

MIPI RFFE v2.0 Webinar: An Overview of New Features and Implementation Benefits

A decorative graphic on the left side of the purple banner consisting of four spheres: a large teal one at the bottom, a medium purple one above it, a small red one to the right, and a very small orange one at the top right.

Jim Ross, Skyworks Solutions - Chair RFFE WG

John Oakley, Intel – Vice-chair RFFE WG

February 17, 2015

2/17/2015

MIPI Alliance: A Brief Introduction



Peter B. Lefkin
Managing Director

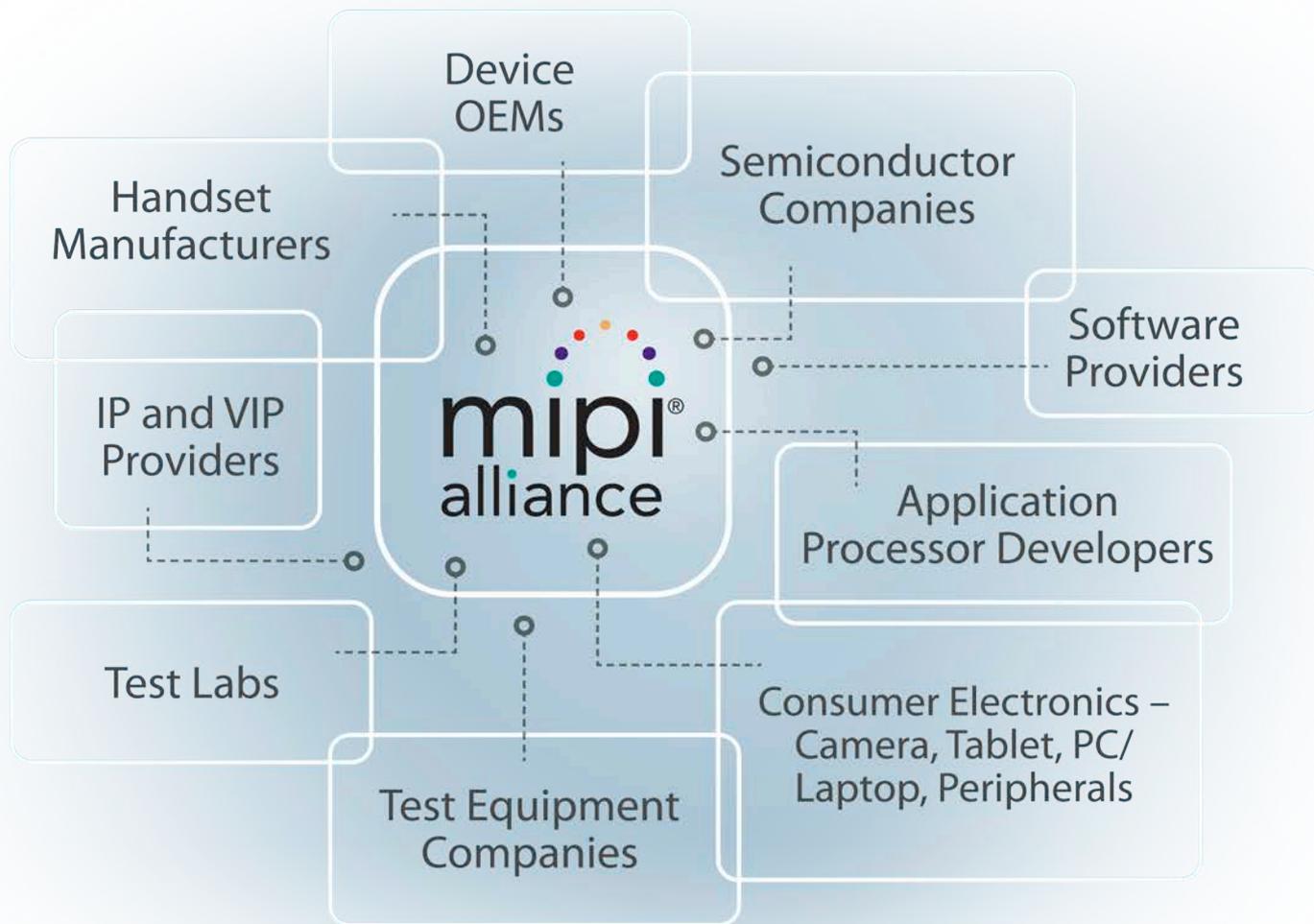


About MIPI Alliance

- 264 Members (as of Jan. 23, 2015)
- 45+ specifications and supporting docs
- We drive **mobile** and **mobile-influenced interface** technology through the development of hardware and software **specifications**
- We work **globally** and **collaboratively** with other standards bodies to **benefit the mobile ecosystem**



MIPI Alliance Member Ecosystem



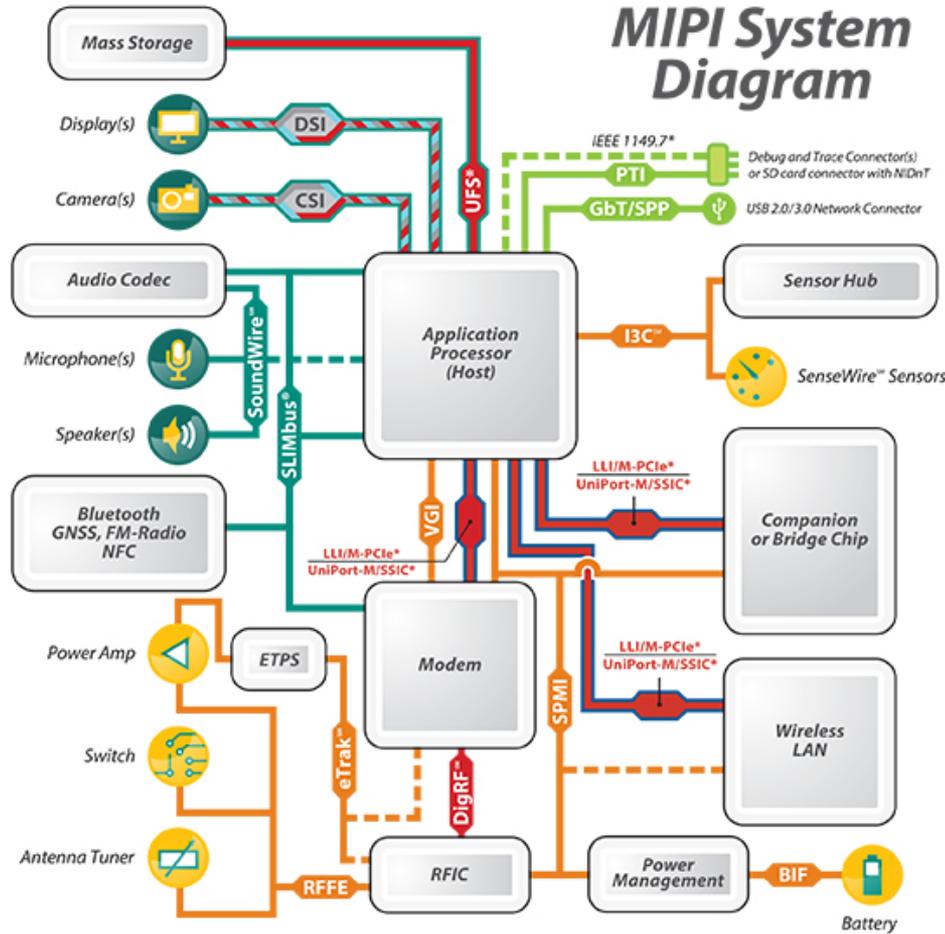


Active MIPI Alliance Working Groups

- Analog Control Interface
- Battery Interface
- Camera
- Debug
- Display High Speed Synchronous Interface
- Low Latency Interface
- Low Speed Multipoint Link (New - SoundWireSM)
- Marketing
- PHY (C / D / M)
- Reduced Input Output (RIO) (New)
- **RF Front-End (RFFESM)**
- Sensor / I3CSM (New)
- Software (New)
- Technical Steering Group
- Test
- UniProSM



