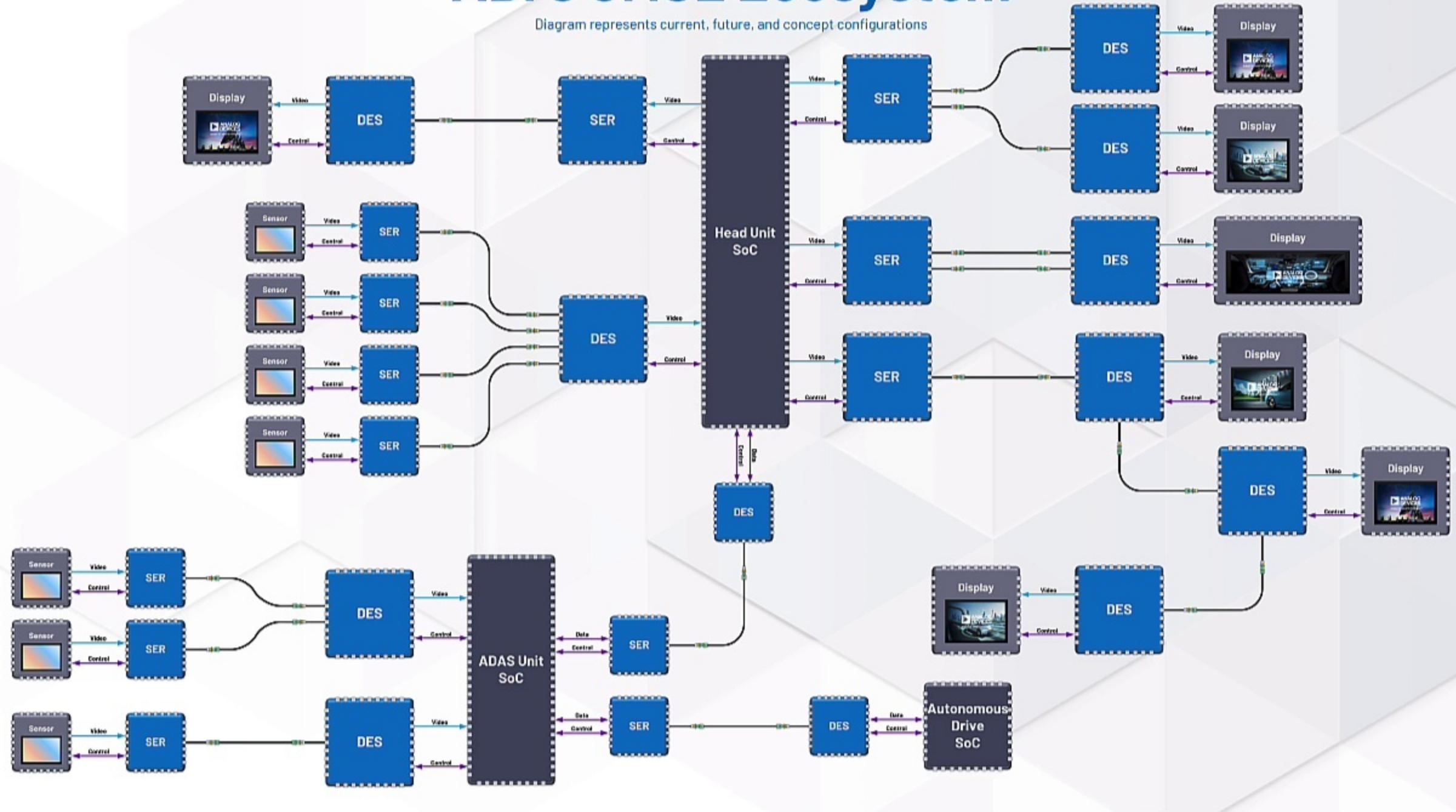
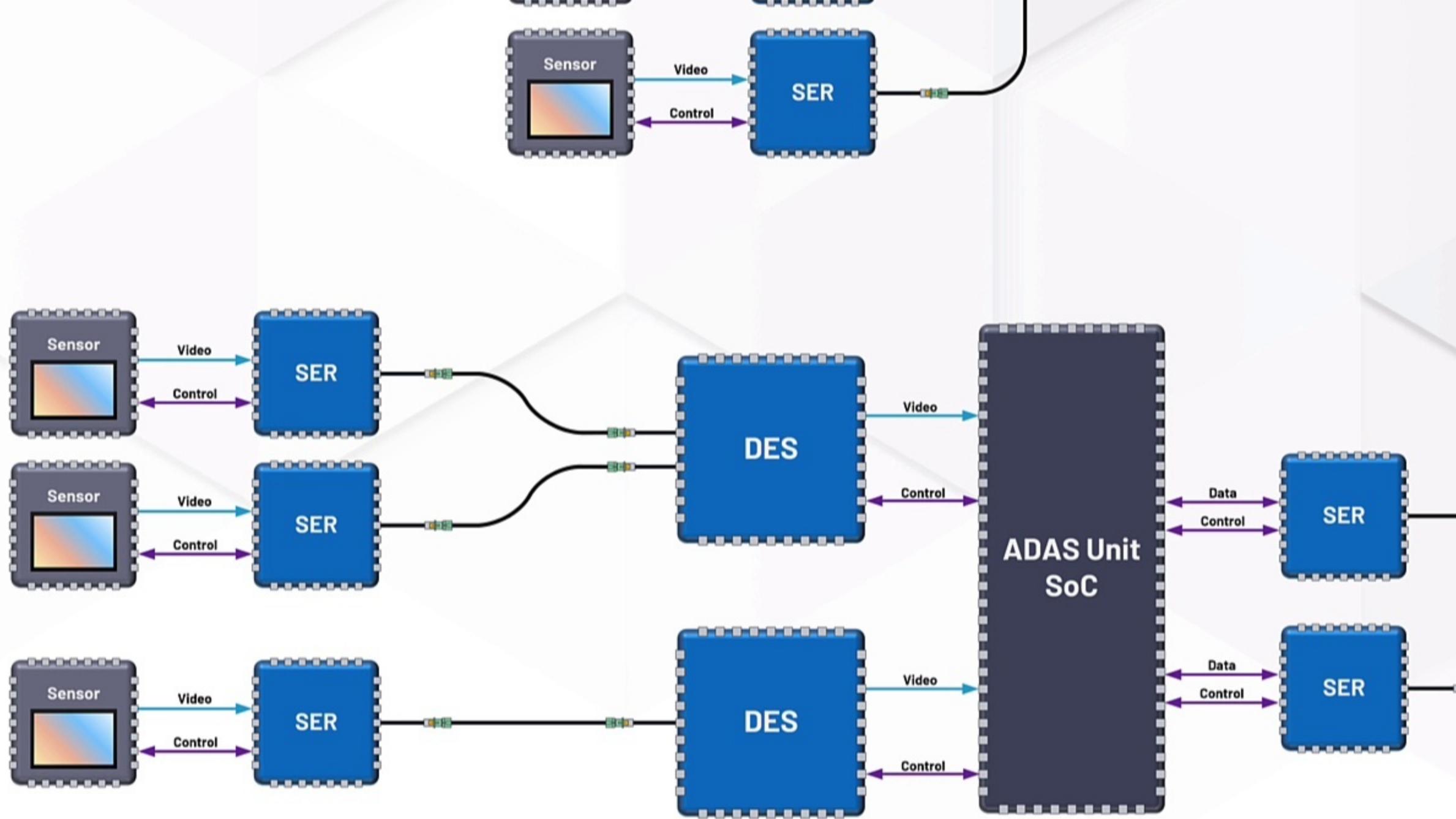


