Radu Pitigoi-Aron
Principal Engineer, Systems Architect
QUALCOMM Technologies, Inc

MIPI I3C℠ Interface – Advanced Features
Outline

• MIPI I3C℠ – intelligent multifeatured interface
• List of main bus management procedures
• Timing Control
  – Problems solved, Challenges, Practical implementation aspects
• Elements of flow control
  – Problems solved, Challenges, Practical implementation aspects
MIPI I3C℠ Bus Clients
MIPI I3C℠ – Intelligent Multifeatured Interface

- MIPI I3C℠ supports several communication formats, all sharing a two-wire interface.
  - The two wires are designated SDA and SCL:
    - SDA (Serial Data) is a bidirectional data pin
    - SCL (Serial Clock) can be either a clock pin or a data pin while in certain HDR Modes

- An MIPI I3C℠ Bus supports the mixing of various Message types:
  - I²C-like SDR Messages, with SCL clock speeds up to 12.5MHz
  - Broadcast and Direct Common Command Code (CCC) Messages that allow the Master to communicate to all or one of the Slaves on the I3C Bus, respectively
  - HDR Mode Messages, which achieve higher data rates per equivalent clock cycle
  - I²C Messages to Legacy I²C Slaves
  - Slave-initiated requests to the Master, for example for In-Band Interrupt or to request the Master role
MIPI I3C℠ BUS Management Features

• Dynamic Address Assignment
• Hot-Join
• In-Band Interrupt
• Secondary Master
• In-Band Hard RESET
• Timing Control
• Common Command Codes
• Error detection and Recovery
• Elements of Flow Control
Timing Control

• Complex applications require several Sensors on a common timeline

• Synchronous Systems and Events
  – Controlling the sampling moments has the potential of drastically reducing the system energy expenditure

• Asynchronous Systems and Events
  – The accuracy of the timestamps of events matters

• The Synchronous and Asynchronous modes can be used independently and concurrently on the same bus and devices
Synchronous Systems and Events

- **S. RED**
- **S. GREEN**
- **S. BLUE**

SYNC Tick [ST] and Delay Time [DT] in-band, via I3C bus

ALL READ IN SYNC

SENSORS NOT IN SYNC

ALL DATA IN SYNC

Sequence Repetition Period
Adjustable [0.2 ; 5 sec], 1 sec nominal
Synchronous – Multiple Transactions

- **I3C START**: Polling
- **Sync Tick [ST] & Delay Time [DT]**: ST if validated by DT
- **T_Ph start, calculated from ST and DT**: Refresh/Adjust Sensor’s Timer
- **DT between ST and T_Ph Start**: Sequence Repetition Period
  - Adjustable [0.2 ; 5 sec]
- **ST&DT to next ST&DT delay**: Adjustable [0.2 ; 5 sec]
  - Each ST&DT instantiation includes Timer Error Correction data
- **I3C messages, on the bus**: Some I3C transactions START condition is used by the Slaves (sensors) for adjusting their (sensors’) internal timers

1. Sensors Sample unsynchronized
2. Sync Tick [ST] & Delay Time [DT]
3. ST if validated by DT
4. T_Ph start, calculated from ST and DT
5. Refresh/Adjust Sensor’s Timer
6. DT between ST and T_Ph Start
7. Sequence Repetition Period
   - Adjustable [0.2 ; 5 sec]
8. ST&DT to next ST&DT delay
   - Adjustable [0.2 ; 5 sec]
9. Each ST&DT instantiation includes Timer Error Correction data
10. I3C messages, on the bus
    - Some I3C transactions START condition is used by the Slaves (sensors) for adjusting their (sensors’) internal timers

© 2017 MIPI Alliance, Inc.
Synchronous – Common Command Codes

• SETXTIME CCC

• Configuration messages
  – ODR (Output Data Rate)
  – TPH (Procedure Repetition Time)
  – TU (Time Unit)

• Run Time messages
  – SYNC Tick [ST]
  – Delay Time [DT]
Asynchronous Systems and Events

• Four Async Modes
  – Basic – Async Mode 0
  – Enhanced – Async Mode 1, 2 and 3

• SETXTIME is the CCC
  – The defining byte selects the running mode
Async 0 Time Diagram

