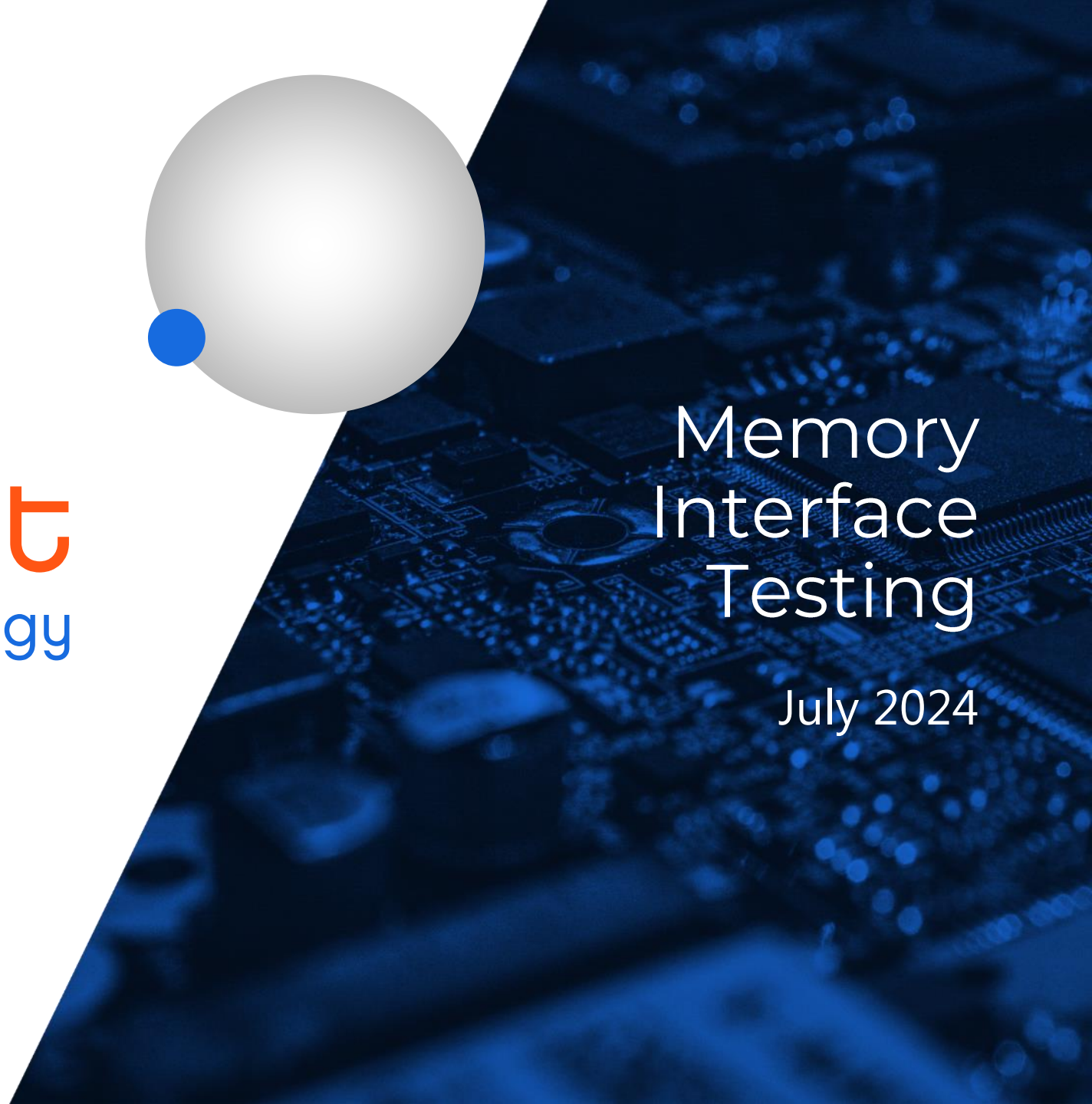
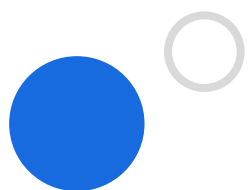


# Memory Interface Testing

July 2024



# Agenda

1. About Introspect Technology
2. Scope of This Presentation
3. ATE on Bench Architecture
4. Memory and Component Testing
5. Host Controller Testing
6. Protocol Analyzer and Interposer Systems
7. SV6E-X SidebandBus Controller and Tester
8. Summary





# About Introspect Technology



# Introspect Makes Tools for Engineers

## ADDRESSING GAP IN TEST EQUIPMENT AVAILABILITY



Bench

~~Costly, few lanes, slow~~



ATE



~~Rigid, low performance~~

- Bench-like accuracy and precision
- EDA style scripting
- Software-style regression and versioning
- ATE-like speed
- Highly parallel
- Designed for automation

# We Test Electronic Interfaces...

**IMAGE  
SENSORS**

**IR RANGE  
FINDERS**

**DISPLAYS**



**MOTION SENSORS,  
MICROPHONES,  
SPEAKERS**

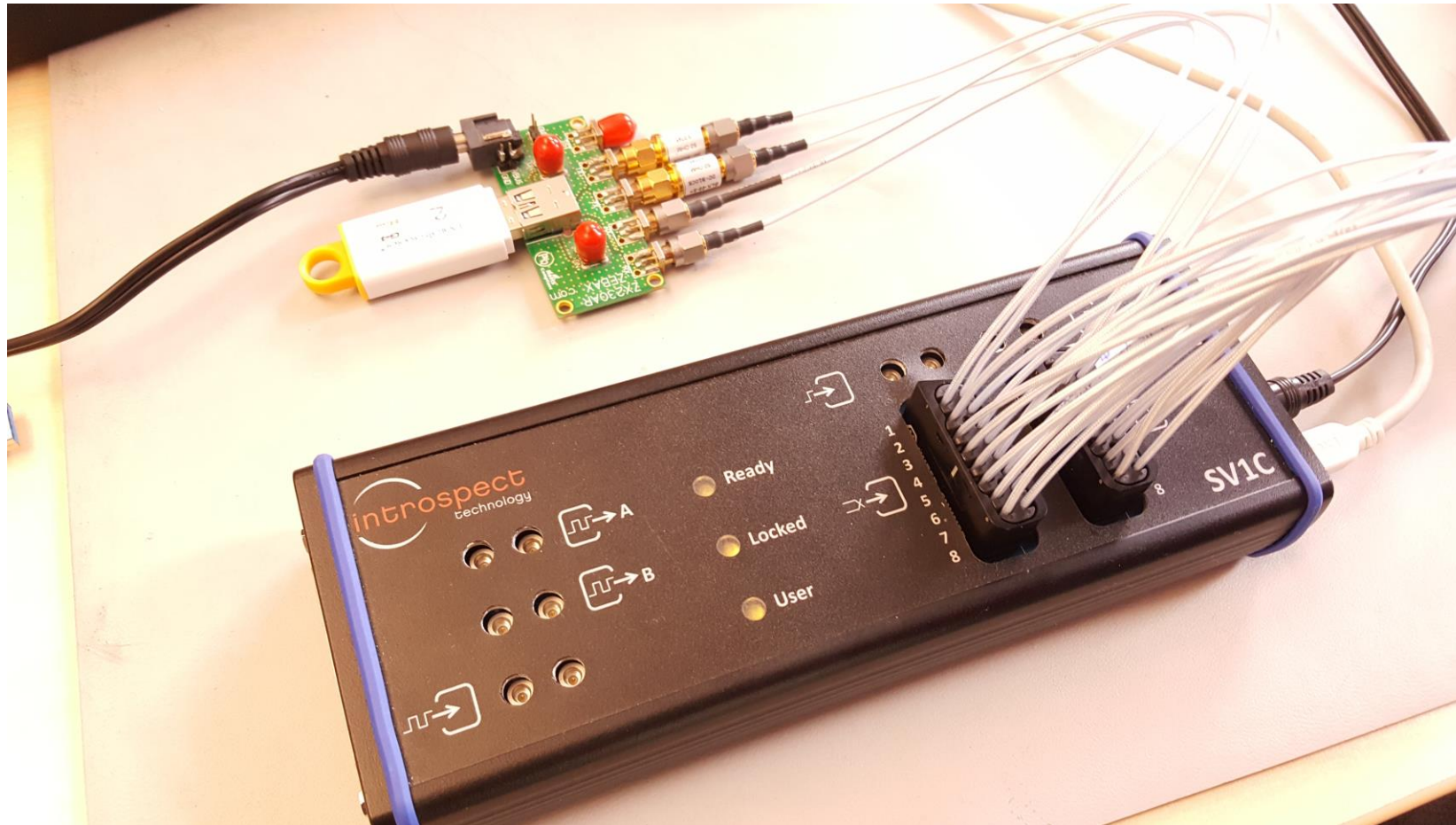
**POWER  
MANAGEMENT ICs**

**APPLICATIONS  
PROCESSORS**

**MEMORIES**

**RF ICs**

# We Act as a Link Partner / Exerciser...





# And We Probe a Live System



# Company Facts

- Founded in **2012**
- Offices in **Montréal, Québec** and **Vancouver, British Columbia**
- **Global, outsourced** sales and distribution channel
- Manufacturer of capital equipment used in the **design validation** and **mass production testing** of electronic components that contain high-speed communications interfaces
  - Smartphones
  - Personal computers and tablets
  - Augmented reality headsets
  - Automotive systems and self-driving systems
  - Data center server racks
  - Medical equipment



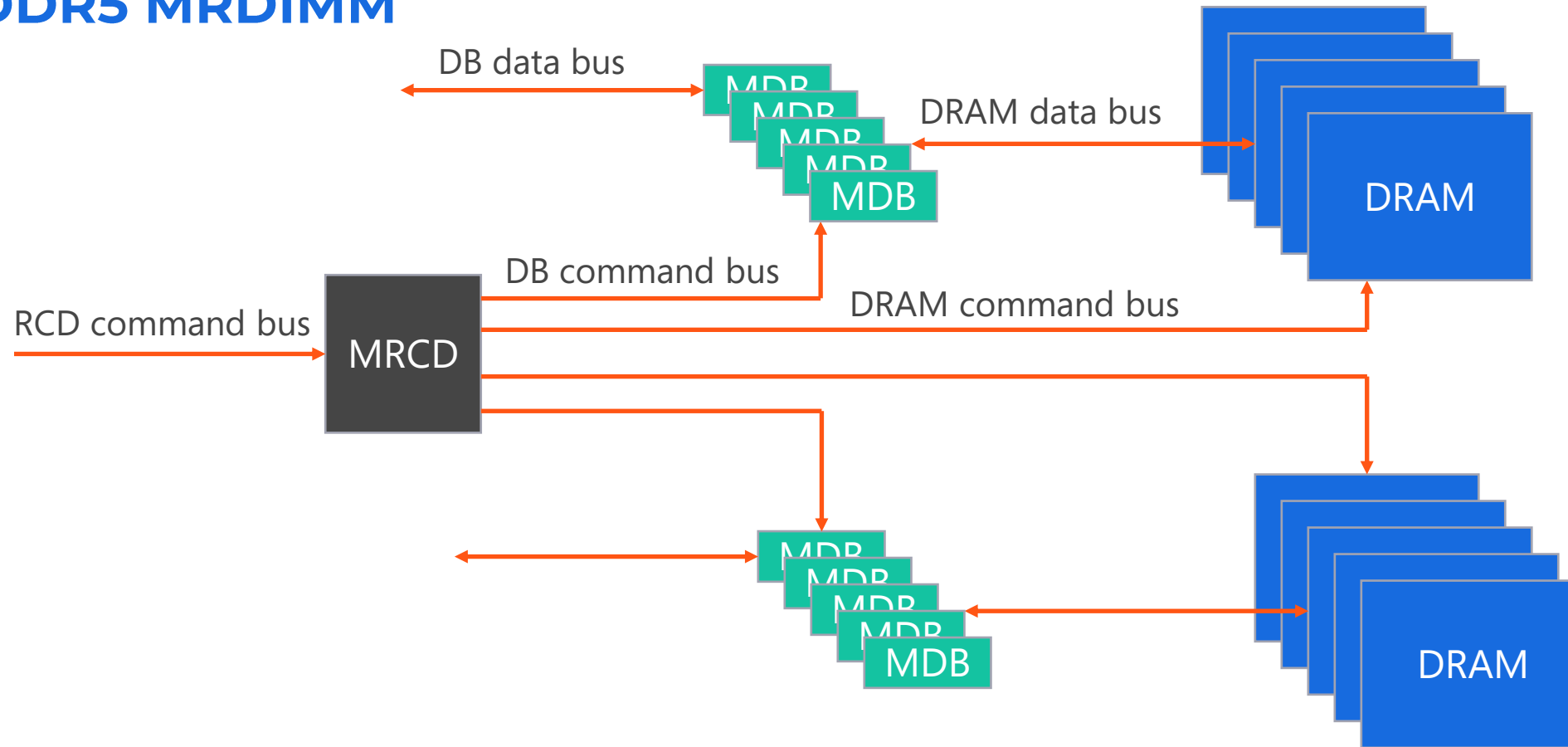


# Scope of This Presentation



# Validating Memory Interfaces

## DDR5 MRDIMM

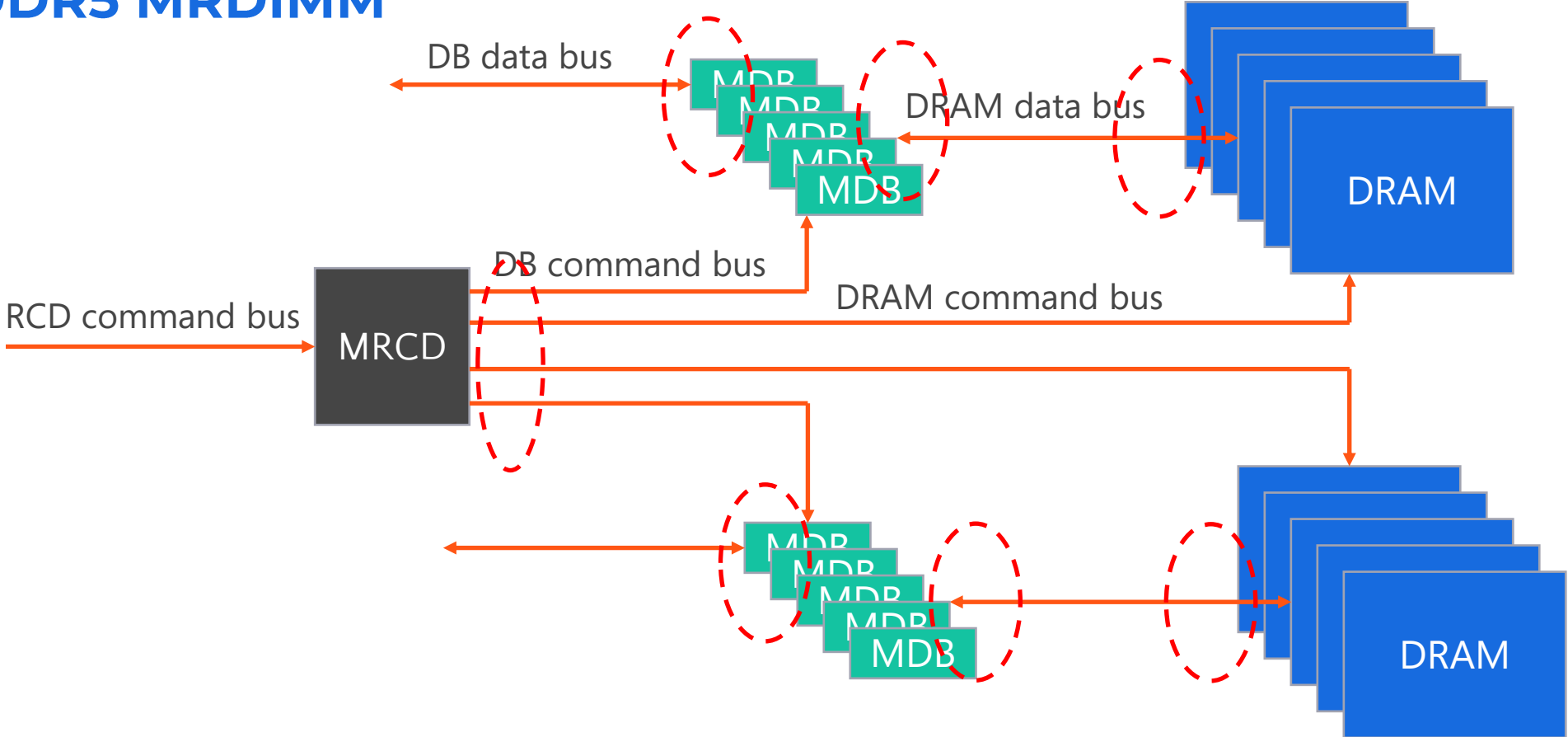


Using the above typical DDR5 DIMM architecture, we will highlight the memory interfaces that are tested by Introspect Technology