# ADI's GMSL Ecosystem

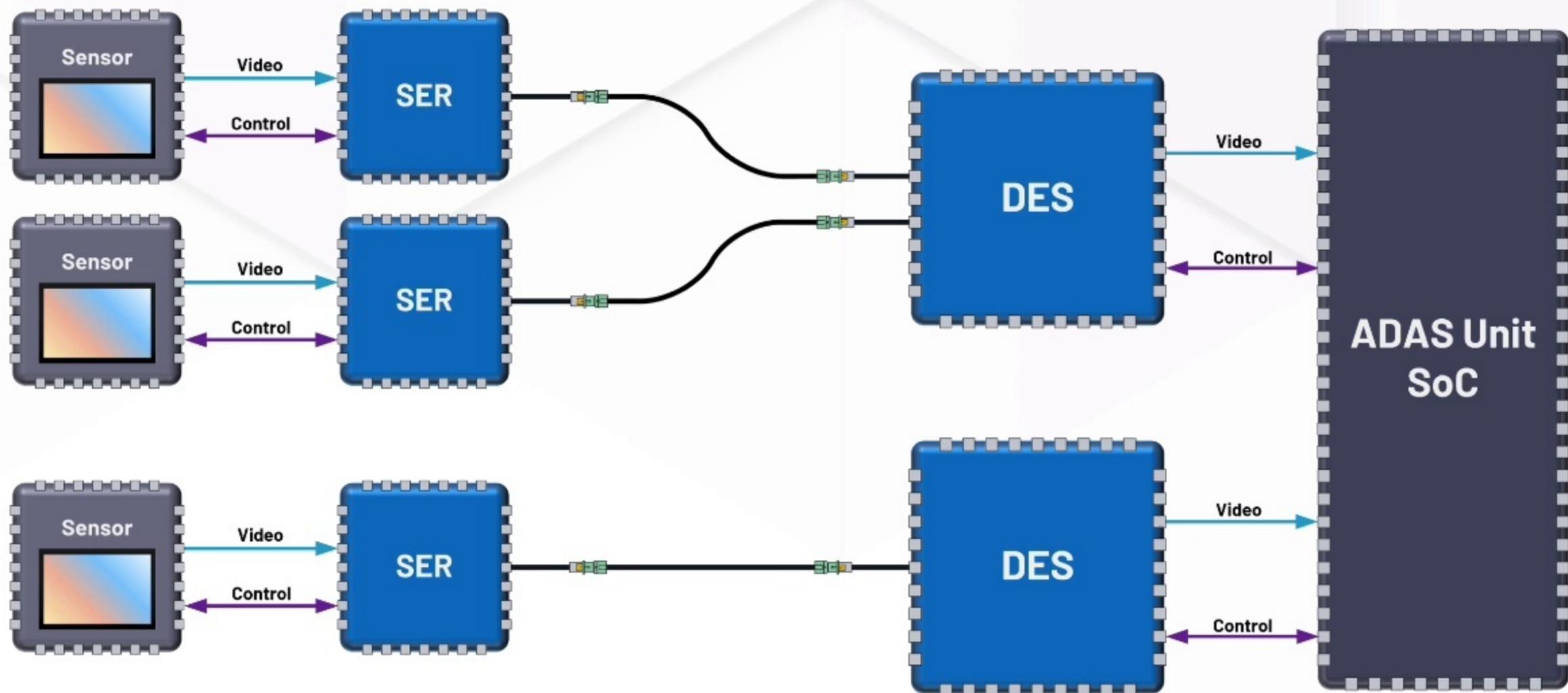
Diagram represents current, future, and concept configurations



