



USER GUIDE

SV2C In-Field Calibration

Standard Operating Procedures

SV2C Personalized SerDes Tester

C SERIES







Table of Contents

Introduction	3
Overview	
Hardware Requirements	4
Firmware / Software / Test Folder Requirements	5
Calibration / Validation Use Cases	6
Step-By-Step Calibration Procedures	8
RX Threshold Voltage Calibration / Validation	8
RX Common Mode Voltage Calibration	
RX Equalization (CTLE) Calibration	
Internal Frequency Validation	43
Recapitulation of Calibration and Validation Procedures	



Introduction

OVERVIEW

This document provides the full set of instructions for performing in-field calibrations of an SV2C Personalized SerDes Tester within a customer production environment. There are four types of calibrations and validations which may be performed, as listed below:

- 1. SV2C RX Threshold Voltage Calibration
- 2. SV2C RX Common Mode Voltage Calibration
- 3. SV2C RX Equalization (CTLE) Calibration
- 4. SV2C Internal Frequency Validation

There are currently three different SV2C firmware releases for use with the calibrations listed above. This document will specify the required combinations of firmware / software / test procedures for each type of in-field calibration.

There are specific use-cases for each of the above calibrations and validations. For example, a particular calibration may be required when the SV2C is being upgraded to a more recent firmware version or may be required when the SV2C is reverted to a previous firmware version. This document will provide the full use-case requirements for each type of calibration and validation.

The main body of this document will provide full, step-by-step instructions for executing each type of infield calibration.

Note: to improve the ease of reading in this document, the following three abbreviations for will frequently be used to designate the three versions of firmware:

FWIESPSV2C03A009 will be abbreviated as "FW09" FWIESPSV2C03A011 will be abbreviated to "FW11" FWIESPSV2C03A016 will be abbreviated to "FW16"



HARDWARE REQUIREMENTS

- (QTY = 1) SV2C Personalized SerDes Tester
- (QTY = 1) 12 V power supply units (manufacturer part number CUI SDI65-12-UDC-P5)
- (QTY = 2) MXP to SMA cable assemblies (manufacturer part number: Huber and Suhner MF53/2x8A_21MXP/21SMA/152).
- (QTY = 16) SMA 50 ohm terminator (example part number Mini-Circuits ANNE-50+)
- (QTY = 16) SMA DC block (example part number Mini-Circuits BLK-18-S+)
- (QTY = 16) SMA adapters, male pin to male pin (example part number Amphenol RF 132168)
- (QTY = 1) USB2 cable for connection between SV2C and a PC
- (QTY = 1) Personal computer, with Windows 7, 8, or 10 installed.

There are two TX to RX loopback configurations used for the procedures in this document: (a) fully differential configuration and (b) single-ended configuration. Connection diagrams for each are shown below in Figure 1. Refer to example part numbers given above.





FIRMWARE / SOFTWARE / TEST FOLDER REQUIREMENTS

There are currently four production calibration and/or validations which may be performed in the field. The required pairing of firmware, software, and test procedure folder combinations for each calibration type are provided below in Table 1 to Table 4.

TABLE 1: RX THREHSOLD VOLTAGE CALIBRATION

FIRMWARE RELEASE	SOFWARE RELEASE	TEST PROCEDURE FOLDER
FWIESPSV2C03A016	IESP 21.1.0	rxThresholdCal_FW16
FWIESPSV2C03A011	IESP 3.6.83	rxThresholdCal_FW09
FWIESPSV2C03A009	IESP 3.6.74	rxThresholdCal_FW09

TABLE 2: RX COMMON MODE VOLTAGE CALIBRATION

FIRMWARE	SOFWARE	TEST PROCEDURE FOLDER
	REEEASE	
FWIESPSV2C03A016	IESP 21.1.0	SV2RxCommonModeCalibration_HiSilicon_21p1p0
FWIESPSV2C03A011	IESP 3.6.83	SV2RxCommonModeCalibration_HiSilicon3p6p74_3p6p83
FWIESPSV2C03A009	IESP 3.6.74	SV2RxCommonModeCalibration_HiSilicon_3p6p74_3p6p83

TABLE 3: RX EQUALIZATION (CTLE) CALIBRATION

FIRMWARE RELEASE	SOFWARE RELEASE	TEST PROCEDURE FOLDER
FWIESPSV2C03A016	IESP 21.1.0	HiSiliconCTLECalibration_21p1p0
FWIESPSV2C03A011	IESP 3.6.83	HiSiliconCTLECalibration_3p6p74_3p6p83
FWIESPSV2C03A009	IESP 3.6.74	HiSiliconCTLECalibration_3p6p74_3p6p83

TABLE 4: INTERNAL FREQUENCY VALIDATION

FIRMWARE RELEASE	SOFWARE RELEASE	CORRESPONDING TEST SCRIPT
FWIESPSV2C03A016	IESP 21.1.0	Test_20201217-Python-freqMeasurement.py
FWIESPSV2C03A011	IESP 3.6.83	Not supported
FWIESPSV2C03A009	IESP 3.6.74	Not supported



CALIBRATION / VALIDATION USE CASES

There are currently four in-field calibration and validations which may be performed. The use cases for each are described below.

1. RX THRESHOLD VOLTAGE CALIBRATION / VALIDATION

The RX threshold voltage calibration and validation procedures in this test folder need to be performed on an SV2C in the following cases:

- Each time the firmware is upgraded between FW09/FW11 and FW16
- Each time the firmware is reverted from FW16 to FW09/FW11

Note that this procedure does not need to be performed when switching between FW09 and FW11.

The RX threshold voltage calibration and validation procedures must be performed in either of "Differential" or "Single-ended" mode. For example, if the SV2C is being calibrated for the "Differential" test case, then the user must execute the following test procedures:

- 1. rxThresholdCal_SV2,
- 2. sv2ProgramFlash, and
- 3. rxThresholdVal_SV2

and if the SV2C is being calibrated for the "Single-ended" test case, then the user must execute the following test procedures:

- 1. rxThresholdCal_SV2_singleEnded,
- 2. sv2ProgramFlash, and
- 3. rxThresholdVal_SV2_singleEnded

The differential calibration is not valid for single-ended operation, nor is the single-ended calibration valid for differential operation. Therefore, the calibration and validation procedures in this test folder need to be performed on an SV2C in the following cases:

- Each time SV2C operation is switched from differential to single-ended usage
- Each time SV2C operation is switched from single-ended to differential usage

Step-by-step procedures for performing RX threshold calibrations are found starting on page 8 of this document.



2. RX COMMON MODE VOLTAGE CALIBRATION

First, note that the common mode voltage calibration is only required if the SV2C is used for singleended operation. For differential operation, the default common mode voltage value of "15" should always be used.

The common mode voltage calibration procedure needs to be performed on an SV2C in the following cases:

- Each time the firmware is upgraded between FW09/FW11 and FW16
- Each time the firmware is reverted from FW16 to FW09/FW11

Note that the common mode voltage calibration needs to be performed at least once per product development cycle (ie, when the SV2C is configured to operate with a new type of DUT device). This will ensure the optimal setting of the SV2C common mode level for the new DUT.

Step-by-step procedures for performing RX common mode voltage calibrations are found starting on page 29 of this document.

3. RX EQUALIZATION (CTLE) CALIBRATION

The RX Equalization calibration procedure must be to be performed each time the SV2C module is powered up. This calibration may be performed for either the differential or single-ended test case.

Step-by-step procedures for performing RX common mode voltage calibrations are found starting on page 34 of this document.

4. INTERNAL FREQUENCY VALIDATION

The internal frequency validation check needs to be performed periodically. Introspect recommends that this validation be performed at the start of each new wafer under test. This internal frequency validation can only be performed when using FW16.

Step-by-step procedures for performing RX common mode voltage calibrations are found starting on page 43 of this document.



Step-By-Step Calibration Procedures

RX THRESHOLD VOLTAGE CALIBRATION / VALIDATION

This section of the document will provide the procedures the following four cases of RX threshold voltage calibration:

Case 1: Differential RX threshold calibration, firmware = FW16

Case 2: Single-ended RX threshold calibration, firmware = FW16

Case 3: Differential RX threshold calibration, firmware = FW09 or FW11

Case 4: Single-ended RX threshold calibration, firmware = FW09 or FW11

Please refer to Table 1 for the valid firmware / software / test procedure folder combinations for this calibration and refer to the step-by-step procedures below.

CASE 1: PROCEDURES FOR DIFFERENTIAL THRESHOLD CALIBRATION WITH FWIESPSV2C03A016

1. Ensure that you are using the firmware / software / test procedure folder combination:

Firmware = FW16 Software = IESP 21.1.0 Test Folder = rxThresholdCal_FW16

2. Load the default RX threshold calibration file. To do this, from the GUI version listed above, open the test procedure "Sv2ProgramFlash". Under "calOptions" change the following settings to enter the module serial number (printed on the back of the module) and point the test procedure to the default calibration text files provided by Introspect Technology:

calOption.serialNumber = <module serial number> calOption.rxThreshFilePath = C:\ **<your full path here>** \defaultSv2ThresholdData_32G.txt calOption.jitterFilePath = C:\ **<your full path here>** \SV2C03_jitterData.txt



File Edit IESP/SV2C_8C32G Wi	zards ControlPanels Too	ls Results Help			
Params	Log *	F	lesults		
Components		calOptions prop	erties (class: DataRecord	i)	
calCptions globalCockConfig efclockaConfig wrtePbatGoTabah wrtePbatGoTaBah wrteFkacGDData wrteFkCalData wrteFkCThreehCalData	rxLinChannels serialNumber rx.ThreshiFlePdh jitterFilePath methods		[1, 2, 3, 4, 5, 6, 7, 8] <svzc num<br="" serial="">C:\xyour path>\d C:\xyour path>\d [_customInit]</svzc>	ber> lefaultSv2ThresholdData_3 V2C03_jitterData.txt	2G.txt
Add Remove Config	serialNumber				
Add Remove Config Test Procedure write/DataToRiash write 1 # / EXPECTED FIRMWARE F	serial Number HeaderCalData writeJitterCalDa WIESPSV2C03A016	ta writeFixThreshCalD/	ata		

