PGY I3C-LITE EX-PD

USER'S MANUAL Ver1.1

PRODIGY TECHNOVATIONS PVT LTD
BENGALURU, INDIA | CONTACT@PRODIGYTECHNO.COM

Contents

Table

Chapter 1
Product Overview

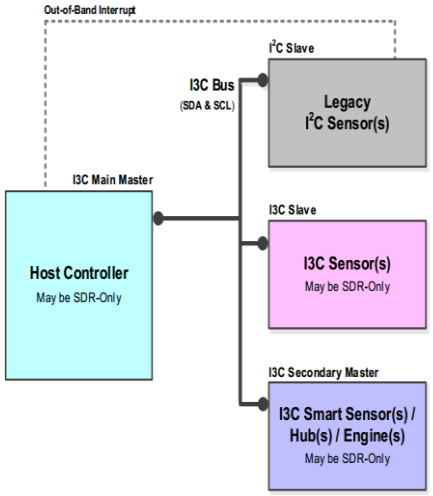
Introduction to I3C

MIPI I3C interface has been developed to ease sensor system design architectures in mobile wireless products by providing a fast, low cost, low power, two-wire digital interface for sensors.

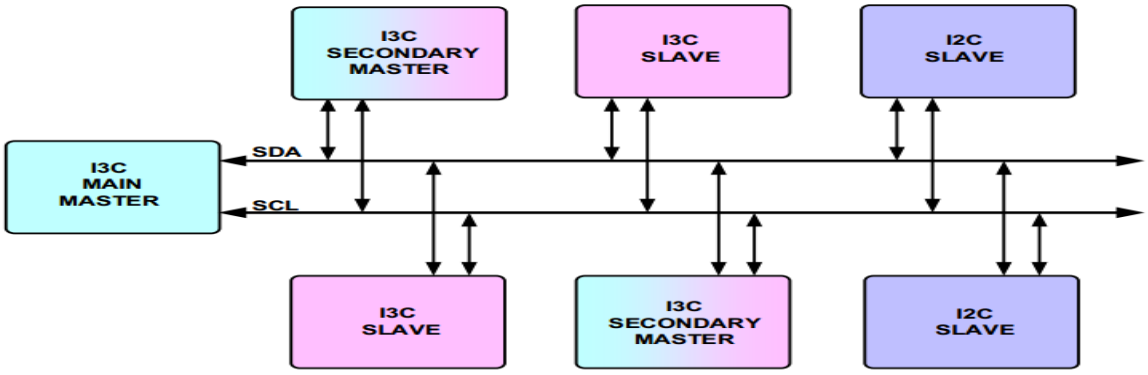
The I3C interface is intended to improve upon the features of the I2C interface, preserving backward compatibility. This Specification defines a standard Multi-Drop interface between Host processors and peripheral sensors.

I3C interface features:

- Two wire serial interface up to 12.5 MHz using Push-Pull
- Legacy I2C Device co-existence on the same Bus (with some limitations)
- Dynamic Addressing while supporting Static Addressing for Legacy I2C Devices
- Legacy I2C messaging
- I2C-like Single Data Rate messaging (SDR)
- Optional High Data Rate Messaging Modes (HDR)
- Multi-Drop capability
- Multi-Master capability
- IBI and Hot-Join support
- Synchronous Timing Support and Asynchronous Time Stamping



The I3C interface provides major efficiencies in Bus power while providing greater than 10x speed improvements over I2C. 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.



Introduction to PGY I3C EX-PD Lite

PGY-I3C-EX-PD Lite is the leading instrument that act as Analyzer, Exerciser and Test Device, enables the design and test engineers to test the I3C/I2C designs for its specifications by configuring device as Master or Slave or both.

Figure 1:PGY-I3C-Lite EX-PD

The device allows to

- Monitor the I3C/I2C bus traffic in real time.
- Allows to view the waveform plot and capture for long duration.
- Exerciser allows to generate custom defined bus traffic, configure response in both Master and Slave.
- Device will be connected to the PC through USB 3.0 and can be connected to any Master/Slave devices

Hardware Features

Protocol	i) I2C ii) I3C (SDR, HDR – DDR, TSP & TSL) iii) IBI, Hotjoin, MasterShip Request
PC Connection	USB 3.0
Device Capabilities	i) I3C Main Master ii) I3C Secondary Master iii) I3C Slave iv) I2C Slave
Exerciser Device Configuration	Configurable 1 Main Master+ 1 Slave (OR) Slave with Secondary master
SCL Duty Cycle Variation	User Defined, 100kHz to 12.5MHz
Voltage	1V,1.2V ,1.8V ,2.5V or 3.3V
SCL and SDA Delay	User Defined
Frame Delay	User Defined

Software Features

Multi Domain View	i) Trigger View ii) Protocol Listing iii) Plot View iv) Exerciser View
Protocol Listing	i) Frame Listing ii) Packet Details with Error Detection and Frequency calculation iii) Markers
Capture Duration	Long Capture
Waveform Plot	i) Digital waveform plot ii) Bus diagram, Annotation & Bit overlay representation for Decoded Protocol iii) Pan, Zoom, Cursor & Save image of waveform
Exerciser	i) GUI based for easy to use ii) Script for deep level testing
Others	i) Search on Packet or Frame Level ii) Report with Protocol Listing, Waveform Images

Chapter 2
Installation

Hardware Setup

The Setup comes with

i) PGY I3C Lite EX-PD Device
ii) USB 3.0 Port
iii) Flying lead probe with female header pin

Figure 2: Front panel image

LED 1	**FPGA DONE STATUS**
LED 2	BOARD POWER GOOD STATUS
LED 4	MASTER LED. LED Lights ON when Exerciser is configured as MASTER
LED 5	SLAVE LED. LED Lights ON when Exerciser is configured as SLAVE
LED 6	HBT. Starts blinking once the Analyzer is booted and ready for acquisition.

Step 1: USB Connection

Connect USB 3.0 Type B to the USB3 Connector in PGY-I3C Lite EX-PD Lite unit and USB 3 Type A to the Host Computer.

Note: Unit gets power by connecting USB.

Figure 3 USB 3 Connector

Figure 4 USB Connected to Host PC

Step 2: Probe Connection

The colored tip is for Clock & Data Signal, it will be marked as SCL & SDA respectively. The black tip is for Ground, it will be marked as GND.

Connect those line with external device respectively.

Figure 5 PROBE CABLE

Software Pre-Requirement

Operating System	Windows 7/8/8.1/10
Memory (RAM)	16 GB RAM (Minimum of 8GB Required)
Hard Drive	50 GB
CPU	Intel I3 processor or higher
Interface	USB3.0
.NET Framework	4.5

If .Net Framework is not available in the System, it can be download from below link

https://www.microsoft.com/en-in/download/details.aspx?id=30653

Software Installation

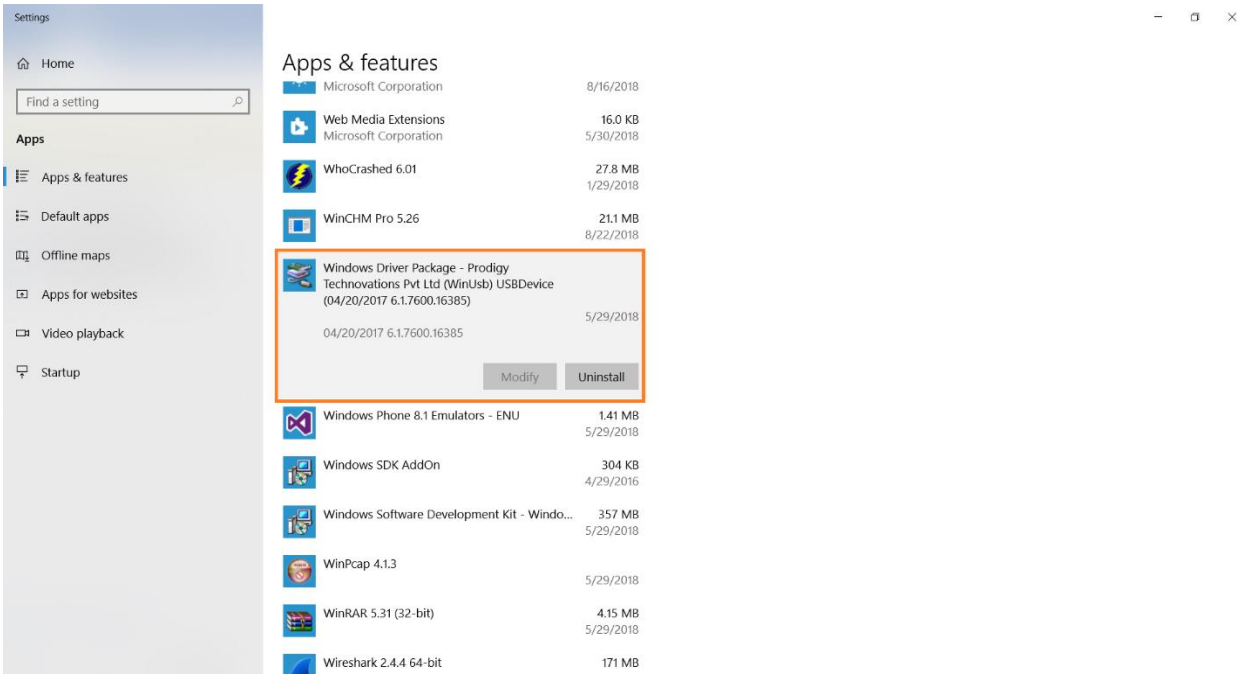
The user will be provided with Installation Software file named ***PGY-I3C Lite Protocol Analyzer and Exerciser.exe*** either through CD or through Internet File Storage (like Google Drive, DropBox).

Double click on the installation file.

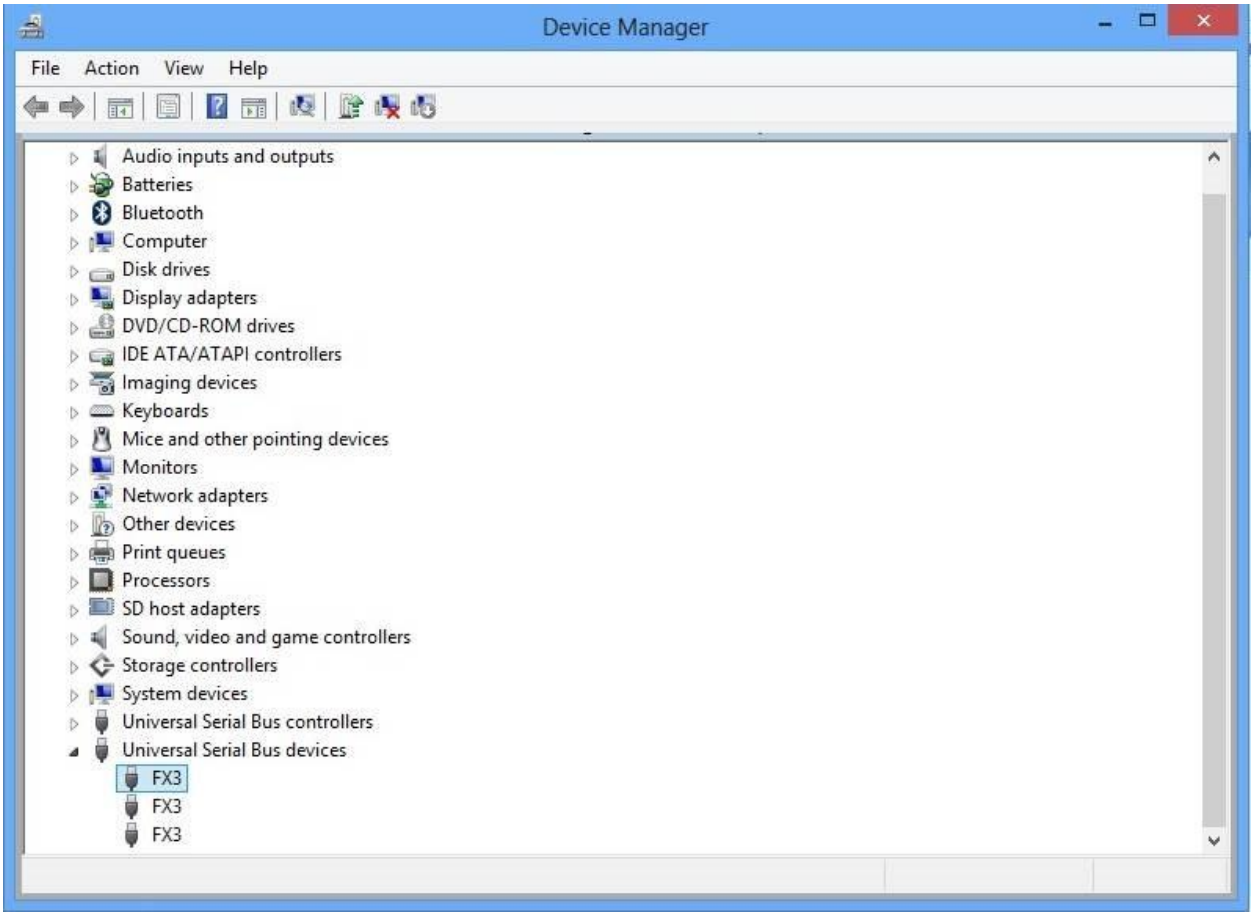
The *Software Installation* screen will appear as shown below and Click Next Button to continue further

Driver Installation

WinUSB driver is required to use PGY-I3C Lite EX-PD over USB3.0 interface. Prodigy WinUSB drivers are automatically installed during installation of PGY-I3C Lite EX-PD software.



On successful installation, FX3 needs to be displayed in Device Manager under USB Devices



Chapter 3

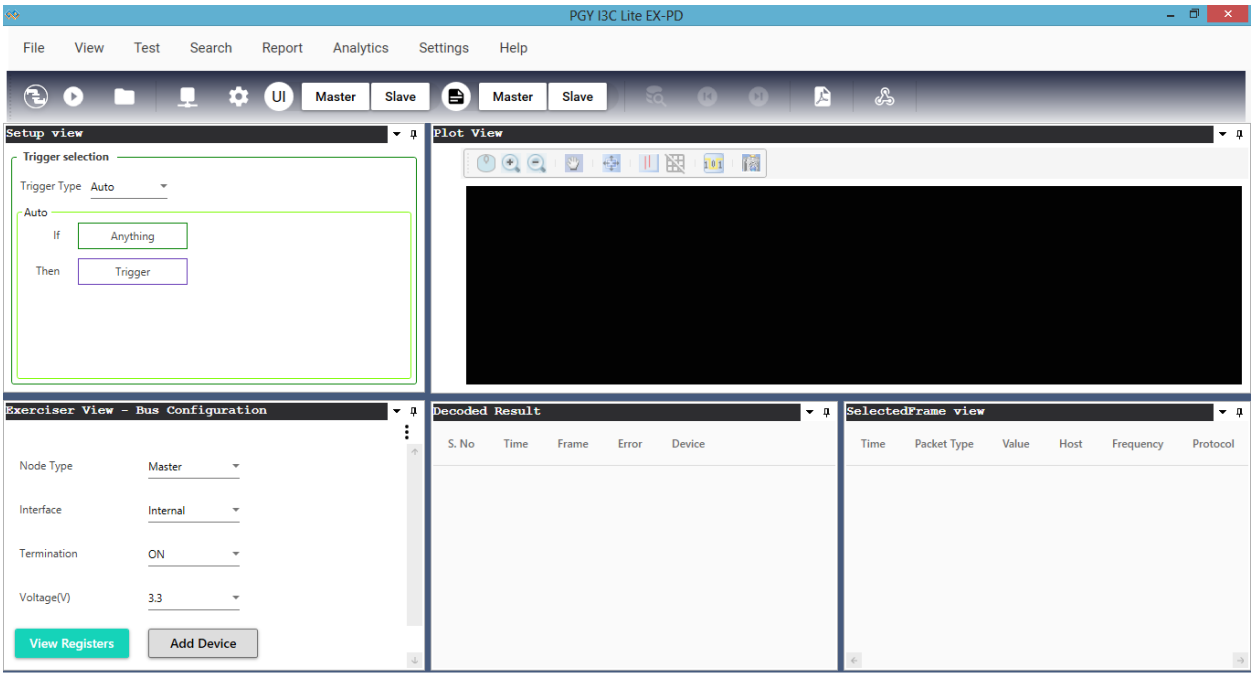
Getting started with Analyzer

Overview

This chapter will quickly introduce you to the basic usage of the PGY-I3C Lite EX-PD Software for generating I3C bus traffic and analysis via the PGY I3C Lite EX-PD. In this chapter we will be covering the following modules

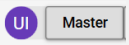
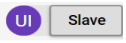
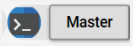
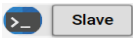
- Establish Connection with Hardware
- Start & Stop Capture
- Protocol View
- Trigger Setup
- Plot Handling
- Report Generation
- Search
- Other Features

Application Map



- Menu Bar
- Tool Bar
- Trigger Setup View
    1. Auto
    2. Simple
- Exerciser View
    1. Bus Configuration
    2. Master UI Command
    3. Master Script Command
    4. Slave UI Command
    5. Slave Script Command
- Plot View
- Protocol view
    1. Frame View
    2. Packet View
    3. Packet Detail View

Tool Bar

Icon	Description	Comments
	Connection Establishment	Allows to Establish connection with hardware
	Start Capture	Start captures the signal and decode packets based on protocol
	Stop Capture	Stop captures the signal.
	Load Trace File	Allows to view offline Trace File decoded result
	View I3C Network	Diagrammatic representation of I3C Network shown with nodes connected on the I3C bus.
	Exerciser - Device Config	Allows to configure the node.
	Exerciser - Master UI Command	UI based Master Command Window
	Exerciser - Slave UI Command	UI based Slave Command Window
	Exerciser - Master Script Command	Script based Master Command Window
	Exerciser - Slave Script Command	Script based Slave Command Window
	Search - Simple	Allows simple search based on Packets – Address/ Data/ Command
	Search - Go to Next Frame	Go to the Next matched search Frame
	Search - Go to Prev Frame	Go to the Previous matched search Frame
	Search - Reset	Reset the search criteria.
	Report	Allows to generate quick PDF report
	State Machine Diagram	Shows the state machine diagram for the selected frame.
	Marker	Shows the Marker information.

Menu Bar

Menu bar has the following menu options:

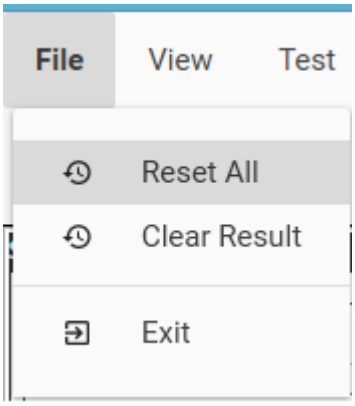
Menu Item	Sub Menu	Description
File	Reset All	Clears Protocol Results from the Analyzer and Removes all node from the Configured I3C Network
	Clear Result	Clears Protocol Results from Analyzer
	Exit	Close the Application
View	Device Configuration	Allows to configure the node.
	Master Script	Script based Master Command Window
	Slave Script	Script based Slave Command Window
	Master UI	UI based Master Command Window
	Slave UI	UI based Slave Command Window
Search	Simple	Allows simple search based on Packets – Address/ Data/ Command
	Advanced	Allows advanced search on Frame with combination of Packets
	Reset	Reset the Search Criteria
Report	Quick Report	Allows to generate quick report in PDF or csv format
	Advanced	Allows to generate quick CSV, PDF with Test Detail with image attachment of waveform.
Analytics	Analytic view	Gives information about Frame, packet and error count and gives information when dynamic address of that particular slave got updated.
Settings	Analytics	Allows to set the Software decoding for I3C/I2C
	FIFO	Exposed internal buffers to user so they can view and analyse further.
Help	About	Gives the product information.