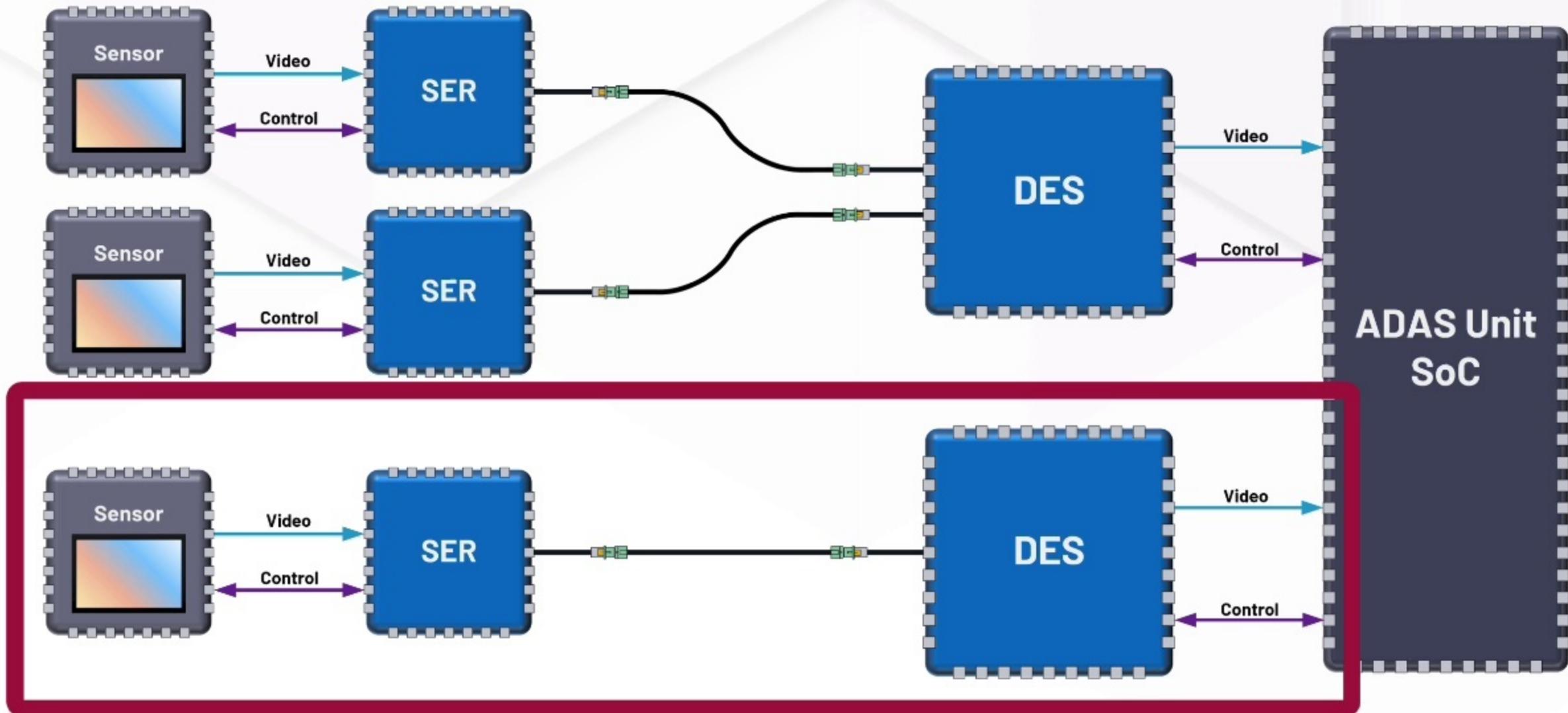
Today we will discuss the benefits and functionality of



GPIO's on our block diagram.



This will be on the bidirectional communication



channel between the serializer and deserializer.



In this course we'll define what the purpose of GPIO's is, define

- ▶ Define the purpose of GPIOs
- ▶ Define the system benefits of GPIOs in a GMSL system

the system benefits of GPIO's, explain the terminology of GPIO

- ▶ Define the purpose of GPIOs
- ▶ Define the system benefits of GPIOs in a GMSL system
- ▶ Explain the terminology of GPIO in the context of a GMSL link

in the context of a GMSL link, explain the methods of

- ▶ Define the purpose of GPIOs
- ▶ Define the system benefits of GPIOs in a GMSL system
- ▶ Explain the terminology of GPIO in the context of a GMSL link
- ▶ Explain the methods of interfacing with the GPIOs

interfacing with the GPIO's, and outline the options for GPIO

Look for the star  
for the answers!

- ▶ Define the purpose of GPIOs
- ▶ Define the system benefits of GPIOs in a GMSL system
- ▶ Explain the terminology of GPIO in the context of a GMSL link
- ▶ Explain the methods of interfacing with the GPIOs
- ▶ Outline the options for GPIO configurations