- 3. Press "Run" from the GUI and wait for the test procedure to finish.
- 4. Very important: when the test procedure has completed, power cycle the SV2C module.
- 5. **Start the RX threshold calibration.** To do this, configure the MXP cables so that all TX signals are looped back to RX signals using the differential configuration (see Figure 1(a) on page 4 for connection details).
- 6. Open the test procedure "rxThreshold**Cal_**SV2". Under "calOptions" change the following settings to enter the module serial number (printed on the back of the module).

calOption.serialNumber = < module serial number>



🔯 Intr	ospect	ESP (v 21.1.0p) - rxTh	resholdCa	I_SV2 (SV2C_8C32G	i)				-	- 0	×
File	Edit	IESP/SV2C_8C32G	Wizards	ControlPanels	Tools	Results	Help				
		Params		Log *			Res	ults			
		Components				calOpti	ons propertie	es (class: DataRe	cord)		
calOpti comput genera globalC pattern plotCre plotCre plotCre plotCre refCloc rxChan txChan	teMean tePlots asureme ClockCo Sync1 ator1 ator2 ator3 atorBas ksConfi nelList1 nelList1	write A write T Intis At Amplitude infig ic 1 g	verag ransfe	moduleName dataRate targetLevels saveEyeScans initialPolynomialEstii commonModeLevel methods	mate Is			<sv2c serial<br="">26000.0 [150.0, 200.0, 29 False [3.4688e-12, -8.6 [0, 1, 2, 3, 4, 5, 6 [_customInit]</sv2c>	number> 50.0, 300.0, 350.0, 400 3346e-11, -2.9656e-06, 5, 7, 8, 9, 10, 11, 12, 1;	.0, 500.0, 600 .4,6669e-05, 3, 14, 15]	0.0] 1.6827,
< Ad	d [Remove Confi	> m	oduleName					-		
1	<pre>''ocedu from SvtBe iesp avera form iesp. iesp. iesp.</pre>	re computeMean ge dftm.comm.comm rrtScan.autoCre = IESP.getInst geVoltageDict 'actor = iesp.g setLimitMinimu setLimitMaximu setLimitStep('	neratePlots AUtils i atePrev ance() = dict m('rxCompa rxCompa	<pre>getMeasurements/F Import * riews = True () Factor() omparatorThre aratorThresho</pre>	shold' shold' ld', l	e writeAve , -203. , 203.2 6)	2)	es writeTransfer	Functions		~
0						Run					

- 7. Press "Run" from the GUI. Ensure that the software connects and proceeds without error before leaving it to execute. Test completion takes approximately 20 minutes.
- 8. When the test procedure is complete it is important to rename results folder from the automatically generated name to the SV2C serial number. The completed test procedure will appear as shown in the figure below.



🔯 Int	rospect	ESP (v 21.1.0p) - rxTh	resholdCal_	SV2 (SV2C_8C32G	i)					-		×
File	Edit	IESP/SV2C_8C32G	Wizards	ControlPanels	Tools	Results	Help					
		Params		Log			Results					
		Defete Result		Notes SV2C20010	usPolyFit	Average	fotageMeas	calCoefficients_SV	20	DataRateTree	dPlot	
0						Run						

9. Load the RX threshold calibration data onto the SV2C. To do this, reopen the test procedure "Sv2ProgramFlash". Under "calOptions" change the following settings to enter the module serial number (printed on the back of the module) and point the test procedure to the default calibration text files provided by Introspect Technology:

```
calOption.serialNumber = <module serial number>
calOption.rxThreshFilePath = C:\ <your full path here> \rxPerPhaseCal_SV2\Results\ [serial
number] calCoefficients_[serial number]\calCoefficients_[serial number].txt
calOption.jitterFilePath = C:\ <your full path here> \SV2C03_jitterData.txt
```



Eile	rospect Edit	ESP (v 21.1.0p) - Sv2P IESP/SV2C 8C32G	rogramFla: Wizards	sh (SV2C_8C32G) ControlPanels	Tools	Results	Help		-		×
		Params		Log *			Resu	ults			
		Components				calOpti	ons propertie	es (class: DataRecord)			
calOpt global refOoc writeD writeH writeR	ions ClockCo xksConfi ataToFla eaderCa terCaID xThresh	nfig g such DData sta CalData		xLinChannels serialNumber wThreahTilePath jtterFiePath methods				[1.2.3.4.5.6.7.8] <sv2c number="" serial=""> <c: <pre="">csubt>vxPerPh C:\spour path>\SV2C03_ [_custominit]</c:></sv2c>	aseCal_SV2\\ jitterData.txt	Results [\]	v (ser
Ad	ld	Remove Confid	rx1	hreshFilePath							
Test F 1 2 3 4 5	Procedu #! EX iesp impor impor	re writeDataToFlash IFECTED FIRMWAR = getIespInsta t dftm.comm.co t dftm.util as	wmteHeade E FWIES nce() mmUtils dftUti	CalData writeJitter PSV2C03A016 as dftComm 1	CalData	writeRxThr	eshCalData				< >
\sim						Pue					

- 10. Press "Run" from the GUI and wait for the test procedure to finish.
- 11. Very important: when the test procedure has completed, power cycle the SV2C module, and close the GUI before proceeding.
- 12. **Start the RX threshold validation.** To do this, reopen the GUI and open the test procedure "rxThreshold**Val_**SV2". Under "calOptions" change the following settings to enter the module serial number (printed on the back of the module)

calOption.serialNumber = <module serial number>

This is as shown in the figure on the following page.

13. Press "Run" from the GUI. The test completion takes approximately 2 minutes.



Eile Edit IESP/SV2C 8C32G	esholdVal_SV2 (SV2C_8C32G)	- ×
Params	lon*	Results
Components	calOptions pro	poerties (class: DataRecord)
calQuion calQuion ompdetMan generateRisa generateRisa generateRisa generateRisa generateRisa generateRisa ploCeator2 ploCeator2 ploCeator3 ploC	moduleName dataRates targetLevels aaveEyeScans initalPohomailEtitate commonModeLevels methods	SV2C20010001 [25000.0] [1500, 200, 250.0, 300.0, 350.0, 400.0, 500.0, 520.0] False [3.4638-12, 2.86346e-11, -2.9656e-06, 4.6669e-05, 1.6827 [15] [150] [_exustomInst]
	moduleName	
	a	
Test Procedure	Parts Dista ant Managements (M. Amerika da antika Amerika M	(h
Test Procedure computeMean get 1 from dftm.comm.comm 2 SvtBertScan.autoCre 3 iesp = IESP.getInst 4 averageVoltageDict 5	<pre>wentePiots getMeasurementsAtAmpltude wnteAverageV Utils import * atcPreviews = True ance() * dict()</pre>	lotages

14. When the validation is complete it is important that you rename results folder from the automatically generated name to the SV2C serial number. The completed test procedure is as shown in the figure below.

		cor (vennop) ivin	reshora rai_		·									
ile	Edit	IESP/SV2C_8C32G	Wizards	ControlPanels	Tools	Results	Help							
		Params		Log *			Res	its						
1202	001000			Average Tran	sferFunc.	Deta R	verage Transformed 200 	Data	Across Con	polyFit	le Levels	(at 26000	Mbps), Ch	
		Delete Result		Notes SV2C20010	001									

15. Verify that the transfer functions are linear (check under the "DataRate Trend Plot" icon) similar to the case shown above. This completes the SV2C RX threshold calibration and validation.



CASE 2: PROCEDURES FOR SINGLE-ENDED THRESHOLD CALIBRATION WITH

FWIESPSV2C03A016

1. Ensure that you are using the firmware / software / test procedure folder combination:

Firmware = FW16 Software = IESP 21.1.0 Test Folder = rxThresholdCal_FW16

2. Load the default RX threshold calibration file. To do this, from the GUI version listed above, open the test procedure "Sv2ProgramFlash". Under "calOptions" change the following settings to enter the module serial number (printed on the back of the module) and point the test procedure to the default calibration text files provided by Introspect Technology:

calOption.serialNumber = <module serial number> calOption.rxThreshFilePath = C:\ **<your full path here>** \defaultSv2ThresholdData_32G.txt calOption.jitterFilePath = C:\ **<your full path here>** \SV2C03_jitterData.txt

🚾 Int File	rospect Edit	ESP (v 21.1.0p) - Sv2P IESP/SV2C_8C32G	rogramFla Wizards	sh (SV2C_8C32G) ControlPanels	Tools	Results	Help			-		×
		Params		Log *			Res	ults				
		Components				calOpti	ons properti	es (class: DataRi	ecord)			
calOpt global refClox wrteD wrteH wrteJi wrteR	tions ClockCo cksConfi lata To Ri leaderCa tterCalD x Thresh	nfig 9 9 9 9 9 1 0 8 4 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9		rxLinChannels serialNumber nr.ThreahFilePath jitterFilePath methods				[1, 2, 3, 4, 5, 6, <sv2c serial<br="">C:\<your pat<br="">C:\<your pat<br="">[_customInit]</your></your></sv2c>	7.8] number> h>\defaultSv h>\SV2C03_j	2Th resh old jitterData.tx	Data_32 t	G.txt
Ac	dd	Remove Config	ser	rial Number								
Test 1 2 3 4 5	Procedu #! E2 iesp impor impor	<pre>wwteDataToHash PECTED FIRMWAR = getIespInsta t dftm.comm.co t dftm.util as</pre>	writeHeade E FWIES nce() mmUtils dftUti	rCalData writeJitter PSV2C03A016 as dftComm 1	CalData	writeRxThr	eshCalData	1				~ ~
5	impor	t dftm.util as	dftUti	1		Run						~

- 3. Press "Run" from the GUI and wait for the test procedure to finish.
- 4. Very important: when the test procedure has completed, power cycle the SV2C module.