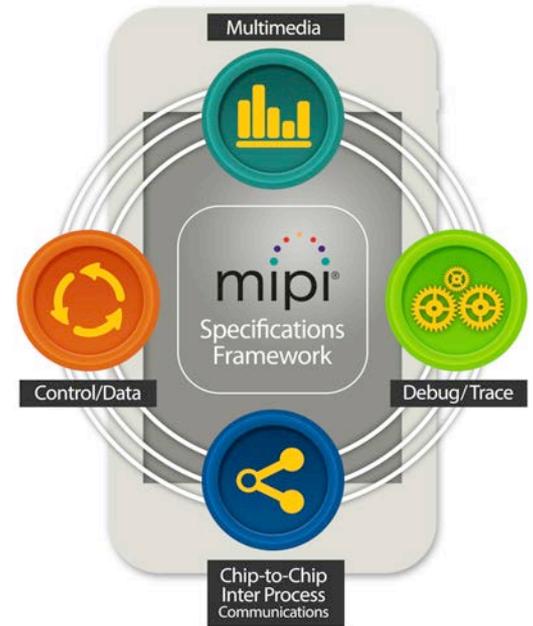
MIPI System Diagram



mipt alliance

* The UFS (Universal Flash Storage) specification from JEDEC, the SSIC (SuperSpeed Inter Chip) specification from USB-IF, and the M-PCIe (Mobile PCI Express) specification from PCI-SIG* all use the MIPI M-PHY* physical layer.

* UFS (Universal Flash Storage) specification is available from JEDEC and uses the MIPI UniPort-M. SLIMbus* and M-PHY* are registered marks and DigRF™, eTrak™ and SoundWire™ are service marks of MIPI Alliance, Inc.





Recent Announcements

- 05 Nov 2014 - [MIPI Alliance Introduces Sensor Interface Specification for Mobile, Mobile-Influenced and Embedded-Systems Applications](#)
- 09 Oct 2014 - [MIPI Alliance Introduces MIPI SoundWireSM, a Comprehensive Audio Interface for Mobile and Mobile-Influenced Devices](#)



The Future of MIPI – Beyond Mobile

- Mobile influences **everything**
- Everything gets faster, smaller and lower power
 - MIPI will continue to evolve specs to take advantage of the evolution of technology in mobile devices



MIPI RFFE v2.0 Webinar: An Overview of New Features and Implementation Benefits

A decorative graphic on the left side of the purple banner consisting of four glossy spheres: a large teal one at the bottom, a medium purple one above it, a small red one to the right, and a very small orange one at the top right.

Jim Ross, Skyworks Solutions - Chair RFFE WG

John Oakley, Intel – Vice-chair RFFE WG

February 17, 2015

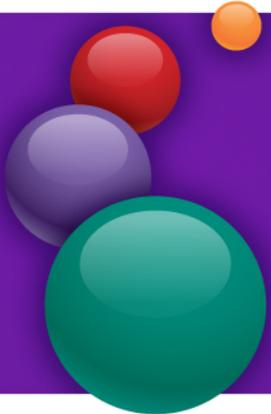
2/17/2015



MIPI RFFE Overview Agenda

- MIPI RFFE v1.x (v1.00.00 & v1.10)
 - What is RFFE?
 - Key Features
- Why RFFE v2.0?
 - Key New Features
 - Multiple devices can control the bus (**Multi-Master**)
 - Additional bus operating frequencies (**Extended Frequencies**)
 - Additional read-back methods (**Synchronous Read**)
 - Support for interrupts (**Interrupt-Capable Slave functionality**)
 - Additional Registers for unified control (New **Reserved Registers** and functions)
- MIPI RFFE Roadmap v2.1?
 - Items up for consideration?

MIPI RFFE v1.x

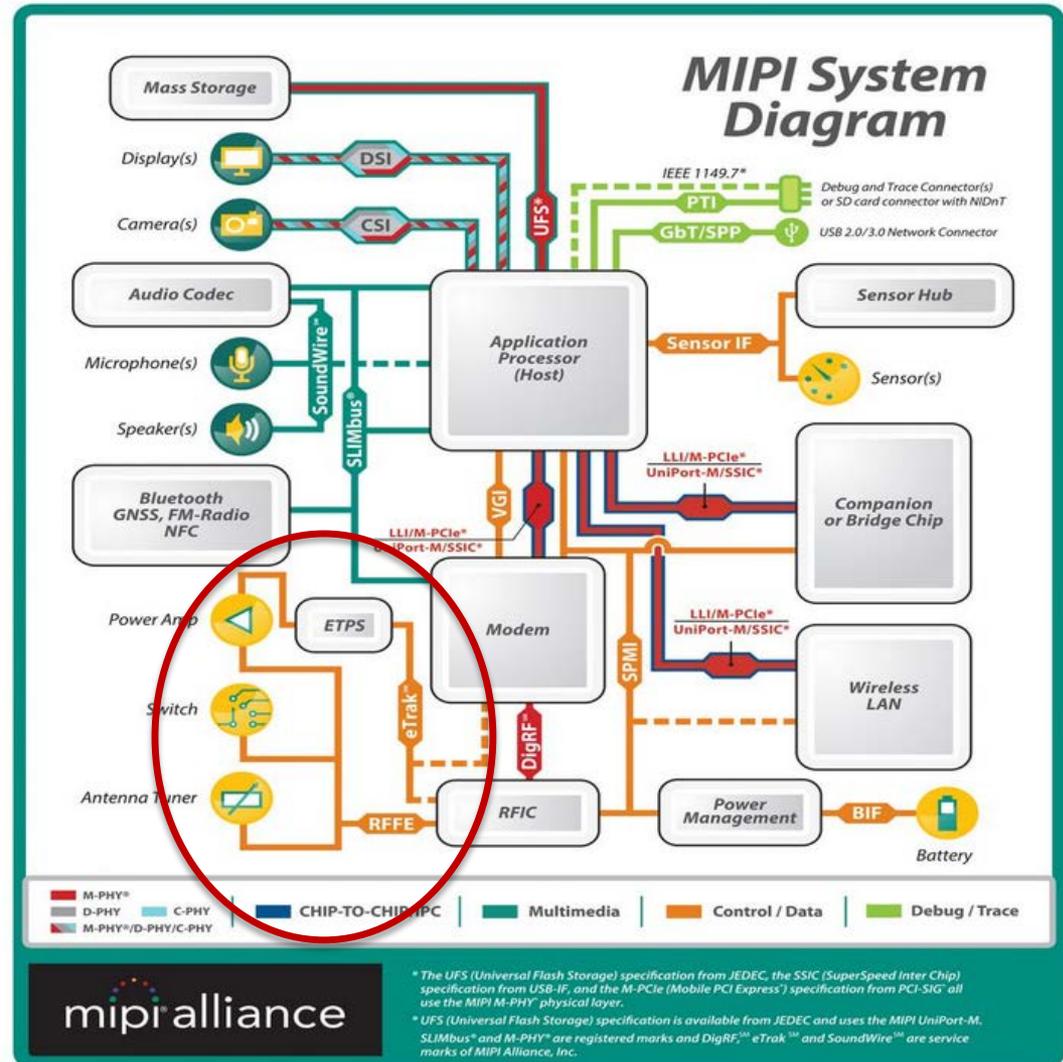




What is RFFE?