This allows the user to access different functionalities of the application

File Menu

File menu option consist of three sub section:

- Reset All
- Clear Result
- Exit



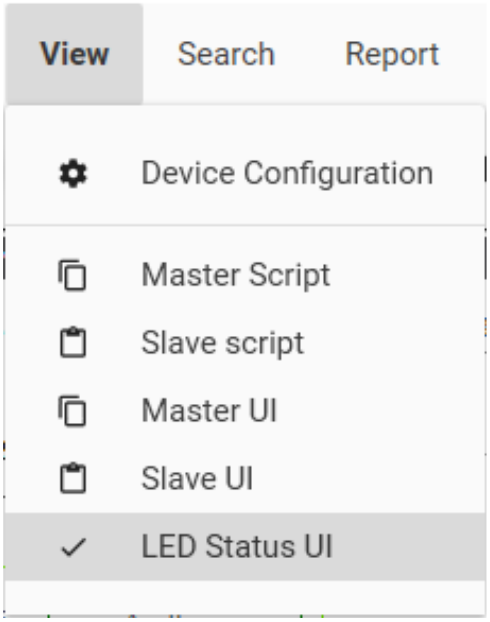
1.Reset All option clears Protocol Results from the Analyzer and Removes all node from the configured I3C Network and does the hardware Reset function.

2.Clear Result option clears Protocol Results from Analyzer

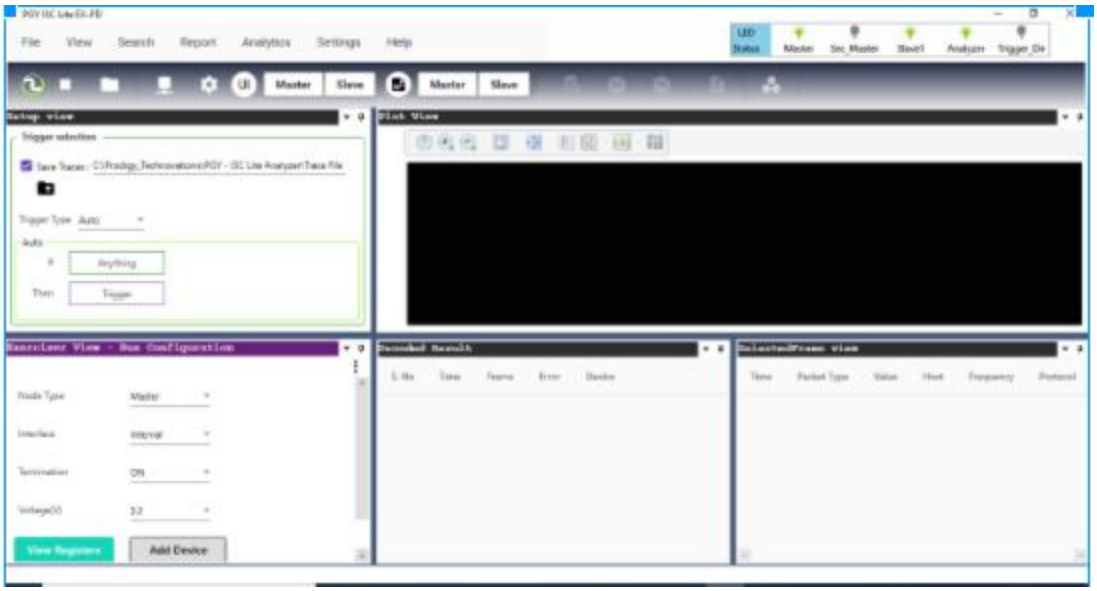
3.Exit option closes the application.

View Menu

View Menu provides different views of Exerciser window.



- Device Configuration option allows to configure the node like I3C Master,I3C/I2C slaves
- Master Script is Script based Master Command window
- Slave Script is Script based Slave Command window
- Master UI is UI based Master Command window
- Slave UI is UI based Slave Command window
- LED Status UI displays the LED Status of the device in GUI as shown below.



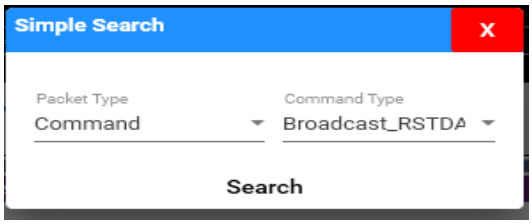
Note: The LED Status indicators will be functional only after the User clicks on Start Capture.
When STOP capture is clicked, the analyser LED will not turn OFF since the hardware STOPS communicating with the software

Search Menu

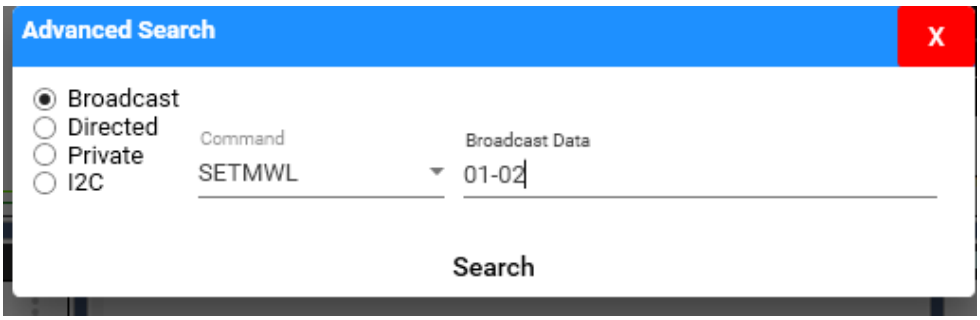
Search menu consists of 3 sub menu:

- Simple
- Advanced
- Reset

Simple search: It allows to search based on Packets – Address/ Data/ Command

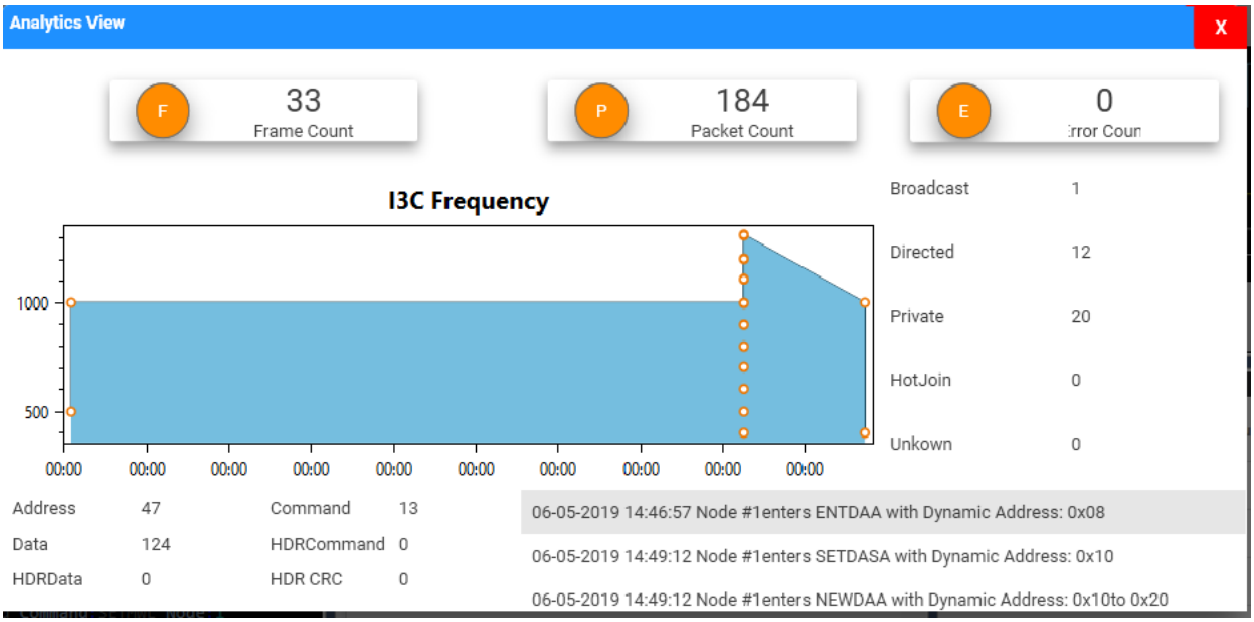


Advance Search :- It allows advanced search on Frame with combination of Packets



Reset : It Resets the Search Criteria.

Analytics Menu



Analytics section basically gives a summarized view of the entire protocol activity captured over a long duration

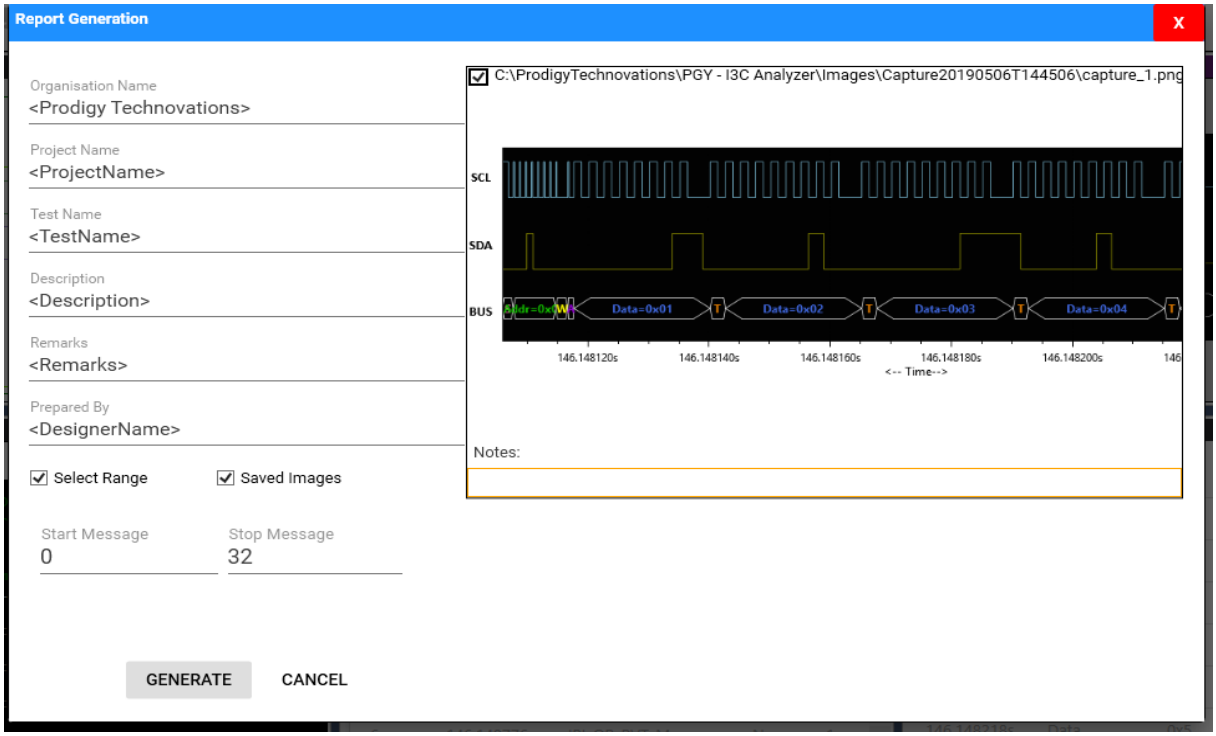
Gives information about

- Frame count
- Packet count
- Error count
- Also gives information when dynamic address of that particular slave got updated.

Report Menu

To generate a report, click Report from ‘Report’ menu. Report generation consists of two sections

- Quick Report : It allows the user to generate a Quick report of the analyzed data in PDF or csv format
- Advanced : It allows to generate the report Allows to quick PDF with Test Detail with image attachment of waveform.



Generation of Report:

1. Select whether any specific range of packets or all the analysed packets needs to be saved.

2.Select the saved images option to attach the Waveform along with generated report.

3. Click on the *Generate Report* button to generate the report.

4. Enables the user to choose the location where the report can be saved in PDF format.

Settings Menu

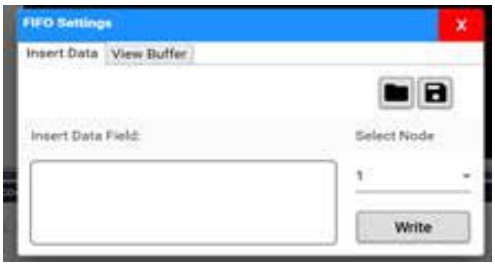
Setting menu has two sub menu.

- Analyzer
- FIFO

1.Analyzer:

By default it will be I3C mode. If you are analysing only I2C traffic on the bus you can select I2C_only. because in I2C 9th bit will be Ack/Nack but in I3C 9th bit will be parity bit so analyser will show error for only I2C packet.

2.FIFO Settings:



FIFO setting has two division

- Insert data:

  Data can be inserted to the buffers and can call data one by one with buffer access command (refer user manual buffer access section) call BUFFERACCESS.txt for demo script.

- View Buffer :

  Exposed internal buffers to user so they can view and analyse further.

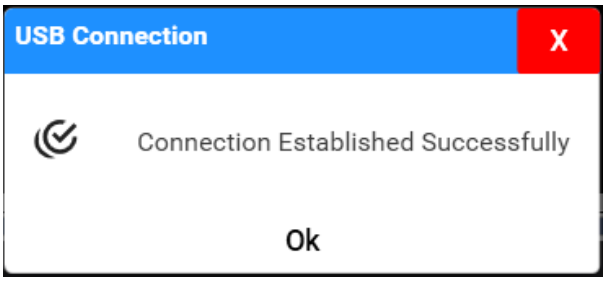
Help Menu

- About

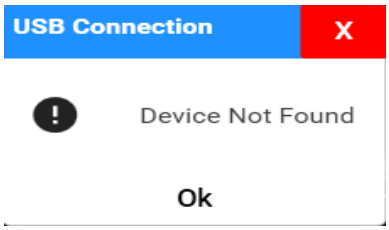
1.About option gives information about the software and hardware version of the product.

Establishing Connection

Click on  *Connection Establishment* icon from Tool Bar, this confirms Software has detected the presence of Hardware. On Successful Connection, the following Popup window will appear on the screen.



On the failure case, the following Popup Window will appear on the screen with the message "*Device Not found*".

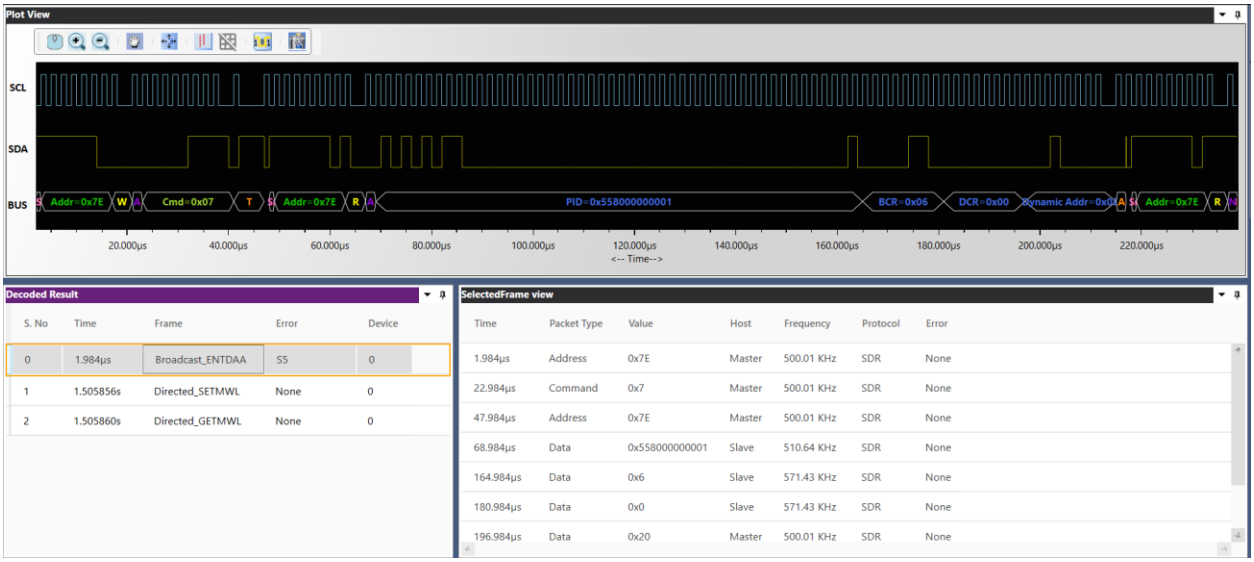


This could be because of one of following reason,

- Make sure the USB cable is connected properly.
- Make sure the Computer is installed with USBPCap driver.
- Device is Power On.
- USB is properly recognized in Device Manager

Start Capture

Once the connection is established with Hardware. User can click on Start Button to Start Capture signal in the Analyzer. Select the ▶ *Start Capture* icon in the Tool Bar of the application. Once it is capturing proper signal, the software will process the signal, decode into protocol result and produce to the user as shown below.



Note: Captured Signal will be stored in the system memory on the folder *C:\Prodigy_Technovations\PGY - I3C Analyzer\Trace File*. File name will be in format of Date Time of when the capture was started E.g. *Capture20180809T160028.*

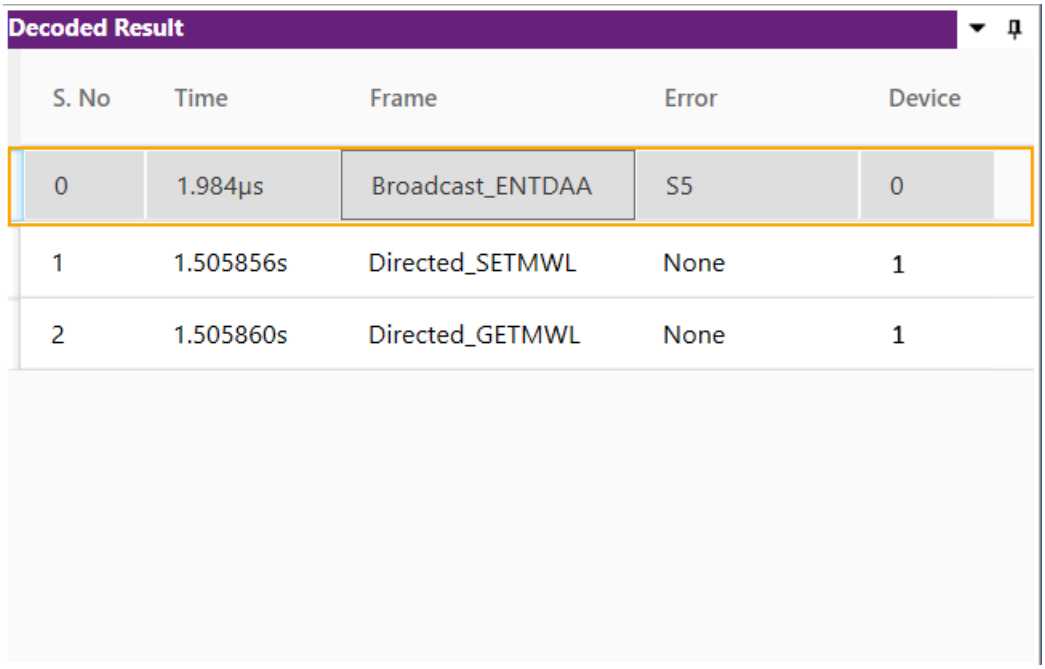
I3C Bus Configuration, Network Traffic generation will be discussed in the next *Chapter 4 Getting started with Exerciser*.

Protocol Listing

The captured signals are decoded and displayed to the end user in a user-friendly manner. This composite of two grids:

- Frame View
- Packet View

Frame View



The frame view gives high level abstract view of the frame, which contains the following fields

HEADER	DESCRIPTION
SL. NO	Frame serial number
TIME	Relative Timestamp at which the packet was captured. This gives you time stamp information. In auto trigger mode times starts from the beginning of the capture. This time resolution is micro second. User can set any row as reference time 0 sec and view the protocol activity with reference to set reference time.
FRAME	It defines the type of Frame Broadcast/ Directed/ Private/ Hot join/ IBI. In case of Broadcast & Directed, it will be displayed along with the command name.
ERROR	In case of any error detected in Frame, it will get displayed over here. It includes the Slave Error (S0 to S5) and Master Error (M0 to M2)
DEVICE	It defines the Node ID assigned during the Bus Configuration which will be discussed in Chapter 4. Node ID will be 0 for Broadcast, Hotjoin & IBI Frame

Chapter 4
Getting started with Exerciser

Overview

PGY I3C EX-PD Lite for MIPI I3C provides advanced test and debug solution for I3C device. It acts as traffic generator in the Bus that you can use to send frame or customize response on the connected network of I3C Devices.