- 5. **Start the RX threshold calibration.** To do this, configure the MXP cables so that all TX signals are looped back to RX signals using the single-ended configuration (see Figure 1(b) on page 4 for connection details).
- 6. Open the test procedure "rxThreshold**Cal_**SV2**_singleEnded**". Under "calOptions" change the following settings to enter the module serial number (printed on the back of the module).

calOption.serialNumber = <module serial number>

🔯 Introspect ESP (v 21.1.0p) - rxThr	esholdCal_	SV2_singleEnded	(SV2C_80	32G)				-			×
File Edit IESP/SV2C_8C32G	Wizards	ControlPanels	Tools	Results	Help						
Params		Log			Resul	lts					
Components				calOpti	ons propertie:	s (class: DataRec	ord)				
caloptions compute/Nean generate/Nois generate/Nois patternsync1 plotCreator1 plotCreator2 plotCreator2 plotCreator3 ncCharneLiut1 bcCharneLiut1 bcCharneLiut1 wrte/werage/votages wrte TransferFunctions		noduleName JataFiate argetLevels saveEyeScans intialPolymomialEsti commonModeLeve methods	mate Is			<sv2c nv<br="" serial="">26000.0 [150.0, 175.0, 200 False [3.4688e-12, -8.63 [15] [customInit]</sv2c>	umber> 0.0, 225.0, 1 346e-11, -2	250.0, 260.	0] 4.6665	№-05, 1.	.6827.
Add Remove Config	mod	urename									
Test Procedure compute Mean gen	eratePlots	getMeasurements/	At Amplitud	e writeAve	rageVoltage	s writeTransferF	unctions		_	_	
<pre>1 from dftm.comm.comml 2 SvtBertScan.autoCre 3 iesp = IESP.getInsta 4 averageVoltageDict = 5</pre>	Utils in atePrevi ance() = dict()	port * ews = True									< >
0				Run							

- 7. Press "Run" from the GUI. Ensure that the software connects and proceeds without error before leaving it to execute. Test completion takes approximately 10 minutes.
- 8. When the test procedure is complete it is important that you rename results folder from the automatically generated name to the SV2C serial number. The completed test procedure will appear as shown in the figure below.



🔯 Inti	rospect	ESP (v 21.1.0p) - rxTh	resholdCal_S	V2_singleEnded	(SV2C_8C	32G)				-		×
File	Edit	IESP/SV2C_8C32G	Wizards	ControlPanels	Tools	Results	Help					
		Params		Log *			Results	1				
		Delete Result		Average Trat Data Verso Notes SV2C20010	usPolyFit	. Average	foltage Meas	CalCoefficients_S	t) 5V2C	DataRateTree	dPlot	
0						Run						

9. Load the RX threshold calibration data onto the SV2C. To do this, reopen the test procedure "Sv2ProgramFlash". Under "calOptions" change the following settings to enter the module serial number (printed on the back of the module) and point the test procedure to the default calibration text files provided by Introspect Technology:

```
calOption.serialNumber = <module serial number>
calOption.rxThreshFilePath = C:\ <your full path here> \rxPerPhaseCal_SV2\Results\ [serial
number] calCoefficients_[serial number]\calCoefficients_[serial number].txt
calOption.jitterFilePath = C:\ <your full path here> \SV2C03_jitterData.txt
```



Eile	rospect Edit	ESP (v 21.1.0p) - Sv2P IESP/SV2C 8C32G	rogramFla: Wizards	sh (SV2C_8C32G) ControlPanels	Tools	Results	Help		-		×
		Params		Log *			Resu	ults			
		Components				calOpti	ons propertie	es (class: DataRecord)			
calOpt global refOoc writeD writeH writeR	ions ClockCo xksConfi ataToFla eaderCa terCaID xThresh	nfig g such DData sta CalData		xLinChannels serialNumber wThreahTilePath jtterFiePath methods				[1.2.3.4.5.6.7.8] <sv2c number="" serial=""> <c: <pre="">csubt>vxPerPh C:\spour path>\SV2C03_ [_custominit]</c:></sv2c>	aseCal_SV2\\ jitterData.txt	Results [\]	v (ser
Ad	ld	Remove Confid	rx1	hreshFilePath							
Test F 1 2 3 4 5	Procedu #! EX iesp impor impor	re writeDataToFlash IFECTED FIRMWAR = getIespInsta t dftm.comm.co t dftm.util as	wmteHeade E FWIES nce() mmUtils dftUti	CalData writeJitter PSV2C03A016 as dftComm 1	CalData	writeRxThr	eshCalData				< >
\sim						Pue					

- 10. Press "Run" from the GUI and wait for the test procedure to finish.
- 11. Very important: when the test procedure has completed, power cycle the SV2C module, and close the GUI before proceeding.
- 12. **Start the RX threshold validation.** To do this, reopen the GUI and open the test procedure "rxThreshold**Val_**SV2**_singleEnded**". Under "calOptions" change the following settings to enter the module serial number (printed on the back of the module)

calOption.serialNumber = <module serial number>

This is as shown in the figure on the following page.

13. Press "Run" from the GUI. Test completion takes approximately 2 minutes.



Introspect ESP (v 21.1.0p) - rxThresho	ldVal_SV2_singleEnded (SV2C_8C	32G)	- 🗆 ×
File Edit IESP/SV2C_8C32G Wiz	ards ControlPanels Tools	Results Help	
Params	Log	Results	
Components		calOptions properties (class: DataRecord)	
calQptions compute/Nean generate/Pois generate/Pois geleAssurements/Amplitude gleAssurements/Amplitude gleAssurements/Amplitude gleCheator1 gleCheator2 gleCheator	moduleName dataPates targetLevels saveEyeScans intiaProynomiaEstimate commonModeLevels methods	SVZ:20010001 (26000 0) [125.0, 150.0, 175.0 Falae [3.4608-12, -8.6346e-1 [15] [.c.ustomlmit]), 200.0, 225.0, 250.0, 260.0] 1, -2.9656e-06, 4.6669e-05, 1.6827,
Add Remove Config Tet Procedure computeMean generate 1 from dtrm.comm.commCtil 2 SvtBertScan.autoCreatef 3 isep = IESP.getInstance 4 average?OltageDict = di 5 swingDict = dict() 6 7 formFactor = isep.getFo	moduleName Plots getMeasurements/AAmpluude import * Previews = True () cct()	writeAverageVoitages	
0		Run	

14. When the validation is complete it is important that you rename results folder from the automatically generated name to the SV2C serial number. The completed test procedure is as shown in the figure below.

🔉 Intr	ospect	ESP (v 21.1.0p) - rxTh	resholdVal_5	V2_singleEnded (SV2C_8	C32G)					×
File	Edit	IESP/SV2C_8C32G	Wizards	ControlPanels Tools	Results	Help				
		Params		Log *		Results				
SV2C2	001000	1		Average TransferFund	DataR	ate TrendPlot	DataVersusPolyFit		olyFit	
				vScan1		verage Transfer F 400 200 200	unctions Across Common Mo	de Leveis (at 26000 Mi	ops), Ch	
				vScan1_run5		-100				
						-300	-200 -100 0 3	100 200 300		
				Notes SV2C20010001						
		Delete Result								
				[Run					

16. Verify that the transfer functions are linear (check under the "DataRate Trend Plot" icon) similar to the case shown above. This completes the SV2C RX threshold calibration and validation.



CASE 3: PROCEDURES FOR DIFFERENTIAL THRESHOLD CALIBRATION WITH FWIESPSV2C03A009 AND FWIESPSV2C03A011

1. Ensure that you are using the firmware / software / test procedure folder combination:

Firmware = FW09 or FW11 Software = IESP 3.6.74 or IESP 3.6.83 Test Folder = rxThresholdCal_FW09

2. Load the default RX threshold calibration file. To do this, from the GUI version listed above, open the test procedure "Sv2ProgramFlash". Under "calOptions" change the following settings to enter the module serial number (printed on the back of the module) and point the test procedure to the default calibration text files provided by Introspect Technology:

calOption.serialNumber = <module serial number> calOption.rxThreshFilePath = C:\ **<your full path here>** \defaultSv2ThresholdData_32G.txt calOption.jitterFilePath = C:\ **<your full path here>** \SV2C03_jitterData.txt

It may also be necessary to change the first line of the test procedure to the correct expected firmware version, either FW09 or FW11 as required. This is all as shown in the figure below.

ile	Edit	IESP/SV2C_8C32G	Wizards	ControlPanels	Tools	Results	Help					
		Params		Log			Result	s				
		Components				calOpti	ons properties	(class: DataRe	ecord)			
alOptic obalCl riteDat riteHei riteJtte riteRx	ons lockCo taToFia aderCa erCaID Thresh	nfig ash JIData ata CalData		rxLinChannels serialNumber rxThreshFilePath jtterFilePath methods			[C 	1, 2, 3, 4, 5, 6, SV2C serial S:\cyour path C:\cyour path _customInit]	7, 8] number> n>\default\$ n>\SV2C03	Sv2Threshd	oldData_ .txt	_32G.t
Add	1	Remove Confi	jitt	erFilePath								
Add	i rocedu	Remove Config re writeDataToFlash	jitt	er FilePath CalData writeJitter	CalData	writeRxThr	eshCalData					

- 3. Press "Run" from the GUI and wait for the test procedure to finish.
- 4. Very important: when the test procedure has completed, power cycle the SV2C module.
- 5. **Start the RX threshold calibration.** To do this, configure the MXP cables so that all TX signals are looped back to RX signals using the differential configuration (see Figure 1(a) on page 4 for connection details).
- 6. Open the test procedure "rxThreshold**Cal_**SV2". Under "calOptions" change the following settings to enter the module serial number (printed on the back of the module).

```
calOption.serialNumber = <module serial number>
```

It may also be necessary to change the first line of the test procedure to the correct expected firmware version, either FW09 or FW11 as required. This is all as shown in the figure below.