# Transmitter Interfaces

## DDR5 MRDIMM

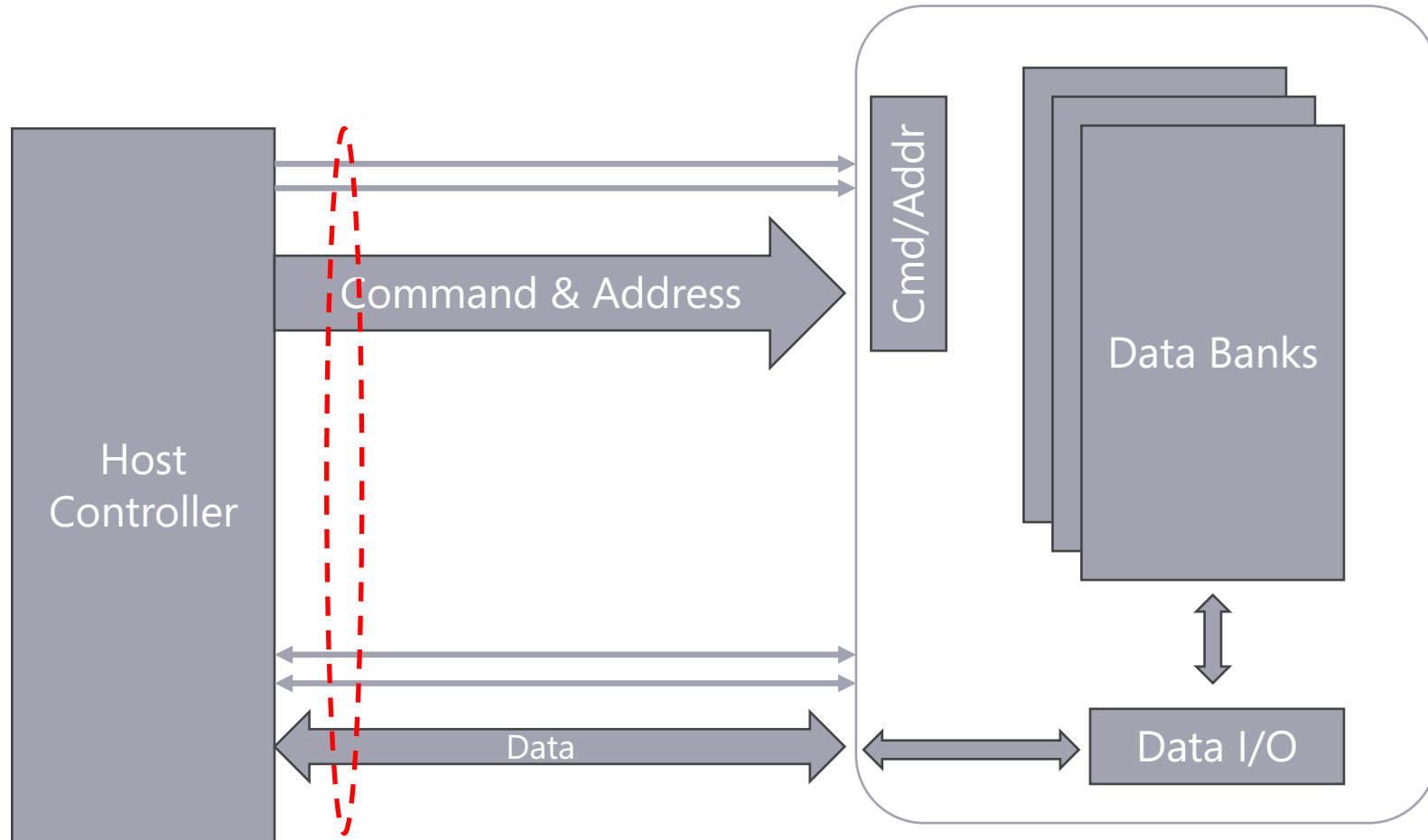


Tx pins on components like RCD, DRAM, and DB

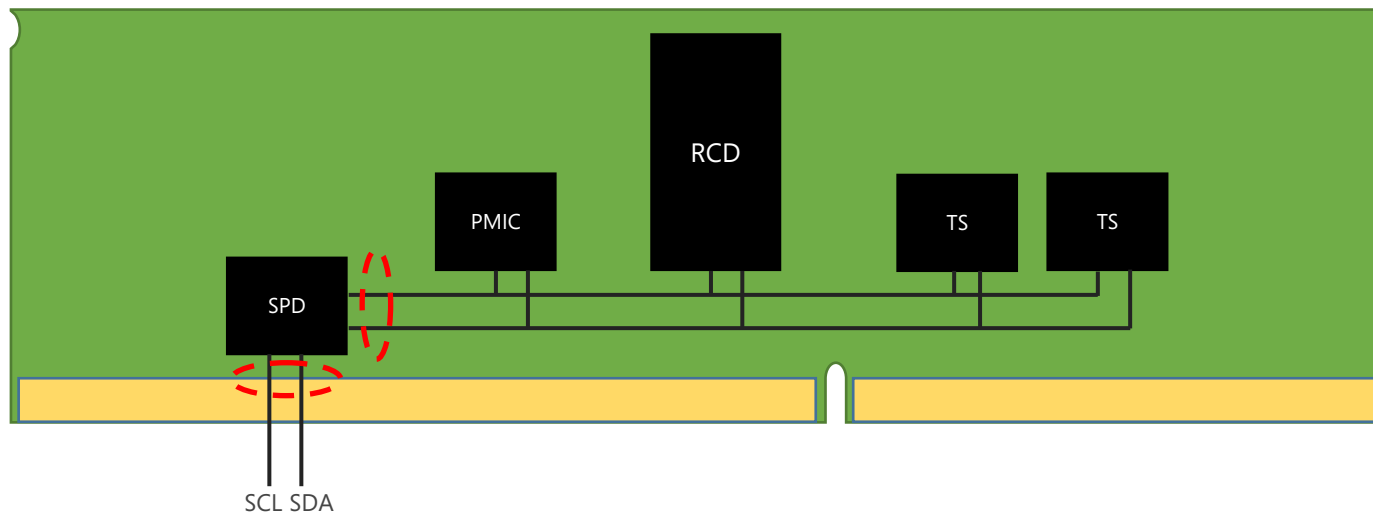




# Validating Controller Components



# Validating SidebandBus Components







# ATE on Bench Architecture



# Really Replace the ATE for DDR Testing

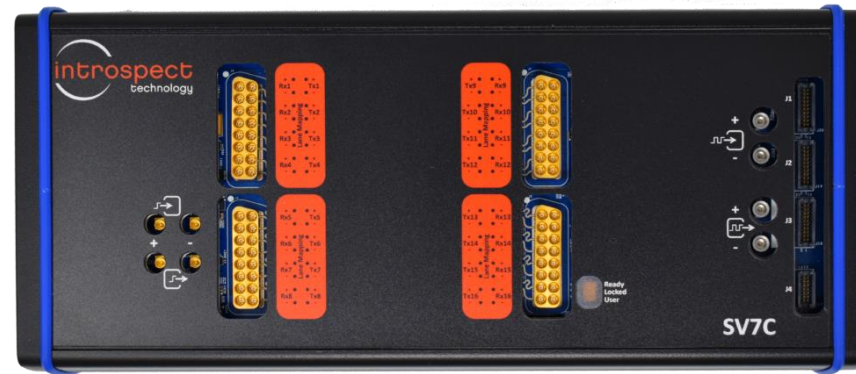


Complete training, shmooring  
capability on all pins



# SV7C-17 ATE on Bench

DEVELOPED FOR DDR/LPDDR/GDDR INTERFACE TESTING



# ATE on Bench

## DEVELOPED FOR DDR/LPDDR/GDDR INTERFACE TESTING

### CAPTURE AND ANALYSIS

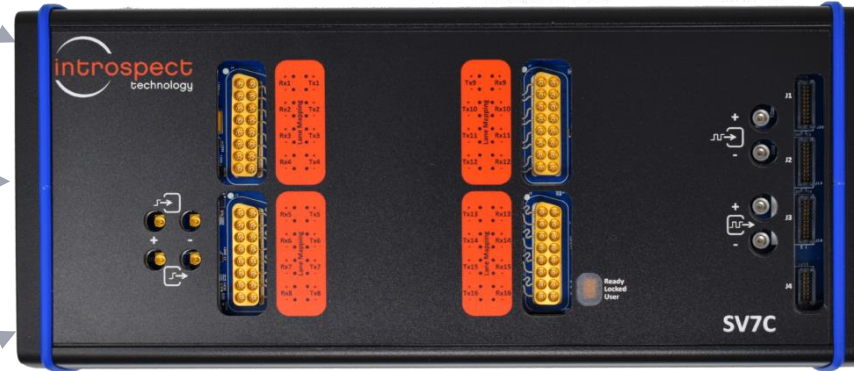
Eye diagrams and BER  
Parallel digital capture  
Analog capture

### 16 RX CHANNELS

Per lane phase control  
Per lane voltage threshold

### 16 ALIGNED RECEIVERS

capable of measuring tx skew



### AUXILIARY GPIO

Provides a host of secondary controls (e.g. triggers, flags)

### 16 TX CHANNELS

Per lane jitter and noise injection  
Per lane slew rate settings  
Per lane skew control  
Per lane voltage control

### 16 PATTERN GENERATORS

Aligned on power up  
Protocol-configurable



# ATE on Bench

## DEVELOPED FOR DDR/LPDDR/GDDR INTERFACE TESTING

### CAPTURE AND ANALYSIS

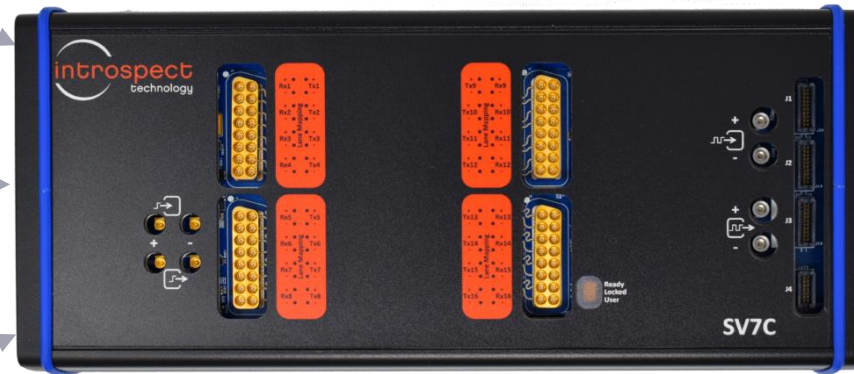
Eye diagrams and BER  
Parallel digital capture  
Analog capture

### 16 RX CHANNELS

Per lane phase control  
Per lane voltage threshold

### 16 ALIGNED RECEIVERS

capable of measuring tx skew



### AUXILIARY GPIO

Provides a host of secondary controls (e.g. triggers, flags)

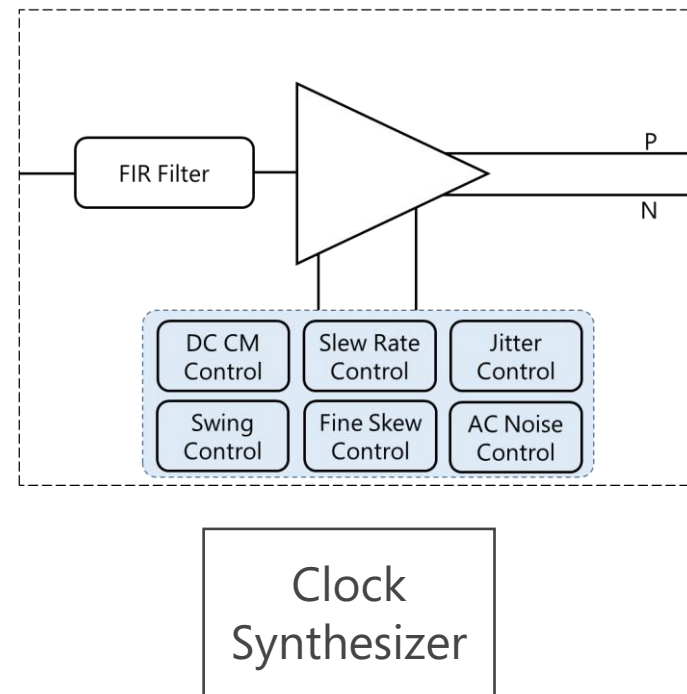
### 16 TX CHANNELS

Per lane jitter and noise injection  
Per lane slew rate settings  
Per lane skew control  
Per lane voltage control

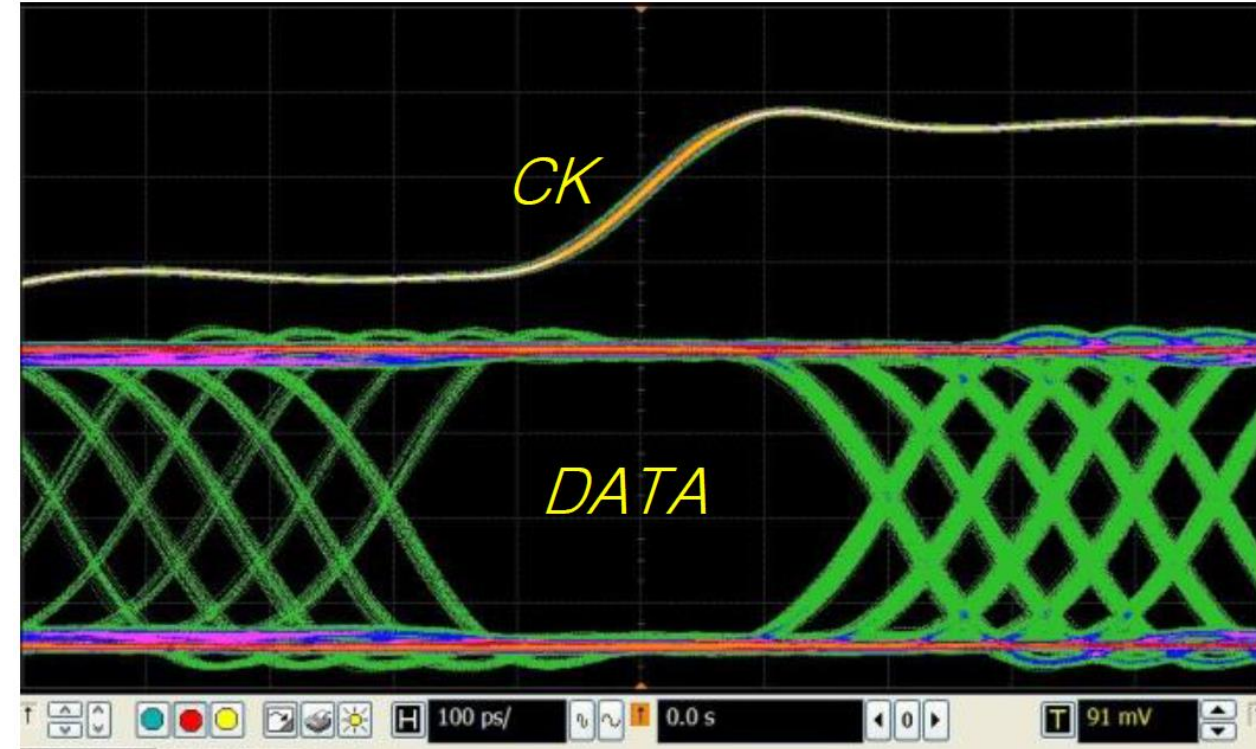
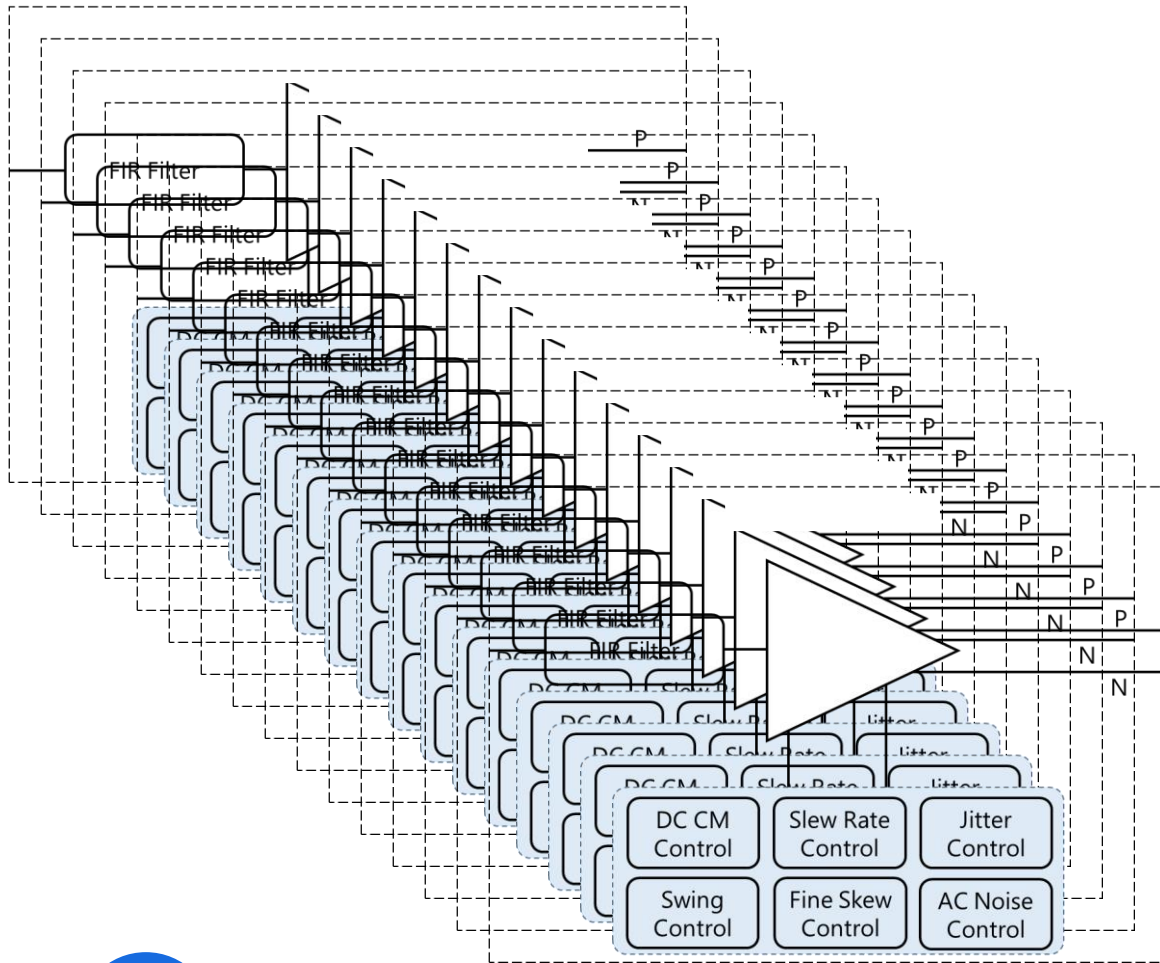
### 16 PATTERN GENERATORS