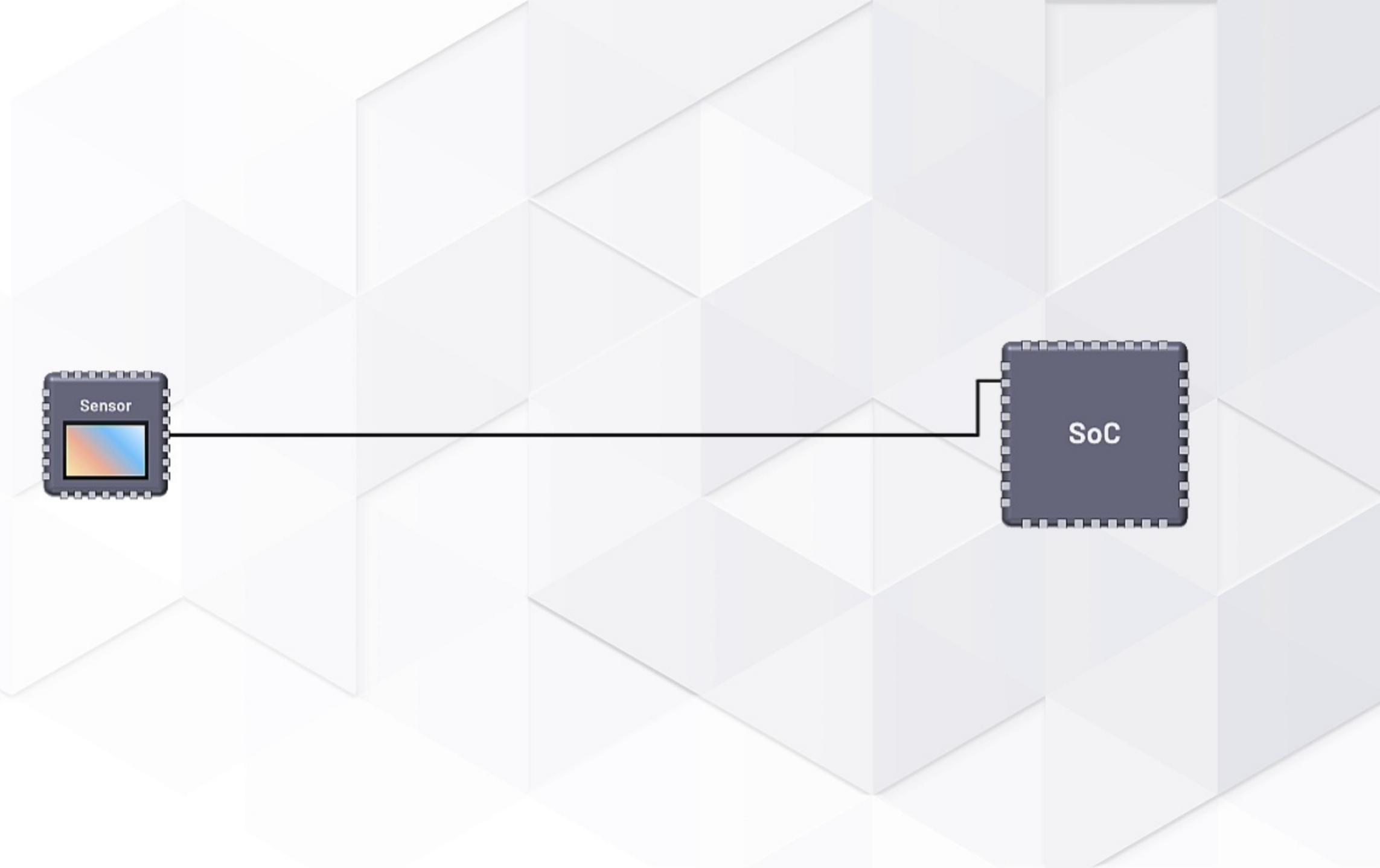
configurations such as pull up or pull downs, transmit and



Look for the star  
for the answers!

- ▶ Define the purpose of GPIOs
- ▶ Define the system benefits of GPIOs in a GMSL system
- ▶ Explain the terminology of GPIO in the context of a GMSL link
- ▶ Explain the methods of interfacing with the GPIOs
- ▶ Outline the options for GPIO configurations

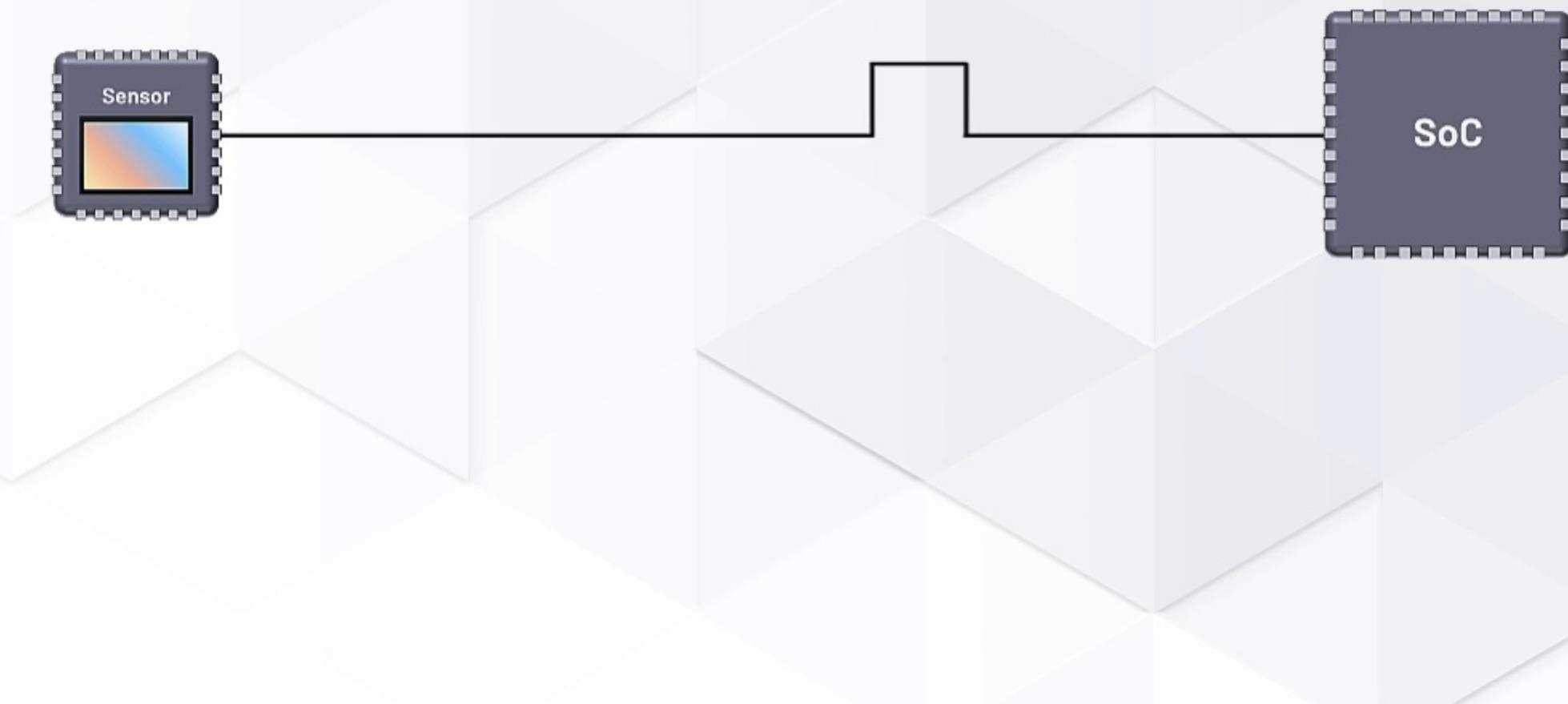
receive IDs, and more.