This release supports both Exerciser and Analyzer. Exerciser can emulate as I3C Master, I3C Slave or as an I2C Slave.

Introduction to Bus Configuration

The Bus Configuration window can be selected either through one of the followings:

- *Menu Bar*: Select View->Bus Configuration
- *Tool Bar*: Click on the following icon  .

Upon the selection, the window will appear on the application as shown below

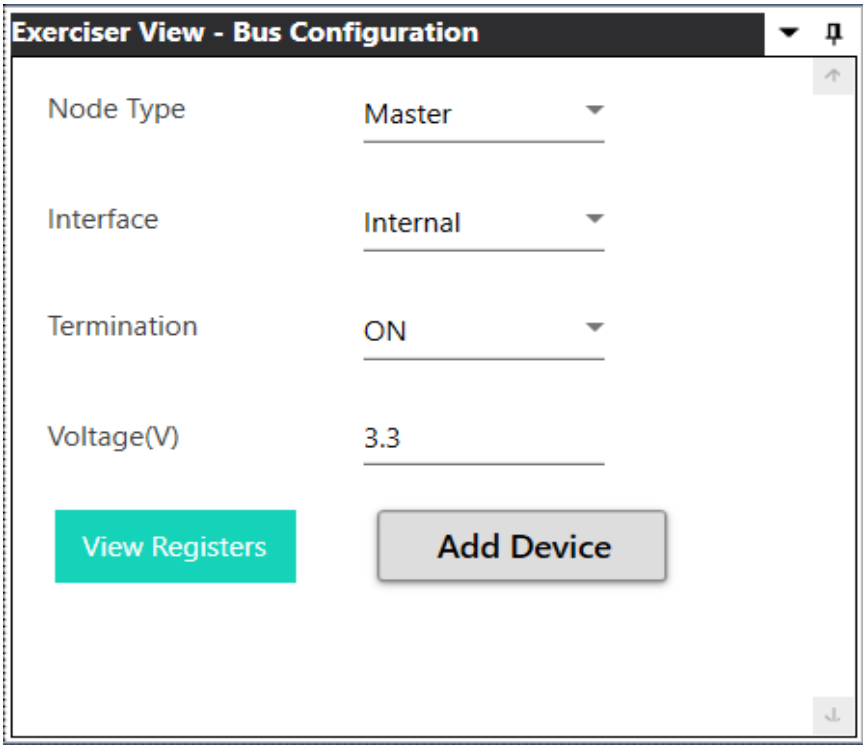


Figure 5: Bus Configuration

Terminology	Description
Node Type	Defines the node type in which it need to be configured. It supports I3C Master, I3C Slave & I2C Slave
Termination	When the connected DUT doesn't have Pull-up resistor, *Termination* needs to be turned *ON*
Voltage	It supports 1.2V ,1.8V & 3.3V
Static Address	It's a unique address defined to slave, through which Master tries to communicate with the device.
Register	Registers available only for I3C device, which includes *PID, BCR, DCR* which defines characteristic value of the device.
Interface	Whether specified device is Internal (Prodigy Exerciser) or External (connected through probe)

Table 1 Bus Configuration Terminology

Configuring Device as I3C Master

Step 1: Select *Node Type* as Master.

Step 2: Select *Interface* as Internal.

Step 3: Select the *Termination* as OFF, if the connected Slave has Pull-up resistor.

Step 4: Enter the *Voltage*, in which Master needs to operate the signal.

Step 5: Click on *View Register*, update the register values for the node.

Step 6: Click on *Add Device*.

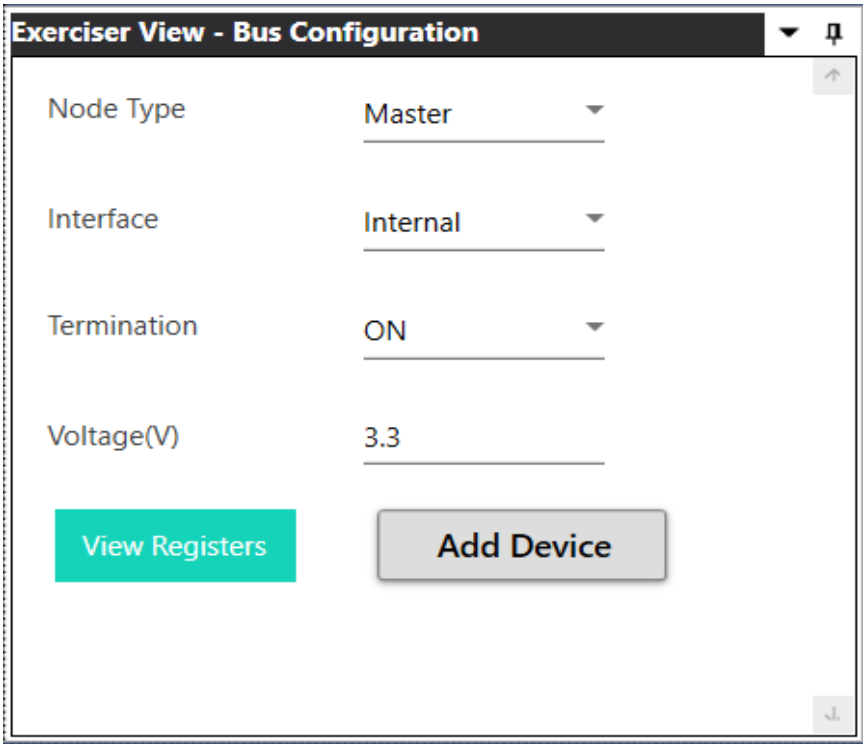


Figure 6: Add Master

Upon successful adding of device, a message with "Device added successfully with node Id #0" will be appeared on screen

Note: Ensure the connection is been established before adding the device. PID should be unique and doesn't match with other devices. More than one master cannot be added in the I3C network.

Configuring Device as I3C Slave

Step 1: Select *Node Type* as I3C Slave.

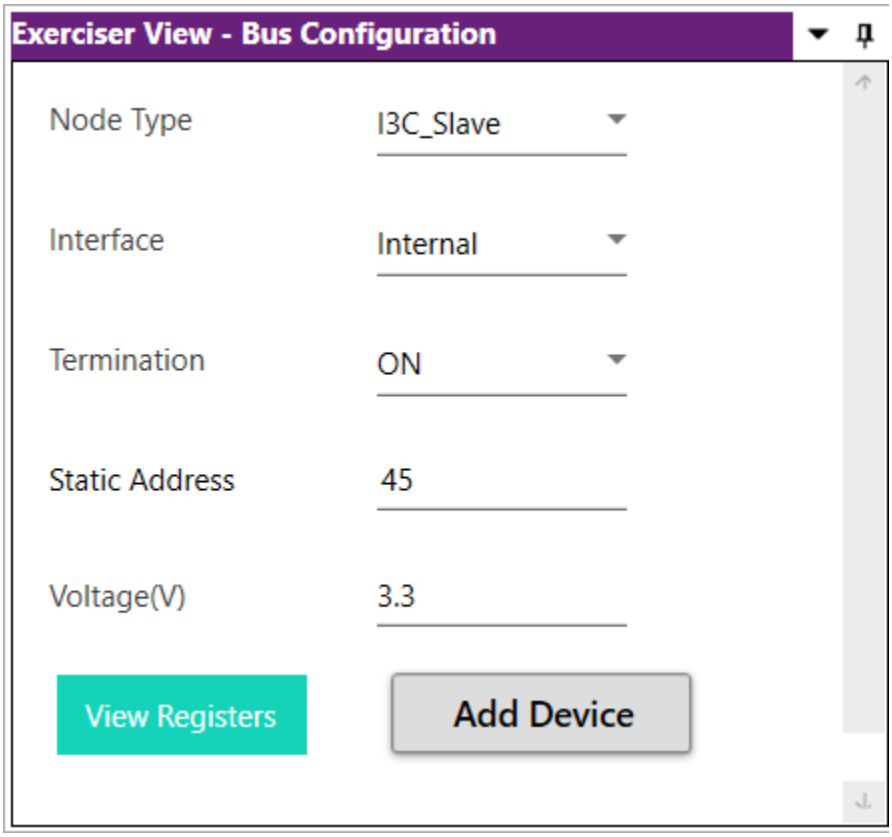
Step 2: Select the *Termination* as OFF, if the connected DUT has Pull-up resistor.

Step 3: Enter the *Static Address*, in which Master need to communicate with the slave.

Step 4: Enter the *Voltage*, in which Slave needs to operate the signal.

Step 5: Click on *View Register*, update the register values (BCR,DCR & PID) for the node.

Step 6: Click on *Add Device*.



Upon successful adding of device, a message with “Device added successfully with node Id #3” will be appeared on screen

Note: Ensure the connection is been established before adding the device. PID, Static Address should be unique and doesn’t match with other devices.

Configuring Device as Secondary Master

Step 1: Select *Node Type* as Sec Master.

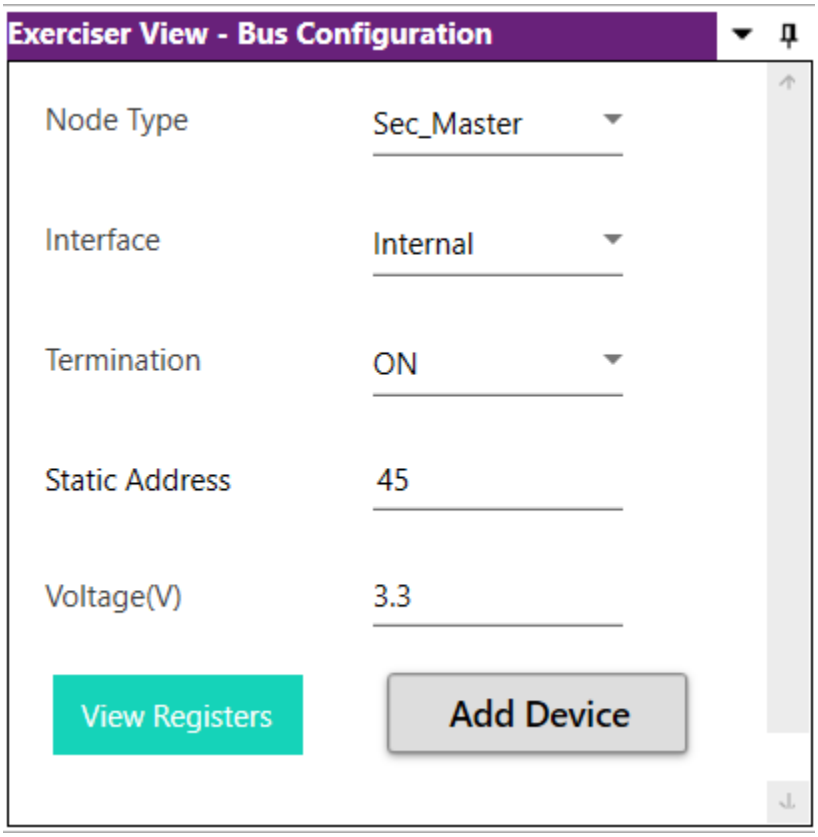
Step 2: Select the *Termination* as OFF, if the connected DUT has Pull-up resistor.

Step 3: Enter the *Static Address*, in which Master need to communicate with the slave.

Step 4: Enter the *Voltage*, in which Slave needs to operate the signal.

Step 5: Click on *View Register*, update the register values for the node.

Step 6: Click on *Add Device*.



Upon successful adding of device, a message with "Device added successfully with node Id #1" will be appeared on screen.

Note: Ensure the connection is been established before adding the device. PID, Static Address should be unique and doesn't match with other devices.

Chapter 4
Exerciser Script

Overview

The large text area constituting at the bottom of the application allows the user to write script to generate the traffic in the I3C network. This text field has the IntelliSense / Code Assist for a variety of script editing features including: keyword prediction, completion, quick info, and option lists.

The Script Window allows to Save the script in form of text file (.txt) extension in Application specific path and load the same file.

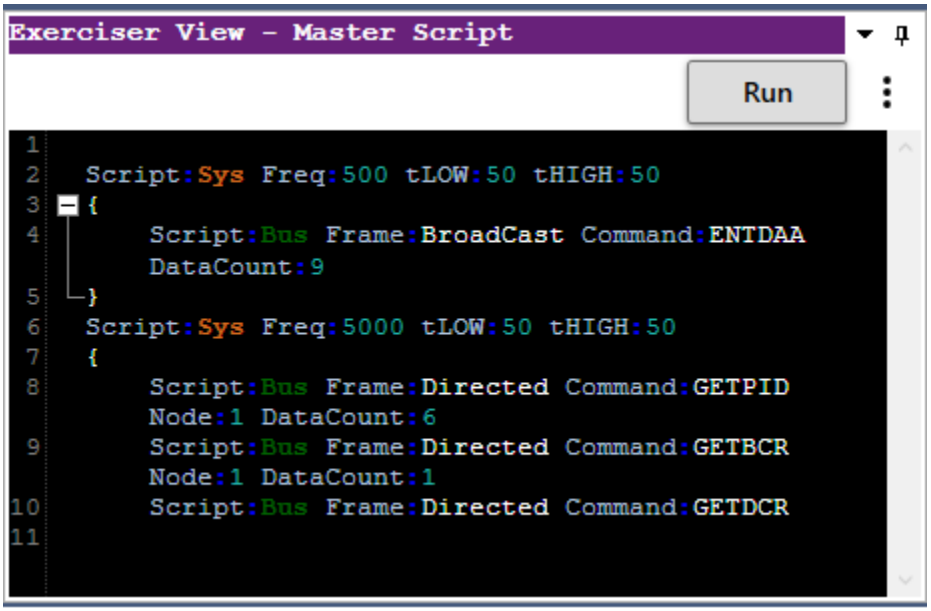


Figure 7: Script Window

Keywords

The script keywords are listed below in the table. The keywords are not case sensitive and are immediately followed by special character ***:*** and parameter. Each keyword-value pair is separated from the next keyword-value pair by space as delimiter.

Script	*Frame*	*Command*	*Transfer*
Node	*Data*	*DataCount*	*Condition*
Error	*HDRCommand*	*HDRData*	*Freq*
tHIGH	*tLOW*	*Stop*	

Table 2: Keyword List

The script will look like below:

```
Script:Sys Freq:400 tLOW:50 tHIGH:50
{
Script:SysOD Freq:500 tLOW:50 tHIGH:50

    Script: Bus Frame:BroadCast Command:ENTDAA DataCount:9
}


Script:Sys Freq:2000 tLOW:50 tHIGH:50
{
    Script: Bus Frame:Directed Command:SETMWL Node:1 Data:10-20
    Script: Bus Frame:Private Node:1 Transfer:WRITE Data:11-22-33-44
    Script: Bus Frame:Private Node:1 Transfer:READ DataCount:4
}

Script:Sys Freq:12500 tLOW:50 tHIGH:50
{
    Script: Bus Frame:BroadCast Command:ENTHDR0 HDRCommand:0402
HDRData:12-34-56-78 Stop: False
    Script:BusExtend HDRCommand:8402 DataCount:4
}
```

Keyword Description

Keyword	Description	Values
`Script`	Defines the script type of the following line	• *Sys* - for duty cycle setup • *Bus* – network traffic • SysOD – for open drain freq setup • Loop - • BufferAcess

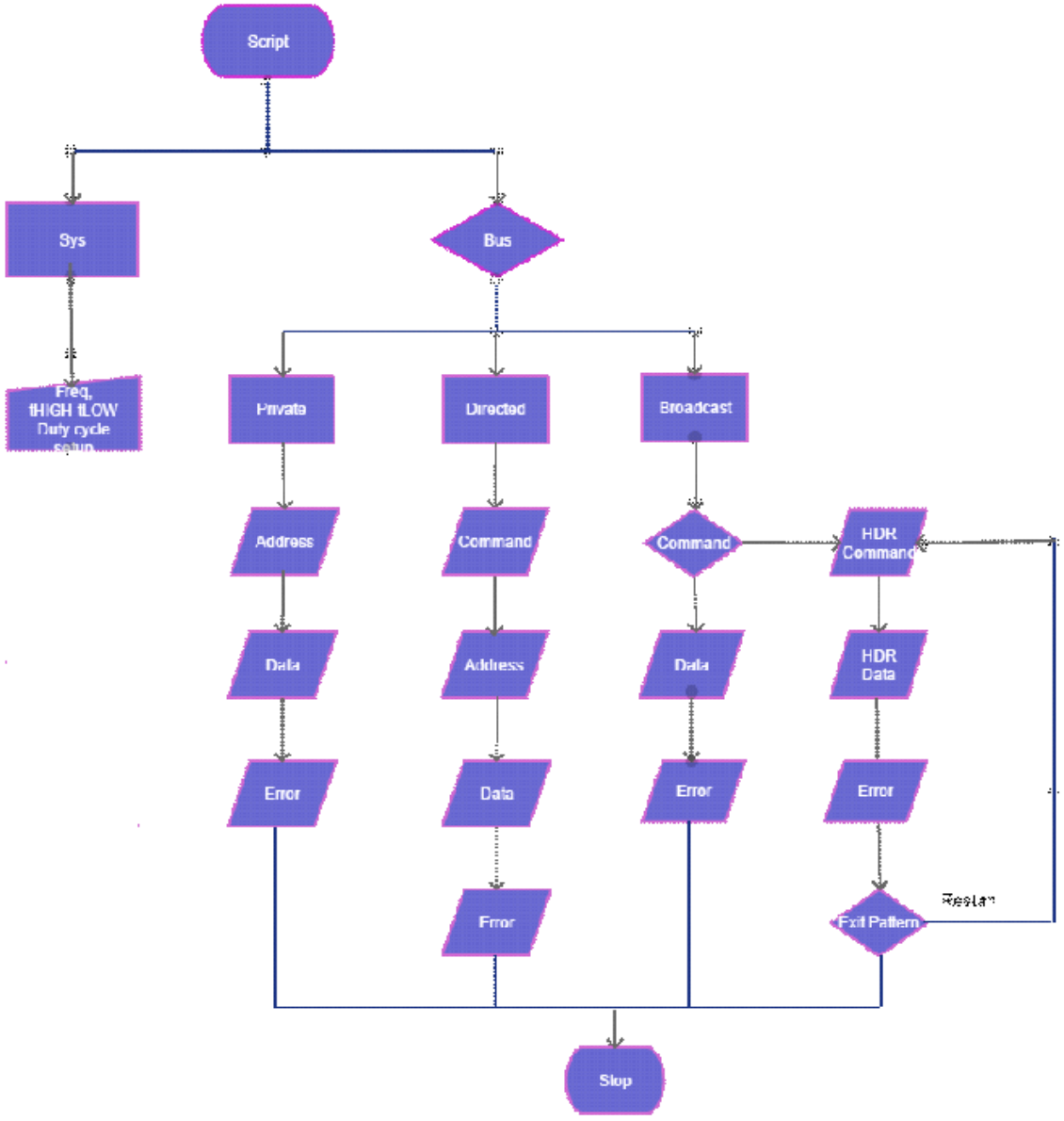
`Frame`	Describe the Frame type of the I3C Message	• *Broadcast*
		• *Directed*
		• *Private*
		• *I2C*
		• *IBI*
		  ○ Slave Script: Interrupt
		  ○ Master Script: Enable / Disable Event
		• *MastershipReq*
		  ○ Slave Script: Interrupt
		  ○ Master Script: Enable / Disable Event
		• *Hot join*
		  ○ Slave Script: Interrupt
		  ○ Master Script: Enable / Disable Event
		• HDRExitPattern