RFFE Introduction

- RFFE WG is the RF Front-End Control Working Group within the MIPI Alliance
 - MIPI System Diagram
- RFFE WG has specified a two-wire control bus to be used (but not limited to) in controlling various RF Front-End devices (e.g. PAs, Filters, Switches, Antennas etc.)
- Work started Sep 2008 and was developed on an accelerated schedule.



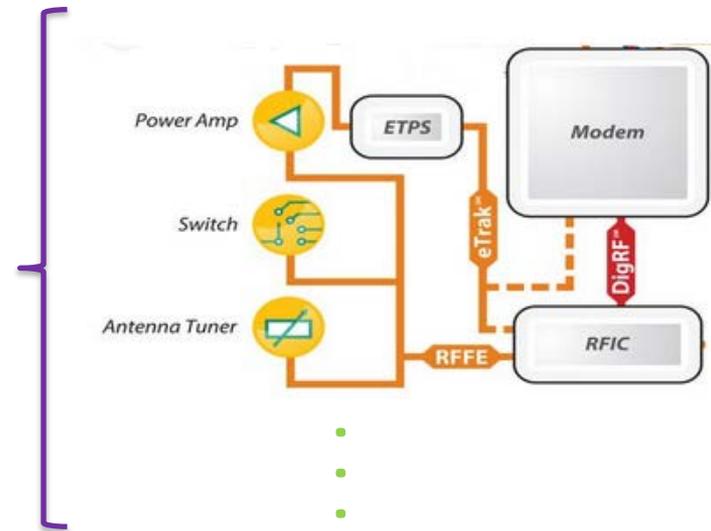


RFFE in the RF Front-End

The RF is essential in conveying the communication over radio waves

- The RF performance and functionality increases the devices versatility by
 - better coverage
 - higher throughput
 - better call connectivity
 - providing international roaming
 - dual or multi SIM configurations
 - providing improved battery life
- Complex RF solutions incorporate a multitude of customized components in the RF Front-End
- Standardized solutions required for control
- RFFE is broadly adopted by the industry being the excellent solution for controlling the RF Front-End

Zooming in...

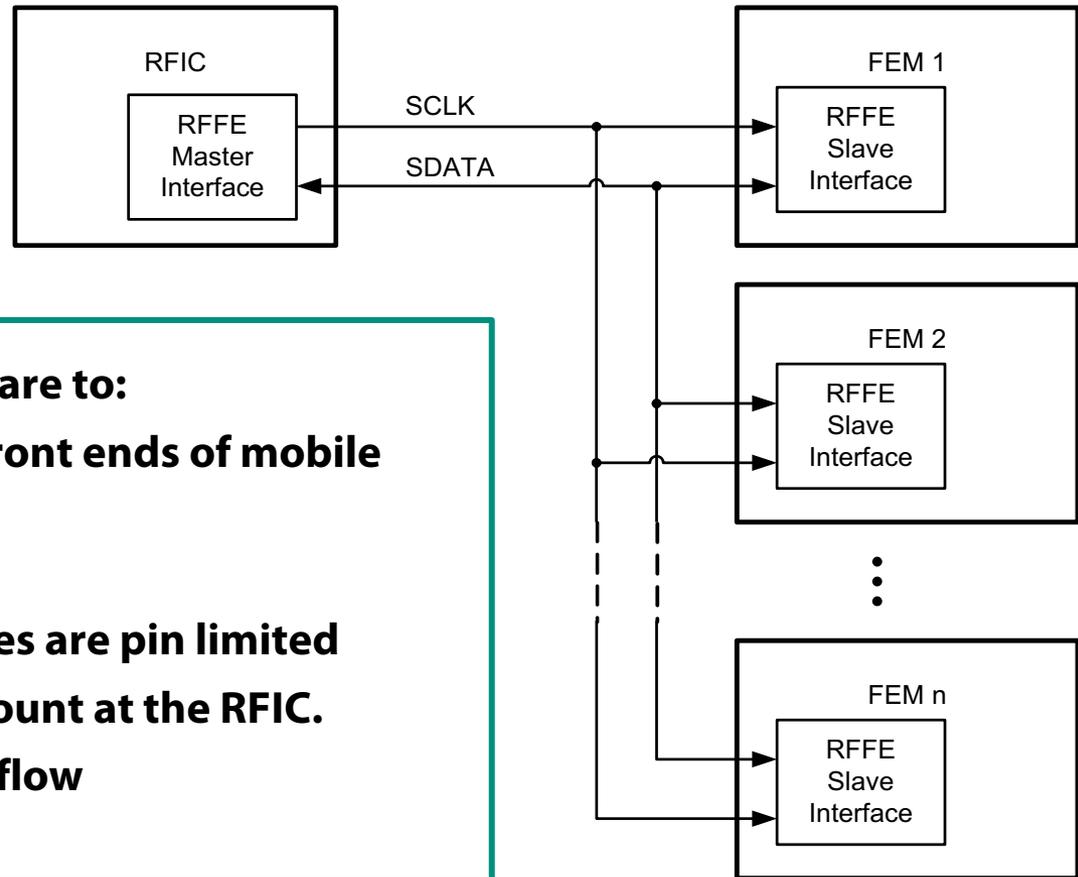


RFFE Control Bus Overview

RFFE Technical Overview

Two signals (+ VIO):

- Master initiated SCLK
- Bi-directional SDATA



- **RFFE key pillars for the design are to:**
 - **Minimize wiring effort in front ends of mobile terminals**
 - **Minimize pin count**
 - Many Frontend devices are pin limited
 - great savings in pin count at the RFIC.
 - **Ease and optimize control flow**