Aligned on power up  
Protocol-configurable

# Single-Lane Pin Electronic Driver

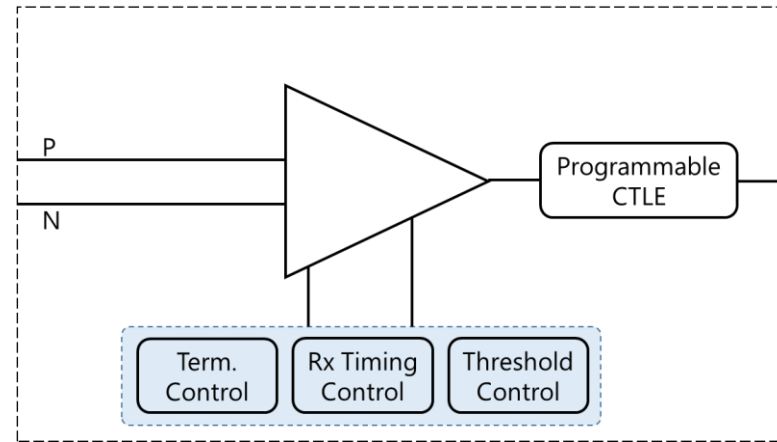


# All Drivers Work as a Bus



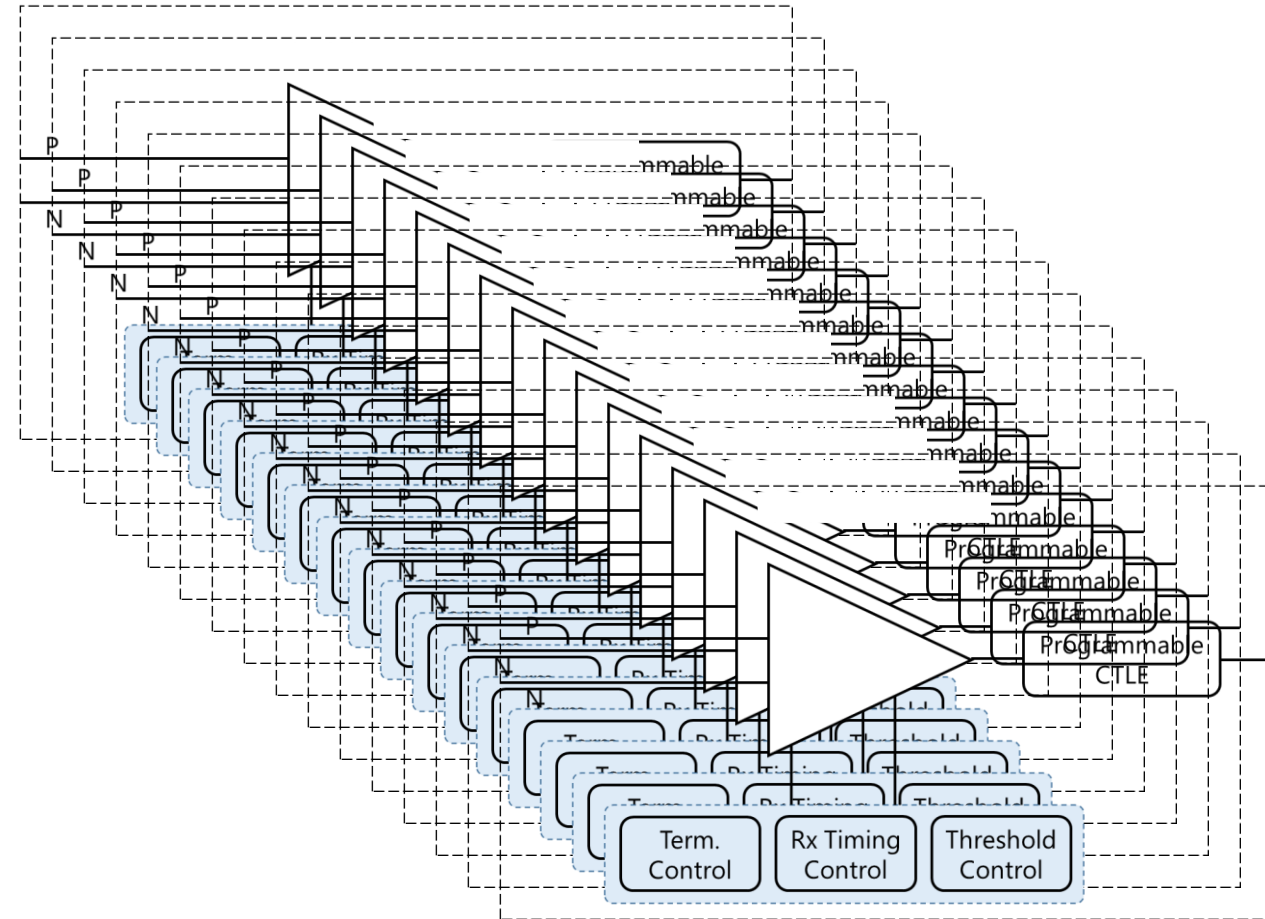
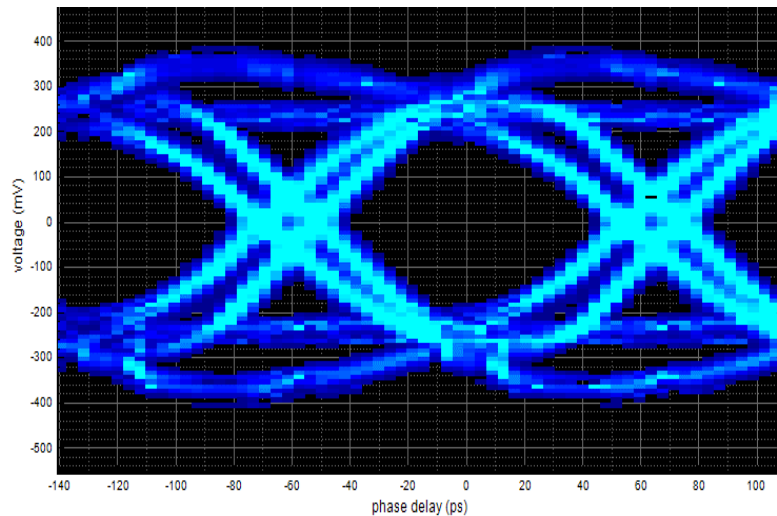
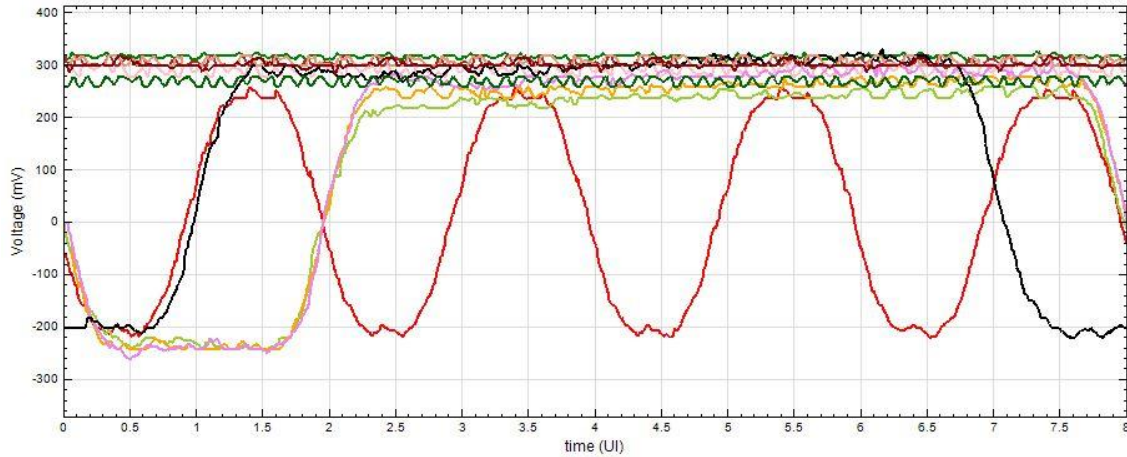
Can produce realistic DDR/LPDDR waveform shapes & skews

# Single-Lane Pin Electronic Receiver

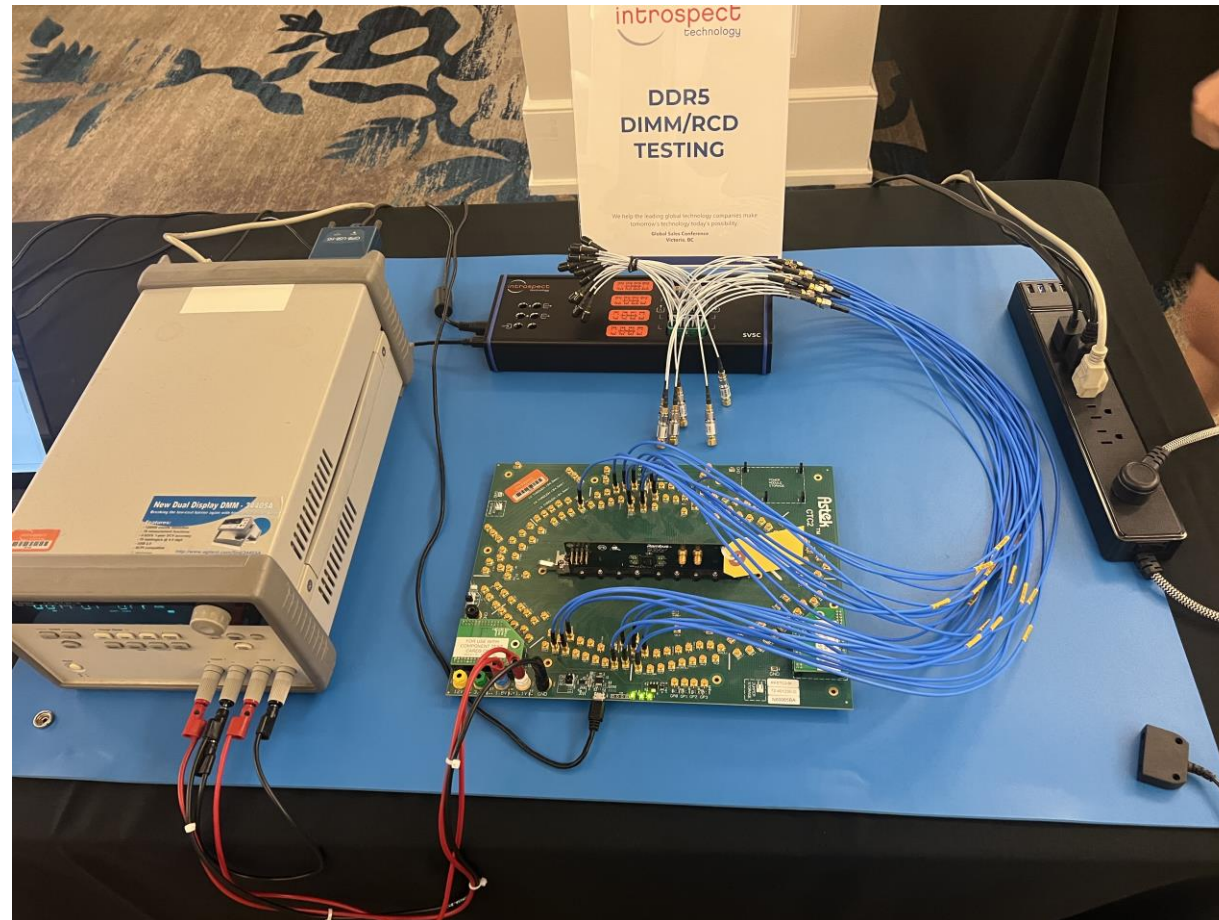




# All Receivers Work as a Bus

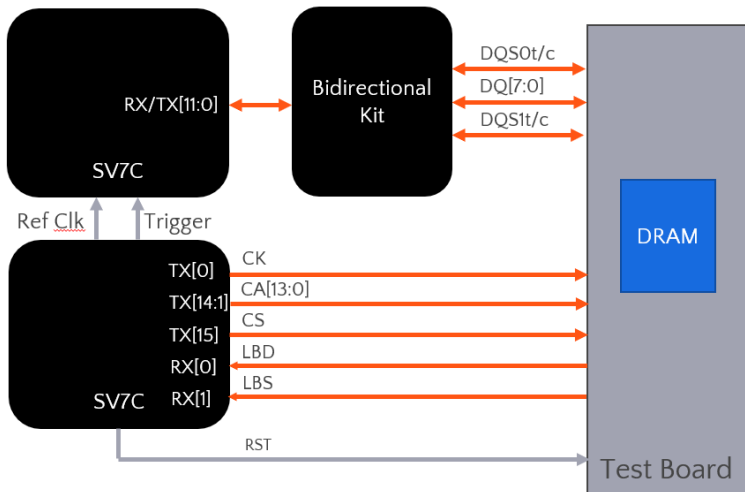


# Size Illustration (DDR and GDDR Component or PHY Test)

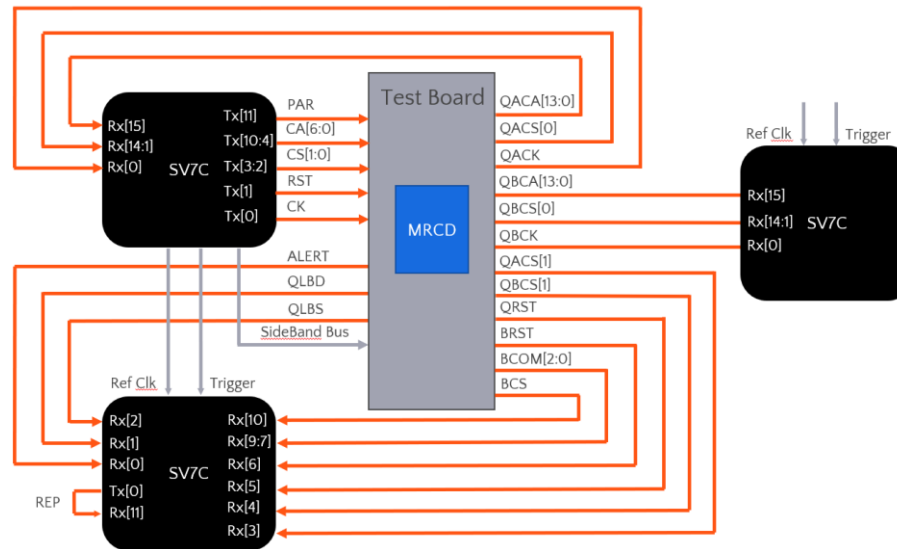


# Typical DDR5 Test Benches

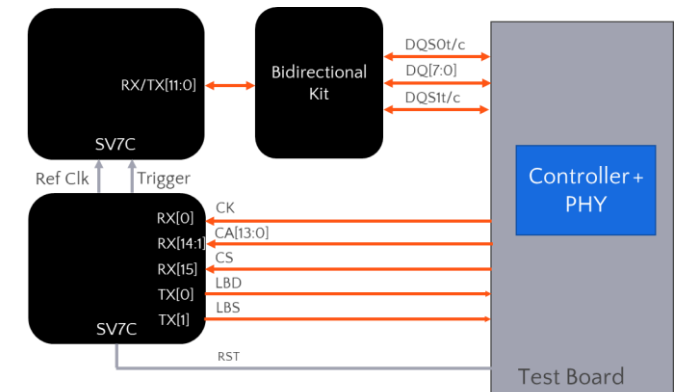
## DRAM



## MRC D/RCD

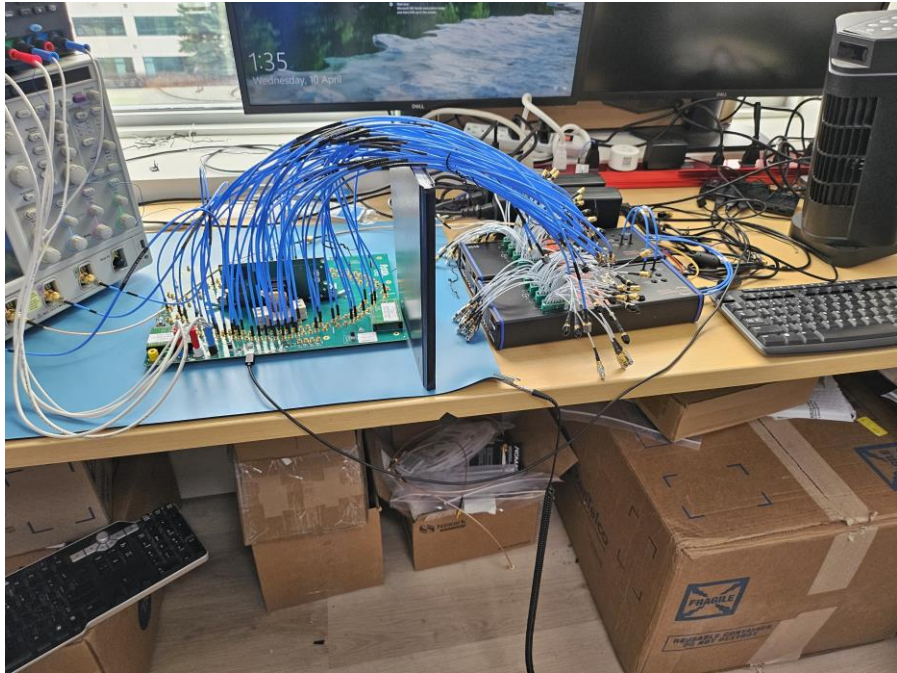


## CONTROLLER

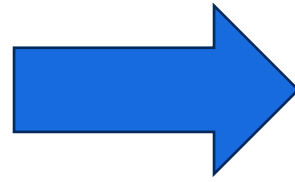




# Evolution to HS Memory ATE – M Series



DDR5 RDIMM Setup



Enclosed System to Hide the Cables and Increase Performance

# M Series Variants

COMPONENT TESTER



GDDR7 PAM3

COMPONENT TESTER



LPDDR5/LPDDR6

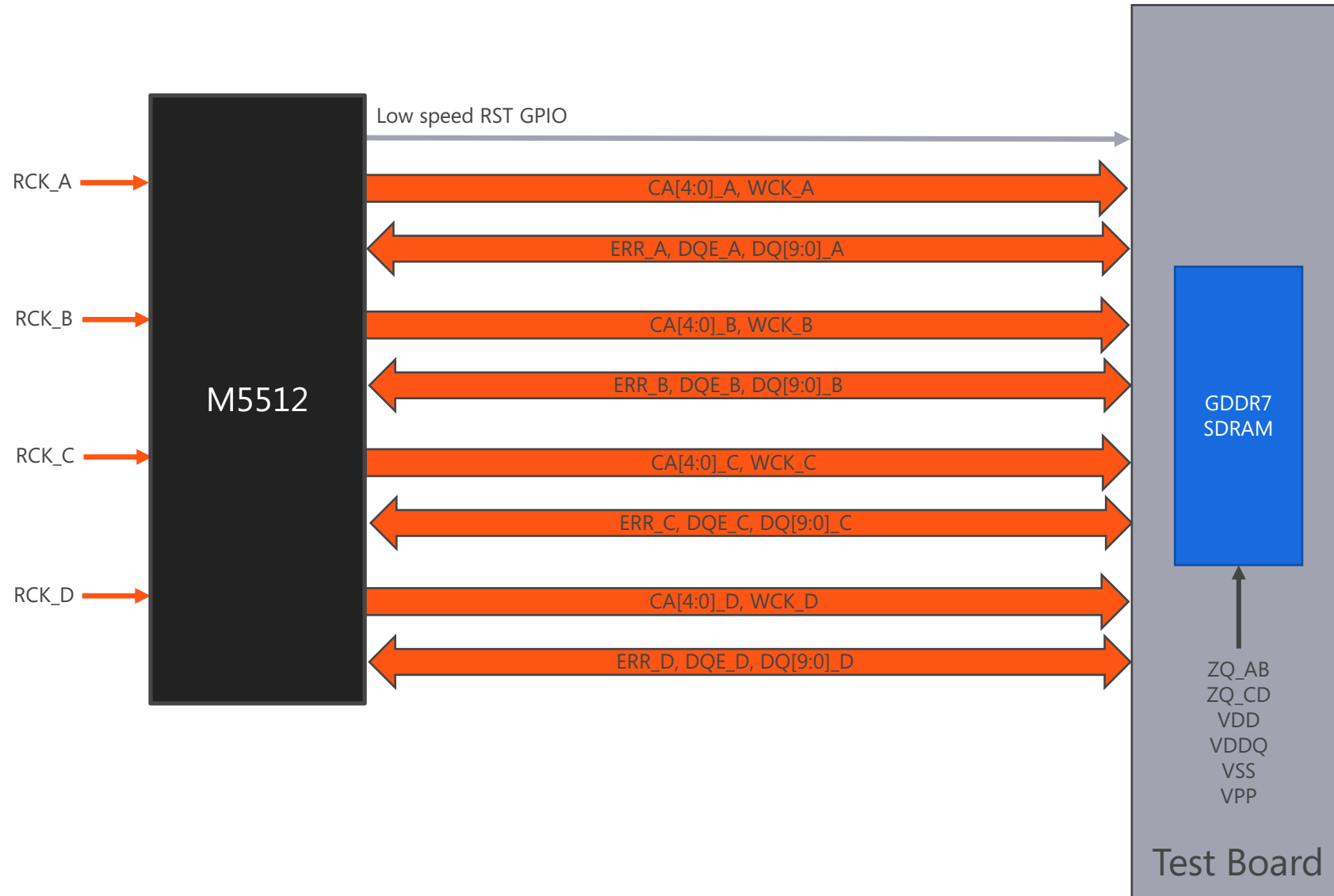
MODULE TESTER



DDR5 MR-DIMM / SO-DIMM / U-DIMM

World-first PAM3 interop test on February 7, 2024 (with leading memory maker)

# Example: GDDR7 4-Channel System







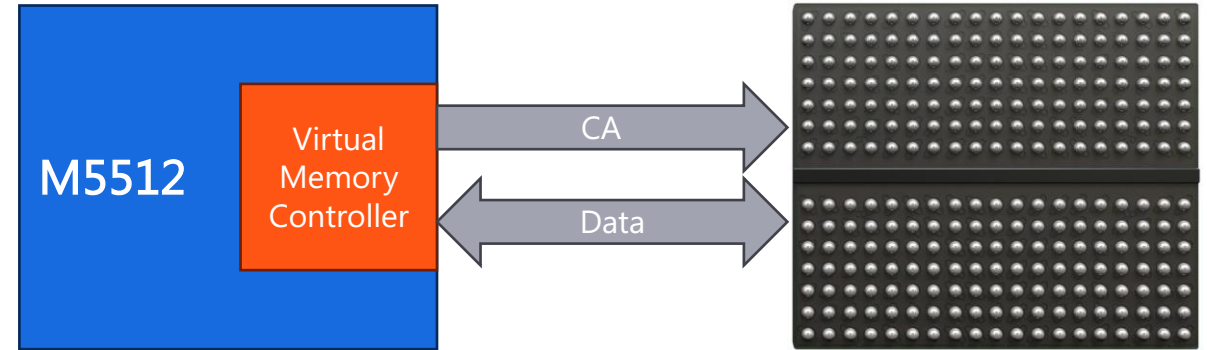
# Memory and Component Testing



# Memory Device Testing

## PHY LEVEL TESTING

Adjustable voltage and timing parameters on all pins  
(including jitter injection)