`Command`	Describes the CCC defined in the I3C spec.	User allowed to enter *CCC* Name or *hex value*
`Transfer`	Defines the transfer type of the Private/I2C message	• *Write* • *Read*
`Node`	Defines the Node ID assigned by the device, unique ID which is used for communicating with device	Anywhere between *0* to *15*
`Data`	Defines 8 bit Hex value during write message.	Multiple data bytes can be supported with special character hyphen – as separator
`DataCount`	8 bit decimal value define the number of data to be read on Read Message	Anywhere between 0 to 255
`Condition`	Defines the initiation of the interrupt from the slave	• *Idle* • *Start*
`HDRCommand`	16bit Hex Value. Which define Transfer Type, Command Code, Node Id	*0000* to *FFFF* • Bit[15] – Transfer Type • Bit[14:8] – Command Code • Bit[7:1] – Node Id • Bit[0] - Reserved
`HDRData`	8 bit Hex Value defined during HDR Write Command	Multiple data bytes can be supported with special character hyphen – as separator. Number of bytes should be in multiple of two

Stop	Issue stop at the end of frame (Optional)	• *True* • *False*
Request	Allows to Enable/Disable – IBI/ MastershipReq/ Hotjoin	• *Enable* • *Disable*
Freq	Defines the frequency of Clock.	Optional data in KHz. Anywhere between *100* to *12500.*
tLOW	Clock Low period	• If Freq is mentioned, the tLOW values will be in % of the total Frequency • Else tLOW will be in Low Time period in *ns,* with minimum value 40ns. (NOTE: For range between 100KHz to 400KHz, period can be changed in steps of 100ns For range between 400KHz to 12.5 MHz, period can be changed in steps of 10ns)
tHIGH	Clock High Period	• If Freq is mentioned, the tHIGH values will be in % of the total Frequency • Else tHIGH will be in Low Time period in *ns,* with minimum value 40ns. (NOTE: For range between 100KHz to 400KHz, period can be changed in steps of 100ns For range between 400KHz to 12.5 MHz, period can be changed in steps of 10ns)

Table 3: Keyword Definition

Script Flow Diagram



Master Script – Broadcast SDR Message

Write Message - Syntax:

`Script:Bus Frame:BroadCast Command:<Value> Data:<Value> Stop:<Value>`

Example:

```
Script:Bus Frame:BroadCast Command:SETMWL Data:00-FF
```

Read Message - Syntax:

`Script:Bus Frame:BroadCast Command:<Value> DataCount:<Value> Stop:<Value>`

Example:

```
Script:Bus Frame:BroadCast Command:ENTDAA DataCount:9
```

Master Script – Broadcast HDR Message

Write Message - Syntax:

`Script:Bus Frame:BroadCast Command:<Value> HDRCommand: <Value> HDRData:<Value> Stop:<Value>`

Example:

```
Script:Bus Frame:BroadCast Command:ENTHDR0 HDRCommand:0402 HDRData:10-20-30-40
```

Read Message - Syntax:

`Script:Bus Frame:BroadCast Command:<Value> HDRCommand: <Value> DataCount:<Value> Stop:<Value>`

Example:

```
Script:Bus Frame:BroadCast Command:ENTHDR0 HDRCommand:8402 DataCount:9
```

HDR with Restart - Syntax

`Script:Bus Frame:BroadCast Command:<Value> HDRCommand: <Value> HDRData:<Value> Stop:False`

Example:

```
Script:Bus Frame:BroadCast Command:ENTHDR0 HDRCommand:0402 HDRData:10-20-30-40 Stop:False
Script:BusExtend HDRCommand:8402 DataCount:9
```

Master Script – Directed Message

Write Message - Syntax:

```
Script:Bus Frame:Directed Command:<Value> Node:<Value> Data:<Value>
Stop:<Value>
```

Example:

```
Script:Bus Frame:Directed Command:SETMWL Node:1 Data:00-FF
```

Read Message - Syntax:

```
Script:Bus Frame:Directed Command:<Value> Node:<Value> DataCount:<Value>
Stop:<Value>
```

Example:

```
Script:Bus Frame:Directed Command:GETMWL Node:1 DataCount:2
```

Master Script – Private Message

Write Message - Syntax:

```
Script:Bus Frame:Private Node:<Value> Data:<Value> Stop:<Value>
```

Example:

```
Script:Bus Frame:Private Node:1 Data:00-FF
```

Read Message - Syntax:

```
Script:Bus Frame:Private Node:<Value> DataCount:<Value> Stop:<Value>
```

Example:

```
Script:Bus Frame:Private Node:1 DataCount:2
```

Master Script – I2C Message

Write Message - Syntax:

```
Script:Bus Frame:I2C Node:<Value> Data:<Value> Stop:<Value>
```

Example:

```
Script:Bus Frame:I2C Node:1 Data:00-FF
```

Read Message - Syntax:

```
Script:Bus Frame:I2C Node:<Value> DataCount:<Value> Stop:<Value>
```

Example:

```
Script:Bus Frame:I2C Node:1 DataCount:2
```

Master Script – BufferAccess

BufferAccess Private - Syntax:

```
Script:BufferAccess Mode:Private Node:<Value> DataCount:<Value>
```

Example:

```
Script:BufferAccess Mode:Private Node:1 DataCount:4
```

BufferAccess DDR - Syntax:

```
Script:BufferAccess Frame:DDR Node:<Value> Command:<Value> DataCount:<Value>
```

Example:

```
Script:BufferAccess Mode:DDR Node:1 Command:4 DataCount:4
```

BufferAccess TSP - Syntax:

```
Script:BufferAccess Frame:TSP Node:<Value> Command:<Value> DataCount:<Value>
```

Example:

```
Script:BufferAccess Mode:TSP Node:1 Command:4 DataCount:4
```

Master Script – IBI

IBI - Syntax:

```
Script:Bus Frame:IBI Request:<Value>
```

Example:

```
Script:Bus Frame:IBI Request:Enable
Script:Bus Frame:IBI Request:Disable
```

Master Script – Hotjoin

Hotjoin - Syntax:

```
Script:Bus Frame:Hotjoin Request:<Value>
```

Example:

```
Script:Bus Frame:Hotjoin Request:Enable
Script:Bus Frame:Hotjoin Request:Disable
```

Master Script – MastershipReq

MasterShipreq - Syntax

```
Script:Bus Frame:MastershipReq Request:<Value>
```

Example:

```
Script:Bus Frame:MastershipReq Request:Enable
Script:Bus Frame:MastershipReq Request:Disable
```

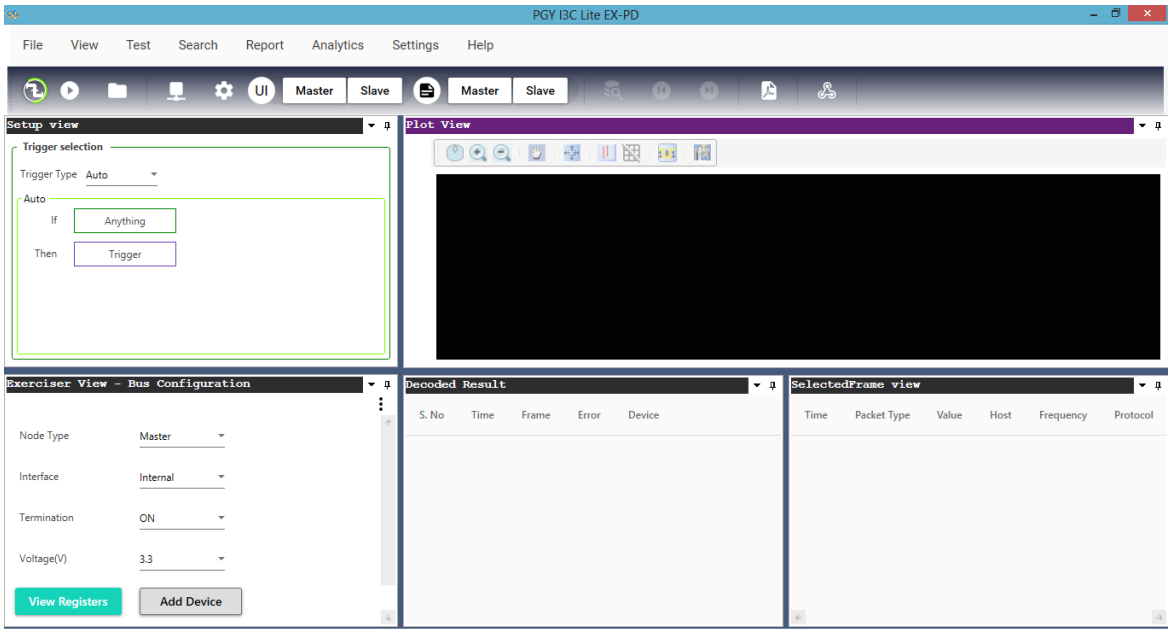
Chapter 5
I3C Demo Scripts

- Ensure that the system meets the requirements as mentioned in the software requirements section.
- Setup the PGY I3C EX-PD Lite unit as described in the hardware setup section.
- Install the PGY I3C Lite EX-PD software and USB drivers. For installation steps click here.

Using Master scripts

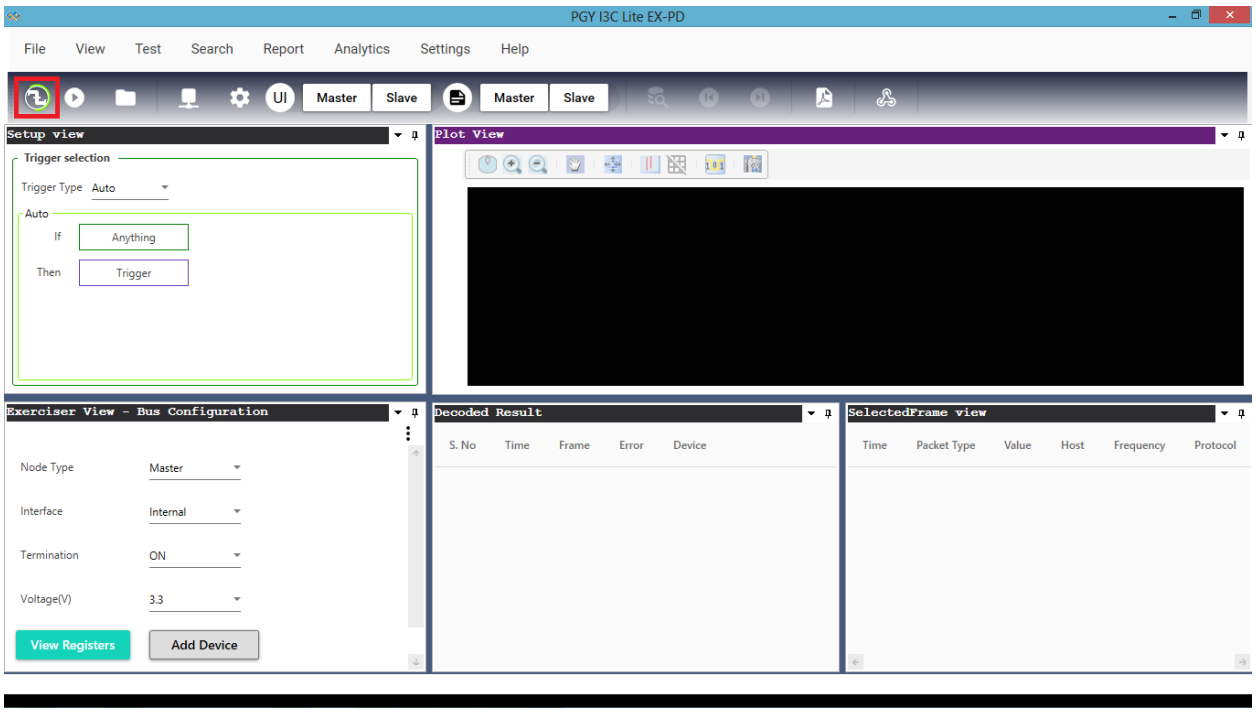
Note: Before using any of the below example scripts, please ensure to add a master and one I3C slave on bus. Also ensure dynamic address is allocated to the added slave.

1. Open the PGY-I3C Lite EX-PD Software. Software is launched as below



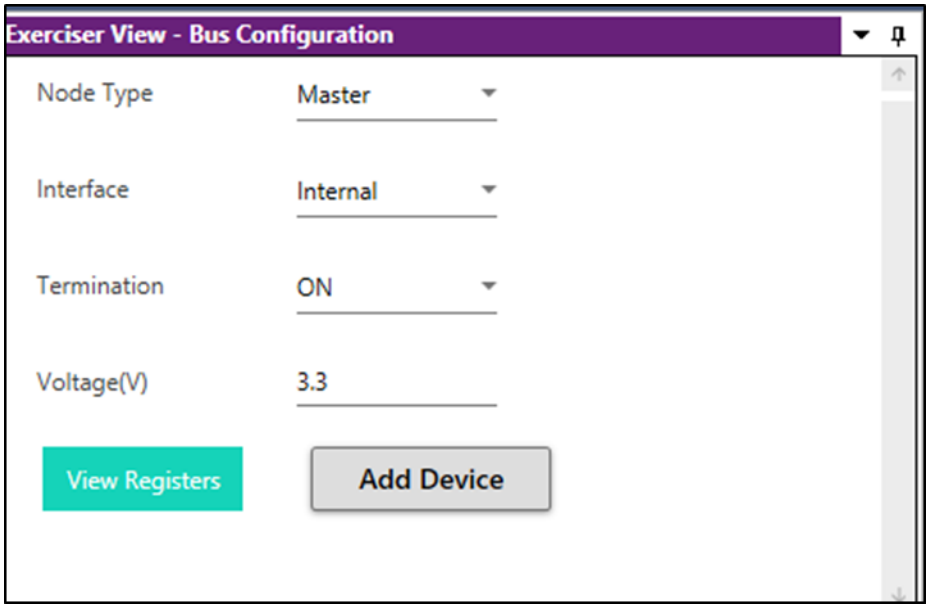
2. Make sure PGY-I3C-EX-PD Lite unit is connected to host computer using USB3.0. interface and powered ON.(Click on hardware reset switch, so it is reset condition)

3. Click on *Test Connection* on the software screen



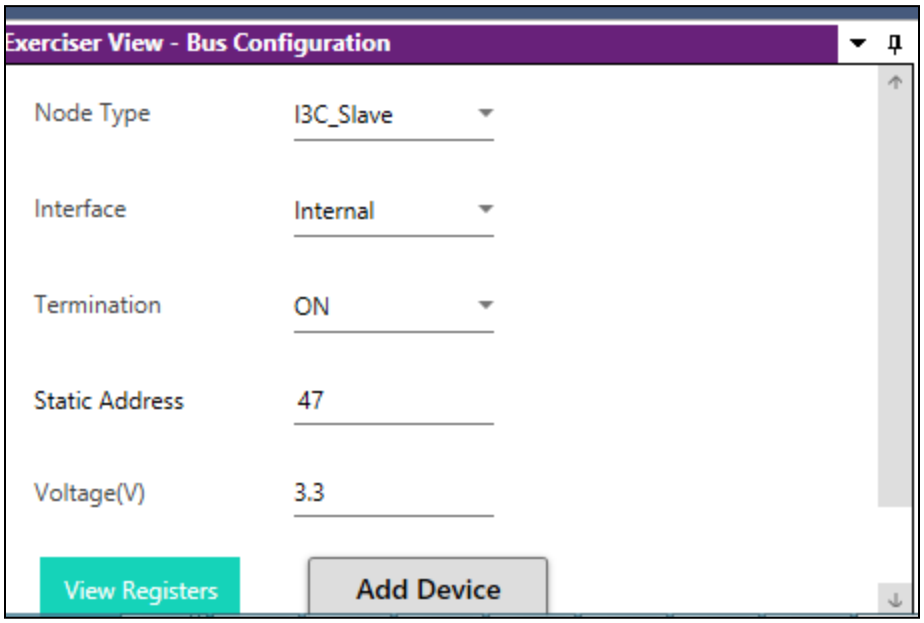
Connection between PGY-I3C-EX-PD Lite hardware unit and the PC will be established. On successful connection, the user will be prompted with a message. If connection is not established, please report to contact@prodigytechno.com

4. Make the following setup in exerciser section to configure the unit as Master

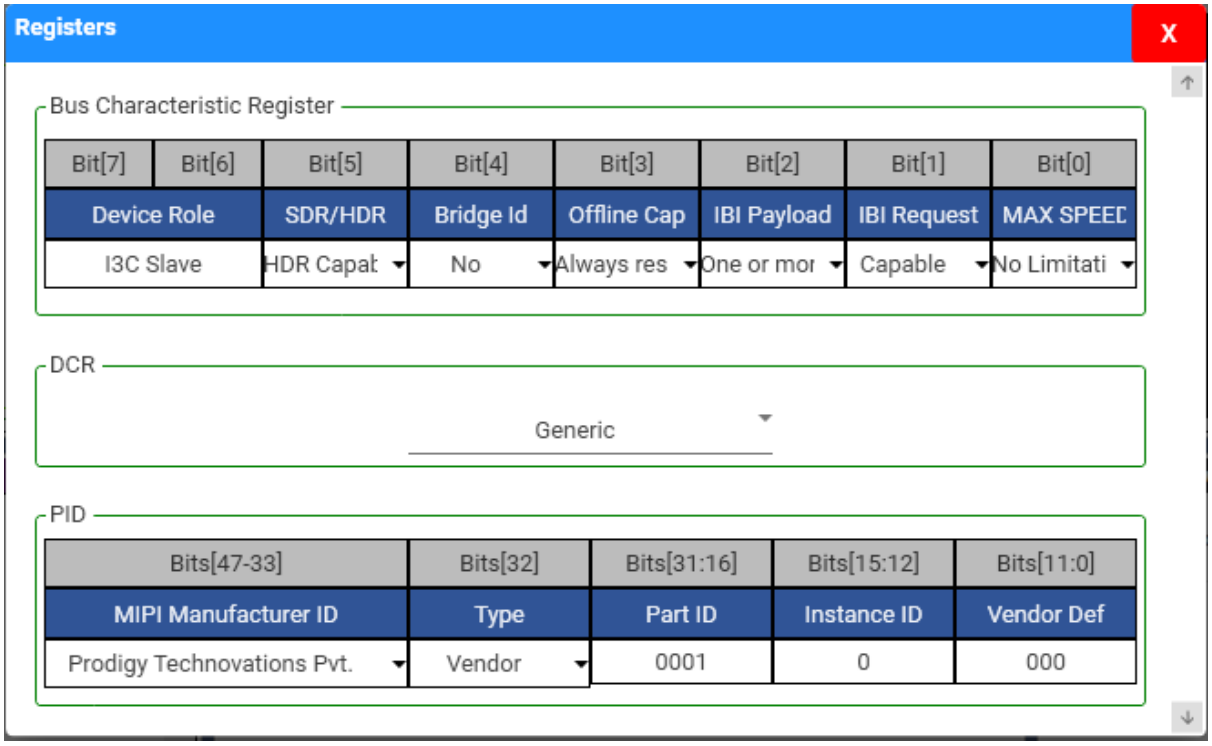


Click on the *Add Device* button. A popup message will be displayed to confirm that the device has been added successfully.