Params Log Results Components calOptions properties (class: DataRecord) Options moduleName < moduleName wear/ents/R4mplitude moduleName ba8CbockConfig targetLavels [150, 0, 200, 0, 250, 0, 300, 0, 500, 0,		ards ControlPanels lools Results He	ар	
Components calOptions properties (dass: DataRecord) MCptions mysteMean neretePots moduleName <svzc number="" serial="">- dataRate 26000 0 MeasurementSRAmplitude targetLevels [150,0,200,0,250,0,300,0,350,400,0,500,0,520,0] badCockConfg saveEyeScans False commonModeLevels [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15] x(Creator 2 x(Creator 2 x(Creator 3 x(Creator 3 x(Creator 3 x(Creator 3 x(Creator 4 x(Creator 3 x(Creator 4 x(Creator 3 x(Creator 4 x(Creator 3 x(Creator 4 x(Creator 3 x(Creator 4 x(Creator 4 x) LocustomInit] ChannelList 1 ChannelList 1 Scan 1 te Transferfunctions moduleName</svzc>	Params	Log	Results	
Woptions mystel/Mean nerete/Ptots module/Name <svzc number="" serial=""> dataRate 26000.0 Measurements/Amplitude saveEpteScans False False 26000.0 bal/CockConfig titemSync1 saveEpteScans False False common/ModeLevels [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15] common/ModeLevels [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15] common/ModeLevels [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15] common/ModeLevels [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15] common/ModeLevels [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15] common/ModeLevels [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15] common/ModeLevels [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15] common/ModeLevels [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15] consolitit1 methods LocustoremInit1]</svzc>	Components	calOptions p	roperties (class: DataRecord)	
modulename	alOptions ompute/Mean encrate/Pota tetMeasurements/AAmpitude tobalOoAcConfig attemSync1 tolOceator2 tolOceator2 tolCeator2 tolCeator2 (charnelLat1 CharnelLat1 CharnelLat1 CharnelLat1 trite/Average/toLages rite/average/toLages	moduleName dataRate targetLevels saveEyeScans intialPoynomalEstimate commonModeLevels methods	<sv2c number="" serial=""> 2600.0 [150.0, 200.0, 250.0, 300.0, 350.0, 400.1 False [3.4688e-128.6345e-11, -2.9655e-06, [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, custominit]</sv2c>	0, 500.0, 520.0] 4.6669e-05, 1.68 14, 15]
	3 from dftm.comm.commUti: 4 SvtBertScan.autoCreate	<pre>ls import * Previews = True s() ict()</pre>		

7. Press "Run" from the GUI. Ensure that the software connects and proceeds without error before leaving it to execute. Test completion takes approximately 10 minutes.



8. When the test procedure is complete it is important to rename results folder from the automatically generated name to the SV2C serial number. The completed test procedure will appear as shown in the figure below.



9. Load the RX threshold calibration data onto the SV2C. To do this, reopen the test procedure "Sv2ProgramFlash". Under "calOptions" change the following settings to enter the module serial number (printed on the back of the module) and point the test procedure to the default calibration text files provided by Introspect Technology:

calOption.serialNumber = <module serial number> calOption.rxThreshFilePath = C:\ **<your full path here>** \rxPerPhaseCal_SV2\Results\ [serial number] calCoefficients_[serial number]\calCoefficients_[serial number].txt calOption.jitterFilePath = C:\ **<your full path here>** \SV2C03_jitterData.txt

It may also be necessary to change the first line of the test procedure to the correct expected firmware version, either FW09 or FW11 as required. This is all as shown in the figure below.



The Edit IESP/SV2C_0C520 VVI	zards ControlPanels Tools	Results Help	
Params	Log	Results	
Components		calOptions properties (class: DataRecord)	
alloptions meData Johash meData Johash meLasadec.3D eta meLasadec.3D eta meLasec.3D eta meLasec.3D eta meLasec.3D eta	nLinChannels seialNunber n.ThreahlePath jiterfiePath methods	[1, 2, 3, 4, 5, 6, 7, 8] <svzc number="" serial=""> C:\opur path>v2PerPhase C:\opur path>VSV2C03_jit [_customInit]</svzc>	sCal_SVZ\Results∖[scr terData.txt
	jitterFilePath		
Add Remove Config			
Fest Procedure writeDataToFlash write	HeaderCalData writeJitterCalData	writeRxThreshCalData	
1 #! EXPECTED FIRMWARE F 2	WIESPSV2C03A011		^

- 10. Press "Run" from the GUI and wait for the test procedure to finish.
- 11. Very important: when the test procedure has completed, power cycle the SV2C module, and close the GUI before proceeding.
- 12. **Start the RX threshold validation.** To do this, reopen the GUI and open the test procedure "rxThreshold**Val_**SV2". Under "calOptions" change the following settings to enter the module serial number (printed on the back of the module)

calOption.serialNumber = < module serial number>

It may also be necessary to change the first line of the test procedure to the correct expected firmware version, either FW09 or FW11 as required. This is all as shown in the figure below.

13. Press "Run" from the GUI. The test completion takes approximately 2 minutes.



Introspect ESP (V 3.6.83) - rxThreshold	Val_SV2 (SV2C_8C32G)	Hala	- • ×
Params		Results	
Components	calOntion	ns properties (class: DataRecord)	
Calipotens calipotens computelken genketerhos genkete	Calipton moduleName dataRates targetLevels saveEyeScans initalPolynomialEstimate commonNodeLevels methods	s projenes (cass. Calanetodo) sector (2000.00) (1500.00, 000, 2500, 3000, 3500, 5406 [3.4689e-12, -8.6346e-11, -2.9656e [0, 7, 15] [_ccustominit]	400.0, 500.0, 520.0] -06, 4.6665±-05, 1.6827
	moduleName		
Add Remove Config	moduleName		
Add Remove Config Test Procedure computeMean generatef	moduleName	sgeVoltages	
Add Remove Config Test Procedure compute/Mean generate 1 #1 EXPECTED FIRMARE FW 2 3 from dftm.comm.commUtil 4 4 SvtDertScan.autoCreateP 5 5 lesp = LSP.getInstance 5	moduleName Nos getMeasurementsAtAmpltude writeAvers IESPSV2C03A011 s import * reviews = True ()	ageVokages	

14. When the validation is complete it is important that you rename results folder from the automatically generated name to the SV2C serial number. The completed test procedure is as shown in the figure below.

le Edit	IESP/SV2C_8C32G	Winnede								
202001000		Wizdius	ControlPanels 1	Tools F	Results	Help				
222001000	Params		Log			Results				
202001000	n		Average Transfe	erFunc	DataRa	Average Transfe	Data Venus Poly Fit	Common Mode	7]. Ch 1	
	Delete Result		Notes SV2C2001000	01		-800	-809200 0 290	400 6		

15. Verify that the transfer functions are linear (check under the "DataRate Trend Plot" icon) similar to the example shown above. This completes the SV2C RX threshold calibration and validation.



CASE 4: PROCEDURES FOR SINGLE-ENDED THRESHOLD CALIBRATION WITH

FWIESPSV2C03A009 AND FWIESPSV2C03A011

1. Ensure that you are using the firmware / software / test procedure folder combination:

Firmware = FW09 or FW11 Software = IESP 3.6.74 or 3.6.83 Test Folder = rxThresholdCal_FW09

2. Load the default RX threshold calibration file. To do this, from the GUI version listed above, open the test procedure "Sv2ProgramFlash". Under "calOptions" change the following settings to enter the module serial number (printed on the back of the module) and point the test procedure to the default calibration text files provided by Introspect Technology:

calOption.serialNumber = <module serial number> calOption.rxThreshFilePath = C:\ **<your full path here>** \defaultSv2ThresholdData_32G.txt calOption.jitterFilePath = C:\ **<your full path here>** \SV2C03_jitterData.txt

Introspect ESP (v 3	.6.83) - Sv2Pro	gramFlash	n (SV2C_8C32G)					-		×
File Edit IESP/S	W2C_8C32G	Wizards	ControlPanels	Tools	Results	Help				
Params			Log			Resu	lts			
Comport	ents				calOpti	ons propertie:	s (class: DataRecord)			
calabtani globalDocKcnfg wrteDtaaToRath wrteHeadercAlbta wrteHeadercAlbta wrteHteCalBta wrteRtThreahCalDta			xLinChannels serialNumber or ThreahFilePath jitterFilePath methods				[1, 2, 3, 4, 5, 6, 7, 8] CSVZC serial number CsVqour path>Vdef. C:\qour path>\SVZ [_custominit]	r> auft Sv2Thresholdf 2C03_jitterData.txt	Data_32	G.Ixt
Add Remove	Config	jitte	erfilePath							
Test Procedure write	DataToFlash v	vriteHeade	CalData writeJitter	CalData	writeRxThr	eshCalData				^
2 3 iesp = get 4 import dft 5 import dft	IespInstar m.comm.com m.util as	nce() mUtils dftUti	as dftComm							~
•					Run					

- 3. Press "Run" from the GUI and wait for the test procedure to finish.
- 4. Very important: when the test procedure has completed, power cycle the SV2C module.



- 5. **Start the RX threshold calibration.** To do this, configure the MXP cables so that all TX signals are looped back to RX signals using the single-ended configuration (see Figure 1(b) on page 4 for connection details).
- 6. Open the test procedure "rxThreshold**Cal_**SV2**_singleEnded**". Under "calOptions" change the following settings to enter the module serial number (printed on the back of the module).

calOption.serialNumber = <module serial number>

🔯 Introspect ESP (v 3.6.8	3) - rxThreshol	dCal_SV2_singleEnded (SV2C_8C	32G)				-		×
File Edit IESP/SV20	_8C32G Wiz	ards ControlPanels	Tools	Results	Help					
Params		Log			Resu	lts				
Components				calOpti	ons propertie	s (class: DataReco	rd)			
caloptions compute/Nean generate/Nois generate/Nois getMeasuremtsRi/Ampituu globallock:Corrig patternSync1 plotCreator2 plotCreator3 plotCreator2 plotCreator2 plotCreator3 plotCreator2 plotCreator3 pl	de	moduleName dataFate targetLevels saveEyeScans intilaPoynomiaEati commonModeLeve methods	mate Is			<sv2c nu<br="" serial="">26000.0 [150.0, 175.0, 200. False [3.4688e-12, -8.634 (0, 1.2, 3.4, 5.6, 7, [_customInit]]</sv2c>	mber> 0, 225.0, 250 46e-11, -2.96 7, 8, 9, 10, 11	0.0, 260.0] 1566-06, 4.66 1, 12, 13, 14,	69e-05, 1 15]	.6827,
Add Remove	Config	moduleName								
Test Procedure compute	Mean generate	Plots getMeasurements/	At Amplitud	e writeAve	rageVoltage	s writeTransferFu	nctions			
<pre>1 #! EXPECTED FIRMWARE FWIESPSV2C03A011 2 from dftm.comm.commUtls import * 3 SvtBertScan.autoCreatePreviews = True 4 iesp - IESP.getInstance() 5 averageVoltageDict = dict()</pre> Geoff Duerden (geoff@introspecttechnology) is signed in								< >		
0				Run						

- 7. Press "Run" from the GUI. Ensure that the software connects and proceeds without error before leaving it to execute. Test completion takes approximately 2 minutes.
- 8. When the test procedure is complete it is important that you rename results folder from the automatically generated name to the SV2C serial number. The completed test procedure will appear as shown in the figure below.