RFFE v1.x Overview

- **Electrical & Digital Details**

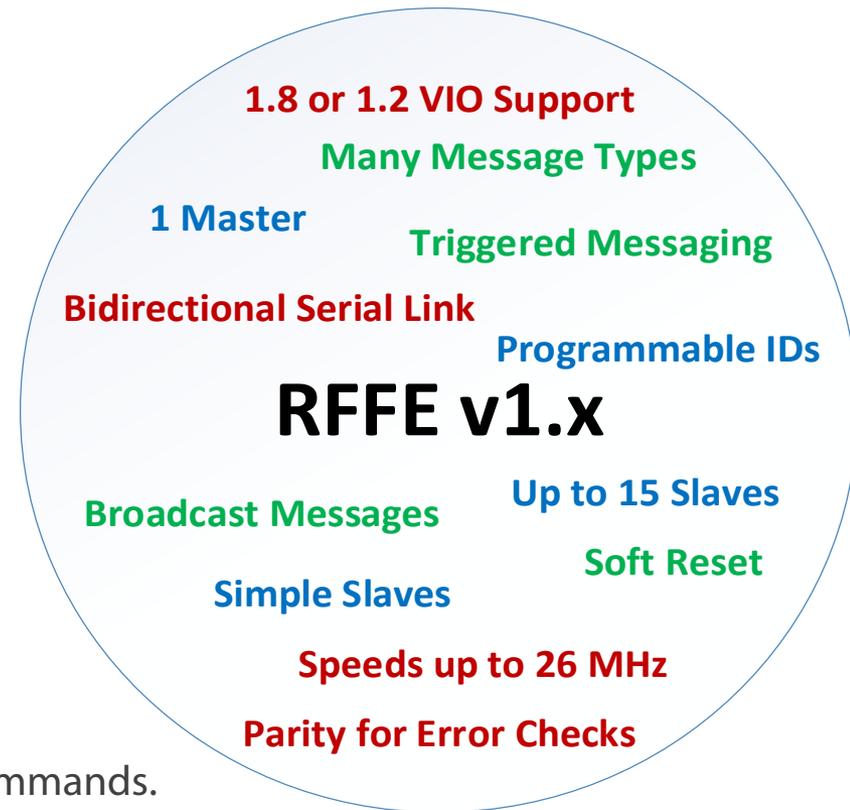
- Up to a 26 MHz bus speed.
- Supports up to an address space of 16 bits.
- Contains parity bits for error checking
- Common voltage reference defined for the interface.

- **Flexible Bus Configuration**

- One master system, which eliminates arbitration for the bus.
- Slave devices are very configurable.
- Slaves support an optional programmable Unique Slave ID.
- Supports user defined group IDs for write commands.

- **Multiple Message types**

- Single byte and multi-byte read and write commands are supported
- Supports broadcast messages over the bus to multiple slave devices.
- An optional trigger feature to solve potential timing issues.
- Supports a command initiated soft reset.

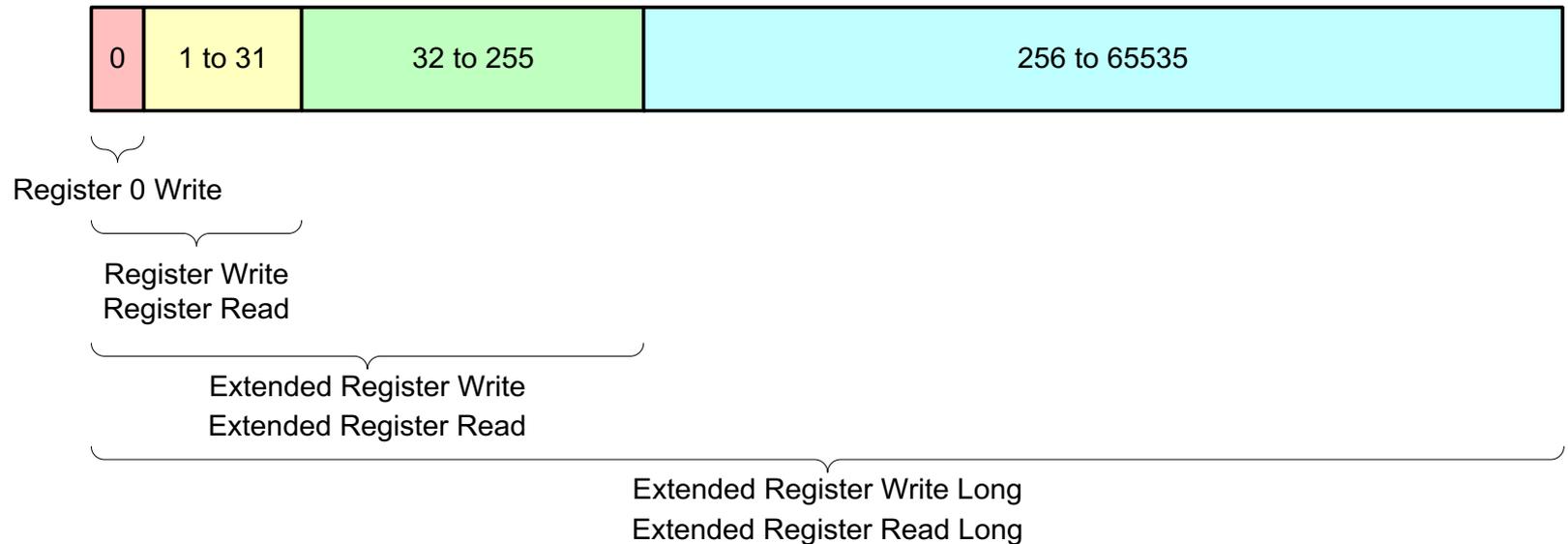




RFFE Register Mapping

Register Space:

- 0x00 – 0x1F (**Basic**)
- 0x20 – 0xFF (**Extended**)
- 0x100 – 0xFFFF (**Extended Long**)



Command Sequence Types



RFFE Control Bus Overview

Basic Register Write and Read Commands

Description	SSC		Command Frame								Data Frame					
Register Write	1	0	SA[3:0]	0	1	0	0	0	0	0	P	Data[7:0]		P	BP	
			Address[4:0]					1	1	1		1	1			
Register Read	1	0	SA[3:0]	0	1	1	0	0	0	0	P	BP	Data[7:0]		P	BP
			Address[4:0]					1	1	1		1	1			