5. An internal I3C Slave can also be added to the unit. Make the following setup in Exerciser view to add an internal slave

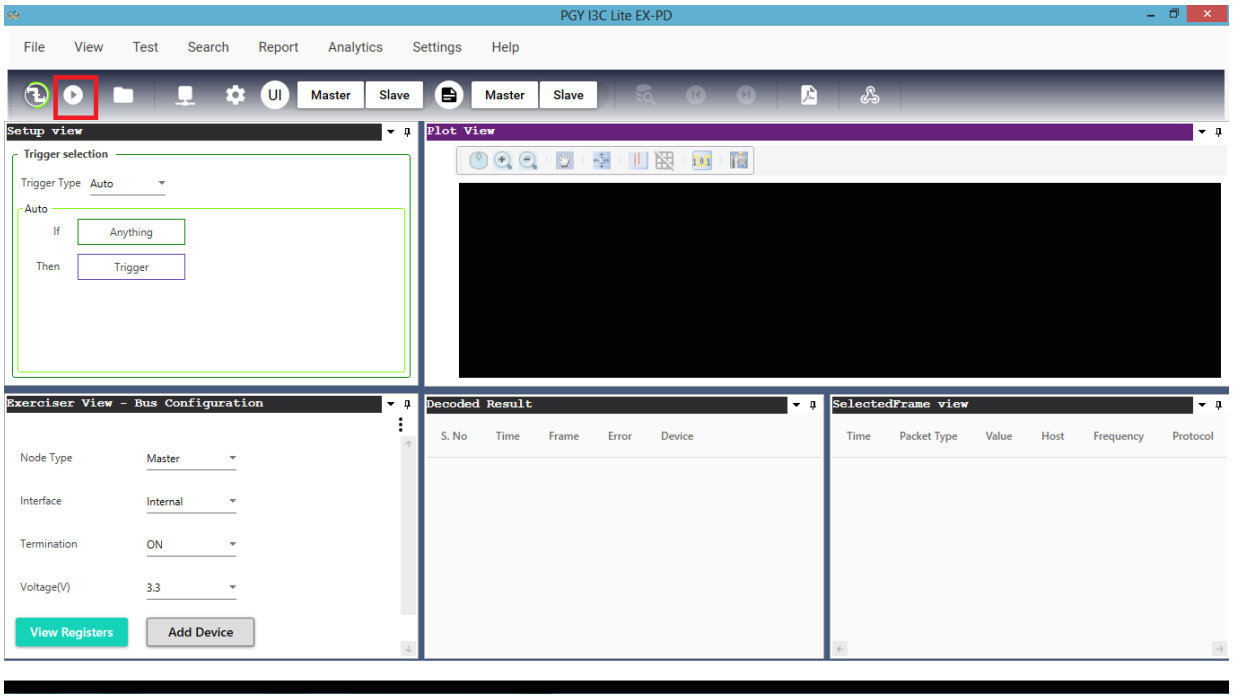


6. Click on *View Registers* button to configure the required settings

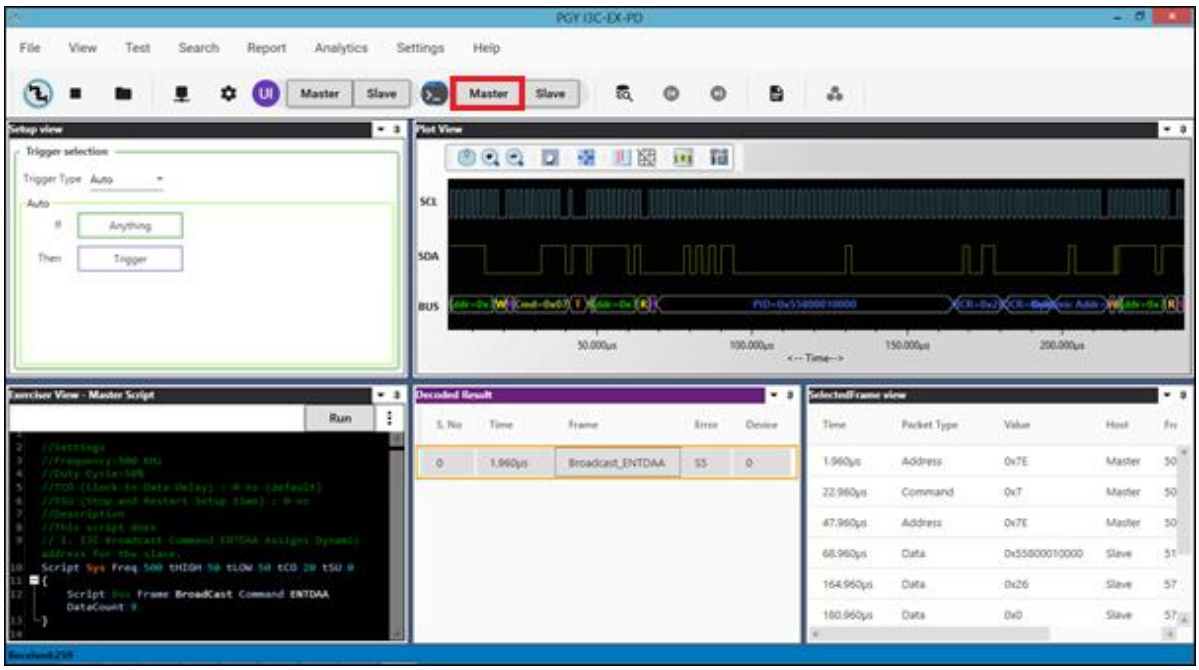


Click on close button to close the Registers view.

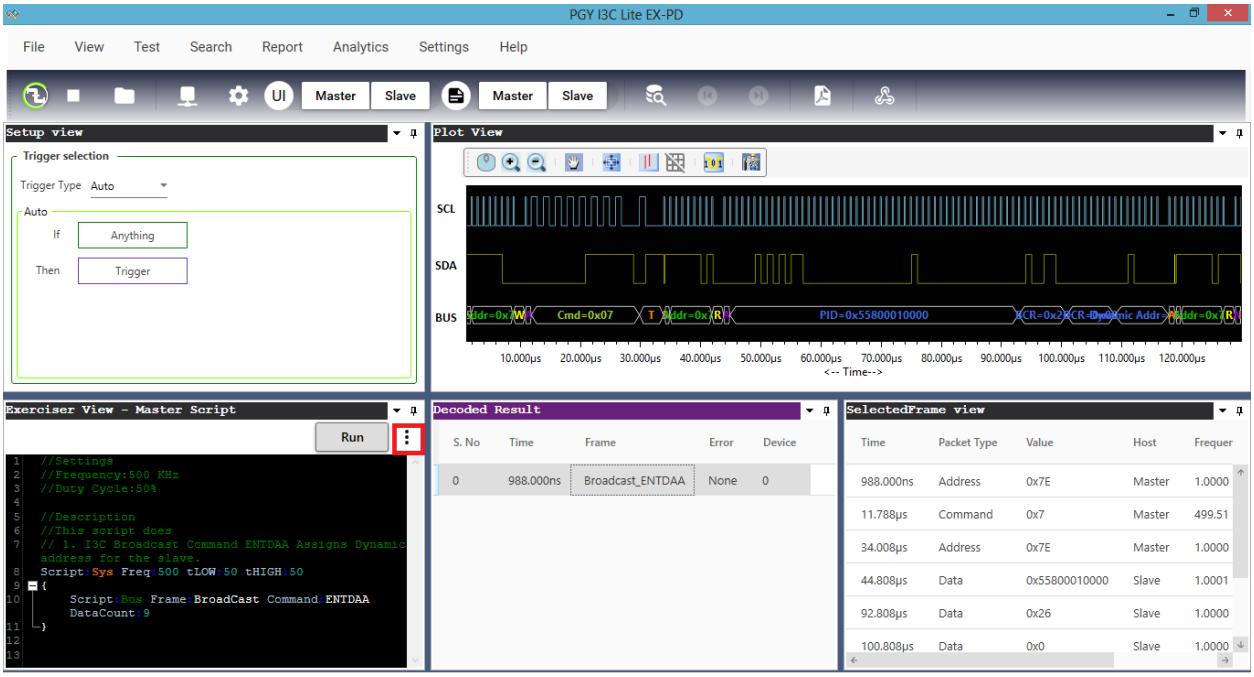
7. Click on *Start Capture* in software to enable the capture process.



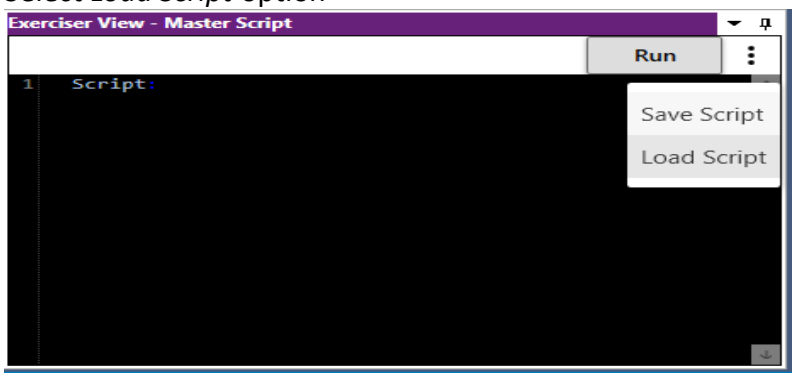
8. Click on *Master* script button to enable the master script window to write/load scripts



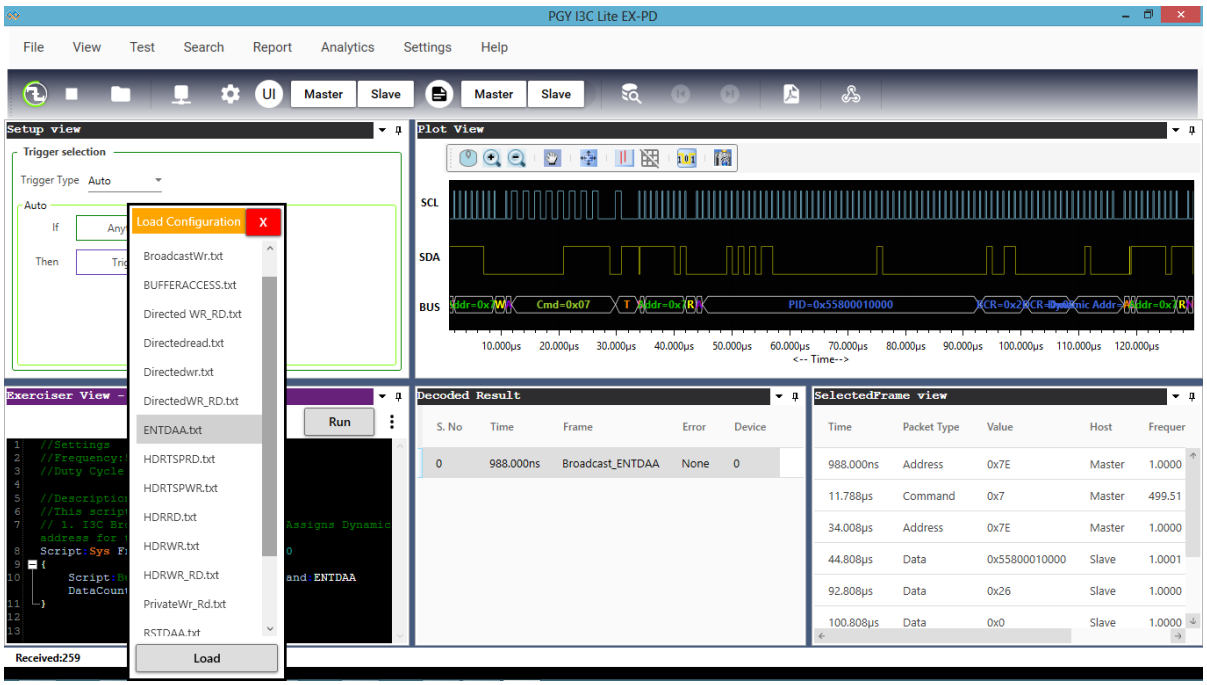
9. Click on the highlighted button and load the ENTDAA script to assign dynamic address to the slave.



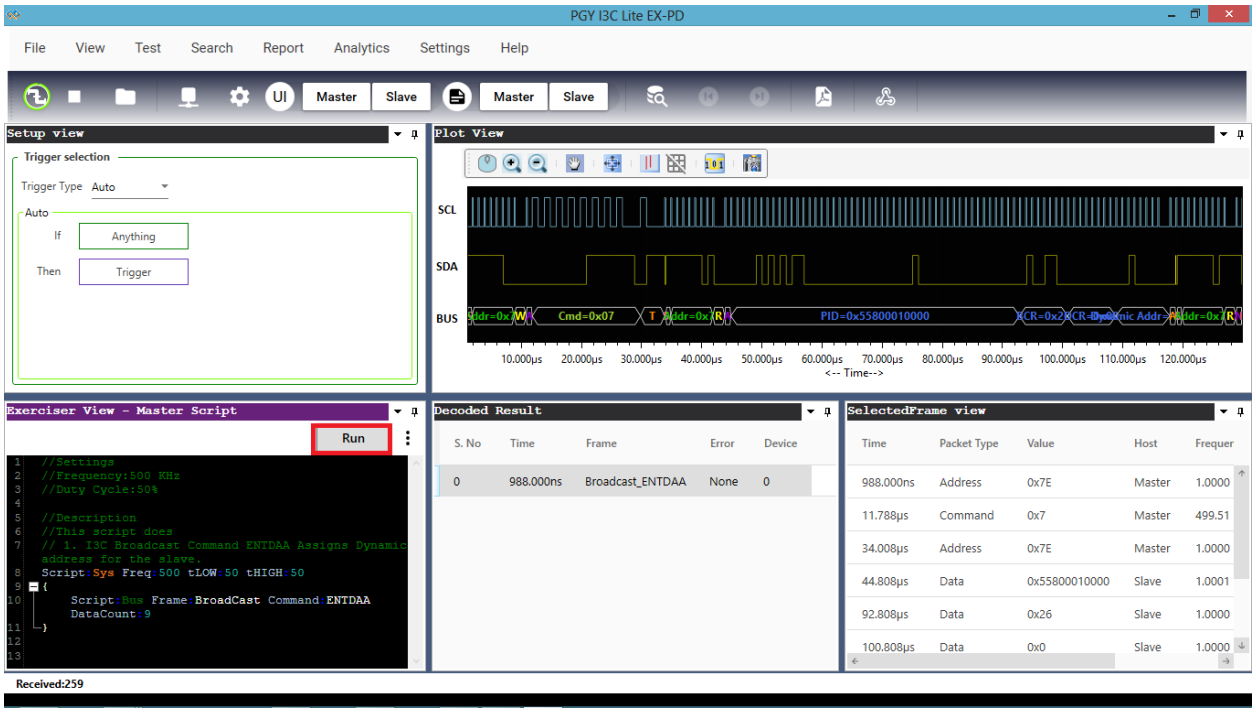
a. Select *Load Script* option



b. Next select *ENTDAA.txt* option and click on the *Load* button to load the script.



10. Click on RUN button to send the loaded script.

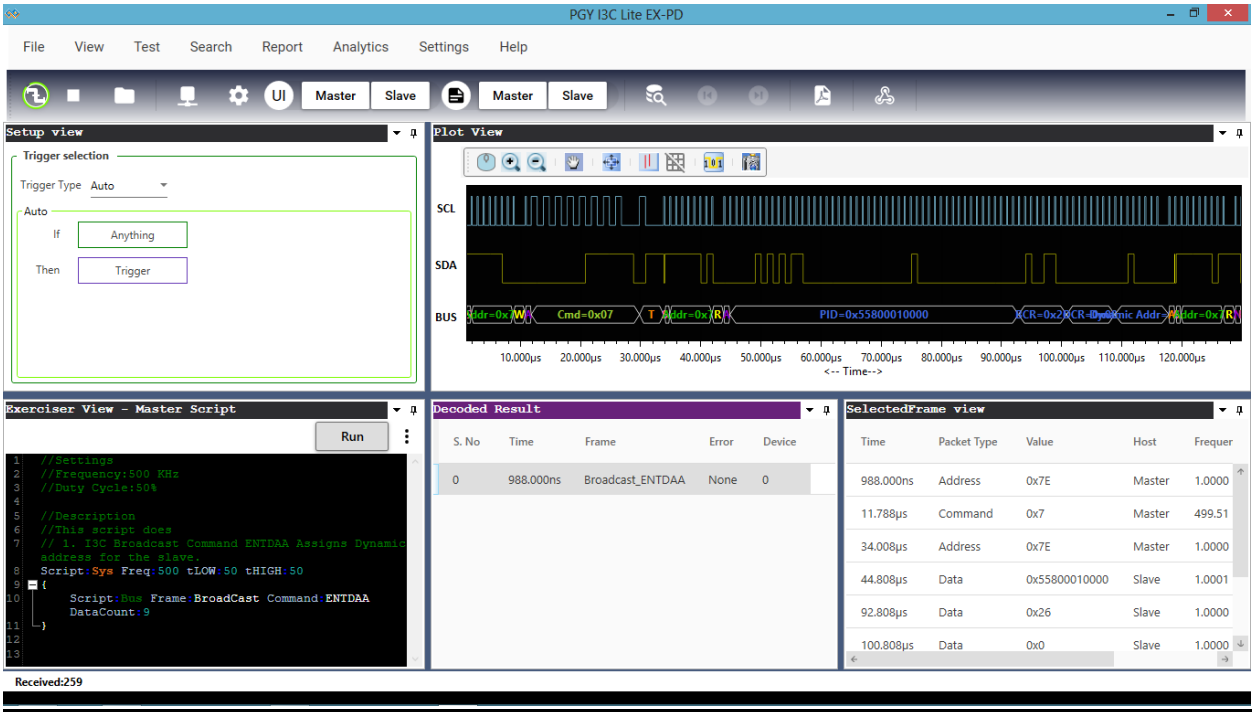


11. Repeat the steps 9 through 10 for other preloaded scripts.

Broadcast ENTDAA (ENTDAA.txt)

//Settings
//Frequency:500 KHz
//Duty Cycle:50%
//Description
//This script does
// 1. I3C Command ENTDAA Assigns Dynamic address for the slave.

```
Script:Sys Freq:500 tHIGH:50 tLOW:50
{
      Script:Bus Frame:BroadCast Command:ENTDAA DataCount:9
}
```



Broadcast SETMWL and SETMRL(BROADCAST_WR.txt)

//Settings

//Frequency:500 KHz

//Clock high time = 40%

//Clock low time = 60%

//Description

//This script does

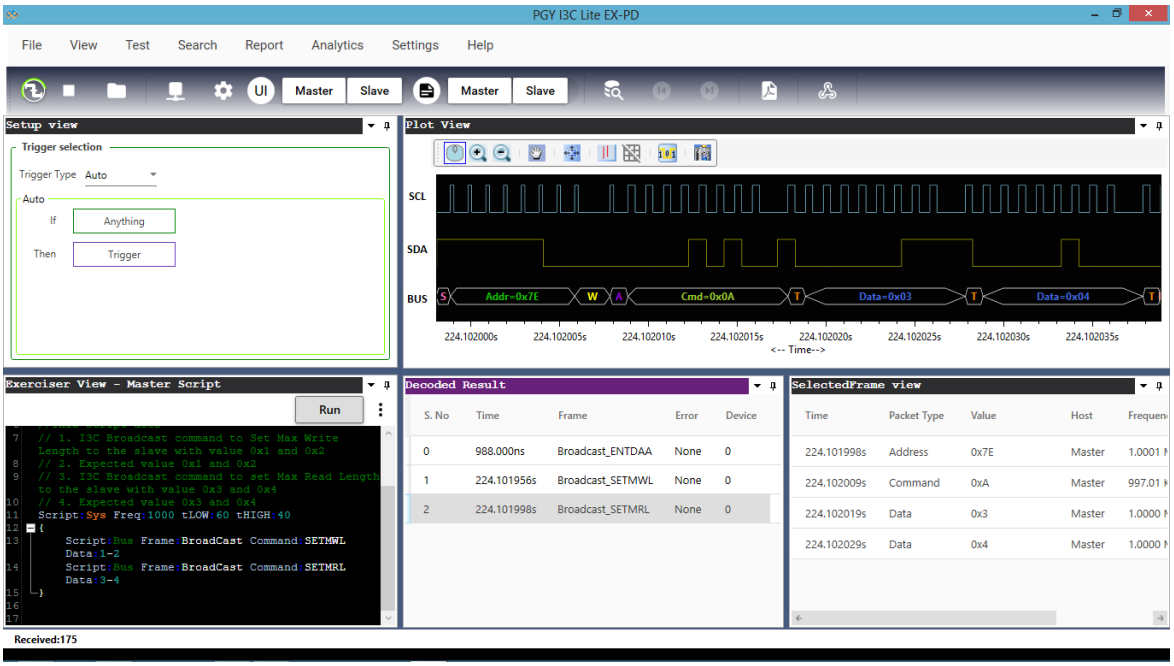
// 1. I3C Broadcast command to Set Max Write Length to the slave with value 0x1 and 0x2

// 2. Expected value 0x1 and 0x2

// 3. I3C Broadcast command to set Max Read Length to the slave with value 0x3 and 0x4