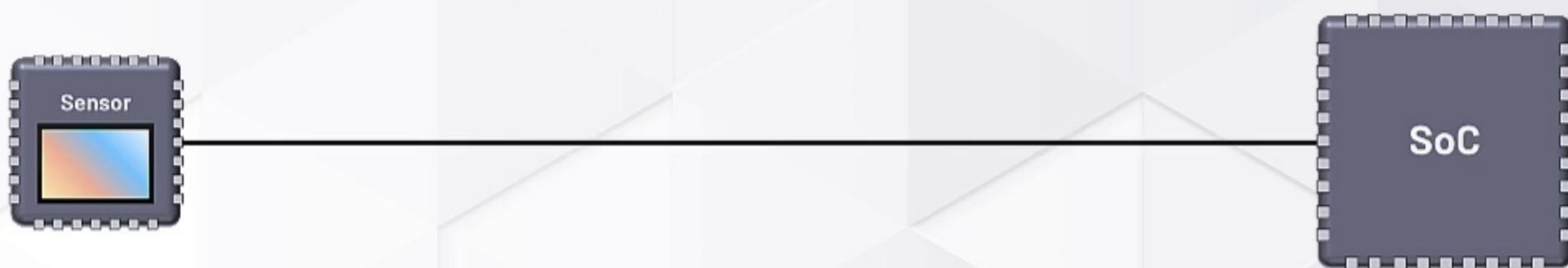
At the core of even the most complicated systems, a simple



pulse like a reset signal, control pulse or synchronization



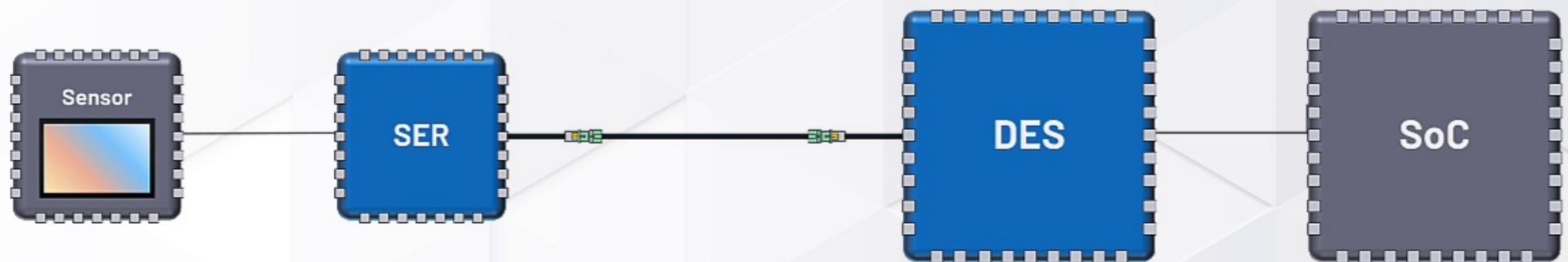
pulse are needed for low level commands.



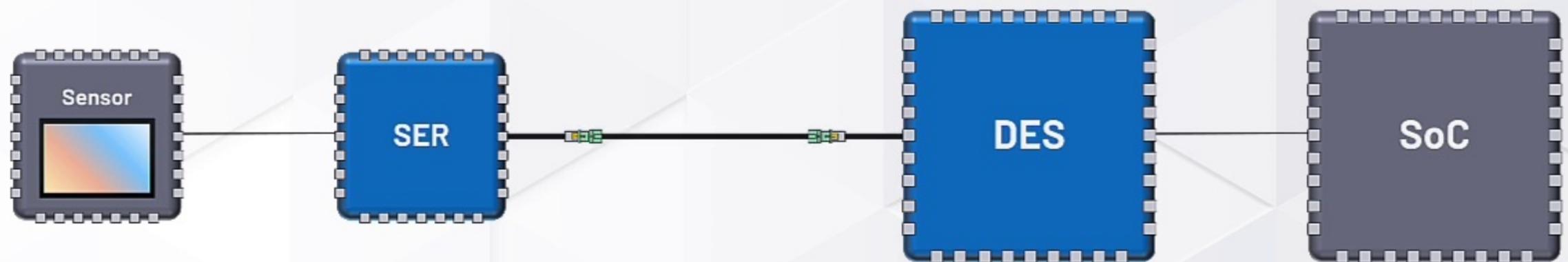
That is where the humble GPIO can be the companion to these



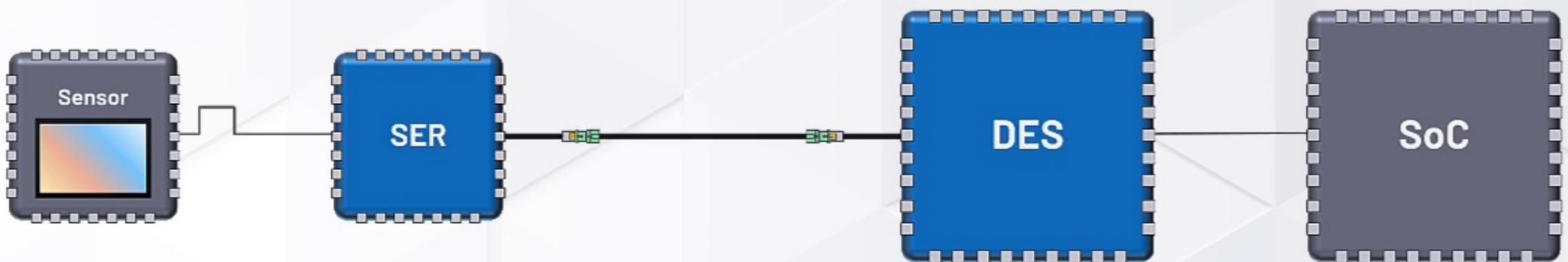
complex systems.