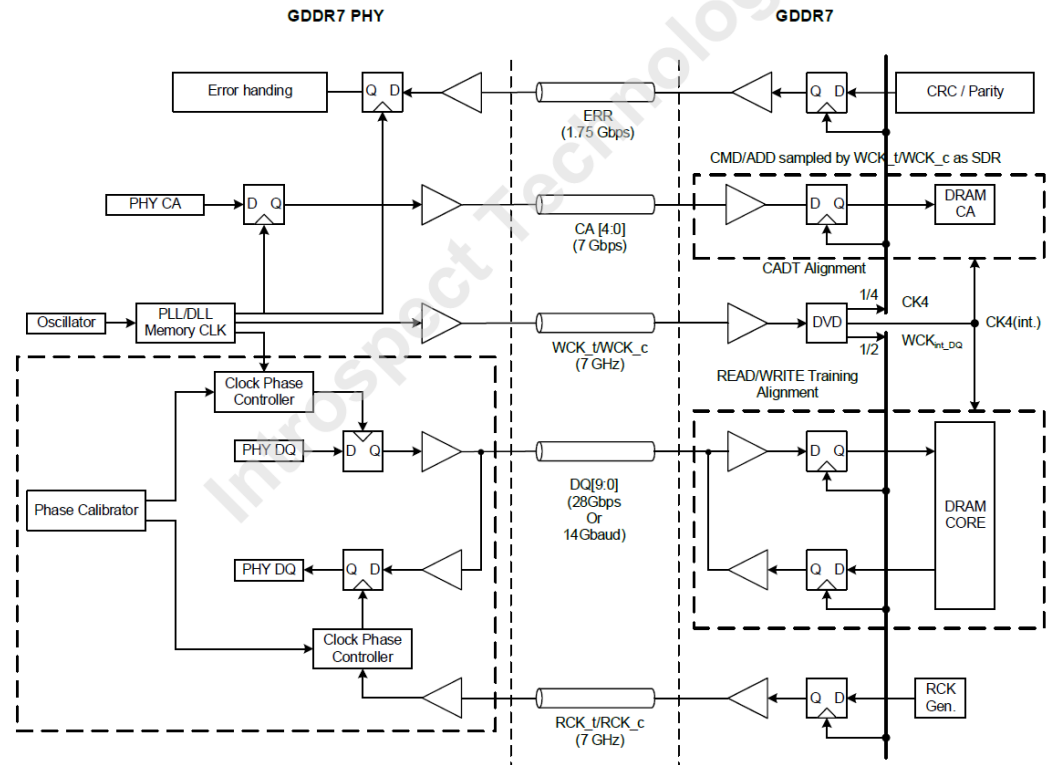


## FUNCTIONAL TESTING

Protocol-compliant stimulus for all memory commands

## FUNCTIONAL STRESS TESTING

What are the limits of the device? Verify functional behaviour while pushing command timings, data rate and other parameters out of spec



# Memory Controller Software (GD7 Example)

The screenshot displays a software configuration interface for DDR memory controller software. The interface is organized into three main sections:

- Left Sidebar (Component Tree):** A vertical list of components under the 'DDR' category. The components include: DramController, DramParams, DramPhyParams, LpDramController, LpDramParams, LpPhyParams, PhyParams, RcdController, RcdParams, RdimController, GddrController, GddrParams, GddrPhyParams, DdrDataCapture, DdrDbCommandPa..., and DdrDramCommand... The 'gddrController1' component is highlighted with a dashed border.
- Central Configuration Panel (gddrController1):** A table of configuration parameters for the selected component. The parameters and their values are:

Parameter	Value
deviceSerialNum	1234
trainingDataFolderPath	
memBusesUnderTest	ABCD
phyParams	gddrPhyParams1
gddrParams	gddrParams1
rxChannelLabeling	gddrChannelLabeling1
txChannelLabeling	gddrChannelLabeling1
calibrateZq	True
trainingDataCaPhase	auto
caPhaseTrainingNumStepsPerUi	32
trainingDataCaVref	auto
caVrefTrainingStepSize	1
trainingDataReadVref	auto
trainingDataReadPam3EyeOffset	auto
trainingDataReadPhase	auto
- Right Panel (Procedure):** A list of four steps for the configuration procedure:
  - 1 gddrController1.run()
  - 2
  - 3 *#Sending MRS command...*
  - 4 gddrController1.sendMrs(13,128)



# Memory Controller Software (GD7 Example)

**Components**

- gddrChannelLabeling1
- gddrController1**
- gddrParams1
- gddrPhyParams1
- pam3Protocol

**gddrController1**

deviceSerialNum	1234
trainingDataFolderPath	
memBusesUnderTest	ABCD
phyParams	<b>gddrPhyParams1</b>
gddrParams	<b>gddrParams1</b>
rxChannelLabeling	<b>gddrChannelLabeling1</b>
txChannelLabeling	<b>gddrChannelLabeling1</b>
calibrateZq	True
trainingDataCaPhase	auto
caPhaseTrainingNumStepsPerUi	32
trainingDataCaVref	auto
caVrefTrainingStepSize	1
trainingDataReadVref	auto
trainingDataReadPam3EyeOffset	auto
trainingDataReadPhase	auto

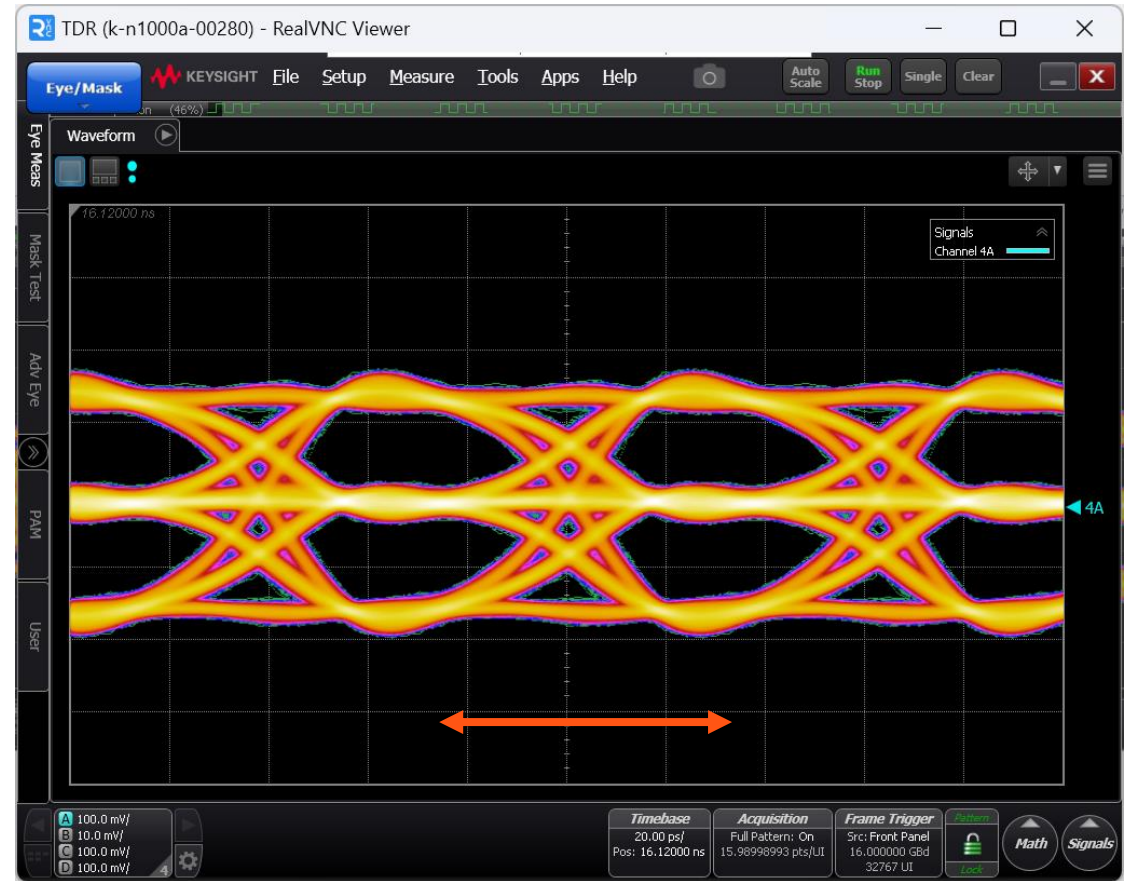
**gddrPhyParams1**

dataRate	16000.0
caVLow	250.0
dqVLow	250.0
wckVLow	200.0
caVHigh	650.0
dqVHigh	650.0
rxVrefInitialValue	600.0
wckVHigh	600.0

**gddrParams1**

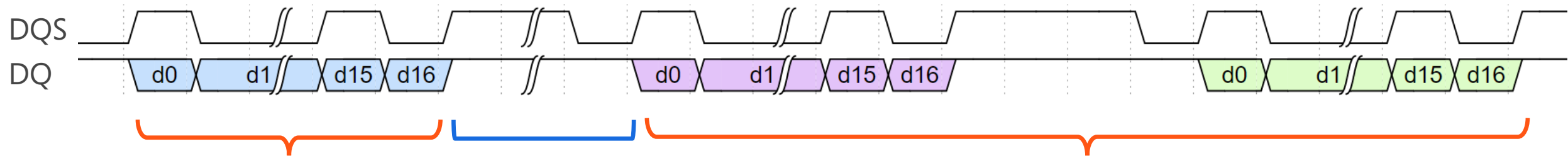
readLatency	19
writeLatency	10
writeCrc	False
readCrc	False
cabi	False
vrefDqInitialValue	100
vrefDqhInitialValue	100
wrCrc2Err	10
dqeRL	1
dqDqeRckDriverStrength	term40Ohm
dqDqeTermination	termOff
errDriverStrength	term40Ohm
calUpd	allEnabled
sev2Err	False
dqeHighZ	False

# Driver Performance – 32 Gbps



# Pattern Generation Architecture

## HOLD PATTERNS DRIVE IDLE STATES IN BETWEEN PATTERN SEQUENCES



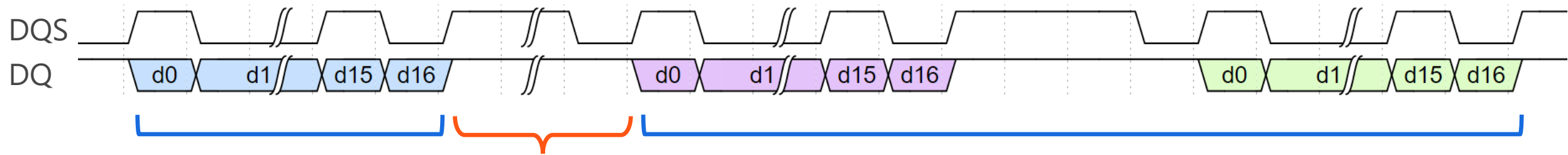
```
busPatternTimeline.addPatterns(burstPattern,1)  
busPatternTimeline.endWithPattern('HoldPattern')
```

```
busPatternTimeline.addPatterns(burstPattern1,1)  
busPatternTimeline.addPatterns(idlePattern,1)  
busPatternTimeline.addPatterns(burstPattern2,1)  
busPatternTimeline.endWithPattern('HoldPattern')
```



# Pattern Generation Architecture

## IN BETWEEN PATTERN SEQUENCES...



Start jitter injection on WCK...

Modify signal voltages levels on CA...

Control timings between DQ signals...

Change slew rate on DQ signals...

# Tuned Virtual Memory Controllers

## DDR5 DRAM

Used to connect to a single  
DDR5 DRAM

Component test paradigm

## RCD/MRCD

Used to connect to a single  
RCD or MRCD device

Component test paradigm

## RDIMM/MRDIMM

Used to connect to an entire  
RDIMM

System-level test paradigm

## LPDRAM

Used to connect to a single  
LPDRAM memory

Component test paradigm

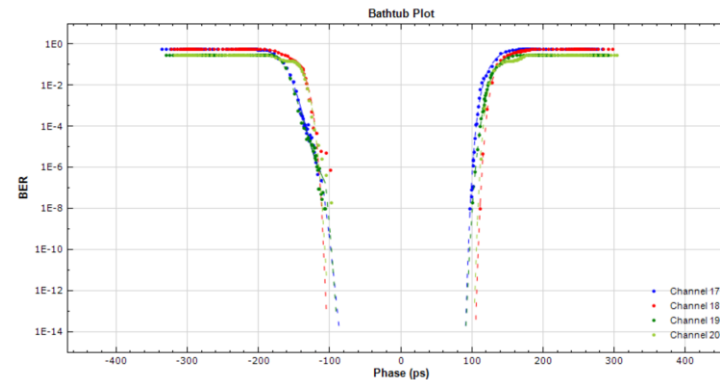
## GDDR

Used to connect to a single  
GDDR memory

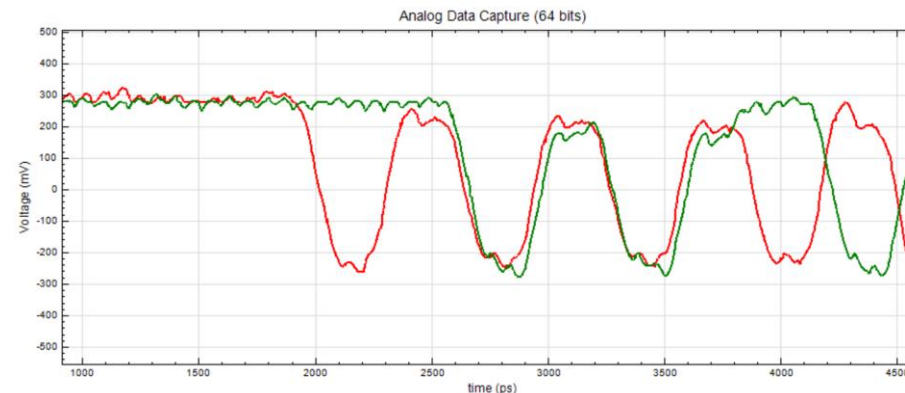
Component test paradigm

# Transmitter Characterization

- Clock to data skew measurement
  - DQS to DQ
  - CK to CA
  - BCK to BCOM
- BERT measurements on clock and data
  - Long duration error rate tests
  - Eye diagrams
  - Jitter measurements
  - Slew rate measurements



DQS and DQ jitter measurement

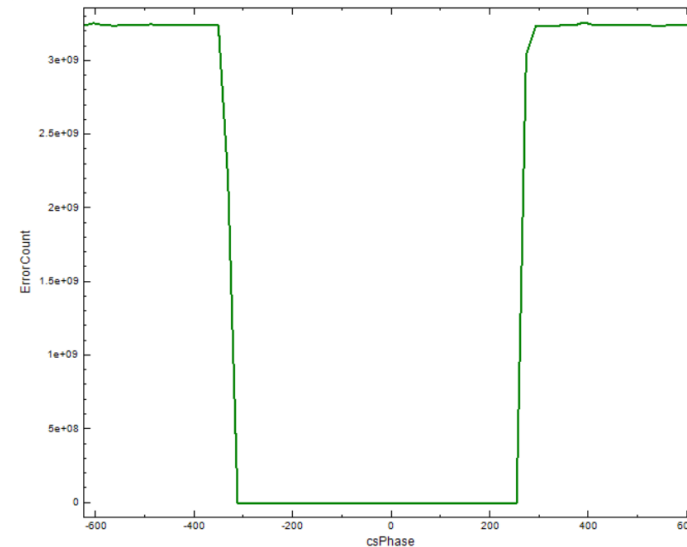
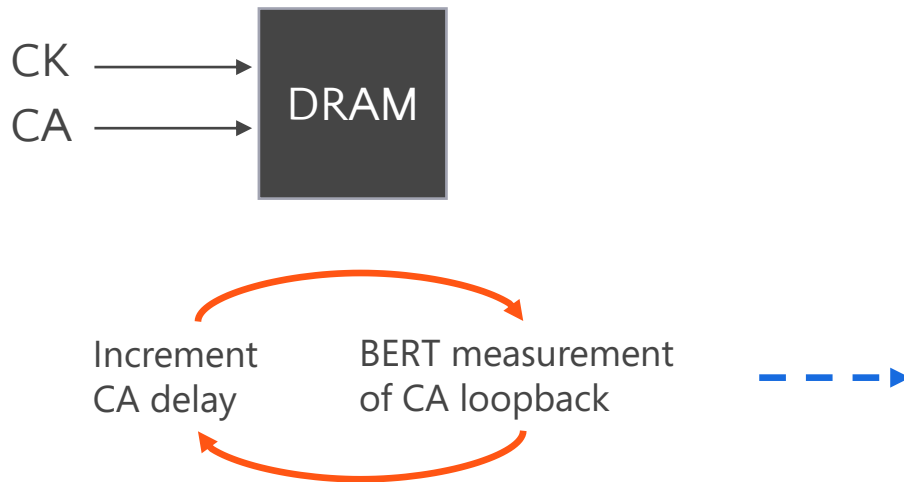


DQ to DQS skew measurement

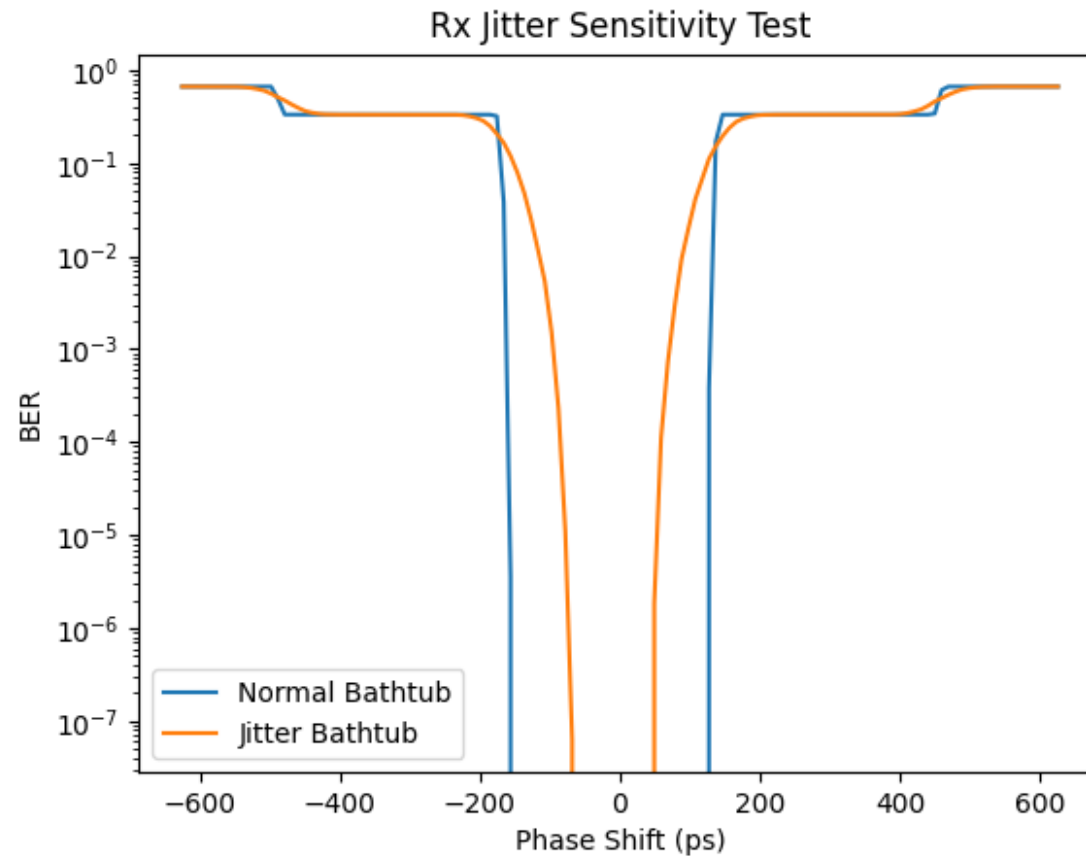


# Receiver Characterization

- Requires support of different loopback/training modes or uses write and read commands
- Measure horizontal and vertical eye opening at receiver while applying different stressors
  - Jitter sensitivity – clock to data skew
  - Voltage sensitivity
  - Stressed eye