// 4. Expected value 0x3 and 0x4

```
Script:Sys Freq:500 tLOW:60 tHIGH:40
{
      Script:Bus Frame:BroadCast Command:SETMWL Data:1-2
      Script:Bus Frame:BroadCast Command:SETMRL Data:3-4
}
```



Directed GETPID , GETBCR and GETDCR (DIRECTED_READ.txt)

//Settings

//Frequency:4 MHz

//Duty Cycle:50%

//Description

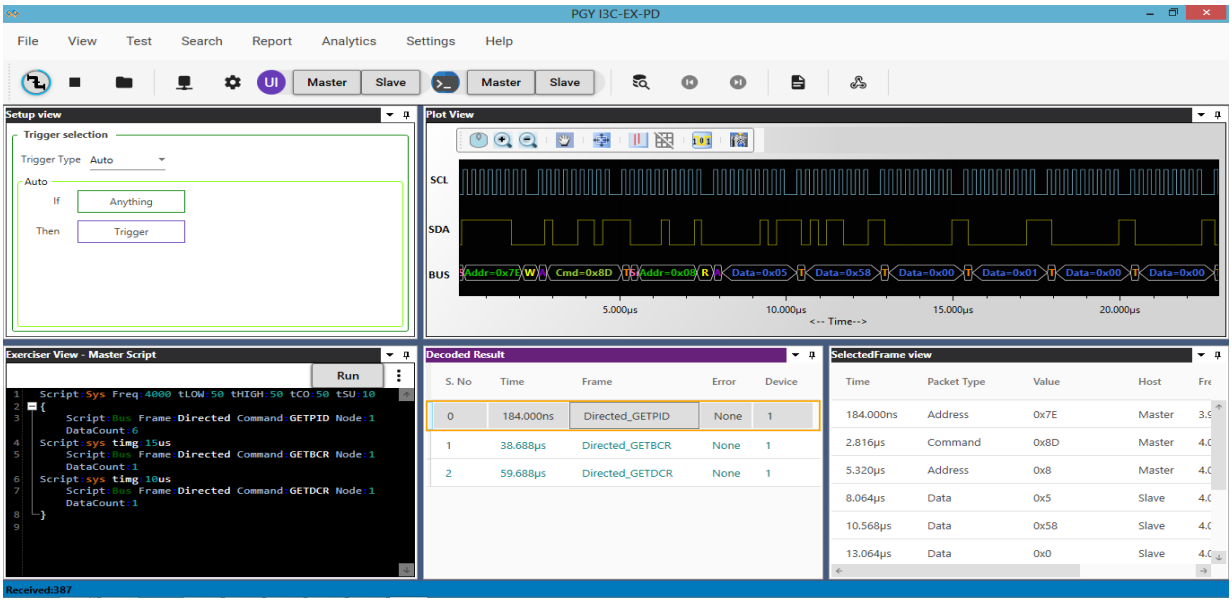
//This script does

// 1. I3C Directed command to get the 48-bit Provisional ID from the slave #1

// 2. I3C Directed command to get the Bus Characteristics Register from the slave #1

// 3. I3C Directed command to get the Device Characteristics Register from the slave #1

```
Script:Sys Freq:4000 tLOW:50 tHIGH:50
{
      Script:Bus Frame:Directed Command:GETPID Node:1 DataCount:6
      Script:Bus Frame:Directed Command:GETBCR Node:1 DataCount:1
      Script:Bus Frame:Directed Command:GETDCR Node:1 DataCount:1
}
```



Directed SETMWL, GETMWL, SETMRL and GETMRL(DIRECTED_WR_RD.txt)

//Settings

//Frequency:4000 KHz

//Duty Cycle:50%

//Description

//This script does

// 1. I3C Directed command to Set Max Write Length to slave #1 with value 0x1 and 0x2

// 2. I3C Directed command to Get Max Write Length from slave #1

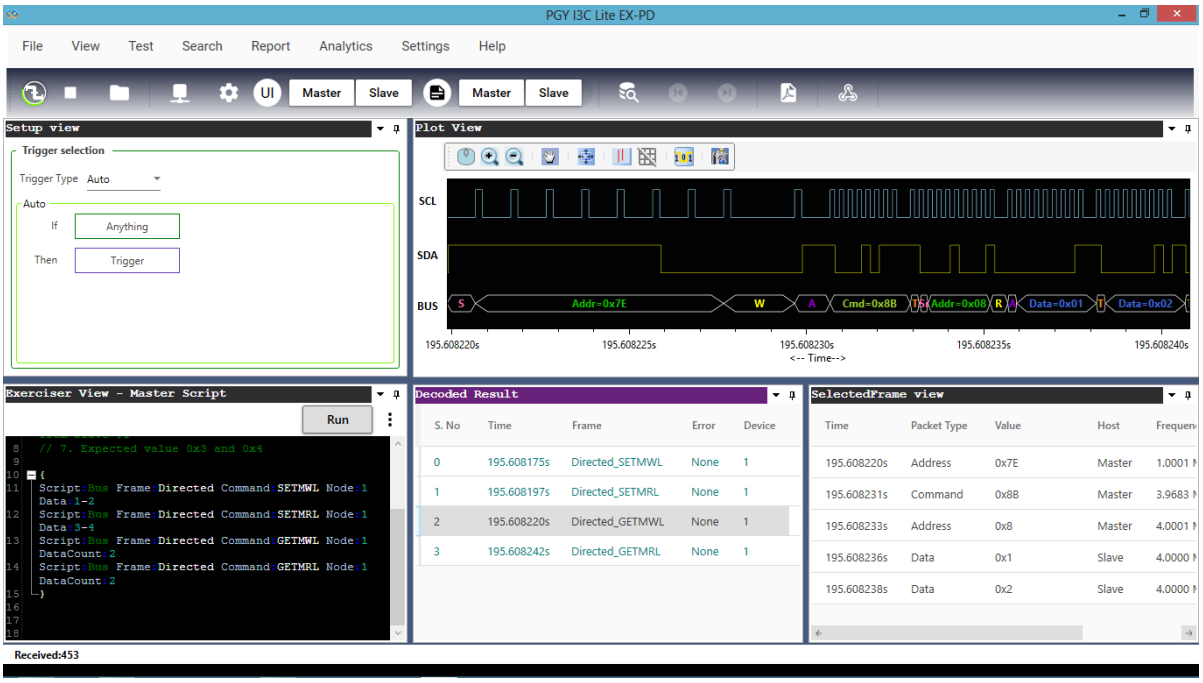
// 3. Expected value 0x1 and 0x2

// 5. I3C Directed command to set Max Read Length to slave #1 with value 0x3 and 0x4

// 6. I3C Directed command to get Max Read Length from slave #1

// 7. Expected value 0x3 and 0x4

```
Script:Sys Freq:4000 tLOW:50 tHIGH:50
{
      Script:Bus Frame:Directed Command:SETMWL Node:1 Data:1-2
      Script:Bus Frame:Directed Command:GETMWL Node:1 DataCount:2
      Script:Bus Frame:Directed Command:SETMRL Node:1 Data:3-4
      Script:Bus Frame:Directed Command:GETMRL Node:1 DataCount:2
}
```



Private WRITE READ (PRVT_WR_RD.txt)

//Settings

//Frequency:5 MHz

//Duty Cycle:50%

//Description

//This script does

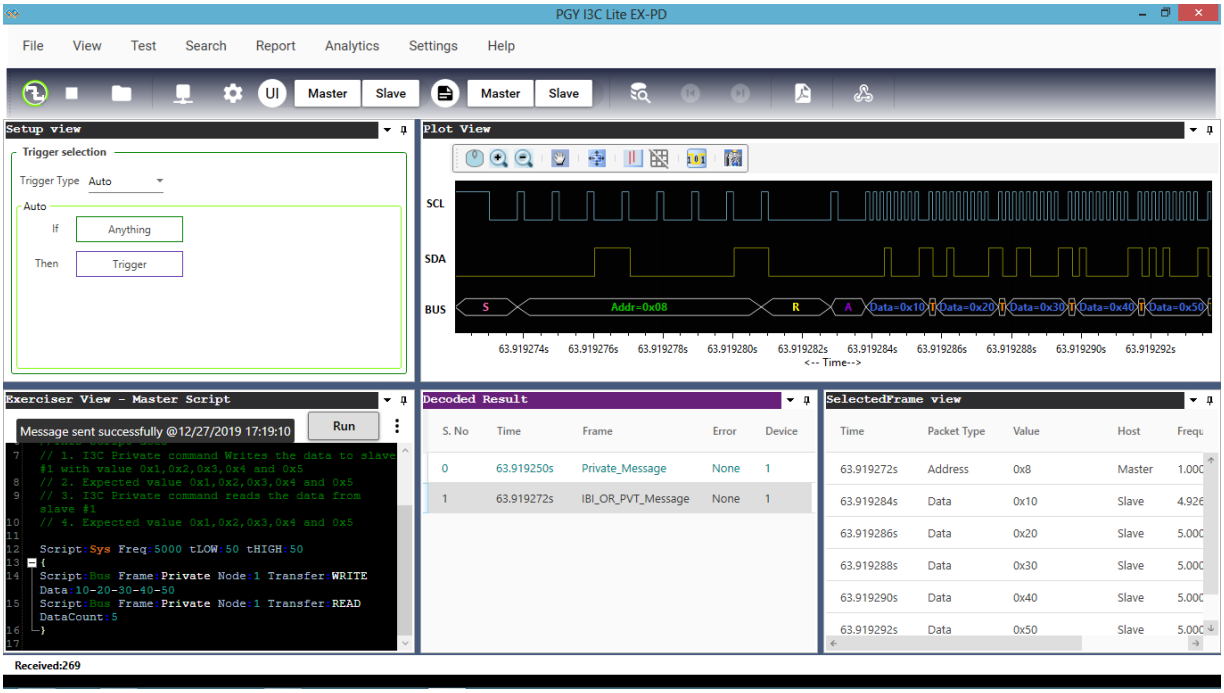
// 1. I3C Private command Writes the data to slave #1 with value 0x1,0x2,0x3,0x4 and 0x5

// 2. Expected value 0x1,0x2,0x3,0x4 and 0x5

// 3. I3C Private command reads the data from slave #1

// 4. Expected value 0x1,0x2,0x3,0x4 and 0x5

```
Script:Sys Freq:5000 tLOW:50 tHIGH:50
{
        Script:Bus Frame:Private Node:1 Transfer:WRITE Data:1-2-3-4-5
        Script:Bus Frame:Private Node:1 Transfer:READ DataCount:5
}
```



Broadcast HDR DDR WRITE (HDRWR.txt)

//Settings

//Frequency:12.5 MHz

//Duty Cycle:50%

//TCO (Clock to Data Delay) : 0 ns

//TSU (Stop and Restart Setup time) : 20 ns

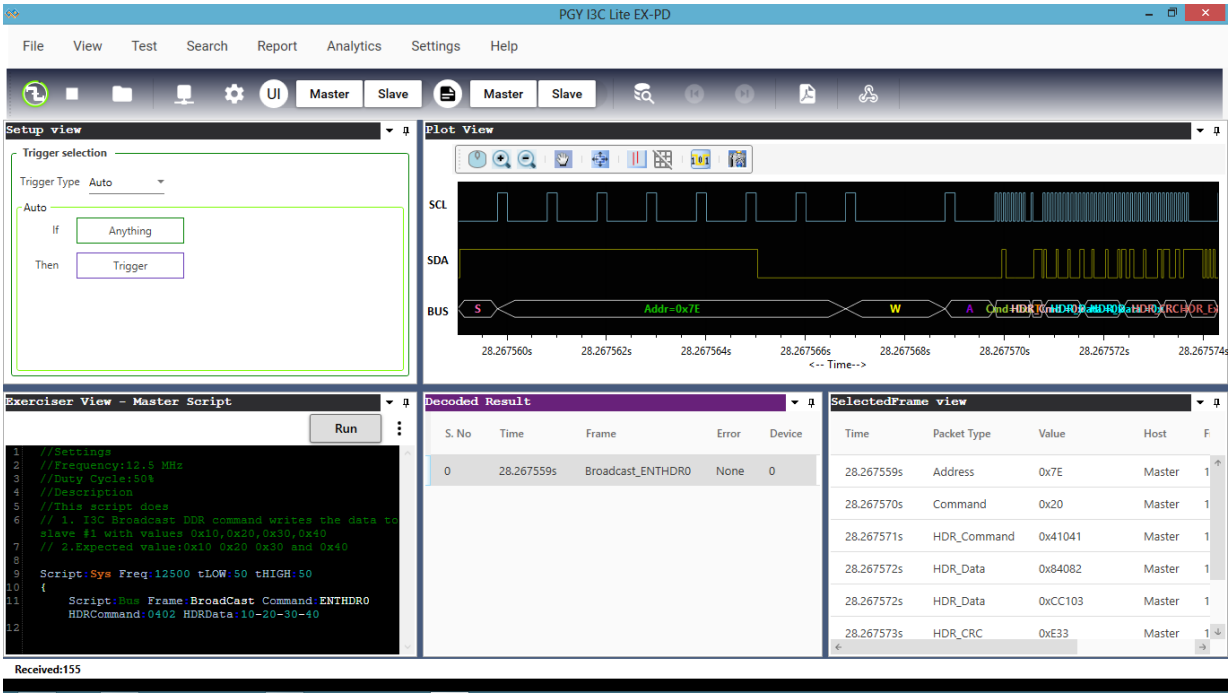
//Description

//This script does

// 1. I3C Broadcast DDR command writes the data to slave #1 with values 0x10,0x20,0x30,0x40

// 2.Expected value:0x10 0x20 0x30 and 0x40

```
Script:Sys Freq:12500 tLOW:50 tHIGH:50
{
      Script:Bus Frame:BroadCast Command:ENTHDR0 HDRCommand:0402
HDRData:10-20-30-40
}
```



Broadcast HDR DDR READ (HDRRD.txt)

//Settings

//Frequency:12.5 MHz

//Duty Cycle:50%

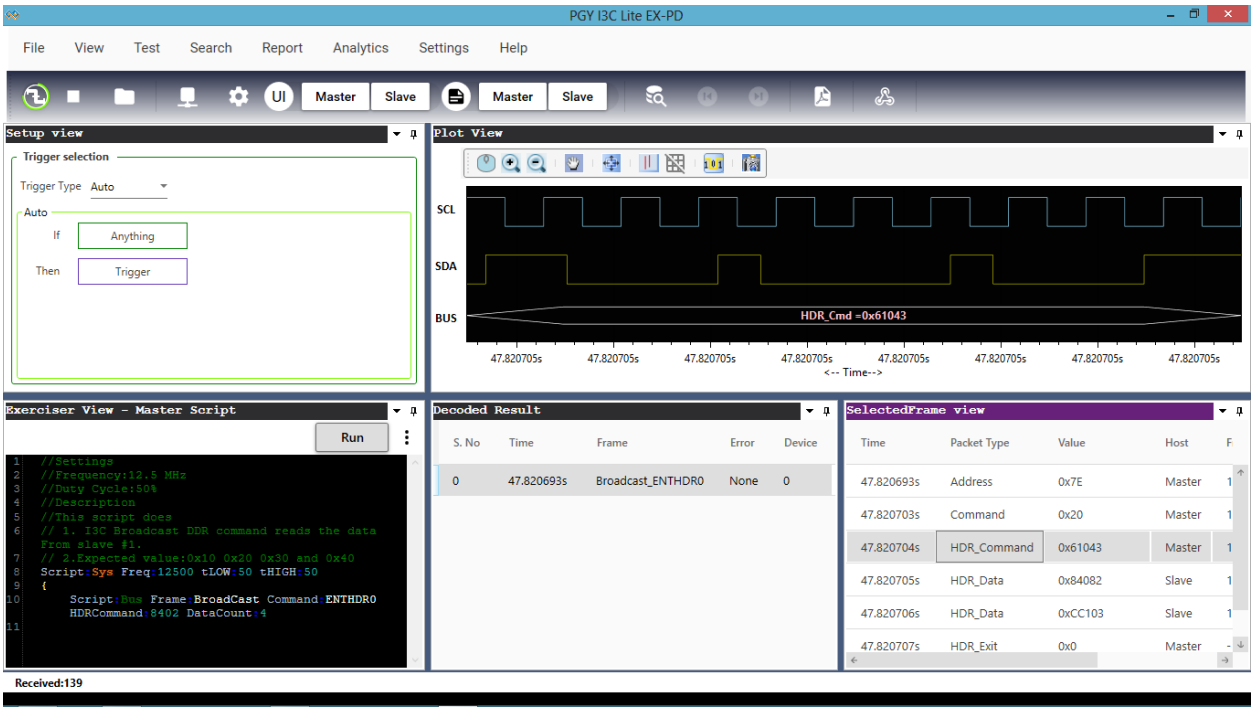
//Description

//This script does

// 1. I3C Broadcast DDR command reads the data From slave #1.

// 2.Expected value:0x10 0x20 0x30 and 0x40

```
Script:Sys Freq:12500 tLOW:50 tHIGH:50
{
      Script:Bus Frame:BroadCast Command:ENTHDR0 HDRCommand:8402
DataCount:4
}
```



Broadcast HDR DDR WRITE READ (HDRWR_RD.txt)

//Settings

//Frequency:500 KHz

//Duty Cycle:50%

//Description

//This script does

// 1. I3C Broadcast DDR command writes the data to slave #1 with values 0x10,0x20,0x30,0x40

// 2.Expected value:0x1020 and 0x3040

// 3. HDR restart

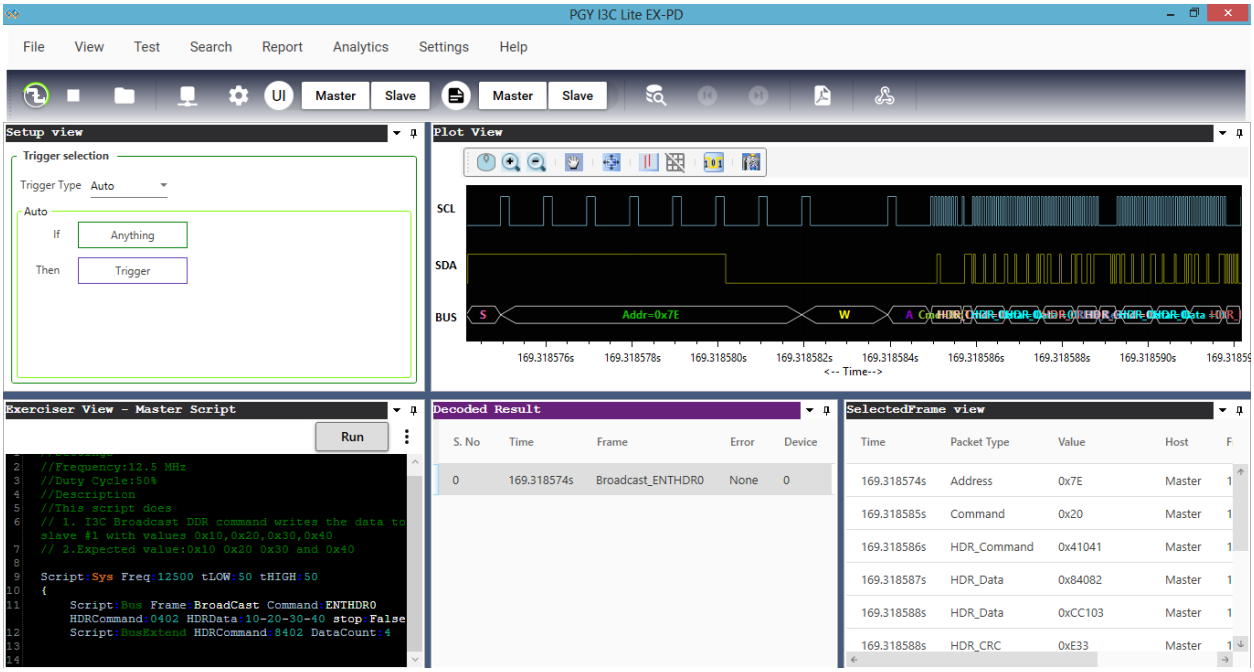
// 4.I3C Broadcast DDR command reads the data from slave #1

