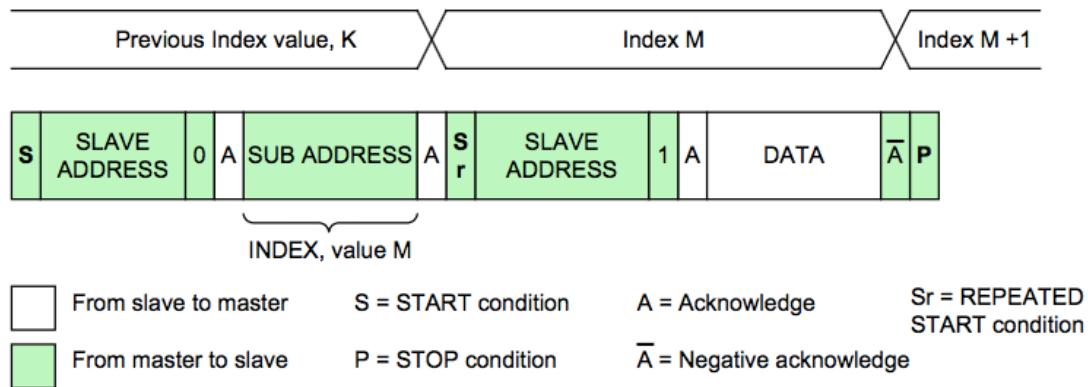


Figure 4. Single Read from a random location

2.2 Sequential Read starting from a Random location

The sequential read starting from a random location is illustrated in Figure 5. The master does a dummy write to the desired index, issues a repeated start condition after an acknowledge from the slave and then addresses the slave again with a read operation. If a master issues an acknowledge after received data it acts as a signal to the slave that the read operation continues from the next index. When the master has read the last data byte it issues a negative acknowledge and stop condition.

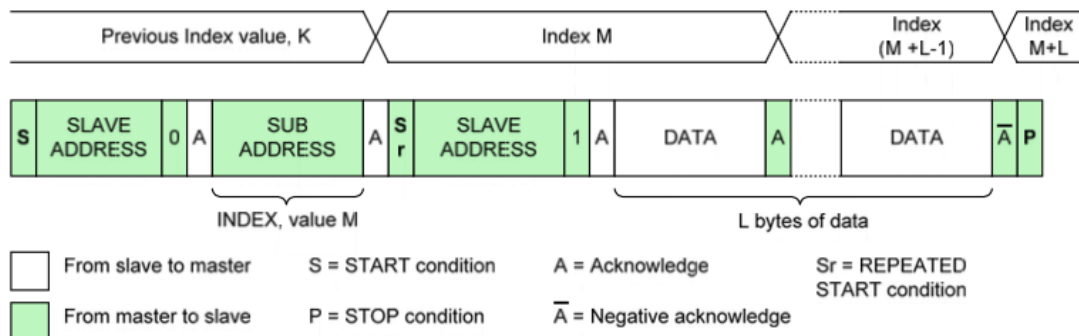


Figure 5. Sequential Read starting from a random location.

2.3 Single Write to a Random location

A write operation to a random location is illustrated in Figure 6. The master issues a write operation to the slave then issues the index and data after the slave has acknowledged the write operation. The write operation is terminated with a stop condition from the master.

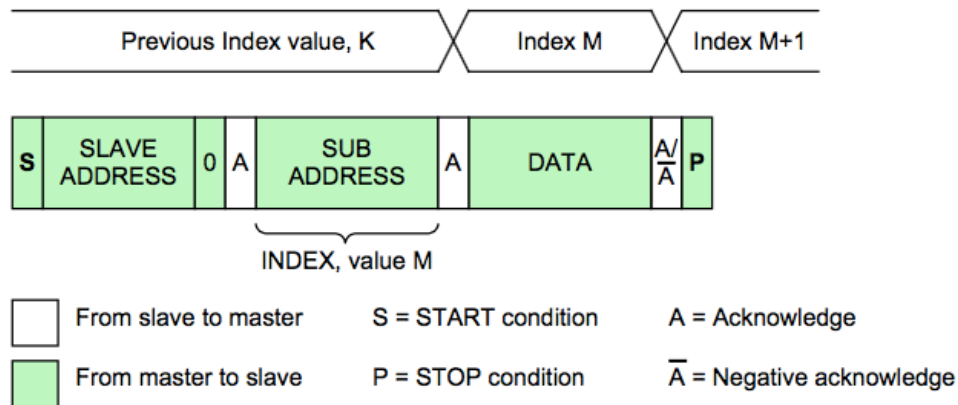


Figure 6. Single Write operation to a random location.

2.4 Sequential Write starting from a Random location

The sequential write operation is illustrated in Figure 7. The slave auto-increments the index after each data byte is received. The sequential write operation is terminated with a stop condition from the master.