# Example Receiver Stress Test Result

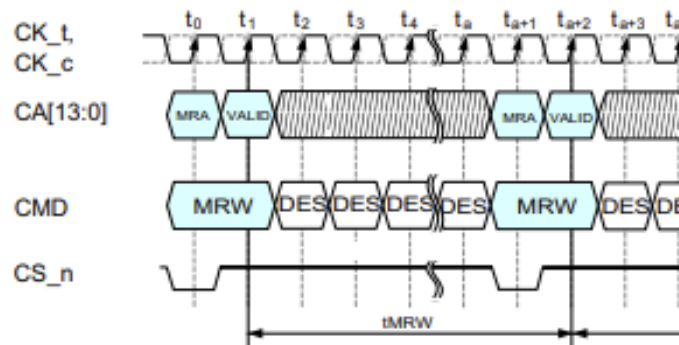


# Functional Testing

- Perform pass/fail checks of device behaviour under normal test conditions
- There is a wide variety of tests in this category, that can be roughly categorized as follows
  - **Functional mode tests** – E.g. Training modes, power down, DDR/SDR operation
  - **Input spec checks** – E.g. Mode register writes, MPC commands
  - **Output spec checks** – E.g. QCS operation, output inversion, Qx output delay

## MODE REGISTER WRITE TEST EXAMPLE

Send MRW commands to RCD → Read RW contents over sideband bus



PASS if read and write data match

FAIL if no match





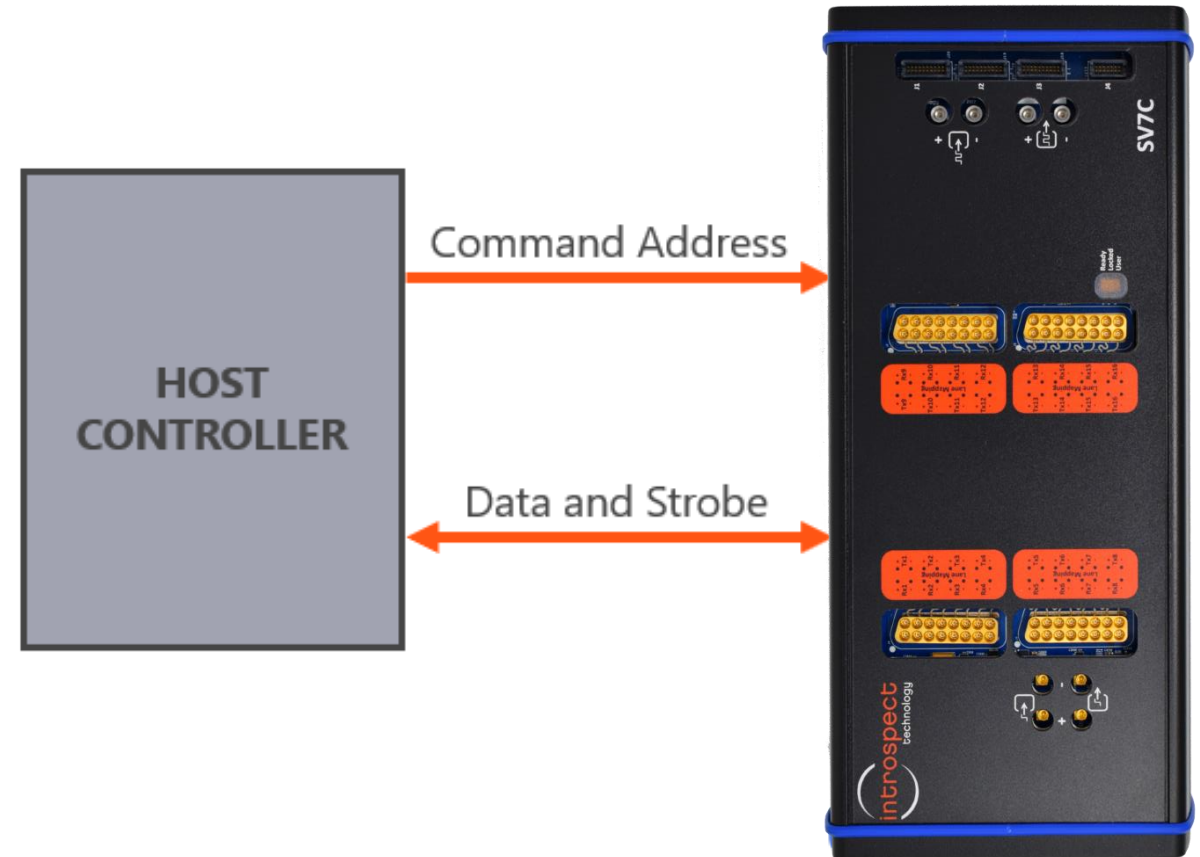
# Host Controller Testing



# Controller Testing

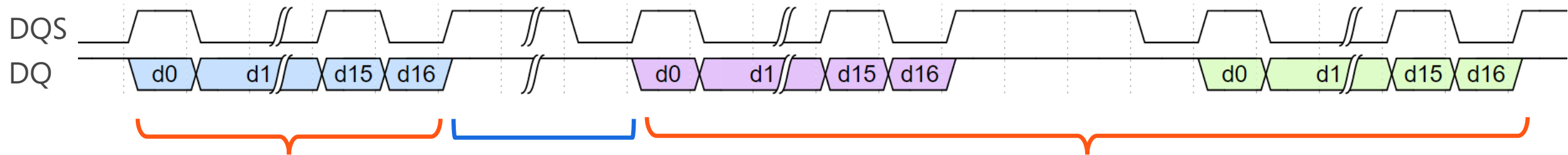
- Supports TX testing of command and data bus
  - Clock to data skew measurements
  - Eye diagrams, BER measurements etc...
- Supports RX testing of data bus using flexible pattern features and analog impairment controls. Requires loopback outputs, built-in error counters, or similar test mode in the host controller.

There is no formal JEDEC spec for the memory controller, so test conditions and pass/fail requirements are not defined.



# Pattern Generation

## HOLD PATTERNS DRIVE IDLE STATES IN BETWEEN PATTERN SEQUENCES

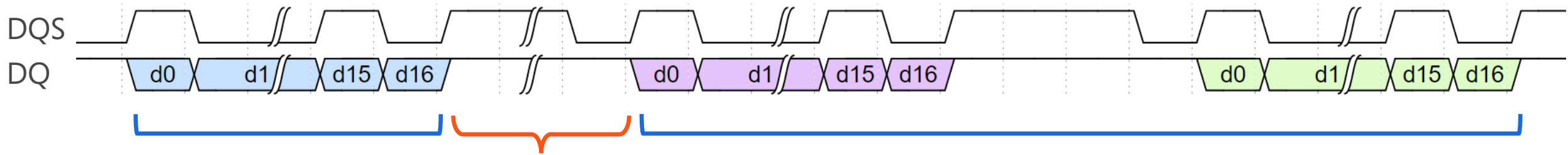


```
busPatternTimeline.addPatterns(burstPattern,1)  
busPatternTimeline.endWithPattern('HoldPattern')
```

```
busPatternTimeline.addPatterns(burstPattern1,1)  
busPatternTimeline.addPatterns(idlePattern,1)  
busPatternTimeline.addPatterns(burstPattern2,1)  
busPatternTimeline.endWithPattern('HoldPattern')
```

# Pattern Generation

IN BETWEEN PATTERN SEQUENCES...



Start jitter injection

Modify signal voltages levels

Control timings between signals

Change slew rate



# Timing and Voltage Control

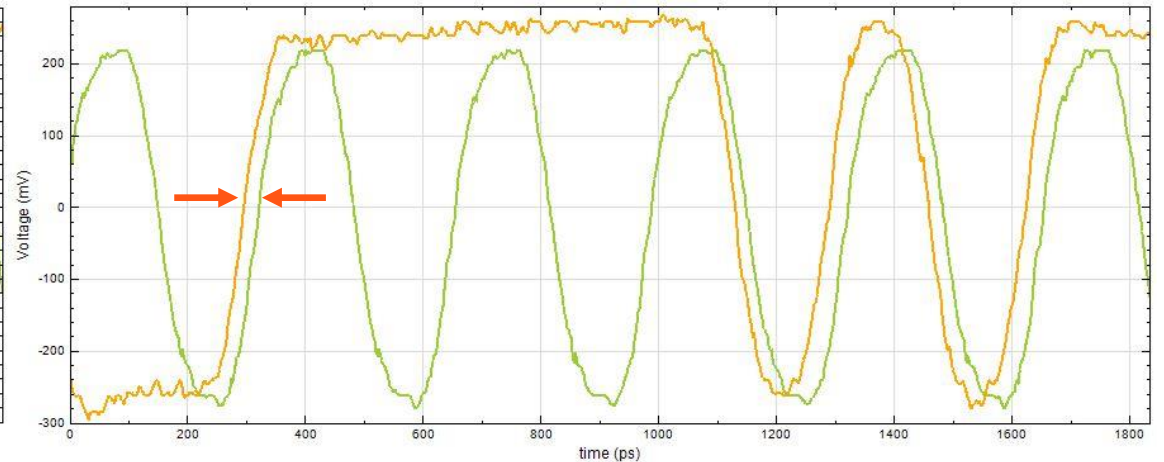
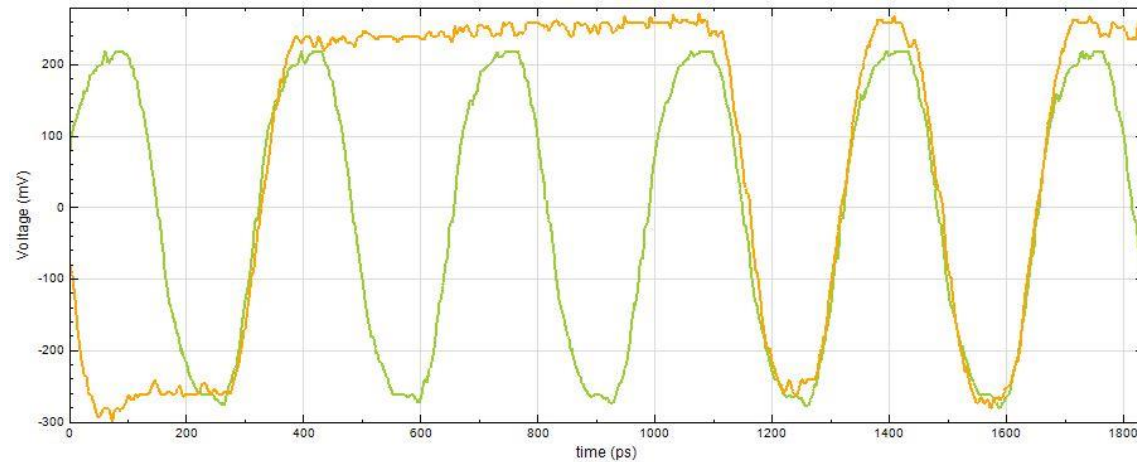
## PER LANE PHASE, AMPLITUDE AND COMMON MODE SETTINGS



# Transmitter Skew Measurement

## MEASURE CHANNEL TO CHANNEL SKEW

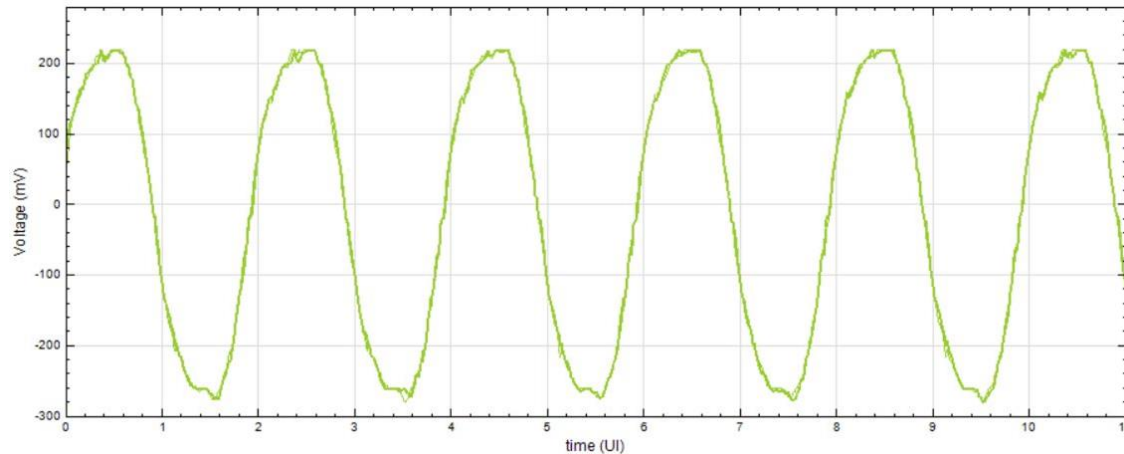
Detect and measure DQ to DQS delay



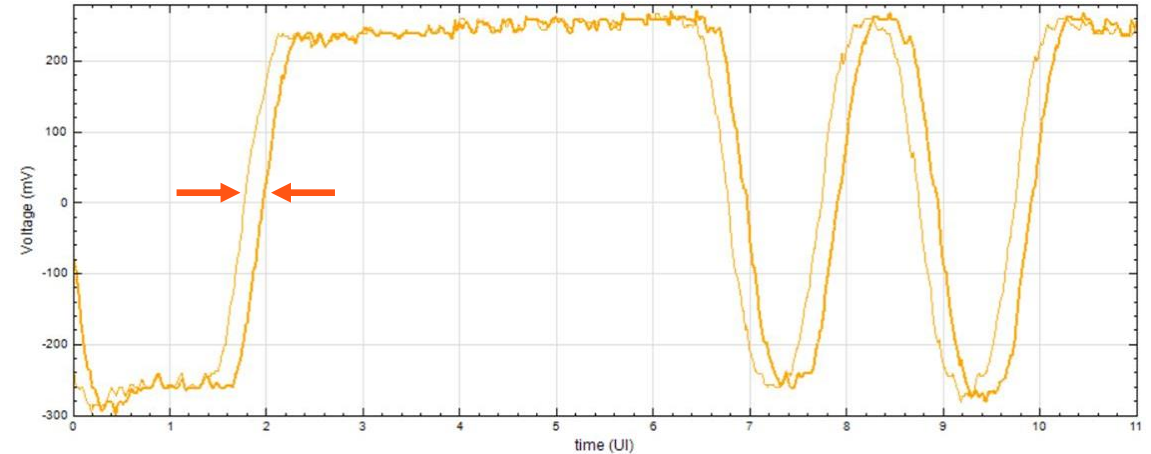
# Transmitter Skew Measurement

VIEW PATTERN ALIGNMENT ON A SINGLE CHANNEL OVER TIME

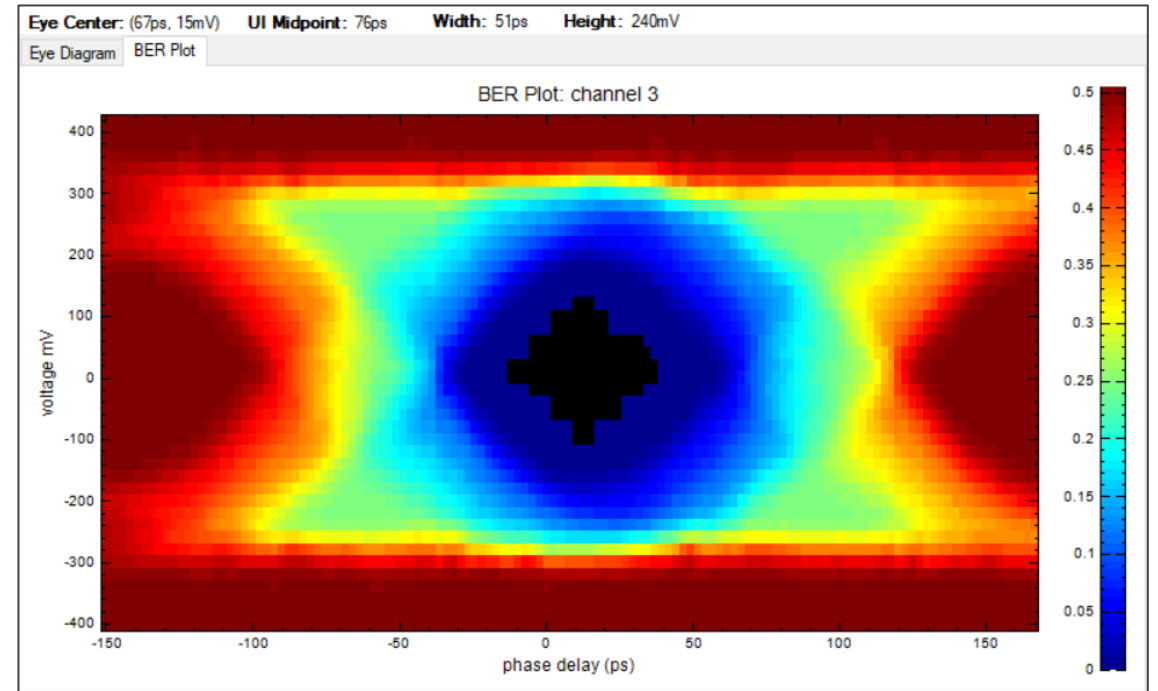
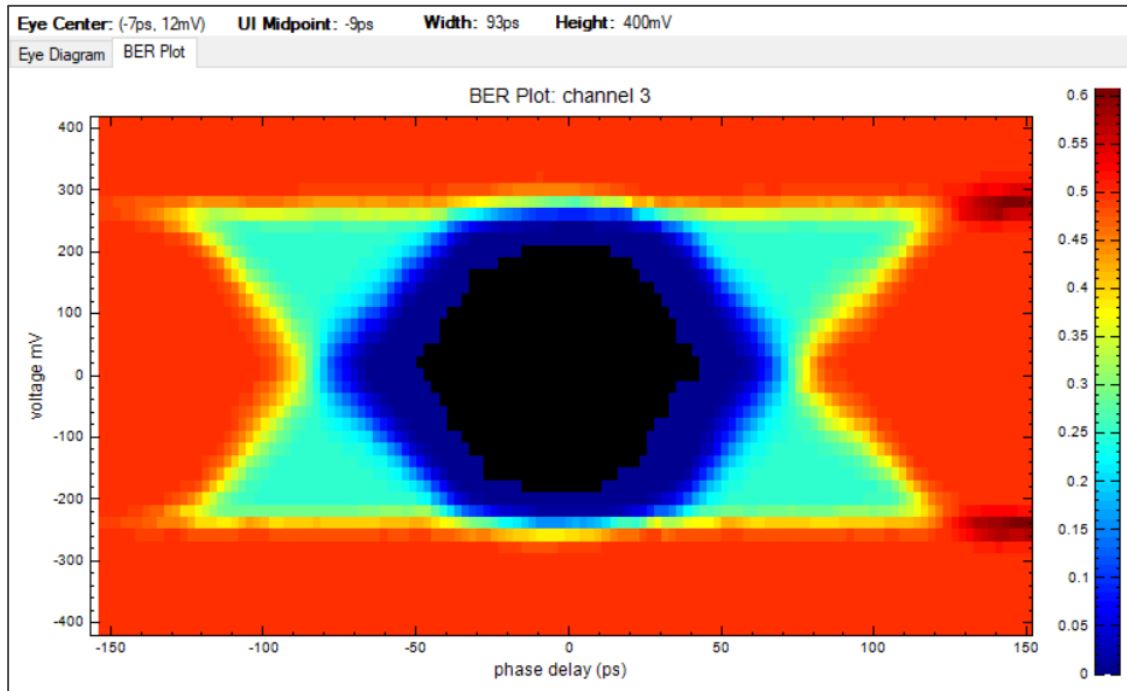
DQS position is constant