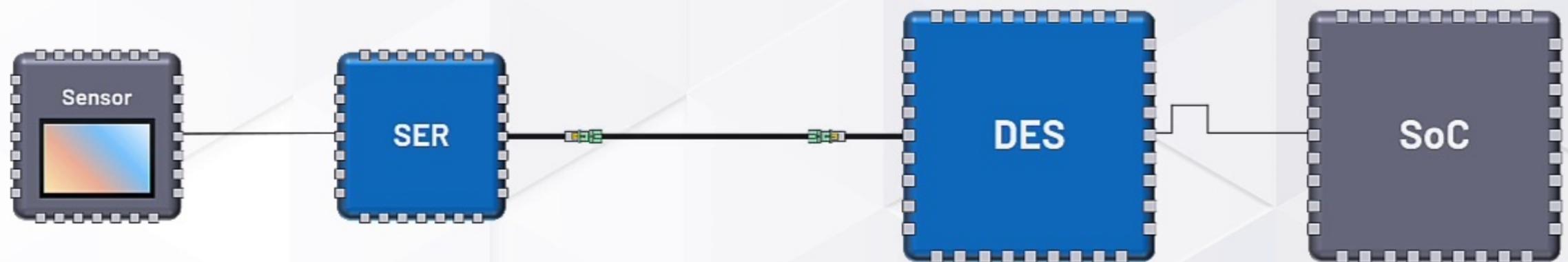
In systems with GMSL, the SerDes devices can transport this



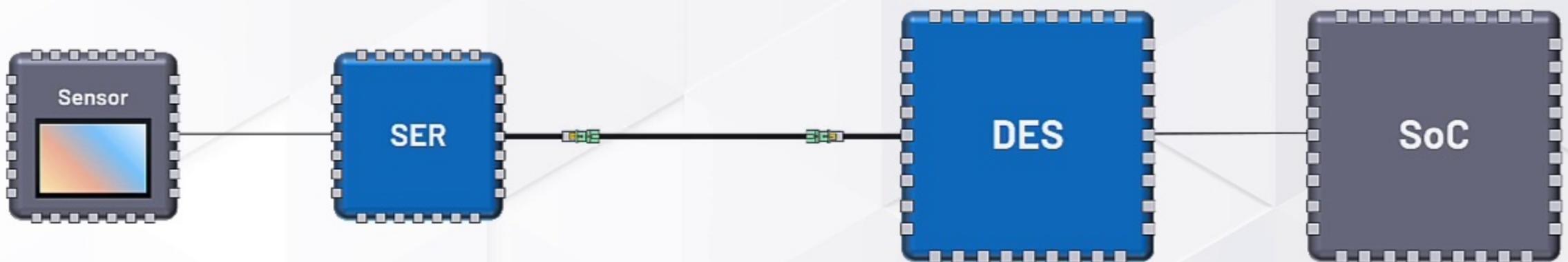
signal over long distances to help a remote sensor communicate



with a local microcontroller.



The SerDes devices convert this discrete signal into a packet to



send over the GMSL link.

G  
P  
I  
O

GPIO stands for General Purpose Input or Output.



It's a simple feature that can be utilized for purposes of



outputting digital signals or receiving digital signal inputs



with a logic high or a logic low voltage.



GPIO's are extremely common and almost assumed to be on every



microcontroller or processor.



In general, GPIO's do not provide significant current to output



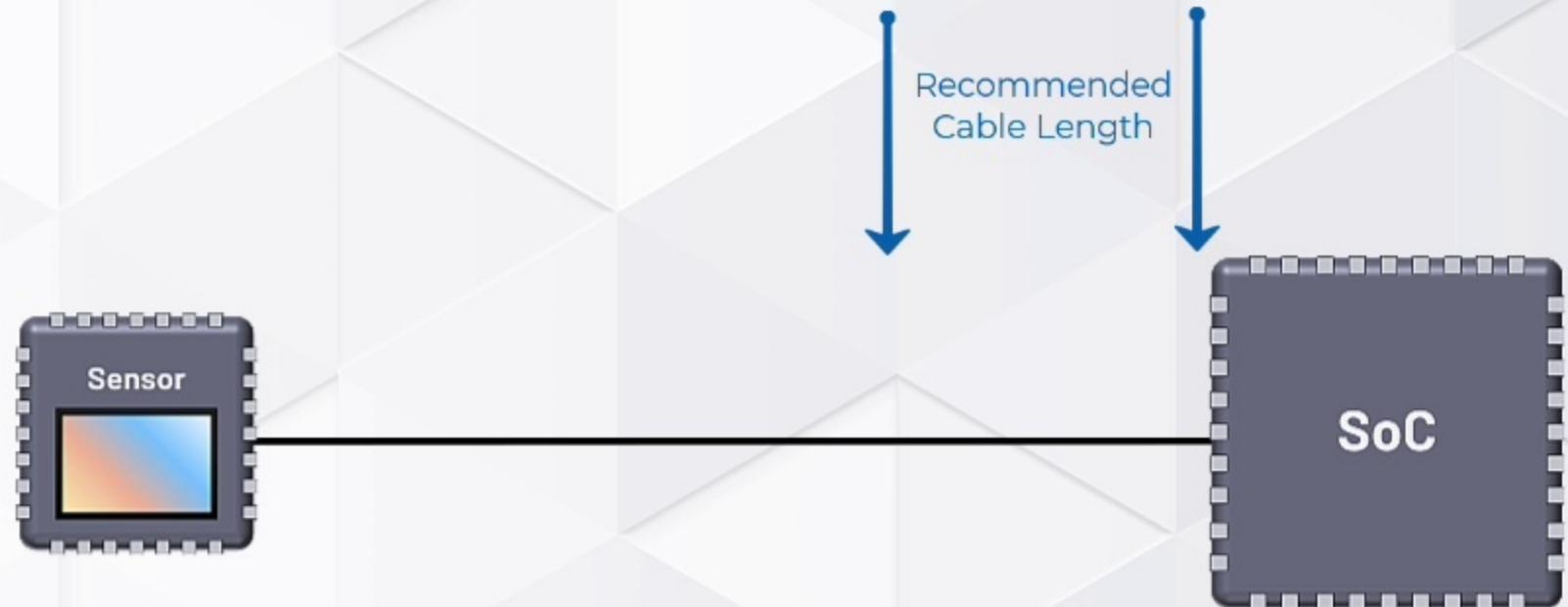
loads and instead operate at standard logic levels as lower