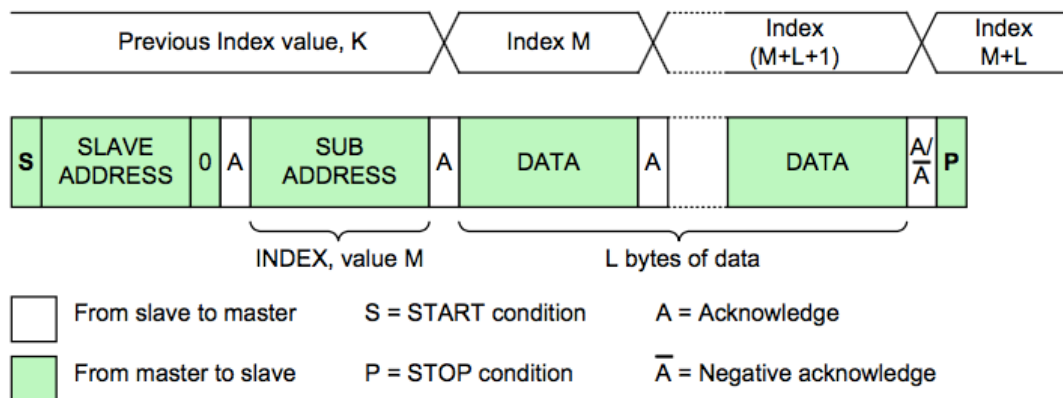


Figure 7. Sequential Write starting from a random location.

3 I²C ADDRESS SPACE

3.1 Device addressing

The addressing procedure for the I2C-bus is such that the first byte after the START condition usually determines which slave will be selected by the master. The exception is the 'general call' address which can address all devices.

The slave devices on the 2S & PS modules are required to support only the 7-bit addressing mode. The 10-bit addressing mode is not supported.

A slave address of a device is made out of a fixed and a programmable part. The fixed part is internally hardwired and designates the device type. The programmable part consists of address input pins which are hardwired on the hybrid or PCB and allows for addressing several devices of the same type on the same bus.

By convention, we allocate 2 fixed address bit at the MSB part of the 7-bit address field [A6-A5] to identify the device type, see Table 1.

Slave Address [A6-A5]	Device Type
00	MPA
01	SSA
10	CBC
11	CIC

Table 1 Device Types

Device types have variable programmable address field in order to meet the system level requirements as depicted in the block diagram of Figure 1.

Table 2 presents the allocation of programmable address field per device type (*Preliminary allocation scheme.*)

Device Type	Programmable Address field	No of devices on same bus
MPA	[A4-A3]='xx', [A2-A0]	8
SSA	[A4-A3]='xx', [A2-A0]	8
CBC	[A4-A0]	32
CIC	Not programmable	1

Table 2 Programmable address field for different device types.

3.2 Reserved addresses

The I2C specification has reserved two sets of eight addresses, 1111XXX and 0000XXX. These addresses are used for special purposes. Table 3 has been taken from the [I2C Specifications](#).

Slave Address	R/W Bit	Description
000 0000	0	General call address
000 0000	1	START byte(1)
000 0001	X	CBUS address(2)
000 0010	X	Reserved for different bus format (3)
000 0011	X	Reserved for future purposes

000 01XX	X	Hs-mode master code
111 10XX	X	10-bit slave addressing
111 11XX	X	Reserved for future purposes

Table 3 I2C reserved addresses

- (1) No device is allowed to acknowledge at the reception of the START byte.
- (2) The CBUS address has been reserved to enable the inter-mixing of CBUS compatible and I2C-bus compatible devices in the same system. I2C-bus compatible devices are not allowed to respond on reception of this address.
- (3) The address reserved for a different bus format is included to enable I2C and other protocols to be mixed. Only I2C-bus compatible devices that can work with such formats and protocols are allowed to respond to this address.

3.3 General call address

The general call address comprises of a slave address equal to 0000 000 followed by R/W = 0 and is reserved to implement special operations of devices. The general call address is for addressing every device connected to the I²C-bus at the same time. However, if a device does not need any of the data supplied within the general call structure, it can ignore this address by not issuing an acknowledgment. If a device does require data from a general call address, it acknowledges this address and behave as a slave-receiver. The master does not actually know how many devices acknowledged if one or more devices respond. The second and following bytes are acknowledged by every slave-receiver capable of handling this data. A slave who cannot process one of these bytes must ignore it by not-acknowledging. Again, if one or more slaves acknowledge, the not-acknowledge will not be seen by the master. The meaning of the general call address is always specified in the second byte.