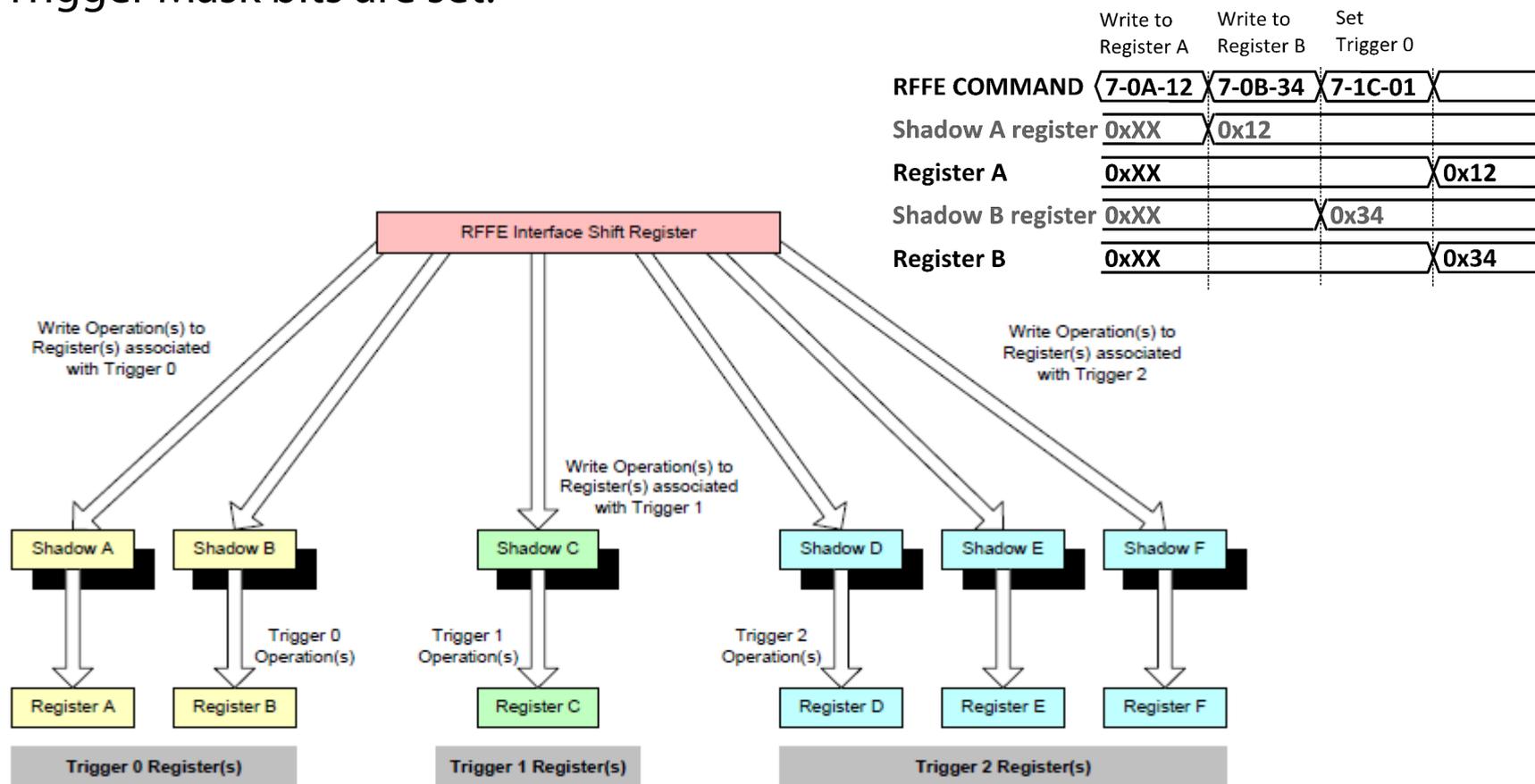
The components of a RFFE message.

- **SSC : Start Sequence Condition**
- **Command Frame**
 - SA[3:0] = Slave Address addressing 15 slaves, SA=0 will broadcast to ALL slaves
- **Parity calculated & inserted for each Frame in a Command Sequence**
- **Data or Address Frame**
- **Bus Park Cycle**



Trigger Registers

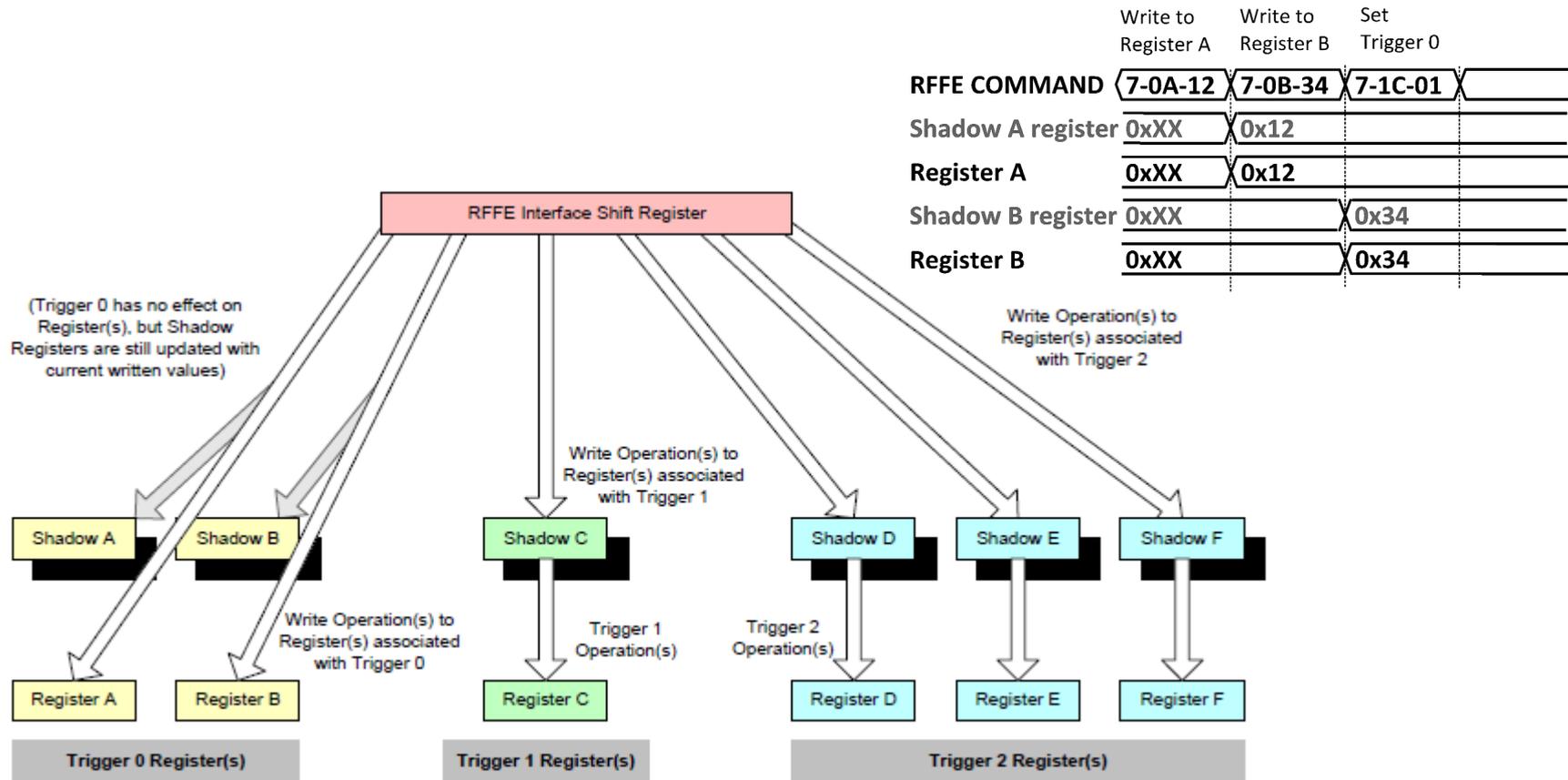
This is an illustration of how Triggers can work as defined by the MIPI RFFE Specification. In this illustration all the Triggers are enabled – in other words no Trigger Mask bits are set.



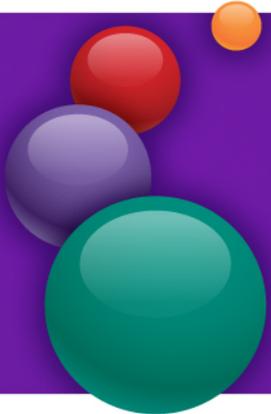


Trigger Registers

In this illustration only Triggers 1 & 2 are enabled. Since the Trigger Mask for Trigger 0 is set, Trigger 0 is disabled and thus data is written directly to the configuration registers, effectively bypassing the shadow registers.