DQ position has changed



# BERT Measurements



Eye diagrams across different transmitter slew rate settings



# Burst Mode Digital Capture

## CAPTURE AND FILTER OUT BURST DATA

DQS

DQ0

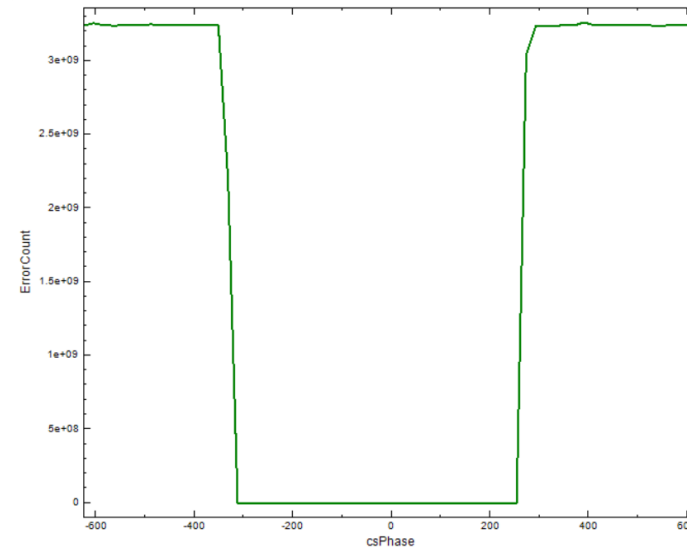
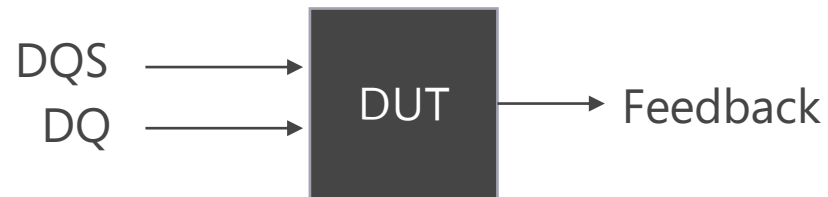
Channel	Burst#	Offset	Status	Bits
1				
	0	2	good	1010101010101010
	1	66	good	1010101010101010
	2	130	good	1010101010101010
	3	194	good	1010101010101010
	4	258	good	1010101010101010
2				
	0	2	good	0111011000110000
	1	66	good	0101000010100100
	2	130	good	0011001010110100
	3	194	good	0111011000110000
	4	258	good	0101000010100100





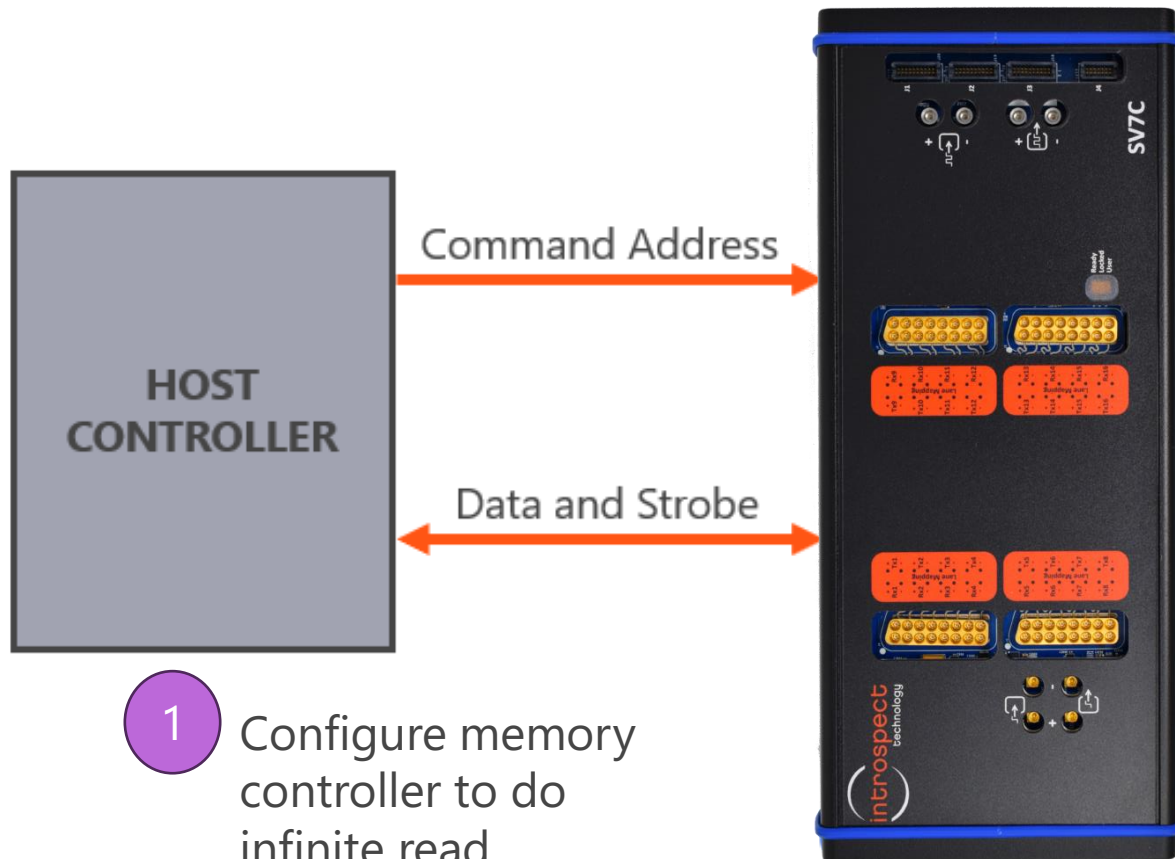
# Receiver Characterization

- Requires support of checking errors within the memory controller component
- Measure horizontal and vertical eye opening at receiver while applying different stressors
  - Jitter sensitivity – clock to data skew
  - Voltage sensitivity
  - Stressed eye





# Receiver Characterization Using BIST



- 1 Configure memory controller to do infinite read operation with BIST checker

- 2 Pre-program read responses and read timings inside SV7C-17 and send the data with impairments

- Two-step receiver characterization based on memory controller BIST mode
- In the first step, the BIST is enabled to perform infinite write/read operations or infinite read operations
- In the second step, the SV7C is programmed in Python to prepare the read response data and then transmit it





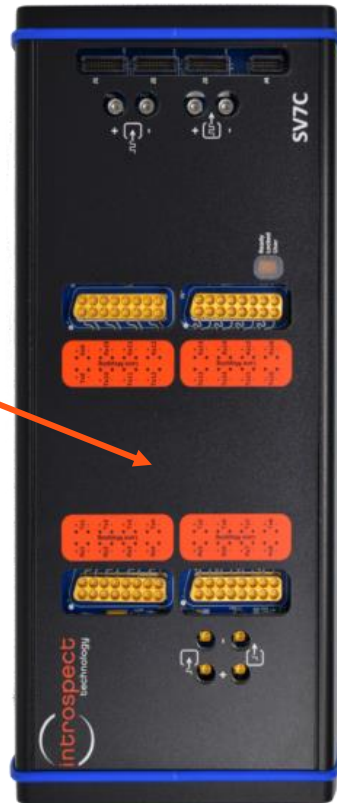
# Protocol Analyzer + Interposer Systems



# DDR5/LPDDR5 Protocol Analyzer

## ANALYZER INSTRUMENT

Same exerciser hardware but with a license for protocol analyzer



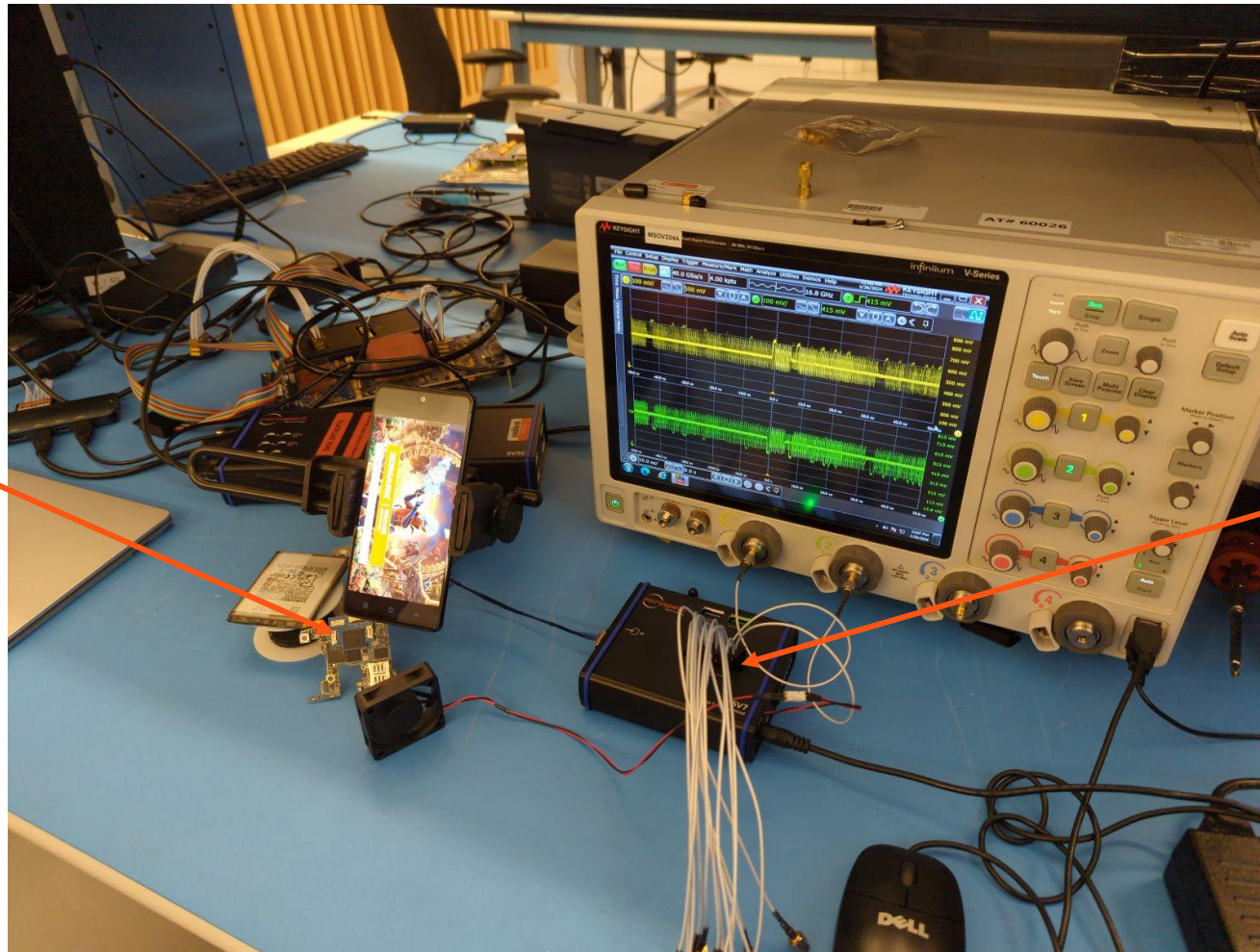
Remote Sampling Heads (Up to 32 Channels)



# DDR5/LPDDR5 Protocol Analyzer

## INTERPOSER SYSTEM

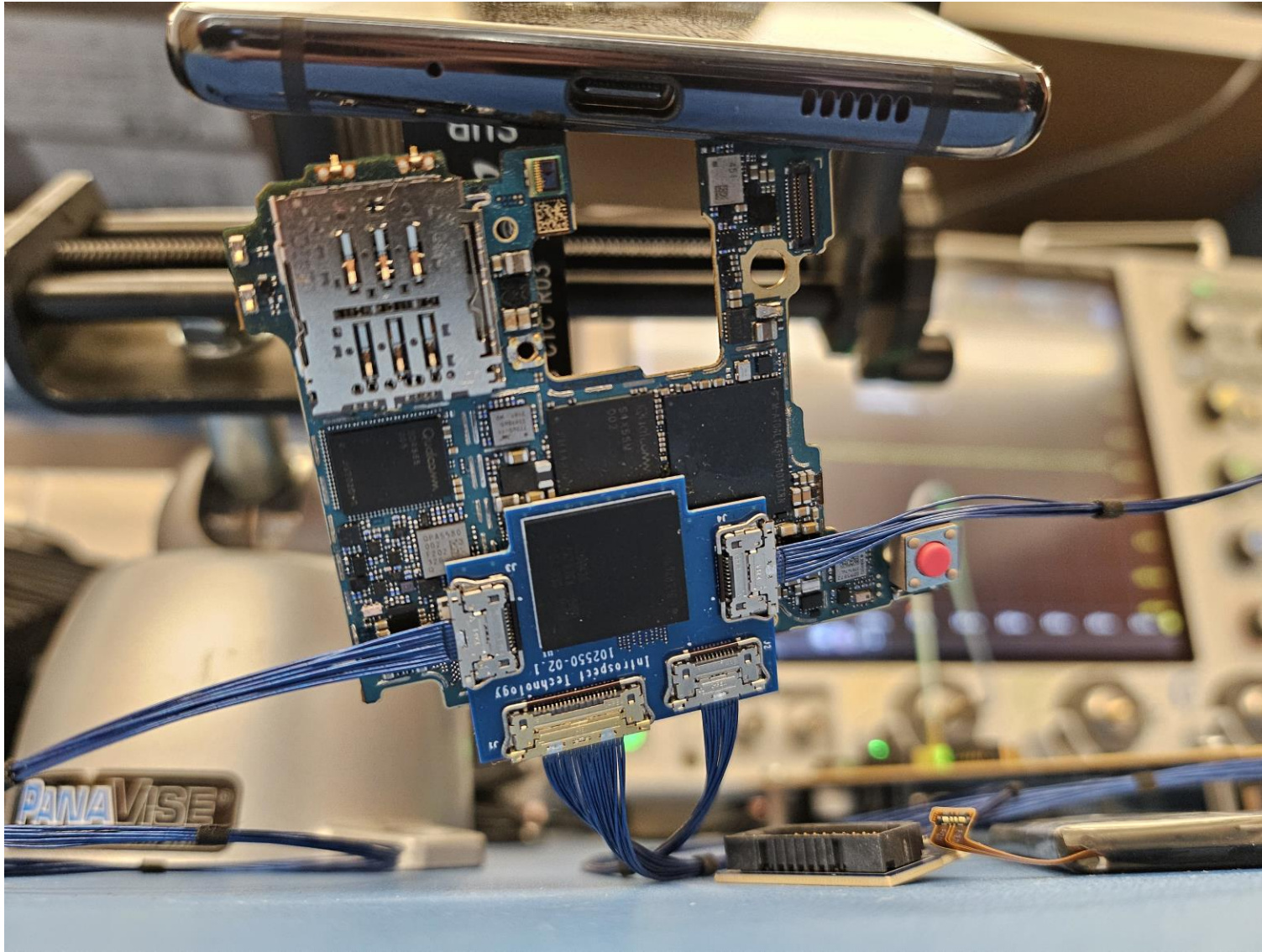
Interposer system  
(example shown is  
PoP LPDDR5)



Remote Sampling  
Head (multi-channel  
active probe)  
connected to  
oscilloscope