speed control signals.



GPIO's are useful, but their usefulness quickly diminishes



when faced with great distance between GPIO interfaces.



Note: The recommended cable length varies due to  
multiple factors not discussed in this video  
See GMSL211B Insertion and Return Loss

How useful are GPIO's in an application where a sensor and



Note: The recommended cable length varies due to  
multiple factors not discussed in this video  
See GMSL211B Insertion and Return Loss

its controller are a long distance apart?



Note: The recommended cable length varies due to  
multiple factors not discussed in this video  
See GMSL211B Insertion and Return Loss

A standard logic level signal over a wire does not travel



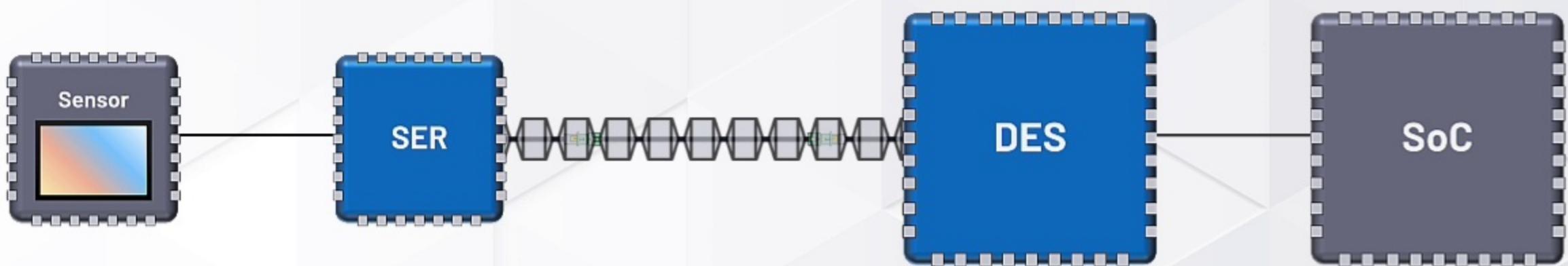
Note: The recommended cable length varies due to  
multiple factors not discussed in this video  
See GMSL211B Insertion and Return Loss

extremely far and it will slowly degrade due to the losses in the

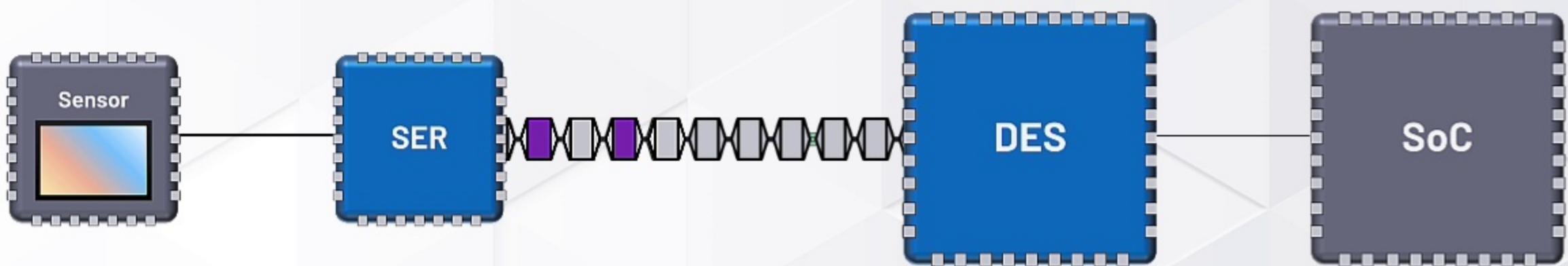


Note: The recommended cable length varies due to  
multiple factors not discussed in this video  
See GMSL211B Insertion and Return Loss

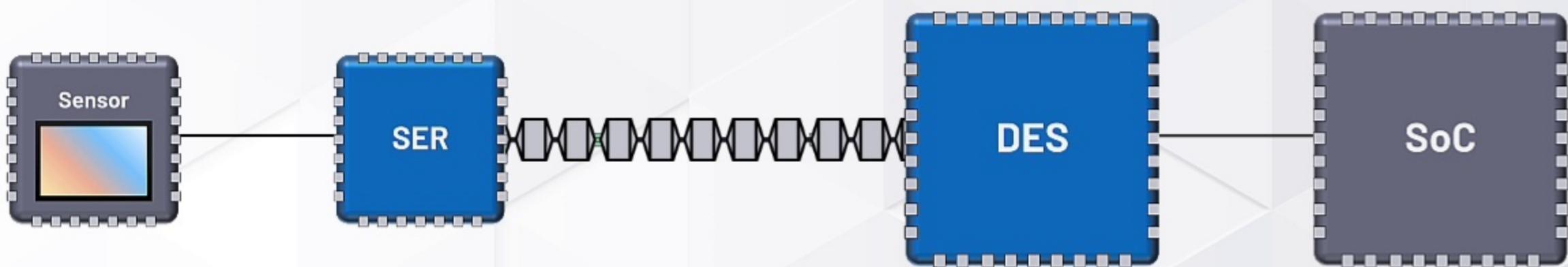
cable.