MIPI RFFE v2.0





Why was RFFE v2.0 developed?

- Background for the RFFE v2.0 development
 - RFFE v1.0 development was on a very aggressive schedule
 - Thus only core features were targeted for this release
 - Other features were intentionally left for later development and inclusion
 - After RFFE v1.0 and v1.1 were released, the WG turned its attention towards addressing potential improvements and extensions to RFFE
 - Forward-looking features
 - Features list was developed by the WG: Inputs from WG members, and from multiple surveys (MIPI and IWPC)
 - WG down-selected features to be included in v2.0 prior to detailed development
 - Backward-compatibility with RFFE v1.x was of prime importance in selecting and developing the new features
 - An RFFE v1.x Slave will still work in RFFE v2.0 systems, although primarily to only v1.x features.
 - RFFE v2.0 expands and improves upon the capabilities provided by RFFE v1.x, and provides a solid foundation for the future of RF front-end architectures.



What's New in MIPI RFFE v2.0?

Key New and Improved Features

- **Electrical & Digital Details**

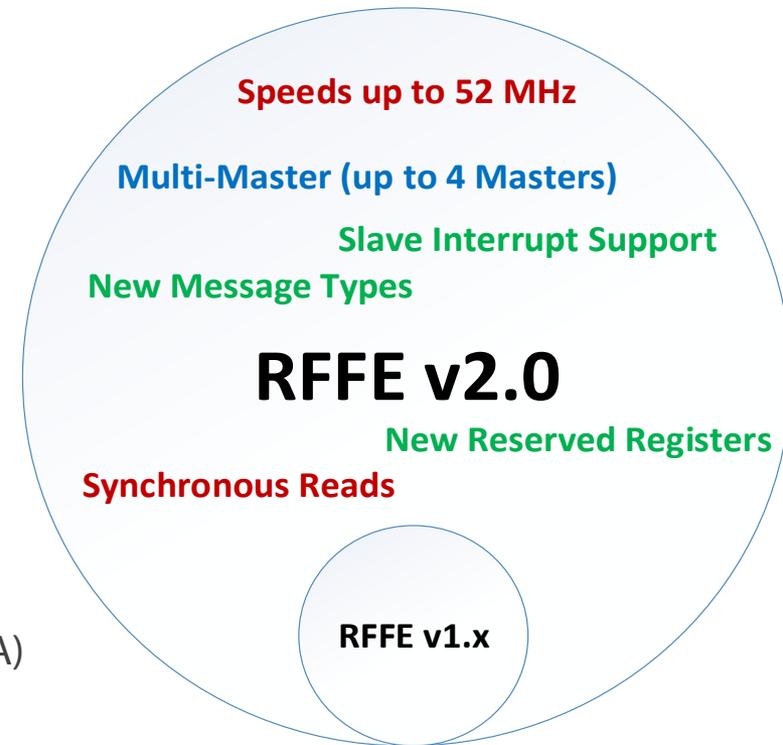
- Extended Frequencies – increased command sequence bandwidth capabilities
- Synchronous Read introduction – Allows for a wider range of bus loading by allowing more time for data propagation on the bus by Slaves, and also enables Extended Frequencies

- **Flexible Bus Configuration**

- Multi-Master - supporting Carrier Aggregation (CA) system architectures

- **Multiple Message types**

- Interrupt-Capable Slave functionality – quicker response opportunities for Slave Devices to report to Master(s)
- New Reserved Registers and functions – Common function register locations easing hardware and software development

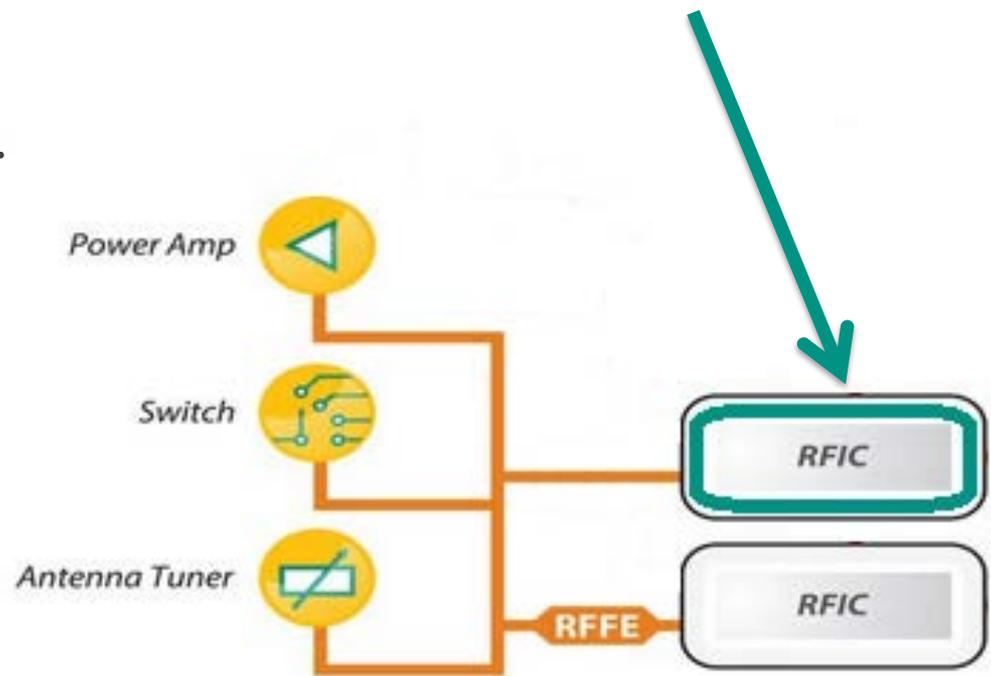


MIPI RFFE v2.0: Multi-Master

Requirements driving the RFFE v2.0 Multi-Master Feature:

- Monitoring of Alternate Bands
 - Carrier Aggregation
 - Transceiver Platform Architecture
- Only one Bus Owner Master (BOM) at any one time. Other Masters may monitor the bus.
 - Arbitration (and associated timing uncertainties) avoided with scheme chosen.
 - RFFE v2.0 Architectures May Remain Single Master

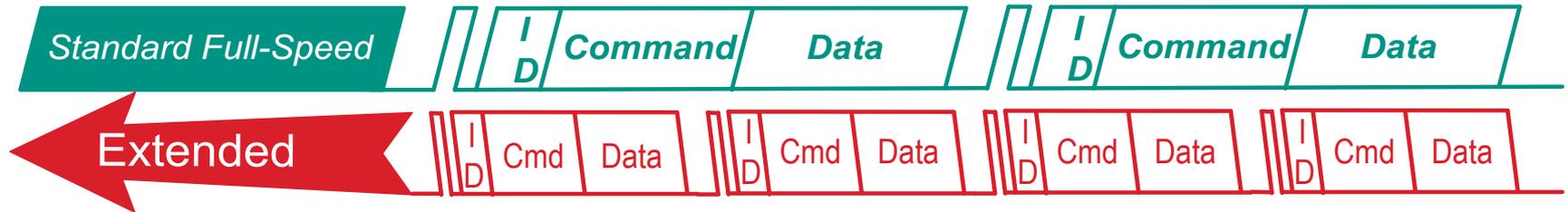
OPTIONAL Multi-Master RFFE Bus Configuration