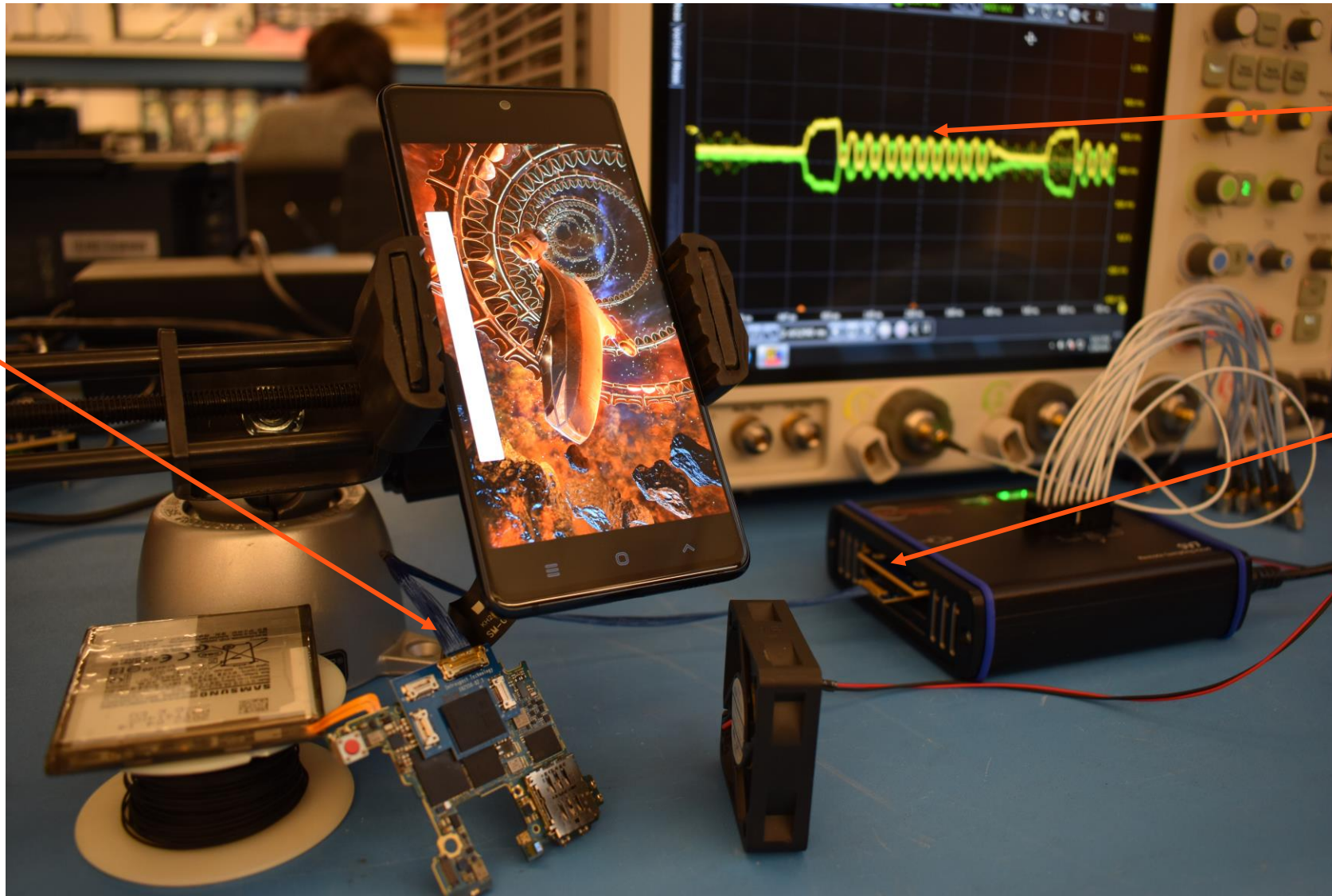


# Zoomed View With All Pins Probed



# Probed Waveform Visible on Scope

Controlled  
Impedance  
Cable

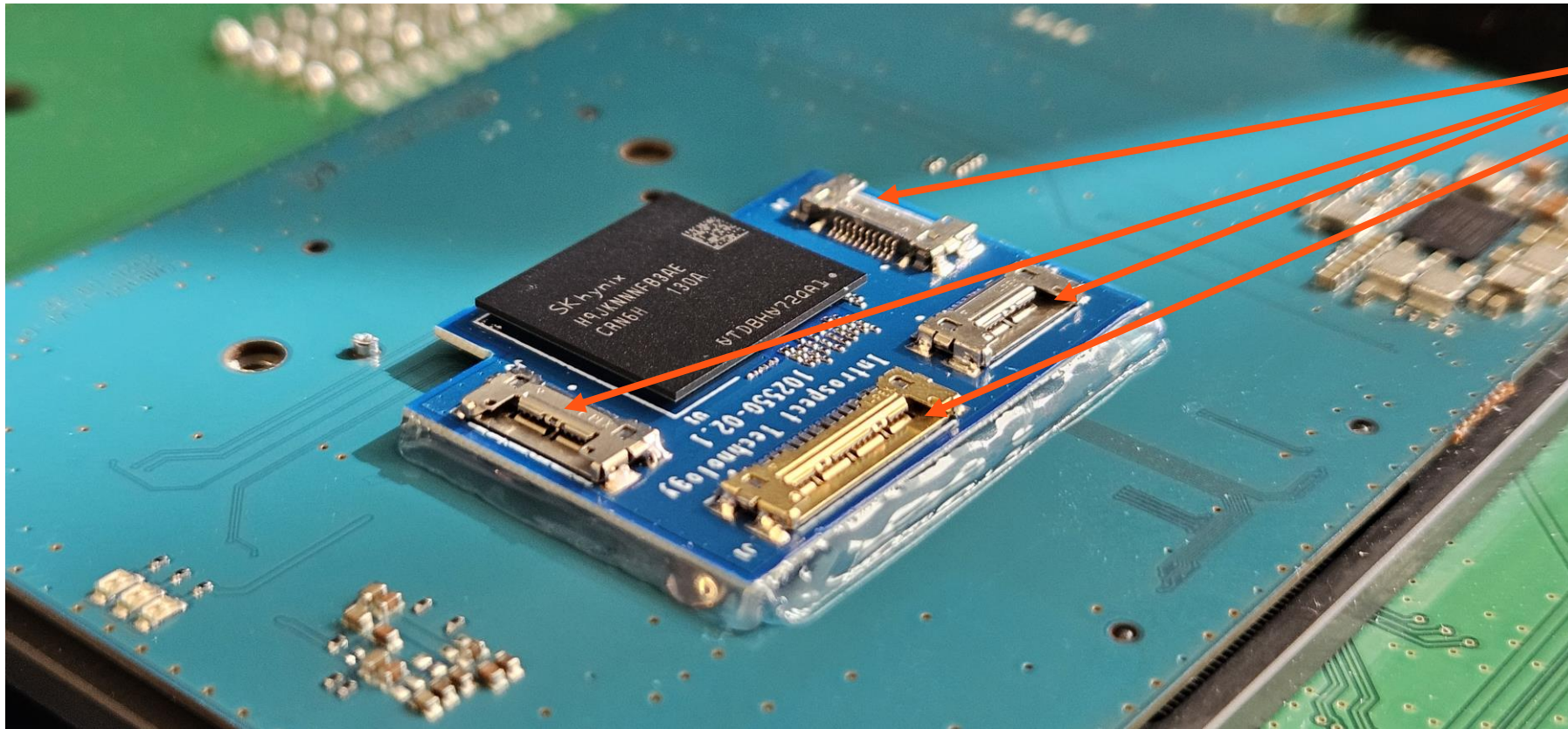


Real Device  
Waveforms

RSH Interface



# Example Evaluation Board



Tip  
attachment  
cable  
connectors

# DDR5/LPDDR5 Protocol Analyzer

## FULL PROTOCOL ANALYSIS

DDR Data Capture Viewer: ddr\_test / ddr\_test\_target

4 Bursts, 38 Commands, 11 Timing Violations

Bursts | Commands

Open Result Folder

Command Selection

Go To:

Command #2: Details

Name: ACTIVATE-2  
 Bits: 

Argument	Value
R	12

CS CA23456  
 11 1100000  
 11 0101000

Command #2: Timings [View Timing Definitions](#)

Ref Command	Value	Min	Max	Unit	Symbol
0 (ACT1)	20.0	-	8.0	nCK	tAAD
4 (ACT2)	48.0	10.0	-	nCK	tRRD
5 (RD)	74.0	36.0	-	nCK	tRCD
6 (PRE)	80.0	84.0	1.40e+05	nCK	tRAS

**Help: Timing Definitions**

The timings are computed between the selected command and the reference commands as shown in the example below

Timing between commands

Command#	TimeStamp	Burst#	Name
0	8		ACT1
1	16		MRR
2	88		ACT2
3	256		ACT1

Command #0: Timings [View Timing Definitions](#)

Ref Command	Value	Min	Max	Unit	Symbol
2 (ACT2)	20.0	-	8.0	nCK	tAAD

Symbol	Description
tAAD	Delay between ACT2 and ACT1
tRRD	Delay between ACT2 and ACT2 (diff. bank)
tRCD	Delay between ACT2 and any read/write/maskedWrite command
tRAS	Delay between ACT2 and PRE (same bank)

30	2232	MRW1
31	2320	MRW2
32	2408	MRW1
33	2576	MRW1
34	2664	MRW2

Bursts Tab: To zoom in, left-click on the plot. Minimum window-length is 8 bits. [Next Tip](#)





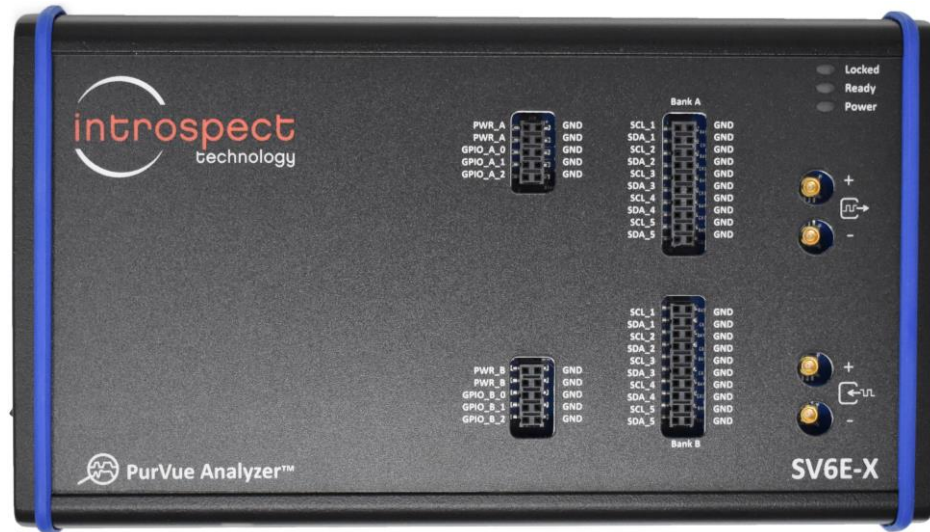
# SV6E-X SidebandBus Controller and Tester



## E SERIES

# SV6E-X – I3C License

Mid-Frequency Digital Test Module



## OVERVIEW

Multi-Purpose Protocol Exerciser, Protocol analyzer, real-time oscilloscope  
Can be licensed for different mid-frequency digital protocols (I3C shown here)

62

## FEATURES

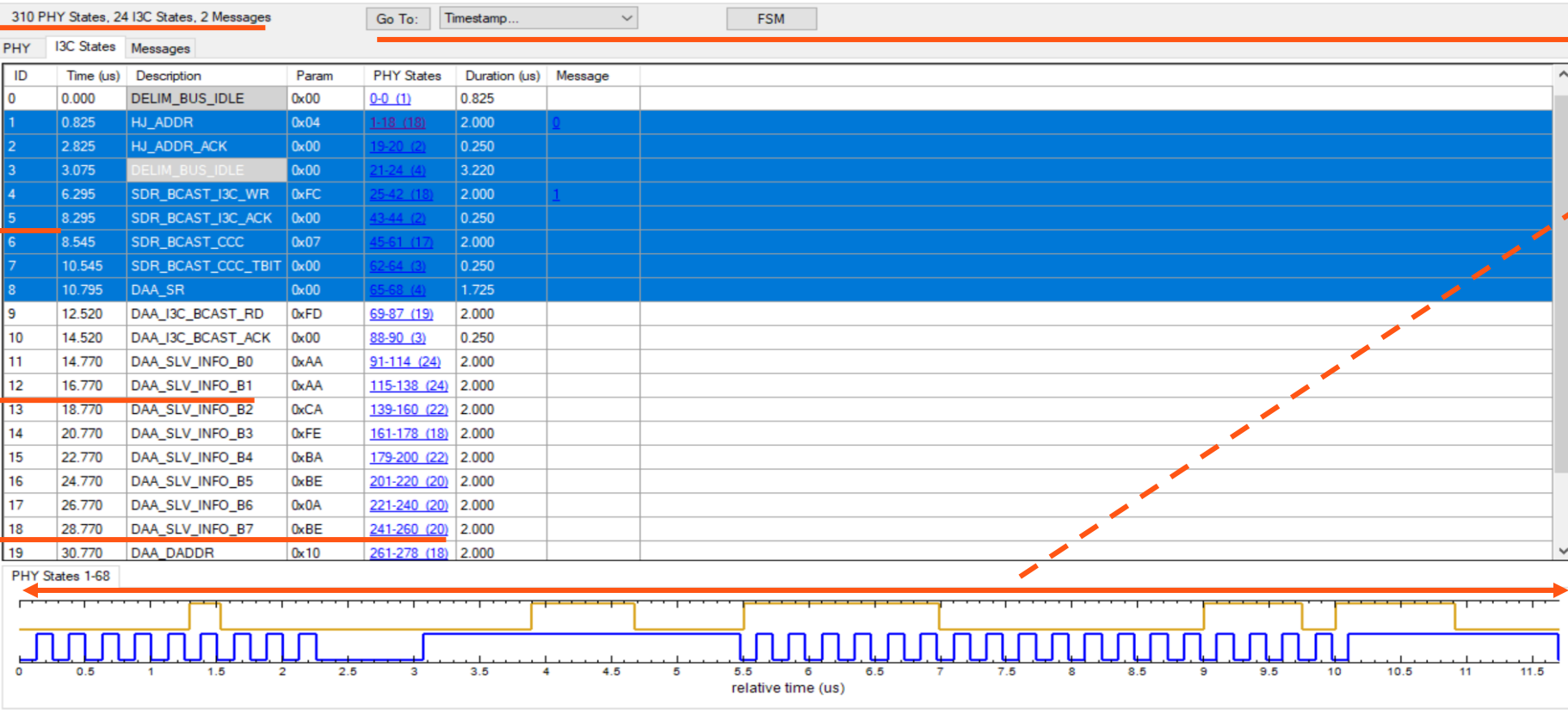
- Two banks of 10 channels each
- 200 MHz operating frequency
- Two programmable, high-current power supplies
- PurVue Analyzer™ technology on any channel

## BENEFITS

- Supports I3C, I3C Basic, JESD403, RCD, PMIC, SPD Hub, DMTF, MCTP
- Enables high-performance testing compared to other solutions
- Replaces racks of bench equipment or PXI test systems

# Detailed Protocol Analyzer View

Capture summary



Powerful search

Precision time stamps

Human-readable event lists

Hyperlinks to toggle views

Dynamically adjust timing diagram span based on selected events

# Powerful Scripting and Logging

Hot Join Example

Interrupt Example

Mastership Request Example

```
File Edit IESP/MIPI_I3C_EXERCISER Wizards ControlPanels Tools Results Help
Params Log Results
Resetting all dynamic addresses on bus...
Assigning dynamic addresses...
No dynamic addresses were assigned during DAA
Master DAA Table (0 entries)

SUCCESS: Offline slave device successfully joined the bus...
8
Reading Bus Characteristics Register for slvAddr=8...
Bus 0 Master : Read BCR as [6] from slvAddr 8

SUCCESS: IBI request was acknowledged by the master...
IBI request was not acknowledged by the master...
IBI request was not acknowledged by the master...
IBI request was not acknowledged by the master...
IBI request was not acknowledged by the master...
IBI request was not acknowledged by the master...
IBI request was not acknowledged by the master...
IBI request was not acknowledged by the master...
IBI request was not acknowledged by the master...
IBI request was not acknowledged by the master...

Querying master for list of interrupts...
Received the following interrupts...
[{'interruptType': 'HJ', 'masterAked': True, 'masterDisabledFuture': False},
{'interruptType': 'IBI', 'slaveAddr': 8, 'masterAked': True, 'masterDisabledFuture':
True, 'hasIbiPayload': False, 'ibiPayloadSize': 0, 'ibiPayload': None}]
SUCCESS: Slave was able to become master...
DAA Table (1 entries)
slvAddr=8, {'hasStaticAddr': False, 'provId': 1, 'bcr': 6, 'dcr': 0}

Test finished
```

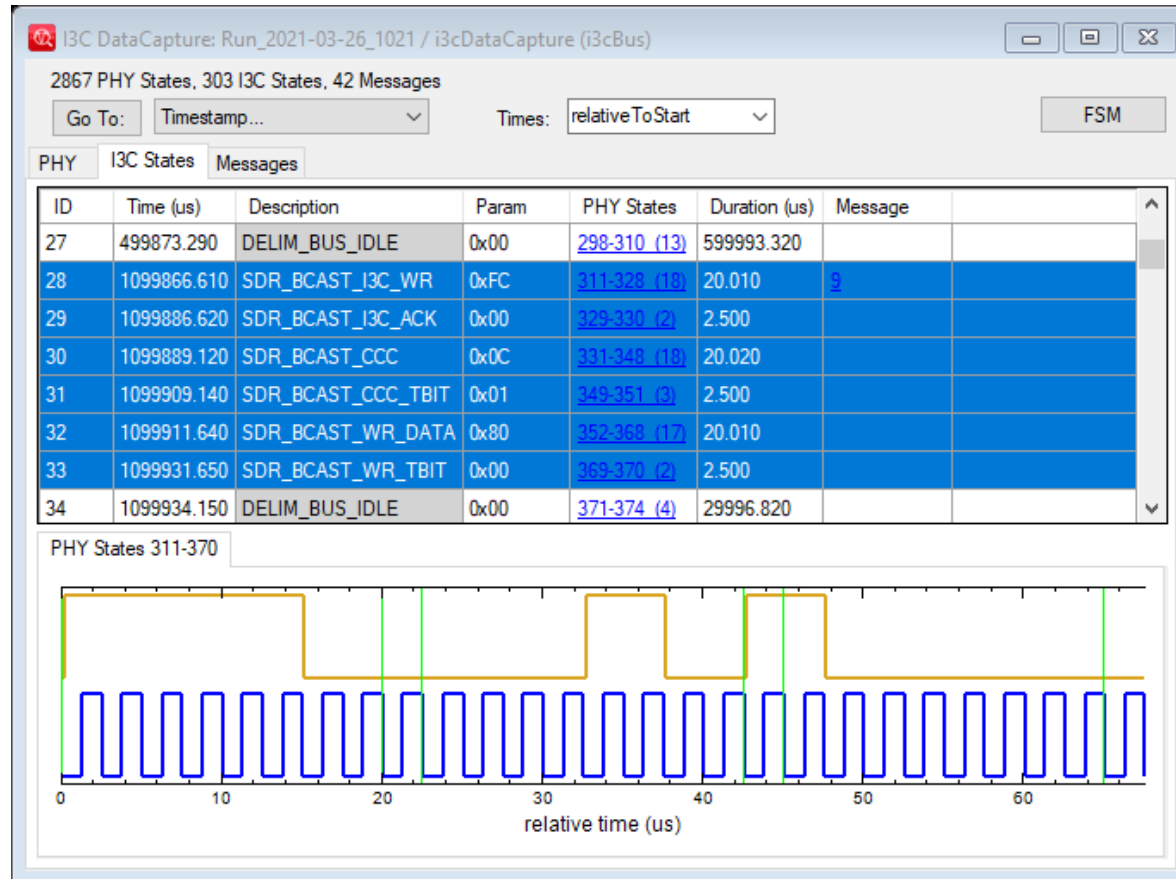


# SidebandBus Controller Component

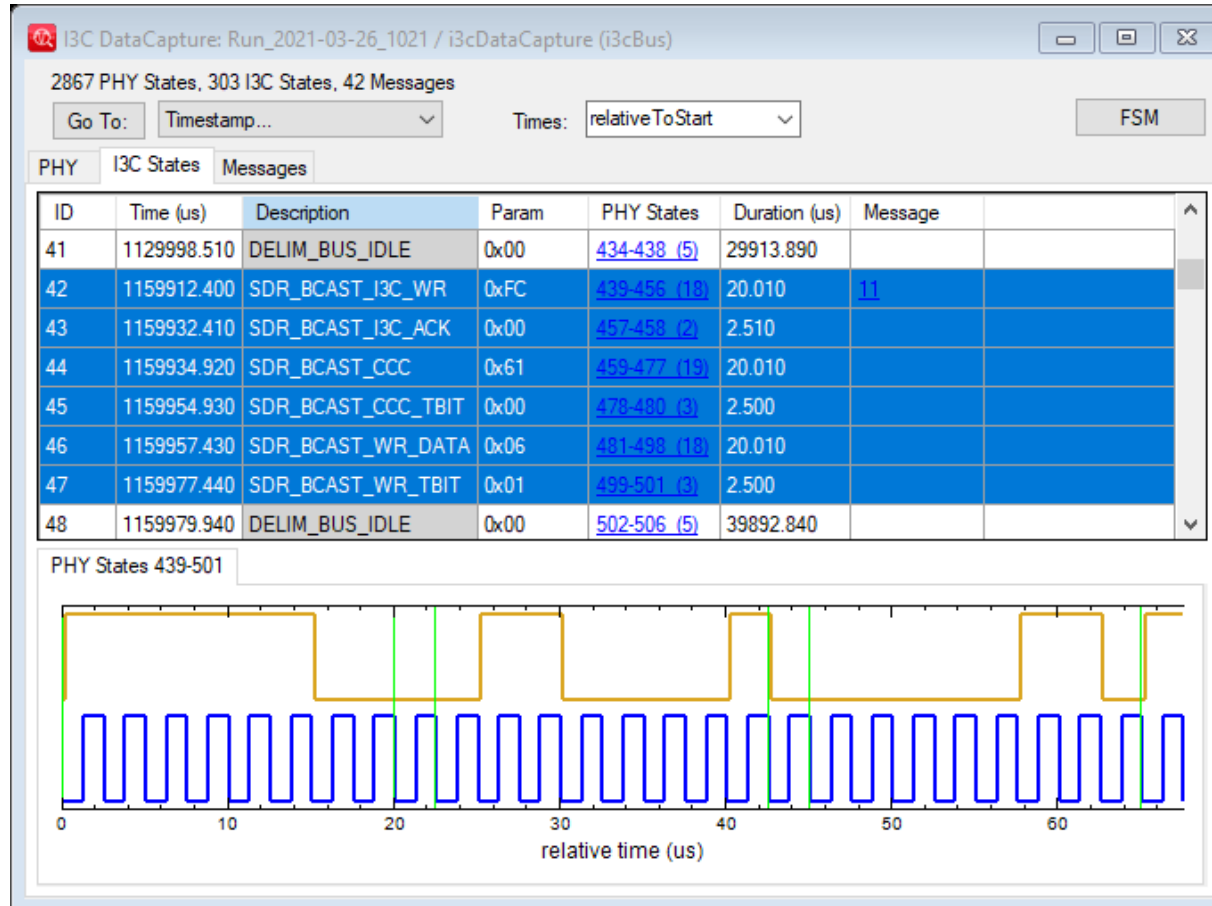
Components	sidebandBusController1 properties (class: SidebandBusController)														
i3cBus i3cDataCapture <b>i3cProtocol</b> jedecSlaveDevice1 jedecSlaveDevice2 jedecSlaveDevice3 jedecSlaveParameters1 jedecSlaveParameters2 jedecSlaveParameters3 masterParams1 sidebandBusController1	<table border="1"><tr><td>startupState</td><td>master</td></tr><tr><td>hid</td><td>3</td></tr><tr><td>autoInitBus</td><td>True</td></tr><tr><td>pecEnabled</td><td>False</td></tr><tr><td>masterModeParams</td><td>masterParams1</td></tr><tr><td>bus</td><td>i3cBus</td></tr><tr><td>codeForConfig</td><td># Define some names for the slave addresses:dtisByName = { 'SPD': 0b1010,</td></tr></table> <p><b>autoInitBus</b> Specifies whether a call to "initializeBus" is part of "setup()" (and "update"). If "autoInitBus" is False, you will need to call "initializeBus()" in the Test procedure after the call to "setup()". If you are using a slave Device component as w...</p>	startupState	master	hid	3	autoInitBus	True	pecEnabled	False	masterModeParams	masterParams1	bus	i3cBus	codeForConfig	# Define some names for the slave addresses:dtisByName = { 'SPD': 0b1010,
startupState	master														
hid	3														
autoInitBus	True														
pecEnabled	False														
masterModeParams	masterParams1														
bus	i3cBus														
codeForConfig	# Define some names for the slave addresses:dtisByName = { 'SPD': 0b1010,														