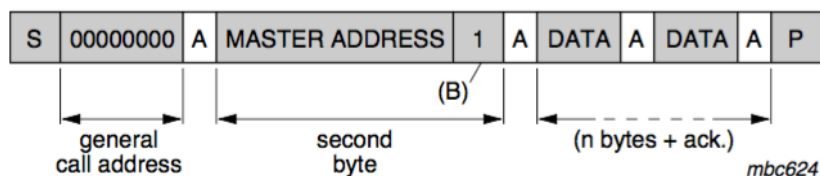


Figure 8 General Call Address data transfer

3.4 Software reset

Following a General Call, (0000 0000), sending 0000 0110 (06h) as the second byte causes a software reset. This feature is optional and not all devices respond to this command. On receiving this 2-byte sequence, all devices designed to respond to the general call address reset and take in the programmable part of their address.

3.5 Device ID

Each device must have hardcoded on silicon a unique 3-byte device identifier. Hardcoded device identifier can be implemented with pre-programmed laser fuses or electrically programmable fuses.

The Device ID is read only and should be mapped in two consecutive addressed device internal registers.

The Device ID read operation as specified in the I²C specification is not supported.

4 ELECTRICAL SPECIFICATIONS AND TIMING FOR I/O STAGES

The presented electrical specifications depict the guidelines for the implementation of I²C busses for the 2S and PS modules. The power supply is 1.2V. Even though the targeted bitrate is 1000 kbit/sec, the document gives timing specifications also for fall-back bitrates of 400 kbit/sec and 100 kbit/sec. It is advised that all devices used in the 2S and PS systems are compatible for all three bit-rates.

4.1 DC Electrical Characteristics

Table 4 presents the DC electrical specifications of the I²C bus implementation for the 2S and PS modules.

Symbol	Parameter	Conditions	
VDD	Supply Voltage		1.2 V
V _{OL}	Maximum Low Output Voltage	I _{out} = 6 mA	0.2 V
V _{OH}	Maximum High Output Voltage	I _{out} = -100 uA	VDD + 0.4V
V _{IL}	Maximum Low Input Voltage		0.6 V
V _{IH}	Minimum High Input Voltage		1.0 V
I _L	Maximum Low Sink Current	V _{OUT} = 0.2V	6 mA

Table 4 DC electrical characteristics

4.2 AC Electrical Characteristics

The minimum HIGH and LOW periods of the SCL clock specified in Table 5 determine the maximum bit transfer rates of 100 kbit/s, 400 kbit/s and 1000 kbit/s while Figure 1 depicts the definition of timing of the I²C bus.

Symbol	Parameter	100 KHz	400 KHz	1000 KHz
t _{LOW} min	Low period of SCL clock	4700 ns	1300 ns	500 ns
t _{HIGH} min	High period of SCL clock	4000 ns	600 ns	260 ns
t _R max	Rise time of SCL & SDA	1000 ns	300 ns	120 ns
t _F max	Fall time of SCL & SDA	300 ns	300 ns	120 ns
t _{SU;DAT}	Data Setup time	250 ns	100 ns	50 ns
t _{HD;DAT}	Data Hold time	5000 ns	900 ns	450 ns

Table 5 AC electrical specifications

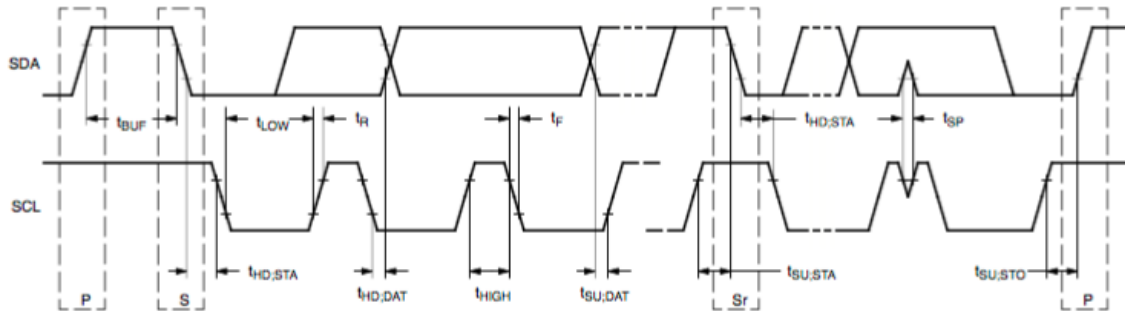


Figure 9 Definition on timing of I²C bus

4.3 Pull Up resistance calculations

In practical systems the pull-up resistance is determined by the rise time limit for I²C systems. For example, as an approximation, for a 400 kbit/sec bus, this limit will be satisfied if the time constant (product of the net resistance and net capacitance) of the total system is set to 1 microsecond. The total time constant may either be set by considering the bus node and choosing pull-up resistors to give time constants of 1 microsecond or by combining the capacitances into an equivalent capacitive loading and calculating the bus pull-up resistor required by this equivalent capacitance.

The pull-up resistor may be calculated as follows:

$$R = \frac{1 \mu Sec}{C_{device} + C_{wiring}}$$

Where: C_{device} = sum of device capacitances connected to each bus.

and C_{wiring} = total wiring and stray capacitance on each bus.