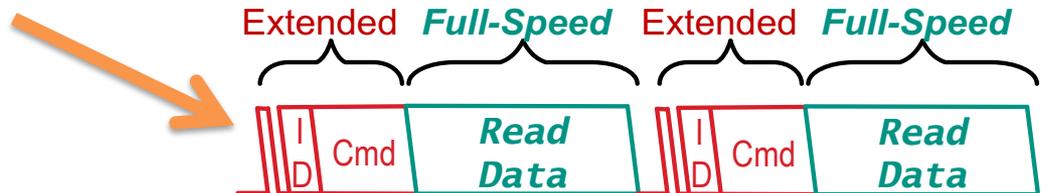


MIPI RFFE v2.0: Extended Frequencies

Extended Frequency Range **DOUBLES** the number of Command Sequences that can be transferred on the bus in a given amount of time.

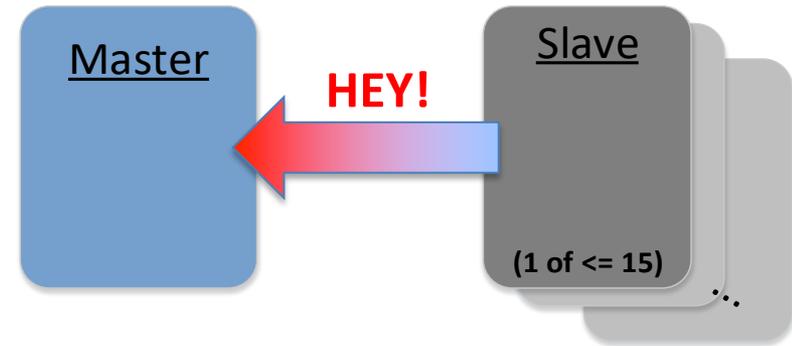


- Double the standard rate exists in the “forward” direction, i.e. Master-to-Slaves for Write types of Command Sequences where the Master (BOM) is driving SDATA.
 - The “forward” direction is typically the most timing-critical control path in an RF Front-End, and also accounts for a majority of bus traffic
- Due to timing limitations in the read-back path, Extended Frequency is not possible when the BOM is not driving SDATA
- However, the RFFE 2.0 sRead feature allows virtually all Slaves to utilize Full-Speed in the existing Standard Frequency Range, thus achieving higher performance in the “reverse” path as well





MIPI RFFE v2.0: Interrupt-Capable Slaves (ICS)



- In RFFE v1.x the only way for a Slave to notify the Master of some condition or communicate back to the Master is through a Read Command Sequence (which may be *initiated **only** by the Master*)
- ICS (Interrupt-Capable Slave) features were developed with the following guidelines:
 - Feature needed to be In-Band : No additional signals or wires were desired
 - (BOM) Master needs to retain control of the bus at all times to ensure that bus timing may remain deterministic
 - Provide the possibility of as close to a “real-time” response as possible
- ICS is an optimized multiple-device polling feature
 - ICS feature is not a “traditional” interrupt - but rather a quick polling method
 - One new RFFE Cmd Seq created for ICS; majority of ICS ops use existing CSs
 - ICS supports up to 16 different interrupts on an RFFE bus (with up to 4 / Slave)



MIPI RFFE v2.0: Interrupt-Capable Slaves (ICS)

Configuration Phase

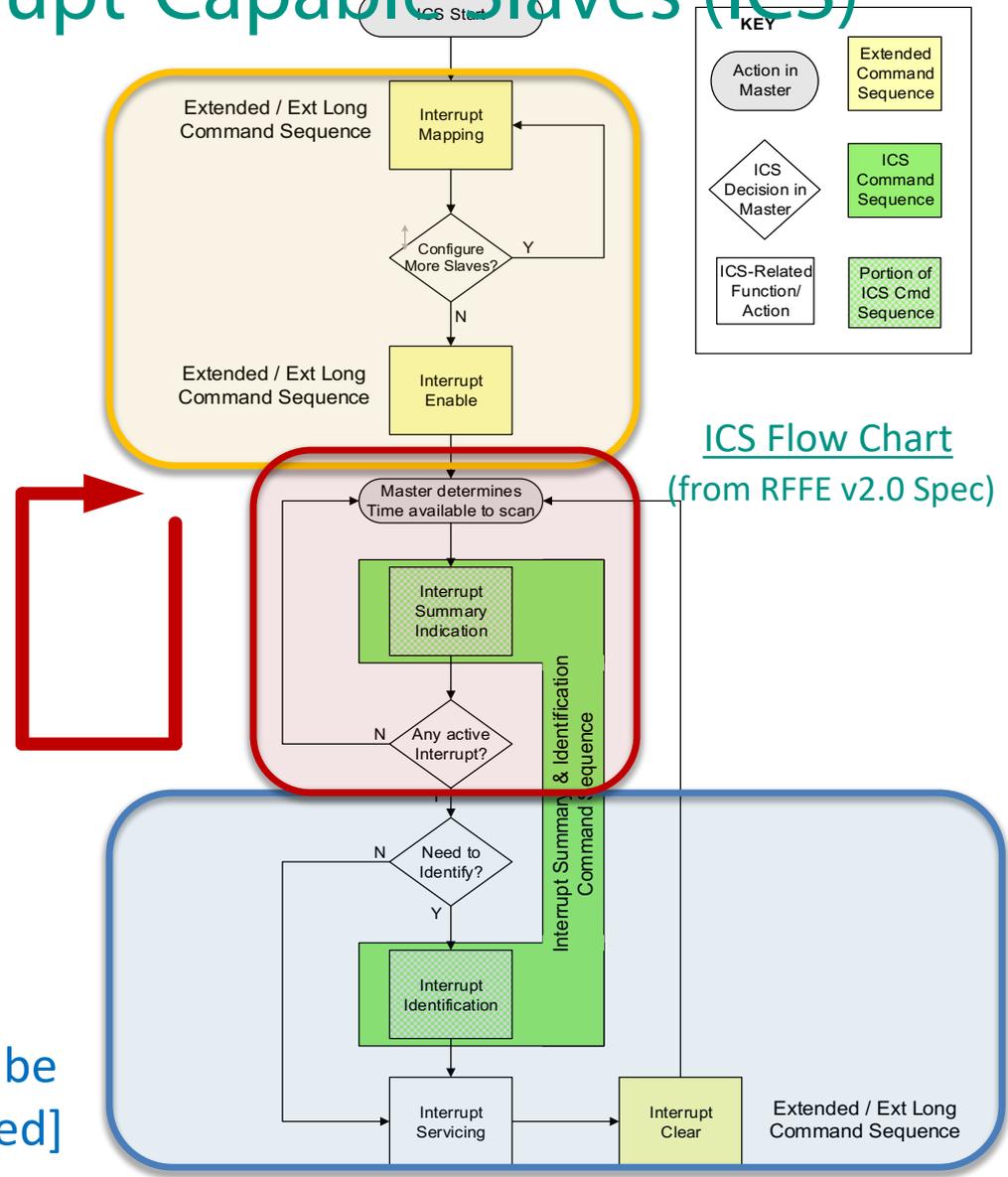
[Only needed at ICS set up or configuration changes]