🔯 Inti	ospect	ESP (v 3.6.83) - rxThre	esholdCal_S	/2_singleEnded (S	5V2C_8C3	2G)				-		×
File	Edit	IESP/SV2C_8C32G	Wizards	ControlPanels	Tools	Results	Help					
		Params		Log *			Results					
SVZCA	2001000	Delete Result		Notes	asPolyFit	. Average	Golage Meas	calCoefficients_SV) //2C	DataRateTren	dPlot	
0						Run						

9. Load the RX threshold calibration data onto the SV2C. To do this, reopen the test procedure "Sv2ProgramFlash". Under "calOptions" change the following settings to enter the module serial number (printed on the back of the module) and point the test procedure to the default calibration text files provided by Introspect Technology:

```
calOption.serialNumber = <module serial number>
calOption.rxThreshFilePath = C:\ <your full path here> \rxPerPhaseCal_SV2\Results\ [serial
number] calCoefficients_[serial number]\calCoefficients_[serial number].txt
calOption.jitterFilePath = C:\ <your full path here> \SV2C03_jitterData.txt
```



Introspect ESP (v 3.6.83) - Sv2Pr File Edit IESP/SV2C 8C32G	ogramFlash (SV2C_8C32G) Wizards ControlPanels	Tools Resu	ults Help		- 0	×
Params	Log		Res	uits		
Components		Ca	Options propertie	es (class: DataRecord)		_
calOptions gdobalCodKonfig wrteDaatToFlash wrteHeadeCalData wrteHteAccalData wrteRxThreshCalData	rxLinChannels serialNumber xrThreahFilePath jtterFilePath methods			[1,2,3,4,5,6,7,8] <sv22 numbers<br="" serial="">C:\cyour path>XrPerP C:\cyour path>SV2C0 [_custominit]</sv22>	haseCal_SV2VResults \ [a 13_jiterData.txt	crip
	jitterFilePath					
Add Remove Confi	9					
Test Procedure writeDataToflash 1 #! EXPECTED FIRMWAR 2 3 iesp = getlespInsta 4 import dftm.com.cc 5 import dftm.util as	writeHeaderCalData writeJitte RE FWIESPSV2C03A011 ince() mmTUtils as dftComm dftUtil	erCalData writeF	RxThreshCalData			< >
						_

- 10. Press "Run" from the GUI and wait for the test procedure to finish.
- 11. Very important: when the test procedure has completed, power cycle the SV2C module, and close the GUI before proceeding.
- 12. **Start the RX threshold validation.** To do this, reopen the GUI and open the test procedure "rxThreshold**Val_**SV2**_singleEnded**". Under "calOptions" change the following settings to enter the module serial number (printed on the back of the module)

calOption.serialNumber = < module serial number>

This is as shown in the figure on the following page.

13. Press "Run" from the GUI. Test completion takes approximately 2 minutes.



File Edit IESP/SV2C_8C32G	Wizards ControlPanels Tools Results	: Help	
Params	Log	Results	
Components	calOp	ptions properties (class: DataRecord)	
alobions generatePita generatePita generatePita generatePita sutemSync1 bidCreator2 bidCreator2 bidCreator2 bidCreator3 bidCreator4 bidCreator5 bidCreator4 bidCreator5 bidCreator4 bidCreator1 bidCreator1 bidCreator1 bidCreator2 bidCreator3 bidCre	moduleName dataRates targetLevels saveEyeScans intalRoPornsilEstimate commonModeLevels methods	SV2:20010001 [26000.0] [125.0, 150.0, 175.0, 200.0, 225.0, 250.0, 2 Faite [3.4688e-128.6346e-112.9656e-06, 4.6669e-05, 1 [15] [_custominit]	60.0
	moduleName		
Add Remove Config			
Add Remove Config			
Add Remove Config Test Procedure computeMean gene	ratePiots getMeasurementsAtAmplitude writeAt	werageVoltages	
Add Remove Config Test Procedure compute/Mean gene 1 #! EXPECTED FIRMWARE 2 3 from dftm.comm.commU	atePlots getMeasurementsAAmplitude witeA FWIESPSV2C03A011 :ils import *	werage Voltages	ŕ
Add Remove Config Test Procedure computeMean gene 1 #1 EXPSCTED FIRMWARE 2 3 firom dftm.comm.commU 4 SvtBettScan.autoCrea 5 iesp = IESP.getInsta 6 averageVoltagePlot	<pre>atePlots getMessurementsAAmplitude writeA FWIESPSV2C03A011 :ils import * ce() uce() dict()</pre>	verage Voltages	,

14. When the validation is complete it is important that you rename results folder from the automatically generated name to the SV2C serial number. The completed test procedure is as shown in the figure below.

tin the	IEED/EV/2C 9C22C	MGanada	CantralDanala	Teele	Desults	Lista				~
ne cuit	Params	Wizarus	Log	IUUIS	Results	Result	s			
V2C200100	1		Average Trans VScar VScar1	ferFunc	DataRe	Average Tran	Deta Versus PolyPr Mer Functions Across Data Rate 	st det Common Made	PolyFit 15), ch 1	
	Delete Result		Notes SV2C200100	01						

17. Verify that the transfer functions are linear (check under the "DataRate Trend Plot" icon) similar to the example shown above. This completes the SV2C RX threshold calibration and validation.



RX COMMON MODE VOLTAGE CALIBRATION

This section of the document will provide the procedures for the following two cases for RX common mode voltage calibration:

Case 1: Single-ended RX common mode calibration, firmware = FW16

Case 2: Single-ended RX common mode calibration, firmware = FW09 or FW11

Please refer to Table 2 for the valid firmware / software / test procedure folder combinations for this calibration and refer to the step-by-step procedures below.

CASE 1: PROCEDURES FOR RX COMMON MODE CALIBRATION WITH

FWIESPSV2C03A016

1. Ensure that you are using the firmware / software / test procedure folder combination:

Firmware = FW16 Software = IESP 21.1.0 Test Folder = SV2RxCommonModeCalibration_HiSilicon_21p1p0

- 2. **Start the RX common mode calibration.** To do this, configure the MXP cables so that all TX signals are looped back to RX signals using the single-ended configuration (see Figure 1(b) on page 4 for connection details).
- 3. Open the test procedure "SV2RxCommonModeCalibration_HiSilicon_21p1p0". There are no "CalOptions" to modify. Simply press "Run" from the GUI as shown in the figure below.



Params Log Results Components globalClockConfig properties (class: GlobalClockSconfig properties), class: GlobalClockSConfig properties (class: GlobalClockSConfig properties), class: GlobalClockSConfig properties (class: GlobalClockSConfig properties), class: GlobalClockSConfig properties, class: GlobalClockSConfig pr	fig)		
Components globalClockConfig properties (class: GlobalClockConfig globalClockConfig dataRate 25780.0 globalClockConfig dataRate 25780.0 atternSync1 ditaRate 38.8 updateDatAteateDependentDefaults True refClockSorfig refClockSyncMode sync refClockSorfig refClockSorfig dataRate ChameList1 ChameList2 True dataRate sync refClockSorfig GataRate Set ithe master operating data rate (Mbps). All channels within the IESP operate at it mn 12:52 Mbps, max 28125 Mbps.	fig)		
etMeasurement 1 obalCockCorfig attemSync1 docksCorfig ChannelList1 ChannelList1 Add Remove Corfig	the same master		
Add Remove Config	the same master		
Add Remove Config	the same master		
ChannelList 1 ChannelList 1 Add Remove Config Config Add Test Ad	the same master		
ChamelList1 referenceClocks refClocksConfig dataRate Sets the mater operating data rate (Mbps). All chamels within the IESP operate at t min 1225 Mbps. max 28125 Mbps.	the same master		
dataRate Sets the master operating data rate (Mbps). All channels within the IESP operate at the min 1225 Mbps, max 28125 Mbps.	the same master		
	and build inductor	data rate.	Range
Test Procedure 1 iesp = getIespInstance() 2 globalClockConfig.setup()			
3 txChannelListl.setup()			
4			
5 for m in range(16):			
6 fail = 0			
<pre>7 print("common mode is: ", m)</pre>			
8 #iesp.writeSubPartRegister(0x0476, 0xF0, m)			_
<pre>9 iesp.setRxCommonModeVoltage(m, [1,2,3,4,5,6,7,8])</pre>			
<pre>10 result = bertMeasurement1.run()</pre>			
11			
<pre>12 if (result != None):</pre>			
<pre>13 for channel in rxChannelListl.channels:</pre>			

4. The test will execute in approximately 20 seconds. The test will finish when it finds the lowest value of common mode voltage which results in error-free BER operation across all channels. The required information is printed in the log window of the test procedure, as shown in the figure below.

Introspect ESP (v 21.1.0p) - SV2Rx	CommonModeCalibration_	HiSilicon	_21p1p0 (§	V2C_8C32G)			-		×
File Edit IESP/SV2C_8C32G	Wizards ControlPanels	Tools	Results	Help					
Params	Log			Results					
patternSyncl: PatternSync time since first measures errorRates (by channel): cumulatedErrors (by channel): Starting Bert Measurement setting up for Bert Measurement setting up for Bert Measurement starting RX Channellist starting RX Channellist common mode is: 10 Starting Bert Measurement setting up for Bert Measurement setting up for Bert Measurement setting up for Bert Measurement setting RX Channellist starting RX Channellist starting RX Channellist patternSyncl: PatternSync time since first measurement setting up for Bert Measurement setting RX Channellist patternSyncl: PatternSync time since first measure First common mode with m Test took 16.8 seconds	<pre>z failed on channe ment: 0.000 s (5: 0.0, 6: 0, irement setup t succeeded on channe ment: 0.000 s (1: 0.0, 5: 0, c irement setup t a succeeded on channe ment: 0.0, 5: 0, c irement setup t a c succeeded on channe ment: 0.0, 2: (1: 0.0, 2: 0, c) (1: 0.2, 2: 0, c) (1: 0, 5: 0, c) (1: 0, 2: 0, c) (2: 0, c) (2: 0, c)</pre>	<pre>ils l=4 0.0, 7 7: 0, innels ils 2-4 0.0, 6 6: 0, innels 0.0, 3 3: 0, chann</pre>	: 0.0, 8: 0) 1, 5-8 : 0.0, 7: 0, 7: 0, 1-8 : 0.0, 4: 0, els : 	8: 0.0} 7: 0.0, 8: 0 8: 0} 4: 0.0, 5: 0 5: 0, 6: 0, 11	0.0) 0.0, 6: 77: 0, 8	0.0, 7: 0. : 0}	O, 8:	0.0}	×