// 5.Expected Result:0x10 0x20 0x30 and 0x40

//6. HDR Exit

```
Script:Sys Freq:500 tLOW:50 tHIGH:50


      Script:Bus Frame:BroadCast Command:ENTHDR0 HDRCommand:0402
HDRData:10-20-30-40 Stop:False
      Script:BusExtend HDRCommand:8402 DataCount:4
}
```



Broadcast HDR TSP WRITE (HDRTSPWR.txt)

//Settings

//Frequency:12.5 MHz

//Duty Cycle:50%

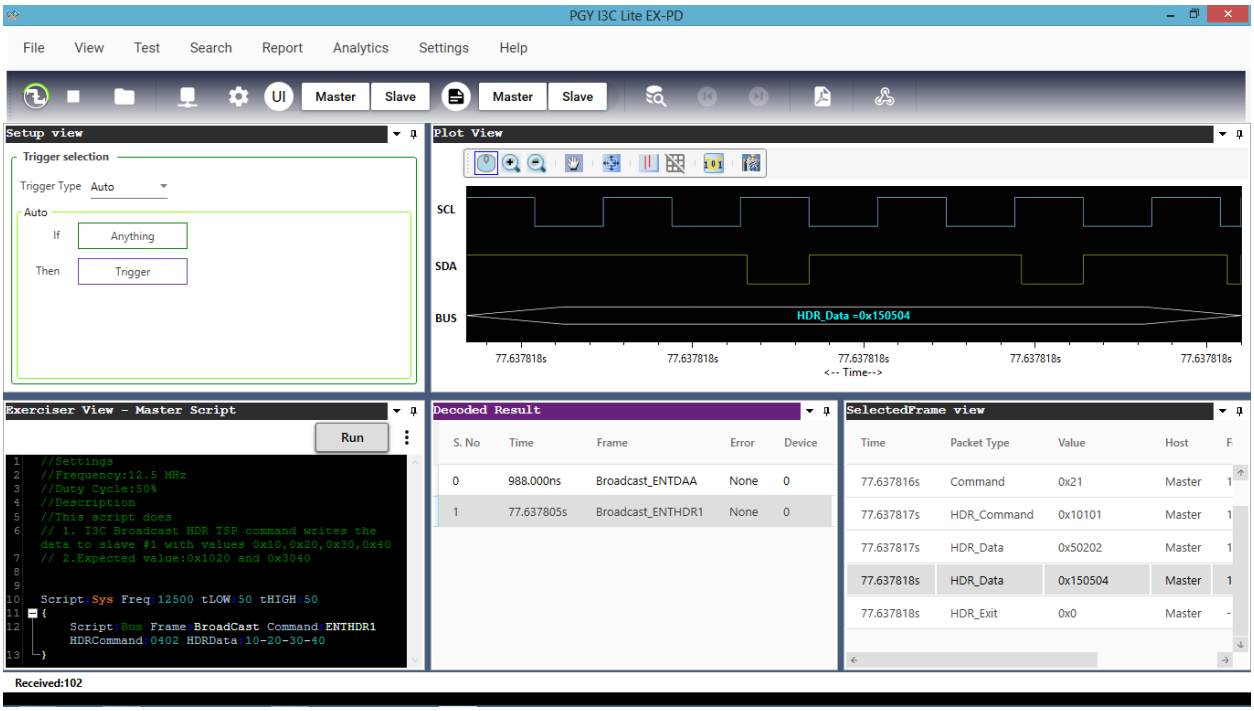
//Description

//This script does

// 1. I3C Broadcast TSP command writes the data to slave #1 with values 0x10,0x20,0x30,0x40

// 2.Expected value:0x10 0x20 0x30 and 0x40 and 0x5060 and 0x7080

```
Script:Sys Freq:12500 tLOW:50 tHIGH:50
{
Script:Bus Frame:BroadCast Command:ENTHDR1 HDRCommand:0402 HDRData:10-20-30-40-50-60-70-80
}
```



Broadcast HDR TSP READ (HDRTSPRD.txt)

//Settings

//Frequency:12.5 MHz

//Duty Cycle:50%

//TCO (Clock to Data Delay) : 0 ns

//TSU (Stop and Restart Setup time) : 20 ns

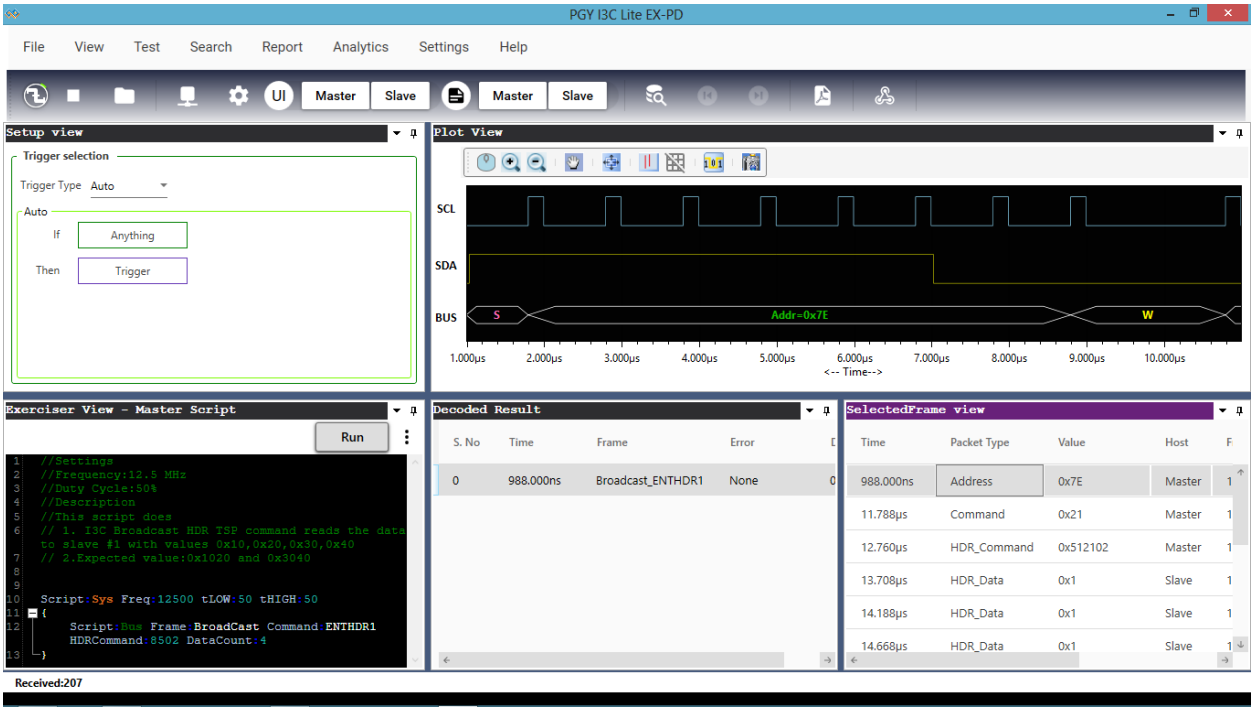
//Description

//This script does

// 1. I3C Broadcast TSP command reads the data from slave #1.

// 2.Expected value:0x10 0x20 0x30 and 0x40 and 0x5060 and 0x7080

```
Script:Sys Freq:12500 tLOW:50 tHIGH:50
{
      Script:Bus Frame:BroadCast Command:ENTHDR1 HDRCommand:8502
DataCount:4
}
```



BUFFER ACCESS (BUFFERACCESS.txt)

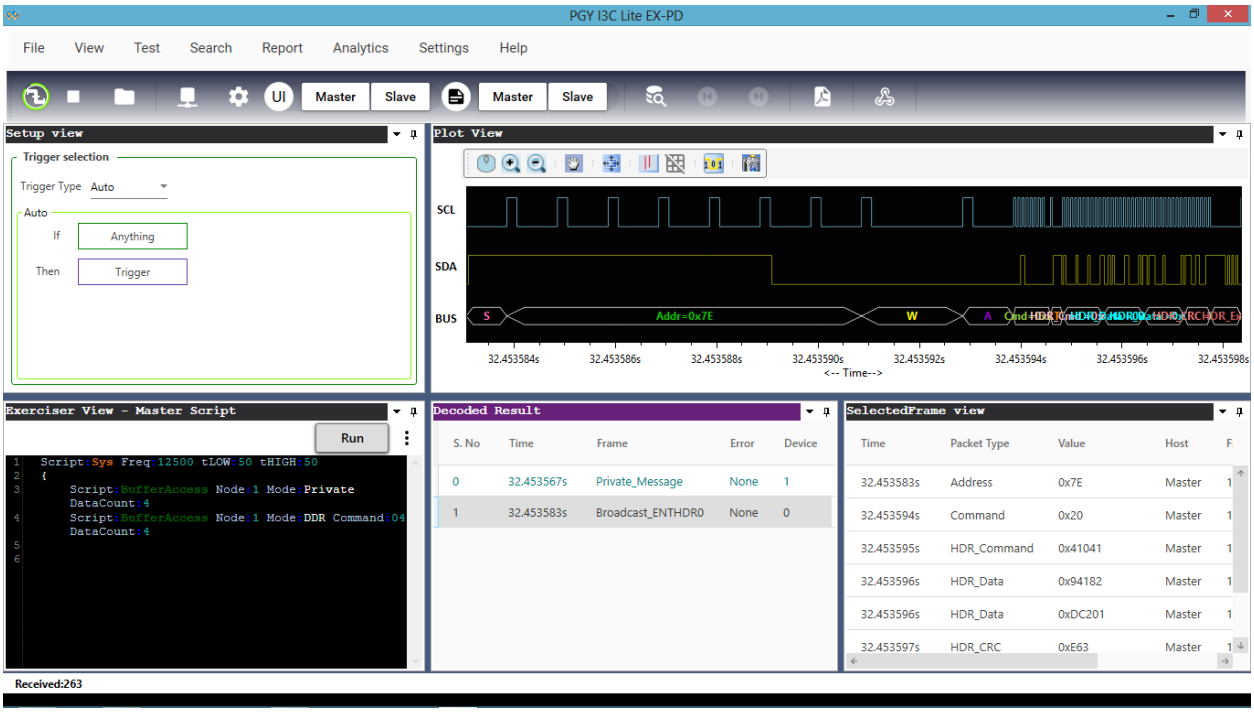
//Description

//This script does

// 1. Data will be read from the FIFO buffer.(Mode:Private)

// 2. Data will be read from the FIFO buffer.(Mode:DDR)

```
Script:Sys Freq:12500 tLOW:50 tHIGH:50 tCO:0 tSU:0
Script:BufferAccess Node:1 Mode:Private DataCount:4
Script:BufferAccess Node:1 Mode:DDR Command:04 DataCount:4
```



Enable /Disable for Inband Interrupt

//This script does

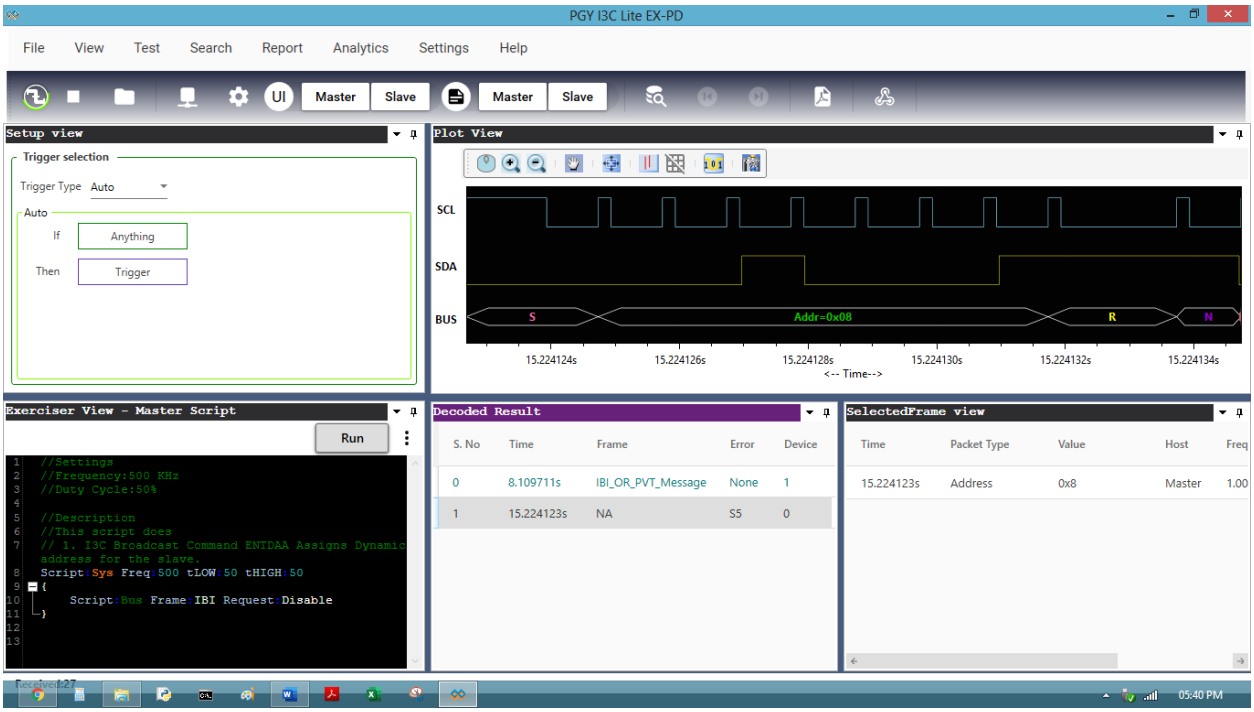
//Master Enables the I3C IBI (inband interrupt) request.

```
Script:BUS Frame:IBI Request:Enable
```

//This script does

//Master Disables the I3C IBI (inband interrupt) request.

```
Script:BUS Frame:IBI Request:Disable
```



Enable/Disable For Hotjoin

//This script does

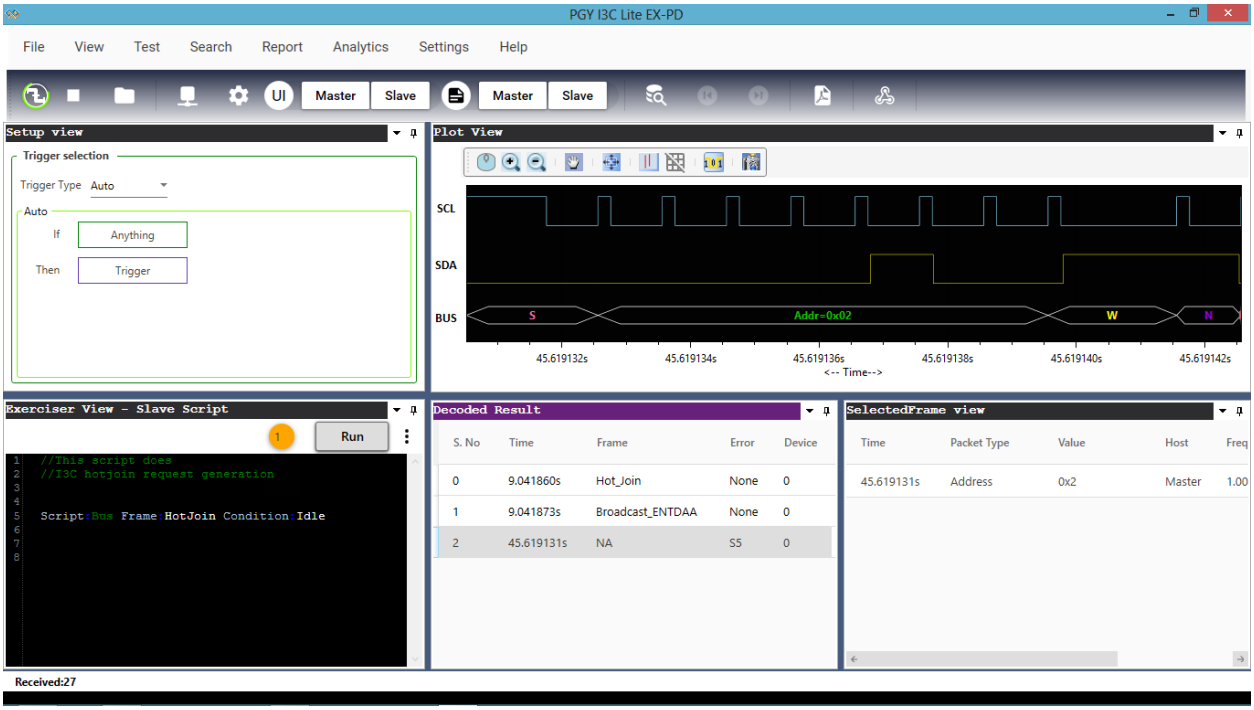
//Master Enables the I3C Hotjoin request.

```
Script:BUS Frame:Hotjoin Request:Enable
```

//This script does

//Master Disables the I3C Hotjoin request.

```
Script:BUS Frame:Hotjoin Request:Disable
```



Enable/Disable For Mastership Request

//This script does

//Master Enables the I3C Mastership request.

```
Script:BUS Frame:MastershipReq Request:Enable
```

//This script does

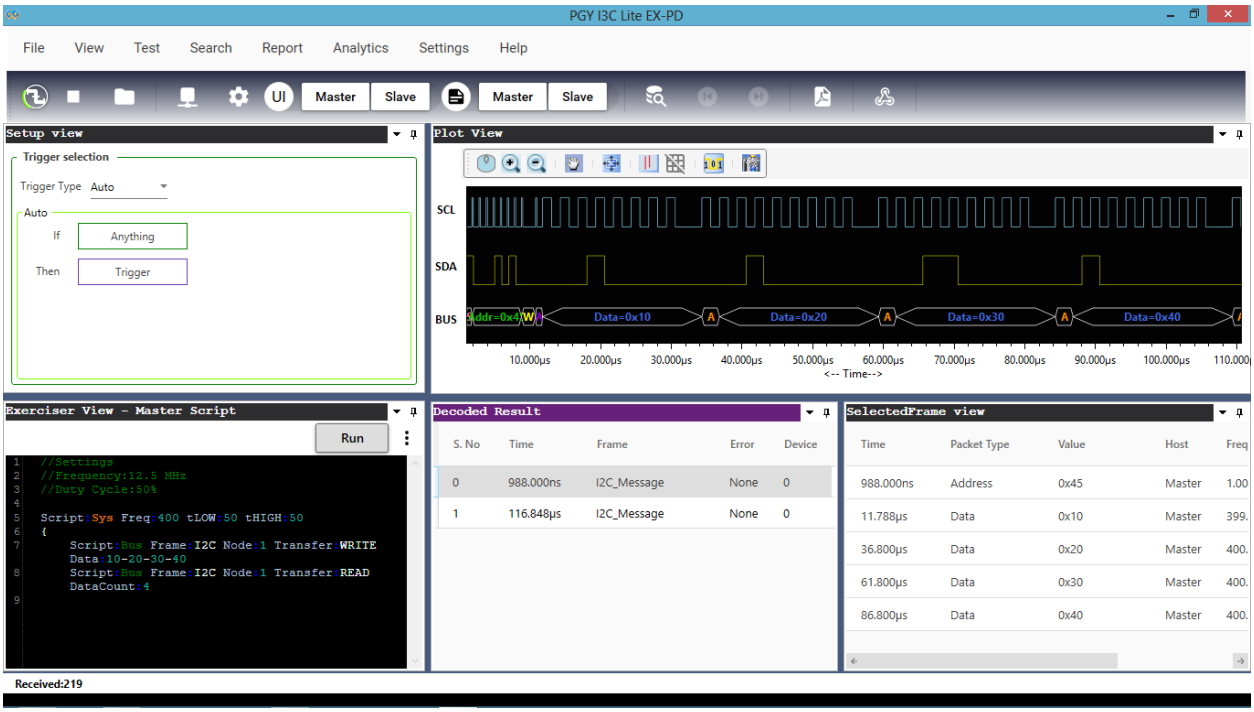
//Master Disables the I3C Mastership request.

```
Script:BUS Frame: MastershipReq Request:Disable
```

I2C WRITE READ (I2C_WR_RD .txt)