Add Remove Config

# Example of Set Bus Config



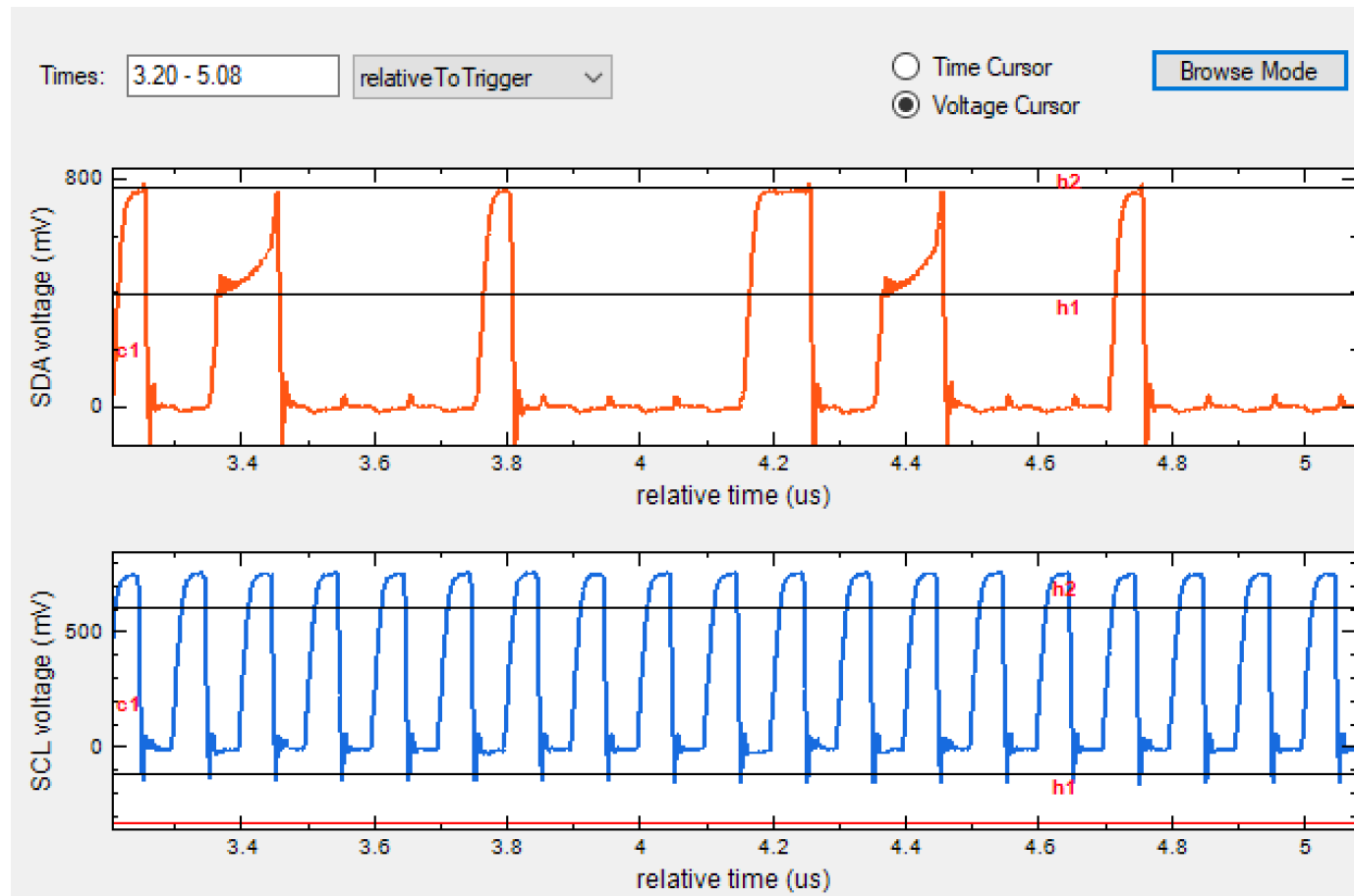
# Example of SETHID





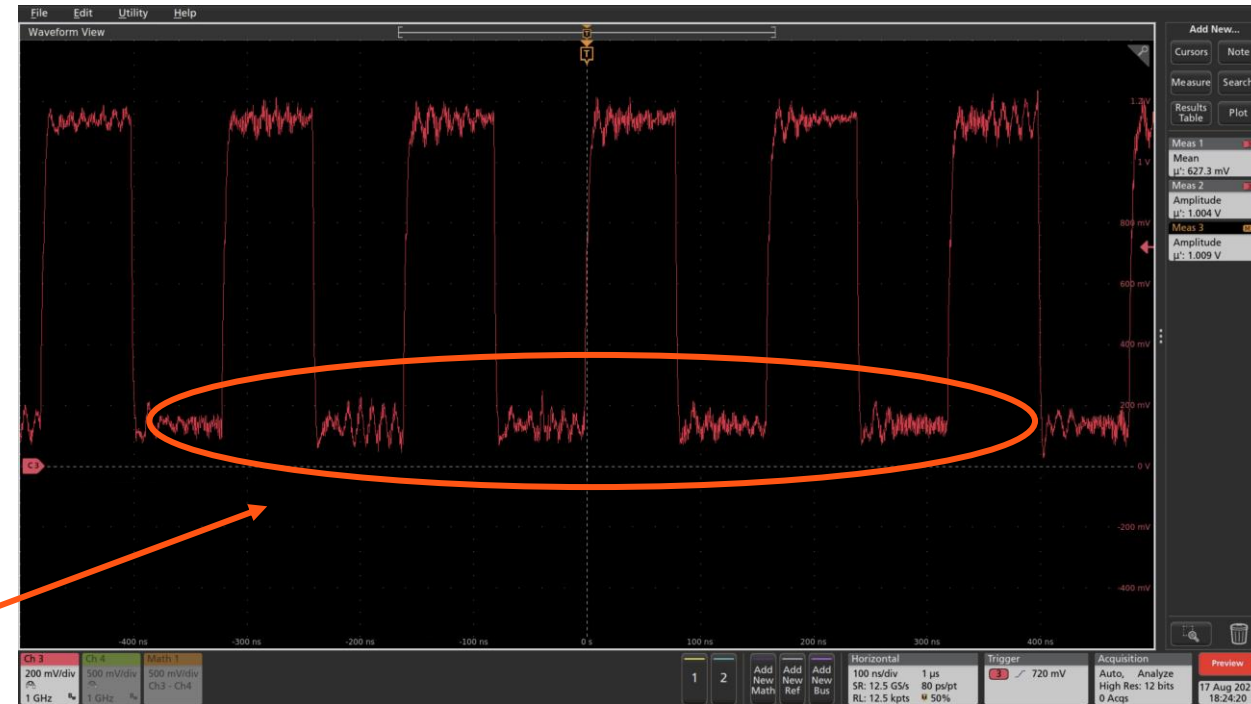
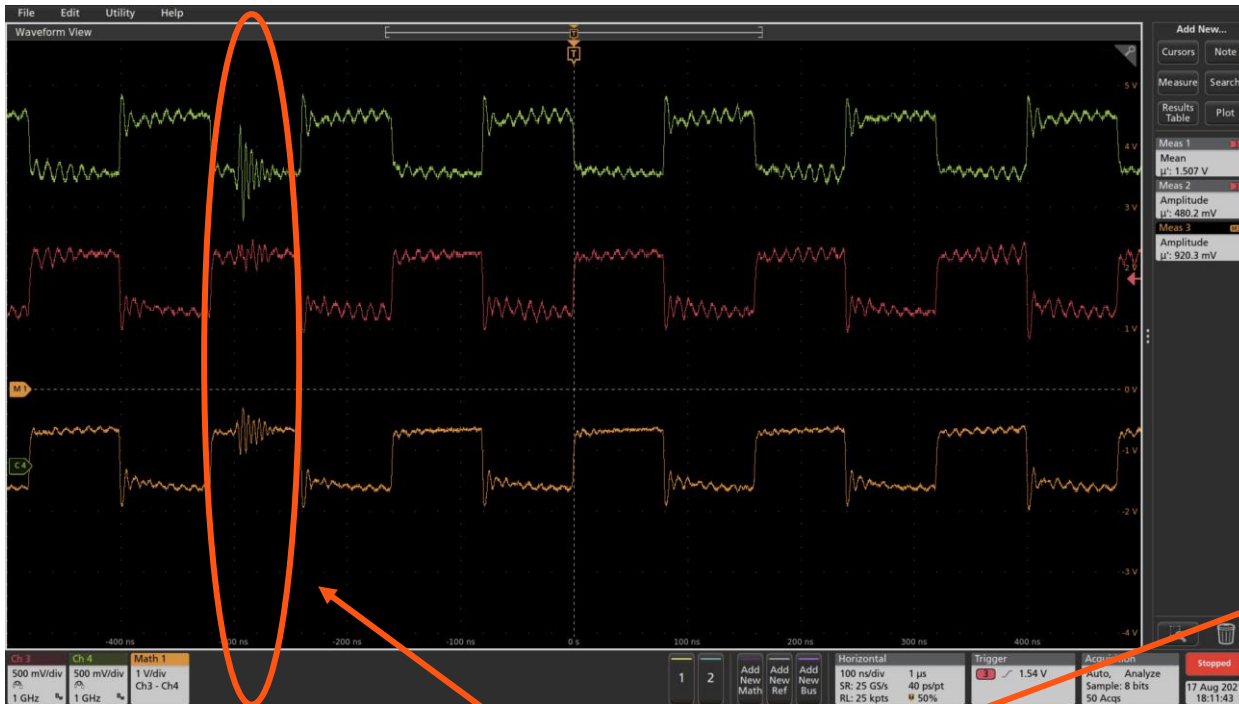
# Real-Time, Single-Shot View of Signals

- I3C PurVue Analyzer™ embedded real-time oscilloscope license provides a real-time, single-shot view of all I3C signals
- It eliminates probing hassles with conventional scopes (attachment issues, cost issues, EMI issues) and completely eliminates the need for using a conventional scope



Pristine signal measurement is shown here where there are no artefacts due to the probes picking up noise from Wi-Fi or RF signals

# Why? Because Probes Are Antennas



69

Most oscilloscope probes result in too much noise pickup  
The SV4E oscilloscope is embedded into the I3C signal plane, so it has far less noise

# PurVue Analyzer™ Capture

- A single component for simultaneous protocol capture (digital capture) and oscilloscope capture (analog capture)
- Familiar user interface (the viewer is almost identical to the I3cDataCapture)
- With the PurVue viewer, users can quickly find interesting parts of the waveform since it is protocol-aware

i3cPurVueCapture1 properties (class: I3cPurVueCapture)	
analogCapturePort	1
digitalCaptureBus	<b>i3cBus</b>
triggerCondition	<b>privateRead</b>
pre TriggerDuration	5000
post TriggerType	numberOfFrames
post TriggerDuration	1
slaveAddrForTrigger	None
saveResults	True

# PurVue Analyzer™ Digital View

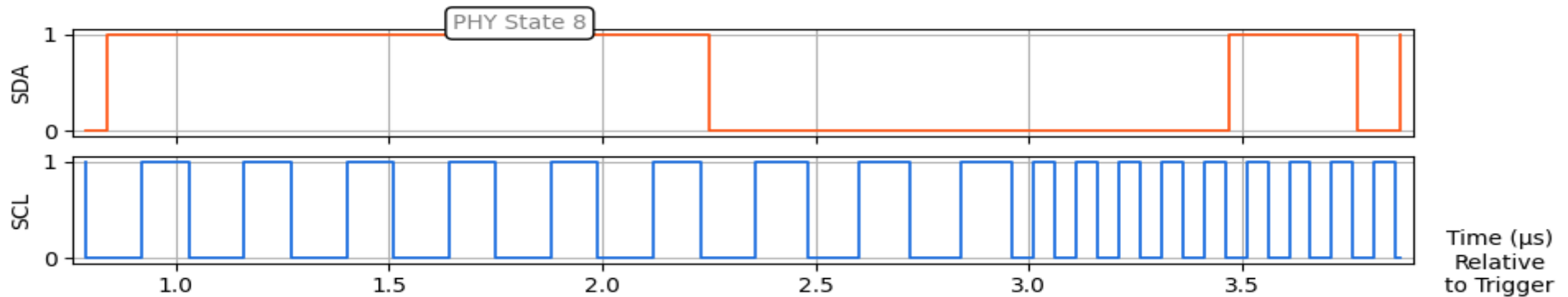
I3C PurVue Capture: Run\_2022-08-08\_1516 / i3cPurVueCapture1

274 PHY States, 21 I3C States, 1 Transactions

Times: RelativeToTrigger

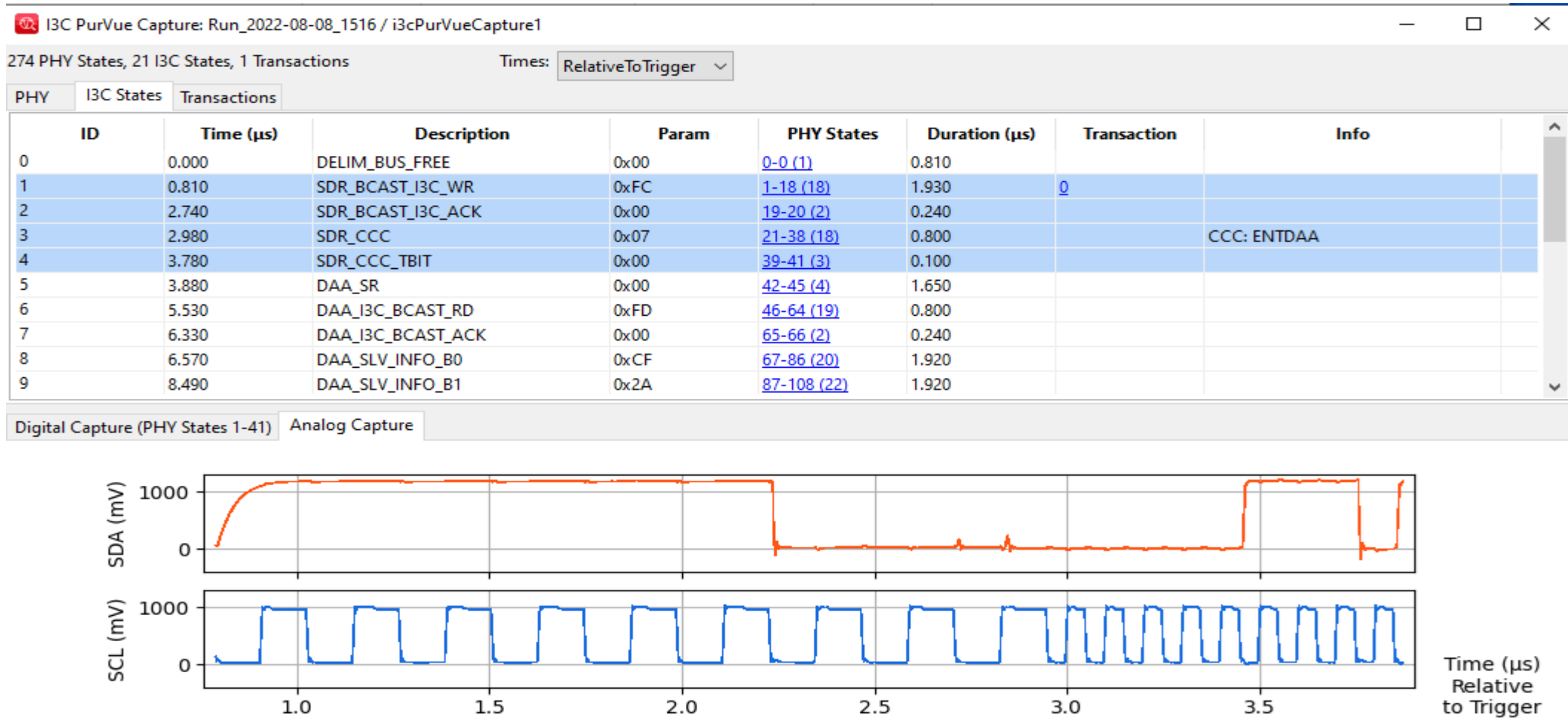
PHY	I3C States	Transactions					
ID	Time (μs)	Description	Param	PHY States	Duration (μs)	Transaction	Info
0	0.000	DELIM_BUS_FREE	0x00	<a href="#">0-0 (1)</a>	0.810		
1	0.810	SDR_BCAST_I3C_WR	0xFC	<a href="#">1-18 (18)</a>	1.930	0	
2	2.740	SDR_BCAST_I3C_ACK	0x00	<a href="#">19-20 (2)</a>	0.240		
3	2.980	SDR_CCC	0x07	<a href="#">21-38 (18)</a>	0.800		CCC: ENTDA A
4	3.780	SDR_CCC_TBIT	0x00	<a href="#">39-41 (3)</a>	0.100		
5	3.880	DAA_SR	0x00	<a href="#">42-45 (4)</a>	1.650		
6	5.530	DAA_I3C_BCAST_RD	0xFD	<a href="#">46-64 (19)</a>	0.800		
7	6.330	DAA_I3C_BCAST_ACK	0x00	<a href="#">65-66 (2)</a>	0.240		
8	6.570	DAA_SLV_INFO_B0	0xCF	<a href="#">67-86 (20)</a>	1.920		
9	8.490	DAA_SLV_INFO_B1	0x2A	<a href="#">87-108 (22)</a>	1.920		

Digital Capture (PHY States 1-41) | Analog Capture





# PurVue Analyzer™ Analog View



Notice how only the shape of the bottom waveform changes from last slide!

# Summary

## HIGHLY DIFFERENTIATED SOLUTIONS

- Introspect develops parallel test instruments for high-speed interfaces
- We have created a rich portfolio of solutions for source-synchronous and DDR interfaces
- Our solutions can be deployed at the component level and at the module level