- 5. In this case, a common mode voltage setting of "11" was the lowest common mode voltage value which resulted in error-free BER operation. Once this common mode voltage setting is reported in the test procedure log, the SV2C RX common mode voltage calibration is complete.
- 6. To incorporate the common mode calibration result into a subsequent test, the following two lines of should be added to the subsequent Python file or test procedure. The example of setting the common mode voltage to "11" is shown below:

iesp = getlespInstance() # if not defined elsewhere
iesp.setRxCommonModeVoltage(11, [1,2,3,4,5,6,7,8])

7. Note: if differential operation is required in a procedure that previously defines this common mode voltage, the default common mode voltage setting of "15" should be used, as shown below:

iesp = getlespInstance() # if not defined elsewhere iesp.setRxCommonModeVoltage(15, [1,2,3,4,5,6,7,8])

CASE 2: PROCEDURES FOR RX COMMON MODE CALIBRATION WITH FWIESPSV2C03A009 AND FWIESPSV2C03A011

1. Ensure that you are using the firmware / software / test procedure folder combination:

Firmware = FW09 or FW11 Software = IESP 3.6.74 or IESP 3.6.83 Test Folder = SV2RxCommonModeCalibration_HiSilicon_3p6p74_3p6p83

- 2. **Start the RX common mode calibration.** To do this, configure the MXP cables so that all TX signals are looped back to RX signals using the single-ended configuration (see Figure 1(b) on page 4 for connection details).
- 3. Open the test procedure "SV2RxCommonModeCalibration_HiSilicon_3p6p74_3p6p83". There are no "CalOptions" to modify. Simply press "Run" from the GUI as shown in the figure below.



File Edit IESP/SV2C_8C32G	Wizards ControlPanels Tools	Results Help				
Params	Log	Results				
Components		lobalClockConfig properties (class: GlobalClo	ckConfig)			
pertMeasurement1	dataRate	25780.0				
attern Sync1	uiWidth	38.8				
xChannelList1	updateDataRateDependent	Defaults True				
(ChannelList1	system RefClock Source	internal				
	refClockSyncMode	sync				
	outputClockAFormat	LVDS				
	outputClock AFreq	100.0				
	outputClockBFormat	LVDS				
	outputClockBFreq	100.0				
Add Remove Config	dataRate Sets the master operating data r min 1225 Mbps, max 28125 Mbp	ate (Mbps). All channels within the IESP open 18.	ate at the same m	aster data	a rate. Ra	ange
Add Remove Config Test Procedure	dataRate Sets the master operating data r min 1225 Mbps, max 28125 Mbp	ate (Mbps). All channels within the IESP open IS.	ate at the same m	aster data	a rate. Ra	ange
Add Remove Config Test Procedure	dataRate Sets the master operating data r min 1225 Mbps, max 28125 Mbp FWIESPSV2C03A011	ate (Mbps). All channels within the IESP oper 18.	ate at the same m	aster data	a rate. Ra	ange
Add Remove Config Test Procedure 1 # : EXPECTED FIRMWARE	dataRate Sets the master operating data r min 1225 Mbps, max 28125 Mbp	ate (Mbps). All channels within the IESP oper 18.	ate at the same m	aster data	a rate. Ra	ange
Add Remove Config Test Procedure 1 # EXPECTED FIRMWARE 2 3 iesp = getlespInstar	dataRate Sets the master operating data min 1225 Maps, max 28125 Map FWIESPSV2C03A011 ce ()	ate (Mbps). All channels within the IESP oper is.	ate at the same m	aster data	a rate. Ra	ange
Add Remove Config Test Procedure 1 #! EXPECTED FIRMWARE 2 3 iesp = getlespInstan 4 globalClockConfig.se	dataRate Sets the master operating data r min 1225 Mbps.max 28125 Mbp FWIESPSV2C03A011 Ce () tup ()	ate (Mbps). All channels within the IESP oper 18.	ate at the same m	aster data	a rate. Ra	ange
Add Remove Config Test Procedure 1 #/ EXPECTED FIRMWARE 2 3 iesp = getIespInstar 4 globalClockConfig.se 5 txChannelListl.setup	dataRate Sets the master operating data min 1225 Mbps, max 28125 Mbp FWIESPSV2C03A011 ce () tup () ()	ate (Mbps). All channels within the IESP oper 18.	ate at the same m	aster data	a rate. Ra	ange
Add Remove Config Test Procedure 1 #: EXPECTED FIRMWARE 2 3 Lesp = getlespInstar 4 globalClockConfig.se 5 txChannelListl.setup 6	dataRate Sets the master operating data min 1225 Mbps, max 28125 Mbp : FWIESPSV2C03A011 ce () tup () ()	ate (Mbps). All channels within the IESP oper is .	ate at the same m	aster data	a rate. Ra	ange
Add Remove Config Test Procedure 1 #! EXPECTED FIRMWARE 2 3 issp = getIespInstan 4 globalClockConfig.se 5 txChannelList1.setup 6 7 for m in range(16):	dataRate Sets the master operating data min 1225 Mbps, max 28125 Mbp FWIESPSV2C03A011 Ce () tup () ()	ate (Mops). All channels within the IESP oper Is.	ate at the same m	aster data	a rate. Ra	ange
Add Remove Corfig 1#1 EXPECTED FIRMWARE 2 3 iesp = getlespInstar 4 [lobalClockConfig.se 5 txChanneLlistl.stur 6 7 for m in range(16): 8 6 fail = 0	dataRate Sets the master operating data min 1225 Maps, max 29126 Map FWIESPSV2C03A011 ce () tup () ()	ate (Mbps). All channels within the IESP oper is.	ate at the same m	aster data	a rate. Ra	ange
Add Remove Config Test Procedure 1 #: First MWARE 2 3 iesp = getIespInstant 4 globalClockConfig.set 5 5 txChannelListl.setup 6 7 for m in range(16): 8 8 fail = 0 9 9 fiesp.vriteSubPa	dataRate Sets the master operating data r mn 1225 Mbps.max 28125 Mbp : FWIESPSV2C03A011 ce() tup() () rtRegister(0x0476, 0x26	ate (Mops). All channels within the IESP oper 18.	ate at the same m	aster data	a rate. Ra	ange
Add Remove Config 1 #i EXPECTED FIRMWARE 2 3 iesp = getIespInstant 4 globalClockConfig.see 5 txChannelList1.setup 6 7 for m in range(16): 8 fall = 0 9 #iesp.vriteSubPt 10 10 setxRxCommon	dstaRate Sets the master operating data min 1225 Maps, max 28125 Maps FWIESPSV3C03A011 ce () () rtRegister(0x0476, 0xE0 ModeVoltage (m, [1,2,3,4	te (Mbps). All channels within the IESP oper s. , m) , 5, 6, 7, 8])	ate at the same m	aster data	a rate. Ra	ange
Add Remove Config Test Procedure 1 #! EXPECTED FIRMWARE 2 3 iesp = getlespInstar 4 globalClockConfig.se 5 5 txChannelListl.setup 6 7 for m in range(16): 8 8 fail = 0 9 #issp.writeSubPa 0 iesp.setRxCompany 10 10 print("command m	dataRate Sets the master operating data m mn 1225 Mbps.max 28125 Mbp : FWIESPSV2C03A011 ce() tup() () rtRegister(0x0476, 0xF0 ModeVoltage(m, [1,2,3,4 ode 1s: ", m)	ate (Mbps). All channels within the IESP oper is. , m) , 5, 6, 7, 8])	ate at the same m	aster data	a rate. Ra	ange

4. The test will execute in approximately 20 seconds. The test will finish when it finds the lowest value of common mode voltage which results in error-free BER operation across all channels. The required information is printed in the log window of the test procedure, as shown in the figure below.

maospect	ESP (v 3.6.83) - SV2R	CommonN	lodeCalibration_H	liSilicon_	_3p6p74_3	p6p83 (SV2)	C_8C32G)		-		×
File Edit	IESP/SV2C_8C32G	Wizards	ControlPanels	Tools	Results	Help					
	Params	[Log		J	Resu	lits				
patternSy time sinc errorRate cumulated command in Starting v Starting patternSy patternSy patternSy time sinc errorRate cumulated starting starting starting starting starting fatternSy time sinc errorRate cumulated First con Test tool	<pre>mol: PatternSy re first measur is (by channel) Errors (by cha- nerrors (by cha- nerrors (by cha- nerrors) Bert Measureme pp for Bert Measureme syncWithDataPa mol: PatternSy re first measures is (by channel) Errors (by channel)</pre>	nc fail ement: (; nnel); nt suremen setup th nc fail ement: (; nnel); nt suremen stup th nc succ nnel); nt suremen succe innel); nt suremen setup th nc succe nnel); nt suremen succe nnel); nt succe nnel); nt succe nnel); nt suremen succe nnel); nt suremen succe nnel); nt succe nnel); nt succe nnel); nt succe nnel); nt succe nnel); nt succe nnel); nt succe nnel); nt succe nnel); nt succe nnel); nt succe nnel); nt succe nnel); setup th nc fail ; nnel); ; nt succe nnel); ; nt succe nnel); ; nt succe nnel); ; nt succe nnel); ; nt succe nnel); ; nnel); ; nnel); ; nnel); ; nnel); ; nnel); ; nnel); ; nnel); ; nnel); ; nnel); ; nnel); ; nnel); ; nnel); ; nnel); ; nnel); ;]	ed on channe (1: 0.0, 5: 0, (1: 0, 5: 0, (1: 0, 5: 0, edded on channe 0.000 s (1: 0, 5: 0, (1: 0, 5: 0, (1: 0, 5: 0, (1: 0, 2: 0, error on al	<pre>x1s 2-4 0.0, 6 6: 0, x1nnels x1s 2-4 0.0, 6 6: 0, x1nnels 0.0, 3 3: 0, 1 chan</pre>	:: 0.0, 7: 0, 1, 5-8 :: 0.0, 7: 0, 1-8 :: 0.0, 4: 0, , mels :	7: 0.0, 8: 0} 7: 0.0, 8: 0} 4: 0.0, 5: 0, 6 10	8: 0.0} 8: 0.0} 5: 0.0, 6: 0 : 0, 7: 0, 8:	.0, 7: 0.1 0}), 8:	0.0}	



- 5. In this case, a common mode voltage setting of "10" was the lowest common mode voltage value which resulted in error-free BER operation. Once this common mode voltage setting is reported in the test procedure log, the SV2C RX common mode voltage calibration is complete.
- 6. To incorporate the common mode calibration result into a subsequent test, the following two lines of should be added to the subsequent Python file or test procedure. The example of setting the common mode voltage to "10" is shown below:

iesp = getlespInstance() # if not defined elsewhere iesp.setRxCommonModeVoltage(10, [1,2,3,4,5,6,7,8])

7. Note: if differential operation is required in a procedure that previously defines this common mode voltage, the default common mode voltage setting of "15" should be used, as shown below:

iesp = getlespInstance() # if not defined elsewhere iesp.setRxCommonModeVoltage(15, [1,2,3,4,5,6,7,8])



RX EQUALIZATION (CTLE) CALIBRATION

This section of the document will provide the procedures for the following four cases for RX equalization (CTLE) calibration:

Case 1: Differential RX CTLE calibration, firmware = FW16

Case 2: Single-ended RX CTLE calibration, firmware = FW16

Case 3: Differential RX CTLE calibration, firmware = FW09 or FW11

Case 4: Single-ended RX CTLE calibration, firmware = FW09 or FW11

Please refer to Table 3 for the valid firmware / software / test procedure folder combinations for this calibration and refer to the step-by-step procedures below.