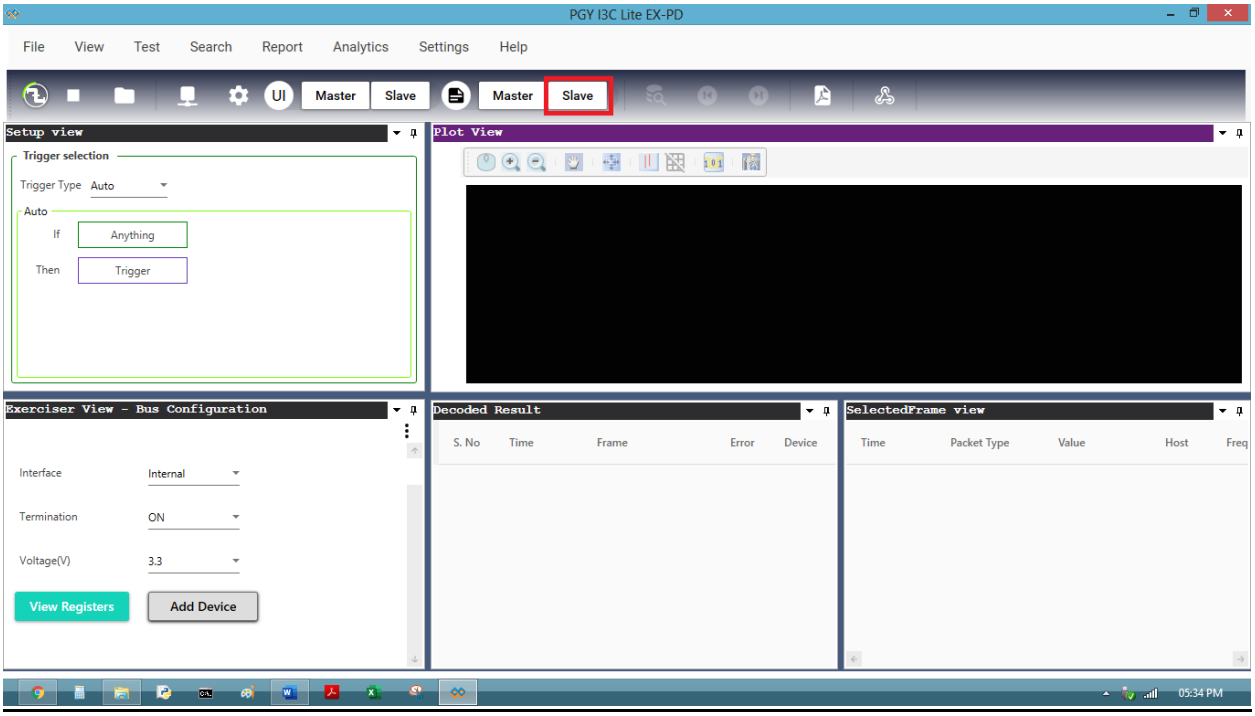
//Settings
//Frequency:500 KHz
//Duty Cycle:50%
//Description
//This script does
// 1. I2C Write command Writes the data to slave #1 with value 0x1,0x2,0x3,0x4 and 0x5
// 2. Expected value 0x10,0x20,0x30,0x40 and 0x50
// 3. I2C read command reads the data from slave #1
// 4. Expected value 0x10,0x20,0x30,0x40 and 0x50

```
Script:Sys Freq:400 tLOW:50 tHIGH:50
{
      Script:Bus Frame:I2C Node:1 Transfer:WRITE Data:10-20-30-40
      Script:Bus Frame:I2C Node:1 Transfer:READ DataCount:4
}
```

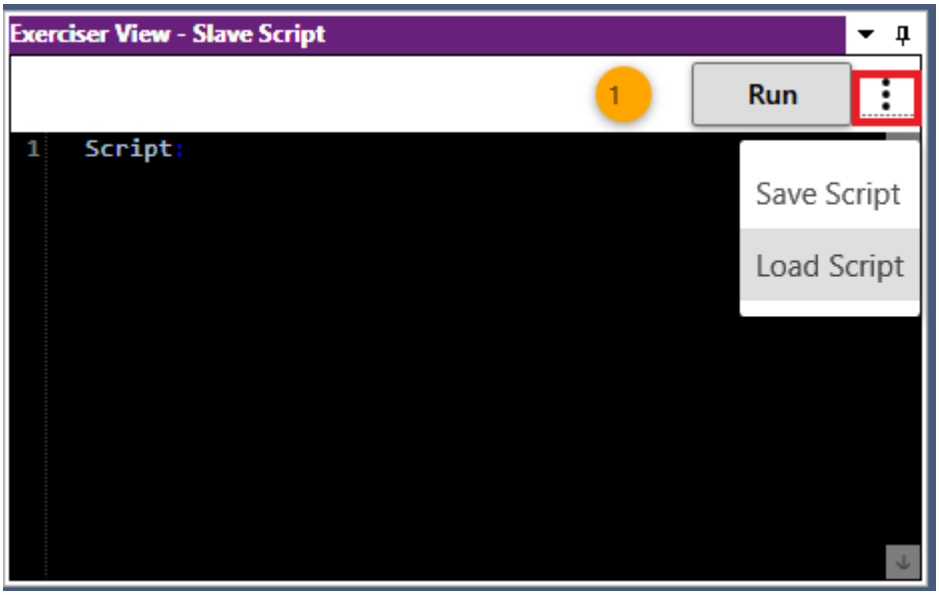


Using Slave Scripts

1. Click on the *Slave script* button.

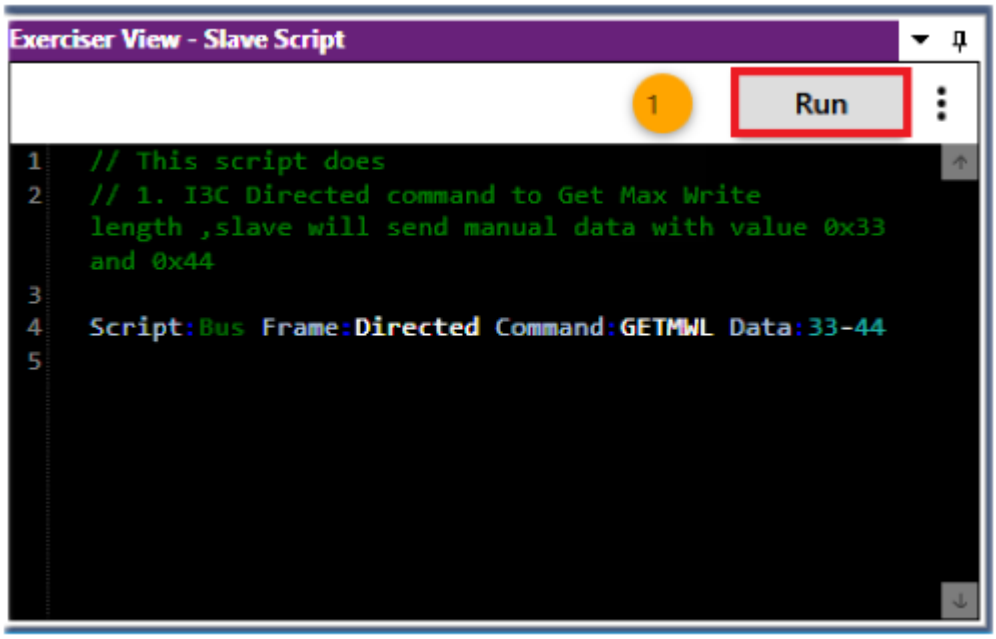


2. Click on the highlighted button and then select the *Load Script* option to load DIR_RD script to get response from slave 1 for Directed Read command.



3. From the options select the *DIRRD.txt* option to load the script.

4. Click on *Run* button to send the script loaded on the script window.



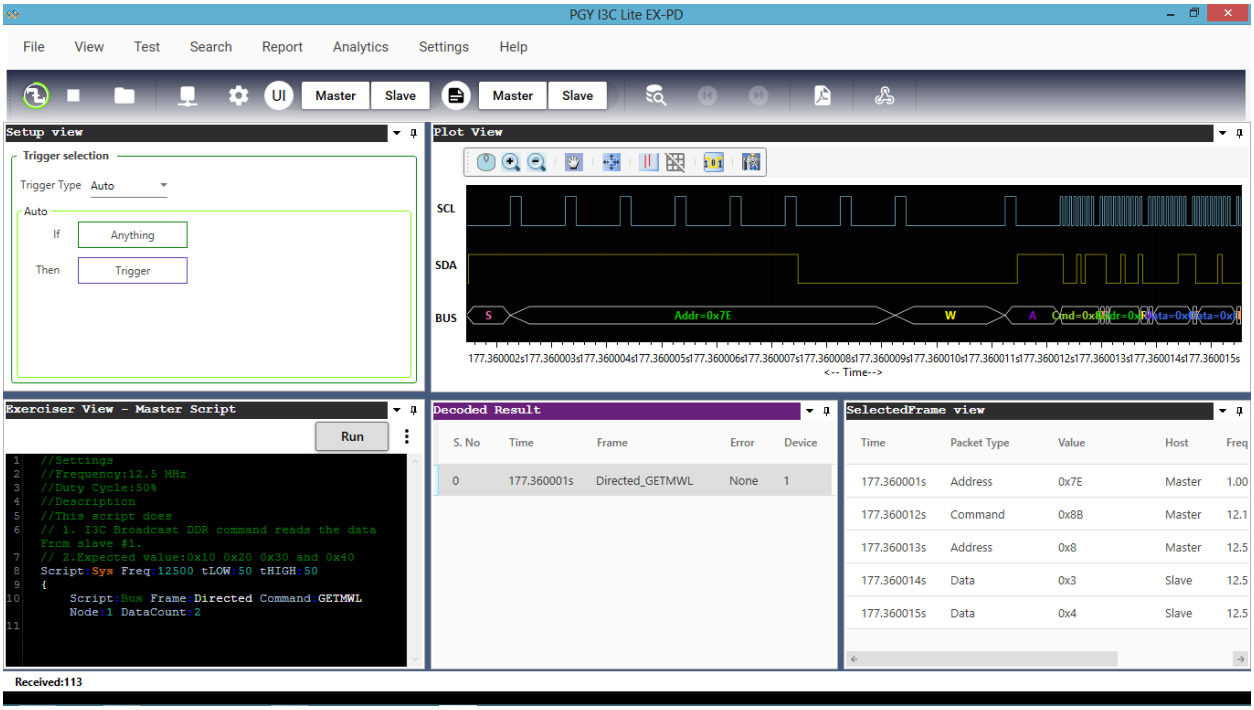
5. Repeat steps 1 through 4 for other preloaded scripts.

Directed Read(DIR_RD.txt)

// This script does

// 1. I3C Directed command to Get Max Write length ,slave will send custom data with value 0x33 and 0x44

```
Script:Bus Frame:Directed Command:GETMWL Data:33-44
```

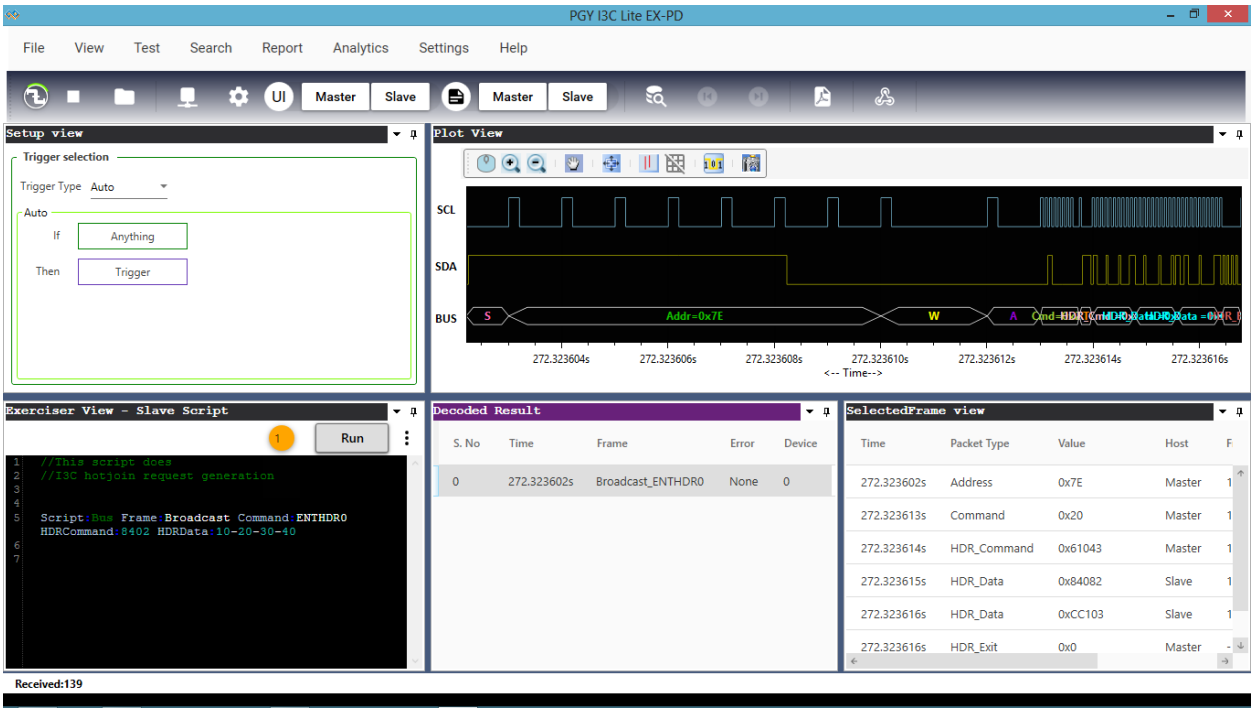


HDR DDR WRITE(HDR_WR.txt)

// This script does

// 1. I3C HDR DDR write ,slave will send custom data with value 0x11 0x22 0x33 and 0x44

```
Script:bus Frame:Broadcast Command:ENTHDR0 HDRCommand:8502 HDRData:11-22-33-44
```

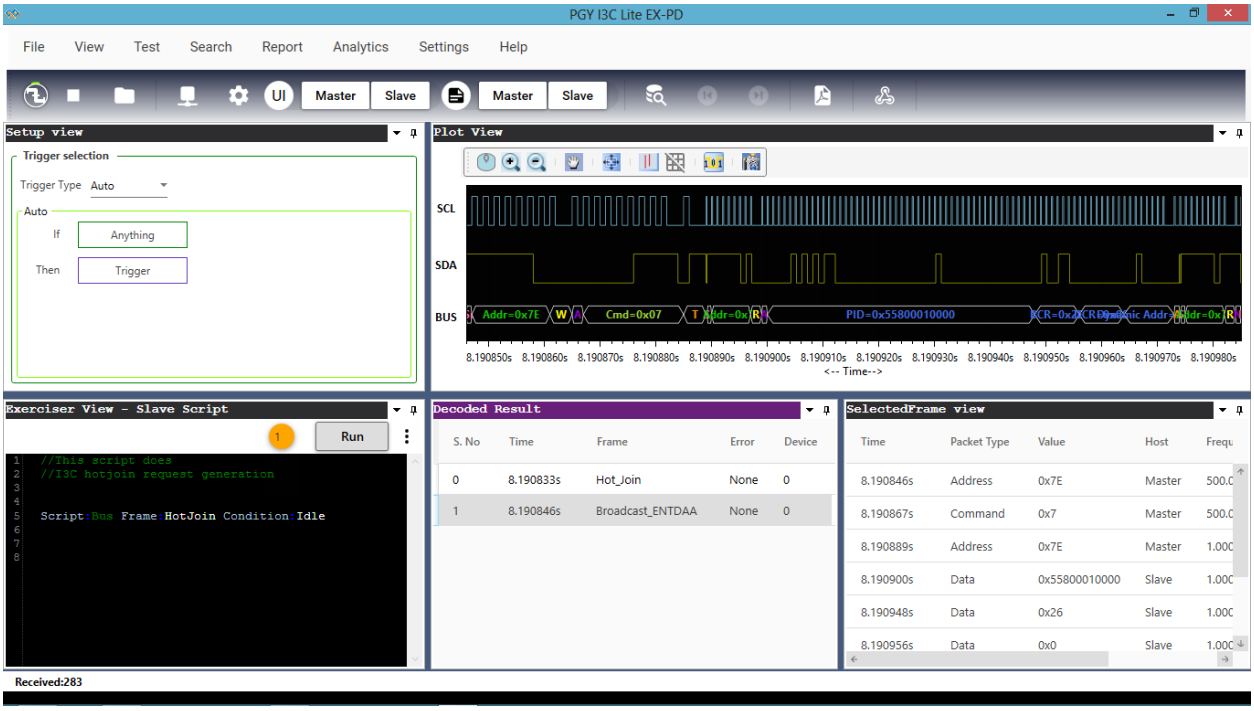


Hotjoin (Hotjoin.txt)

//This script does

//I3C hotjoin request generation

```
Script:Bus Frame:HotJoin Condition:Idle
```

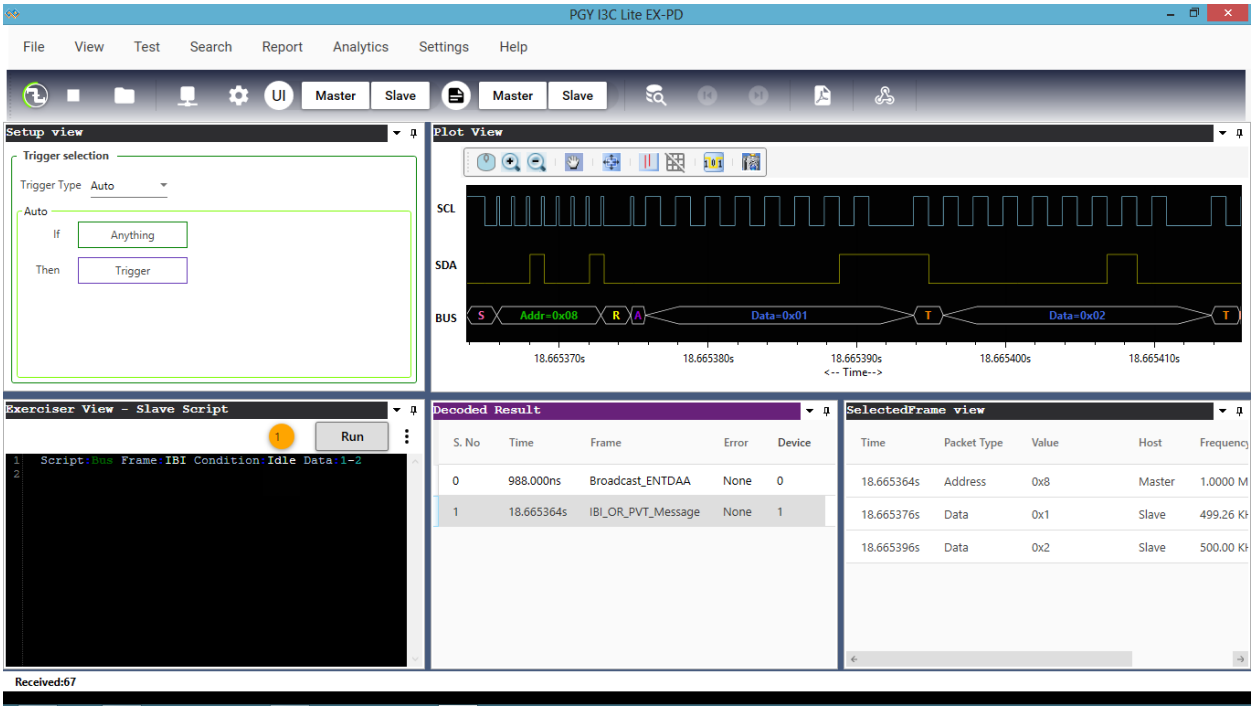


Inband interrupt (IBI.txt)

//This script does

//I3C IBI (inband interrupt) request generation with data 0x01 0x02

```
Script:Bus Frame:IBI Condition:Idle Data:01-02
```



Mastership request(Mastership_req.txt)

//This script does

//I3C MastershipReq request generation

```
Script:bus Frame:MastershipReq Condition:Idle
```