Quick Interrupt Scan

(Of All ICS Enabled Slaves)
[Optimized for Time-Minimized Polling of Any/All Interrupt Requests]

Identification / Servicing / Clearing Phase

[Used only when Interrupts must be Identified, Serviced, and/or Cleared]



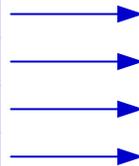


MIPI RFFE v2.0: New Reserved Registers

RFFE v1.x

Register (bits 7:0)

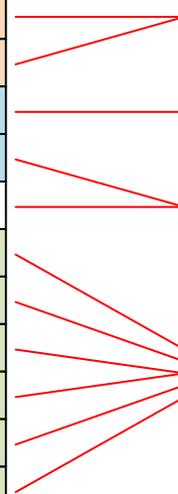
PM_TRIG
PRODUCT_ID
MANUFACTURER_ID
MAN_&_USID



RFFE v2.0

Register (bits 7:0)

PM_TRIG
PRODUCT_ID
MANUFACTURER_ID
MAN_&_USID
EXT_PRODUCT_ID
REV_ID
GSID0-GSID1
UDR_RST
ERR_SUM
INT_MAP0-1
INT_MAP2-3
INT_EN0
INT_EN1
INT_CLR0
INT_CLR1
BUS_LD



NEW REGISTERS

User-Defined Extended Product & Revision ID

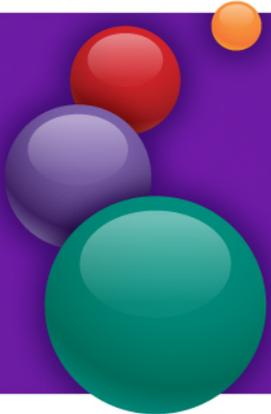
GSID register defined

Error Logging & Software Reset (retain U/GSID, Triggers, etc)

Interrupt Maps, Enables, and Clears

Slave SDATA Bus Load

MIPI RFFE Roadmap v2.1?

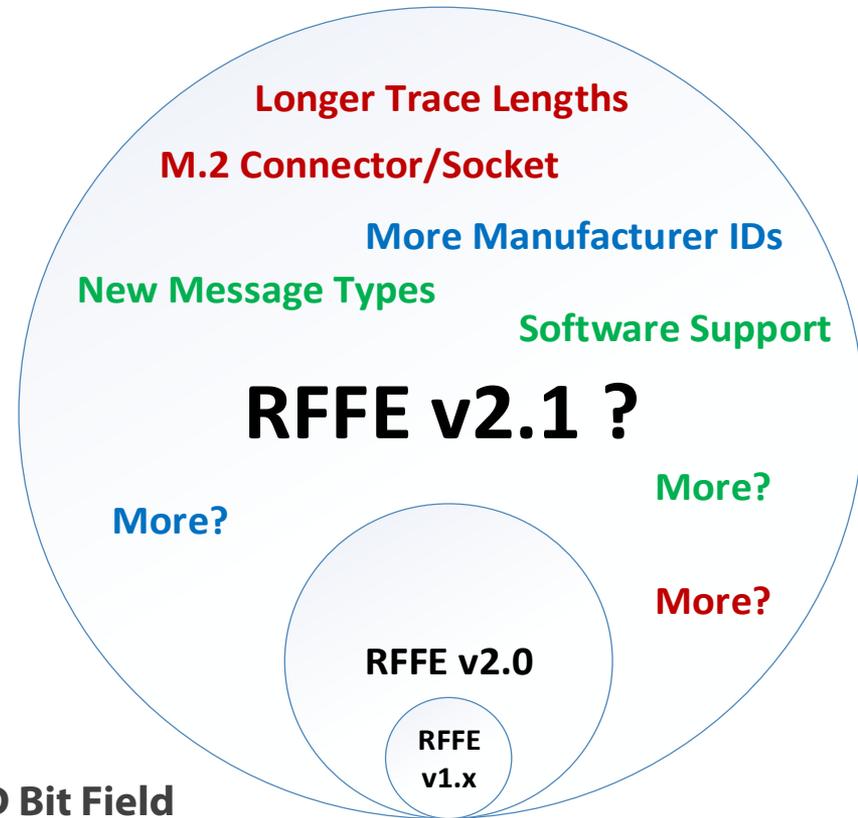




MIPI RFFE v2.1: Future Enhancements

What comes after RFFE v2.0?

- The Working Group has begun to gather ideas for the next release and some of these ideas are outlined below
- **Electrical & Digital Details**
 - Longer Trace Lengths
 - RFFE over M.2 Connector/Socket
- **Flexible Bus Configuration**
 - Potential Extension of the Manufacturer ID Bit Field
- **Multiple Message types**
 - Optional extensions to the Master Write (& Read?) CS(s)
 - Software Considerations
- **The WG welcomes additional members and contributions!**



MIPI RFFE Documents and Website





MIPI RFFE: Documents

RFFE Specification and Supporting Documents (available to all MIPI Members)

- <https://members.mipi.org/wg/All-Members/home/approved-specs#RFFE>

MIPI® Specification for RF Front-End Control Interface (RFFESM) v1.10:

- Specification: Version 1.10 – November 2011
- Application Note: Version 1.10 – November 2011
 - Usage examples (Triggers, Group Slave IDs, Resolving USID Conflicts, etc.)
 - FAQs
- PICS: Version 1.10 – October 2011
 - Protocol Implementation Conformance Statement, Checklist for vendors

MIPI® Specification for RF Front-End Control Interface (RFFESM) v2.0:

- Specification: Version 2.0 – December 2014
- Application Note: Version 2.0 – February 2015
- FAQ: Version 2.0 – February 2015
- Conformance Test Specification (CTS): Version 2.0 – (estimated release: 3Q15)



MIPI/RFFE Website

- **Questions to the WG?** Contact PM: rob.anhofer@mipi.org
- **Questions from Press/other?** Contact Marketing: jennifer.mcaleer@mipi.org
- Public website – RFFE WG: <http://mipi.org/working-groups/rf-front-end>
- MIPI Contributor and Board Members are welcome to join the WG:
 - Member website (request a member login): <https://members.mipi.org/site/login>
 - Access to WG mail reflector and discussions, file repository, calendar, meeting agendas and minutes, schedules and access to Bugzilla change request system
 - Weekly WG meetings: Wednesday's @ 8:30am PST / 11:30am EST / 17:30 CET (2hr)
 - Next RFFE WG conference call is February 25, 2015 (no call Feb-18)
 - RFFE WG calendar: <https://members.mipi.org/wg/RF-FE/calendar>
 - RFFE WG dashboard: <https://members.mipi.org/wg/RF-FE/workgroup>
- Next RFFE WG F2F Meeting:
 - Seattle, USA: Tuesday March 10 to Thursday March 13, 2015
 - Includes MIPI Open Day session with RFFE presentation and Q&A session