Important note: when performing **differential** RX CTLE calibration, as in Cases 1 or 3 above, ensure that **differential** RX threshold voltage calibration data has previously been loaded onto the module. Similarly, when performing **single-ended** CTLE calibration as in cases 2 and 4 above, ensure that the **single-ended** RX threshold voltage calibration data has been previously loaded onto the module. Performing the CTLE calibration with incorrect RX threshold calibration data, or without RX threshold calibration data, will result in an incorrect CTLE calibration.

CASE 1: PROCEDURES FOR DIFFERENTIAL RX CTLE CALIBRATION WITH

FWIESPSV2C03A016

1. Ensure that you are using the firmware / software / test procedure folder combination:

Firmware = FW16 Software = IESP 21.1.0 Test Folder = HiSiliconCTLECalibration_21p10

- 2. **Start the RX CTLE calibration.** To do this, configure the MXP cables so that all TX signals are looped back to RX signals using the differential configuration (see Figure 1(a) on page 4 for connection details).
- 3. Open the test procedure "HiSiliconCTLECalibration_21p10". There are no "CalOptions" to modify. Simply press "Run" from the GUI as shown in the figure below.



Introspect ESP (v 21.1.0p) - HiSilicon File Edit IESP/SV2C 8C32G Wi	CTLECalibration_21p1p0 (SV2C_80	C32G) Results Help	- 0 X
Params	Log	Results	
Components		calibrationOptions properties (class: DataRecor	rd)
bert Measurement 1 bert Scan 1 calibrationOptions eye Scan 1 gobalDockConfig pattemsync 1 ncCharmelList 1 bcCharmelList 1 bcCharmelList 1 vScan 1	channels uesVZTx methods	[1, 2, 3, 4, 5, 6, 7, 8] True [_customhit]	
Test Breachers			
<pre>iestProcedure iestProcedure iestProcedu</pre>	() p() oltage(15, [1,2,3,4,5, Options.channels ,5,6,7,8] mels :	6,7,8])	
•		Run	

4. The test will execute in approximately 12 minutes. When the test finishes, the calibrated CTLE settings will be printed at the end of the test log. An example of the output log of a typical run is shown in the figure below.





5. As shown in the example output log file:

The first row of coefficients contain the CTLE offsets, per channel, the second row of coefficients contain the CTLE high frequency gain settings, per channel, and the third row of coefficients contain the low frequency gain settings, per channel.

- 6. Once the calibration settings are reported in the test procedure log, the SV2C RX differential CTLE calibration is complete.
- 7. The optimal CTLE parameters are set at the end of this calibration procedure and remain in effect until the SV2C module power is cycled. There is no need to reprogram the CTLE settings even if subsequent test procedures are executed, as long as the subsequent test procedures do not attempt to overwrite the calibrated CTLE settings. In general, this SV2C RX differential CTLE calibration procedure should be run once after each SV2C power cycle.
- 8. To incorporate these CTLE values manually into another test procedure, please refer to the Introspect "Help files" available in the Introspect ESP GUI for both Python syntax and examples. The most common iesp functions for setting the CTLE values are listed below:

setCtleOffsetCancellation() setCtleHighFreqGain() setCtleLowFreqGain() setCtleParams()

and the calibration procedure itself, "HiSiliconCTLECalibration_21p10", contains several examples of setting CTLE parameters for reference.

CASE 2: PROCEDURES FOR SINGLE-ENDED RX CTLE CALIBRATION WITH FWIESPSV2C03A016

1. Ensure that you are using the firmware / software / test procedure folder combination:

```
Firmware = FW16
Software = IESP 21.1.0
Test Folder = HiSiliconCTLECalibration_21p10
```

2. **Start the RX CTLE Mode Calibration.** To do this, configure the MXP cables so that all TX signals are looped back to RX signals using the differential configuration (see Figure 1(b) on page 4 for connection details).



3. Open the test procedure "HiSiliconCTLECalibration_21p10". There are no "CalOptions" to modify, but note that a user may wish to explicitly set the RX common mode voltage in this procedure, as shown on line 3 of the test procedure below.

File Edit IESP/SV2C_8C32G	Wizards ControlPanels	Tools Results	Help		
Params	Log		Results		
Components		calibratio	nOptions properties (class	: DataRecord)	
ertMeasurement1	channels		[1, 2, 3, 4, 5	5, 6, 7, 8]	
alibrationOptions	useSV2Tx		True		
veScan1	methods		[_custom]	lnit]	
obalClockConfig					
attern Sync 1					
ChannelList1					
Scan1					
					-
	channels				
Add Remove Config					
art Procedum					-
eschooleure					
liesp = getlespinstan	ce ()				
2 globalClockConfig.se	tup()				
3 iesp.setRxCommonMode	Voltage(15, [1,2,	3,4,5,6,7,8])			
4					
5 channels = calibrati	onOptions.channel	s			
6 allChannels = [1,2,3	,4,5,6,7,8]				
7 offsetsByChannel = {	}				
O b f O a d a a D a O b a a a a 1	}				
s niGainsbychannel = {	}				
9 lfGainsByChannel = {	,				
9 lfGainsByChannel = { 10 for channel in allCh	annels :				

4. Press "Run" from the GUI as shown in the figure. The test will execute in approximately 12 minutes. When the test finishes, the calibrated CTLE settings will be printed at the end of the test log. An example of the output log of a typical run is shown in the figure below.





5. As shown in the example output log file in the previous figure:

The first row of coefficients contain the CTLE offsets, per channel, the second row of coefficients contain the CTLE high frequency gain settings, per channel, and the third row of coefficients contain the low frequency gain settings, per channel.

- 6. Once the calibration settings are reported in the test procedure log, the SV2C RX single-ended CTLE calibration is complete.
- 7. The optimal CTLE parameters are set at the end of this calibration procedure and remain in effect until the SV2C module power is cycled. There is no need to reprogram the CTLE settings even if subsequent test procedures are executed, as long as the subsequent test procedures do not attempt to overwrite the calibrated CTLE settings. In general, this SV2C RX differential CTLE calibration procedure should be run once after each SV2C power cycle.
- 8. To incorporate these CTLE values into another test procedure, please refer to the Introspect "Help files" available in the Introspect ESP GUI for both Python syntax and examples. The most common iesp functions for setting the CTLE values are listed below:

setCtleOffsetCancellation() setCtleHighFreqGain() setCtleLowFreqGain() setCtleParams()

and the calibration procedure itself, "HiSiliconCTLECalibration_21p10", contains several examples of setting CTLE parameters.

CASE 3: PROCEDURES FOR DIFFERENTIAL RX CTLE CALIBRATION WITH FWIESPSV2C03A009 OR FWIESPSV2C03A011

1. Ensure that you are using the firmware / software / test procedure folder combination:

Firmware = FW09 or FW11 Software = IESP 3.6.74 or 3.6.83 Test Folder = HiSiliconCTLECalibration_3p6p74_3p6p83

2. **Start the RX CTLE calibration.** To do this, configure the MXP cables so that all TX signals are looped back to RX signals using the differential configuration (see Figure 1(a) on page 4 for connection details).



3. Open the test procedure "HiSiliconCTLECalibration_3p6p74_3p6p83". There are no "CalOptions" to modify. Simply press "Run" from the GUI as shown in the figure below.

File Edit IESP/3V2C_8C320	vizarus contron-ariels ie	ools Results Help		
Params	Log	Resu	its	
Components		calibrationOptions prope	erties (class: DataRecord)	
bert Measurement 1	channels		[1, 2, 3, 4, 5, 6, 7, 8]	
calibrationOptions	useSV2Tx		True	
yeScan1	methods		[_customInit]	
lobalClockConfig attern Sync 1				
(ChannelList 1				
Scan1				
	channels			
Add Damaura Carfin				
Add Remove Conlig				
Test Procedure				
1 #! EXPECTED FIRMWARE	FWIESPSV2C03A011			
2				
3 iesp = getIespInstand	;e()			
4 globalClockConfig.set	up()			
5 #iesp.setRxCommonMode	Voltage(15, [1,2,3,-	4,5,6,7,8])		
6 channels = calibratio	nOptions.channels			
<pre>7 allChannels = [1,2,3,</pre>	4,5,6,7,8]			
<pre>8 offsetsByChannel = {}</pre>				
<pre>9 hfGainsByChannel = { }</pre>				
10 IfGainsByChannel = {}				

4. The test will execute in approximately 6 minutes. When the test finishes, the calibrated CTLE settings will be printed at the end of the test log. An example of the output log of a typical run is shown in the figure below.

🔯 Int	rospec	ESP (v 3.6.83) - HiSilic	onCTLECali	bration_3p6p74_3	3p6p83 (S	V2C_8C32	G)		-	×
File	Edit	IESP/SV2C_8C32G	Wizards	ControlPanels	Tools	Results	Help			
		Params	[Log			Result	ts		
Star	ting	VScan								^
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
(1:	7, 2	64, 3: 62, 4:	50, 5:	56, 6: 54,	7: 47,	8: 44)				
(1:	11, :	2: 10, 3: 9, 4:	10, 5:	0, 6: 0, 7:	11, 8	: 9}				
{1:	0, 2	0, 3: 6, 4: 0	, 5: 7,	6: 6, 7: 1,	8: 0}					
Test	fin	lshed								
Test	t00	5.4 minutes								
										~
							_			
0						Run				



5. As shown in the example output log file:

The first row of coefficients contain the CTLE offsets, per channel, the second row of coefficients contain the CTLE high frequency gain settings, per channel, and the third row of coefficients contain the low frequency gain settings, per channel.