**MTS** – Master Timestamp, expressed in MASTER’s time units

**MREF** – Master Reference, i.e. Start of Master’s secondary counter, MCNT2

**MC1, MC2** – Master’s counters values, captured at the corresponding HWSE events.

**SC1, SC2** – Slave’s counters values, captured at the corresponding HWSE events.

\[
\text{MTS} = \text{MREF} - \text{MC2} \times \text{SC1/SC2}
\]
Async 0 on SDR

SENSE SAMPLE

Sensor's SCNT1 Start
Sensor Initiates IRQ
Sensor's Clock for Timer/Counter

1st HW EVENT

Master's Timestamp MTS

Virtual MC1

2nd HW EVENT

Sensor's SCNT2 Start
SC1 Captured
Sensor's Clock for Timer/Counter

Master's MCNT2 Start
MREF Captured
MC2 Captured

First SCL Rising Edge After ACK or T Bit

SC1 Byte1

I3C SDR SCL

ISR Read

I3C BUS

LEGEND
- Bus Management
- Master to Slave
- Slave to Master
- T Bit - Transition

© 2017 MIPI Alliance, Inc.
Elements of Flow Control

• The Transmitter drives actively the data lines
  – SDA on HDR-DDR
  – SDA and SCL on HDR-TSx

• The Receiver might need to end the transaction
  – The bus needs to provide the opportunity for the Receiver to change the state of a line, in a pre-established way
# HDR-DDR Transactions

## HDR-DDR Preamble Values

<table>
<thead>
<tr>
<th>Context</th>
<th>Preamble Value and Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2'b00</td>
</tr>
<tr>
<td><strong>After EnterHDR</strong></td>
<td></td>
</tr>
<tr>
<td>Command Word follows</td>
<td></td>
</tr>
<tr>
<td><strong>After Read CMD</strong></td>
<td></td>
</tr>
<tr>
<td>Slave ACK, Data follows</td>
<td></td>
</tr>
<tr>
<td>Slave NACK, Aborted</td>
<td></td>
</tr>
<tr>
<td><strong>After Read DATA</strong></td>
<td></td>
</tr>
<tr>
<td>Reserved for Future Use</td>
<td></td>
</tr>
<tr>
<td>CRC Word follows</td>
<td></td>
</tr>
<tr>
<td>Master Aborts, Slave yields.</td>
<td></td>
</tr>
<tr>
<td>Master drives second 0.</td>
<td></td>
</tr>
<tr>
<td>Data follows. Master does not drive second bit.</td>
<td></td>
</tr>
<tr>
<td><strong>After Write CMD</strong></td>
<td></td>
</tr>
<tr>
<td>Data follows</td>
<td></td>
</tr>
<tr>
<td><strong>After Write DATA</strong></td>
<td></td>
</tr>
<tr>
<td>CRC Word follows</td>
<td></td>
</tr>
<tr>
<td>Data follows</td>
<td></td>
</tr>
</tbody>
</table>

© 2017 MIPI Alliance, Inc.
HDR-DDR – Slave controls DDR READ command
HDR-DDR – Master Controls DDR READ Transaction [1]

Beginning of new DATA Word

Beginning of HDR Restart or HDR EXIT Pattern

Early ending with no CRC
HDR-DDR – Master Controls DDR READ Transaction [2]

Early ending with CRC
HDR-DDR – Slave Requests DDR WRITE Termination [1]

Beginning of new DATA Word

Early ending with no CRC
HDR-DDR – Slave Requests DDR WRITE Termination [2]
HDR-TSx – Master Controls S2M Data Transfer

Beginning of new DATA Word

$T_1\ T_0$
$S_D4$
$S_{SD4}$
$M_{SDA}$
$S_{SCL}$
$M_{SCL}$
$T_1\ T_0$
$C_1\ C_2$
$S_{CI}$
$S_{CJ}$

Beginning of HDR Restart or HDR EXIT Pattern

$S_D4$
$S_{SD4}$
$M_{SDA}$
$S_{SCL}$
$M_{SCL}$
$T_1\ T_0$
$C_1\ C_2$
HDR-TSx – Slave Controls the M2S Data Transfer
ENDXFER CCC – Early Termination Setup and Invocation

- **Defining Bytes**
  - 0x7F – SET/GET Repetition Interval for HDR-TSx
  - 0x55 – Initiates the HDR-TSx with Ending Data Transfer Procedure Enabled
  - 0xF7 – SET/GET CRC Index for HDR-DDR
  - 0xAA – Initiates the HDR-DDR with Ending Data Transfer Procedure Enabled
mipi®
DEVCON
THANK YOU

BANGALORE, INDIA
MIPI.ORG/DEVCON

2017
MIPI ALLIANCE
DEVELOPERS
CONFERENCE