- 6. Once the calibration settings are reported in the test procedure log, the SV2C RX differential CTLE calibration is complete.
- 7. The optimal CTLE parameters are set at the end of this calibration procedure and remain in effect until the SV2C module power is cycled. There is no need to reprogram the CTLE settings even if subsequent test procedures are executed, as long as the subsequent test procedures do not attempt to overwrite the calibrated CTLE settings. In general, this SV2C RX differential CTLE calibration procedure should be run once after each SV2C power cycle.
- 8. To incorporate these CTLE values manually into another test procedure, please refer to the Introspect "Help files" available in the Introspect ESP GUI for both Python syntax and examples. The most common iesp functions for setting the CTLE values are listed below:

setCtleOffsetCancellation() setCtleHighFreqGain() setCtleLowFreqGain() setCtleParams()

and the calibration procedure itself, "HiSiliconCTLECalibration_3p6p74_3p6p83", contains several examples of setting CTLE parameters for reference.

CASE 4: PROCEDURES FOR SINGLE-ENDED RX CTLE CALIBRATION WITH FWIESPSV2C03A009 AND FWIESPSV2C03A011

1. Ensure that you are using the firmware / software / test procedure folder combination:

Firmware = FW09 or FW11 Software = IESP 3.6.74 or IESP 3.6.83 Test Folder = HiSiliconCTLECalibration_3p6p74_3p6p83

2. **Start the RX CTLE Mode Calibration.** To do this, configure the MXP cables so that all TX signals are looped back to RX signals using the differential configuration (see Figure 1(b) on page 4 for connection details).



3. Open the test procedure "HiSiliconCTLECalibration_21p10". There are no "CalOptions" to modify, but note that a user may wish to explicitly set the RX common mode voltage in this procedure, as shown on line 5 of the test procedure below.

	Edit 1251/5120_00520	Wizards ControlPanel	s fools	Results	Help		
	Params	Log			Results		
	Components			calibration	options properties (class: DataReco	rd)	
ert Me	asurement 1	channels			[1, 2, 3, 4, 5, 6, 7, 8]		
alibrat	ionOntions	useSV2Tx			True		
veSca	an1	methods			[_customInit]		
obalC	llockConfig						
ttem	Sync1						
Chan	nelList1						
nan	nelList I 1						
Carr							
		channels					
		. '					
Ad	d Remove Config						
Ad	d Remove Config						
Ade est F	d Remove Config Procedure						
Adi est F	d Remove Config	FWIESDSU2C03300	a				
Add Test P	d Remove Config Procedure #! EXPECTED FIRMWARE	FWIESPSV2C03A00	9				
Add Test F	d Remove Config rocedure #! EXPECTED FIRMWARE	FWIESPSV2C03A00	9				
Ad Test P 1 : 2 3 :	d Remove Config Procedure # ! EXPECTED FIRMWARE iesp = getlespInstanc	FWIESPSV2C03A00	9				
Adi Test F 2 3 : 4 (d Remove Config Procedure #! EXPECTED FIRMWARE iesp = getlespInstanc globalClockConfig.set	FWIESPSV2C03A00	9				
Adi Test F 2 3 4 (5	d Remove Config Procedure #! EXPECTED FIRMWARE iesp = getlespInstanc globalClockConfig.set iesp.setRxCommonModeV	FWIESPSV2C03A00 ce() tup() Voltage(15, [1,2	9	5,7,8])			
Adi Test F 2 3 4 5 5	d Remove Config Procedure #! EXPECTED FIRMWARE liesp getLespInstance globalClockConfig.set iesp.setExCommonMode channels = calibratic	FWIESPSV2C03A00 ce() tup() %Oltage(15, [1,2 onOptions.channe	9 ,3,4,5,1 1s	5,7,8])			
Adi Test P 2 3 5 5 6 7 7	d Remove Config vocedure #! EXPECTED FIRMWARE iesp = getlespInstand globalClockConfig.set iesp.setRxCommonMode channels = calibratic allChannels = [1,2,3,	FWIESPSV2C03A00 De() Voltage(15, [1,2 onOptions.channe ,4,5,6,7,8]	9 ,3,4,5,0 1s	5,7,8])			
Adi Test F 2 3 5 5 6 0 7 8 0	d Remove Config Procedure	FWIESPSV2C03A00 ce() voup() Yoltage(15, [1,2 onOptions.channe (4,5,6,7,8]	9 ,3,4,5,0 1a	5,7,8])			
Adi Test F 2 3 3 4 5 5 5 7 8 9 1	d Remove Config Vocedure <i>if EXPECTED FIRMWARE</i> <i>iesp</i> getlespinstanc globalClockConfig.set <i>iesp</i> .setRxCommonModet channels = [1,2,3, <i>offsetsByChannel</i> = () <i>ifGainByChannel</i> = ()	FWIESPSV2C03A00 ce() tup() Voltage(15, [1,2 onOptions.channe 4,5,6,7,8] }	9 ,3,4,5,0 1a	5,7,8])			
Ad Test F 2 3 3 4 5 5 5 6 0 7 8 0 9 1 10	d Remove Config Procedure #! EXPECTED FIRM#ARE lesp = getlespInstand plobalClocKConfig.set iesp.setRxCommoMdode' channels = calibratic allChannels = (1,2,3, offsetsByChannel = (1) ifGainByChannel = (1)	<pre>FWIESPSV2C03A00 ce() tup() /oltage(15, [1,2 onOptions.channe 4,5,6,7,8]))</pre>	9 ,3,4,5,4	5,7,8])			

4. Press "Run" from the GUI as shown in the figure. The test will execute in approximately 6 minutes. When the test finishes, the calibrated CTLE settings will be printed at the end of the test log. An example of the output log of a typical run is shown in the figure below.

File	Edit	IESP/SV2C_8C32G	Wizards	ControlPanels	Tools	Results	Help			
		Params		Log			Resul	ts		
Star	ting	VScan								^
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
Star	ting	VScan								
(1:	30, 2	2: 83, 3: 76, 4	: 73, 5	: 89, 6: 86,	7: 71	, 8: 89	9)			
(1:	1, 2:	: 0, 3: 0, 4: 0	, 5: 0,	6: 0, 7: 8,	8: 0}					
{1:	14, 2	2: 6, 3: 8, 4:	7, 5: 1,	6: 7, 7: 0	, 8: 8	}				
Test	fini	ished								
Test	tool	k 5.3 minutes								
										~
						Pue				



5. As shown in the example output log file in the previous figure:

The first row of coefficients contain the CTLE offsets, per channel, the second row of coefficients contain the CTLE high frequency gain settings, per channel, and the third row of coefficients contain the low frequency gain settings, per channel.

- 6. Once the calibration settings are reported in the test procedure log, the SV2C RX single-ended CTLE calibration is complete.
- 7. The optimal CTLE parameters are set at the end of this calibration procedure and remain in effect until the SV2C module power is cycled. There is no need to reprogram the CTLE settings even if subsequent test procedures are executed, as long as the subsequent test procedures do not attempt to overwrite the calibrated CTLE settings. In general, this SV2C RX differential CTLE calibration procedure should be run once after each SV2C power cycle.
- 8. To incorporate these CTLE values into another test procedure, please refer to the Introspect "Help files" available in the Introspect ESP GUI for both Python syntax and examples. The most common iesp functions for setting the CTLE values are listed below:

setCtleOffsetCancellation() setCtleHighFreqGain() setCtleLowFreqGain() setCtleParams()

and the calibration procedure itself, "HiSiliconCTLECalibration_3p6p74_3p6p83", contains several examples of setting CTLE parameters.





INTERNAL FREQUENCY VALIDATION

This section will of the document will provide the procedures for the internal frequency validation. This validation is only available when using FW16. The IESP 21.1.0 software must be installed, but the Python file provided by Introspect may be executed from any Python environment. There are no external MXP cable connections required.

1. Ensure that you are using the following firmware / software / Python file combination:

```
Firmware = FW16
Software = IESP 21.1.0
Python File = Test_20201217-Python-freqMeasurement.py
```

2. Modify the "Test_20201217-Python-freqMeasurement.py" file as required to work within the desired Python environment. The only expected modification to be made is on line 6 of this file, where the location of the SvtPython libraries from the IntrospectESP 21.1.0 installation are defined. The modification to the line will be of the form shown below:

```
sys.path.append(r"C:\<your full path here>\IntrospectESP_21.1.0\SvtPython")
```

3. Start the Internal Frequency Validation. An example of executing the test on a PC from a command prompt is show in the figure below. The current working directory in this case is "sv2 calibration scripts\FW16_21p1p0".



4. The resulting output file, "freq.csv" will be written to the same current working directory. This output file currently contains only two comma separated entries, as defined below:

The first entry contains "1" if the high-frequency accuracy check has passed for 25781.25 Mbps, and contains "0" if a failure has occurred.

The second entry contains "1" if the high-frequency accuracy check has passed at 27952.5 Mbps, and contains "0" if a failure has occurred.

5. Once these two results are reported, the SV2C internal frequency validation is complete.



RECAPITULATION OF CALIBRATION AND VALIDATION PROCEDURES

This document has provided the full set of step-by-step instructions for performing in-field calibrations for the SV2C. There are four types of calibrations and validations which may be performed in the field:

- 1. SV2C RX Threshold Voltage Calibration
- 2. SV2C RX Common Mode Voltage Calibration
- 3. SV2C RX Equalization (CTLE) Calibration
- 4. SV2C Internal Frequency Validation

and there are currently three different SV2C firmware releases for use with the calibrations listed above. A user only needs to execute the combination of procedures and firmware versions relevant to their testing application, according to the use cases described within this document.



Revision Number	History	Date
1.0	Document Release	March 25, 2021

The information in this document is subject to change without notice and should not be construed as a commitment by Introspect Technology. While reasonable precautions have been taken, Introspect Technology assumes no responsibility for any errors that may appear in this document.



© Introspect Technology, 2021 Published in Canada on March 25, 2021 EN-G055E-E-21083

INTROSPECT.CA