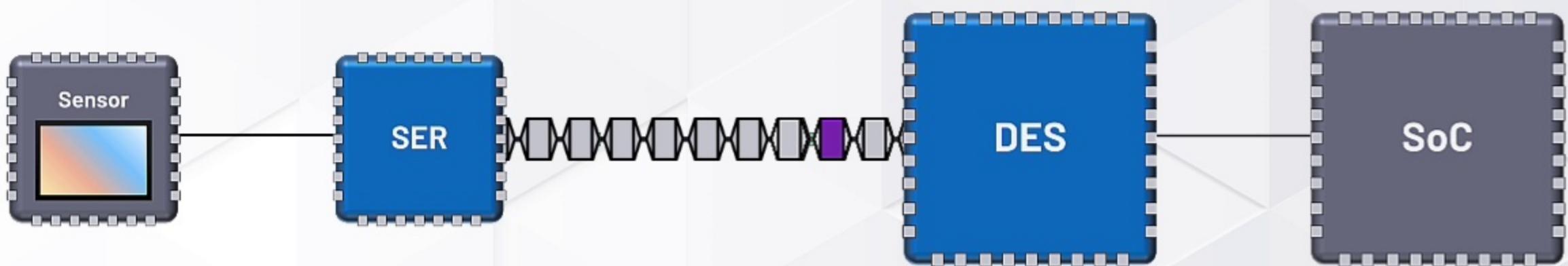
However, with GMSL in the system, there are already long



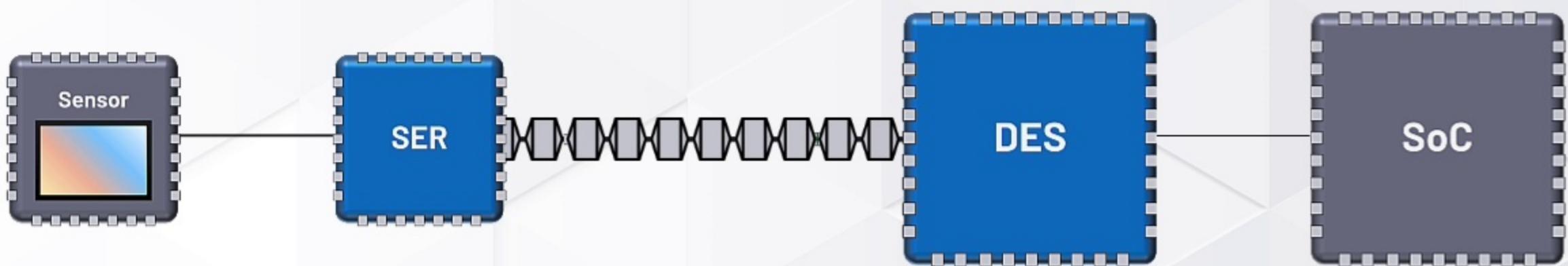
video and data paths that go back and forth between these



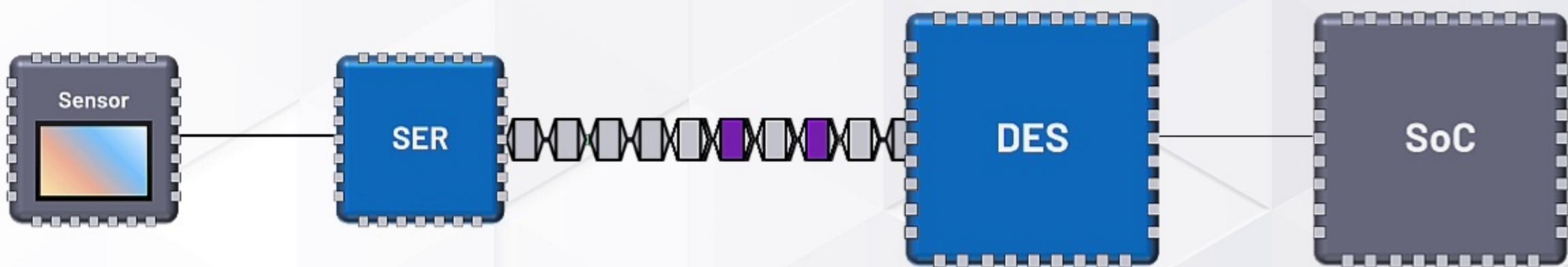
applications.



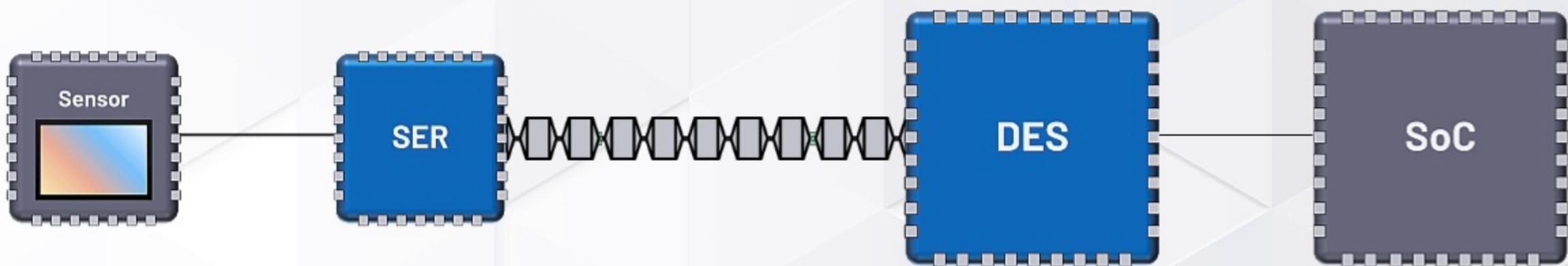
So is there a way to utilize the GMSL link?



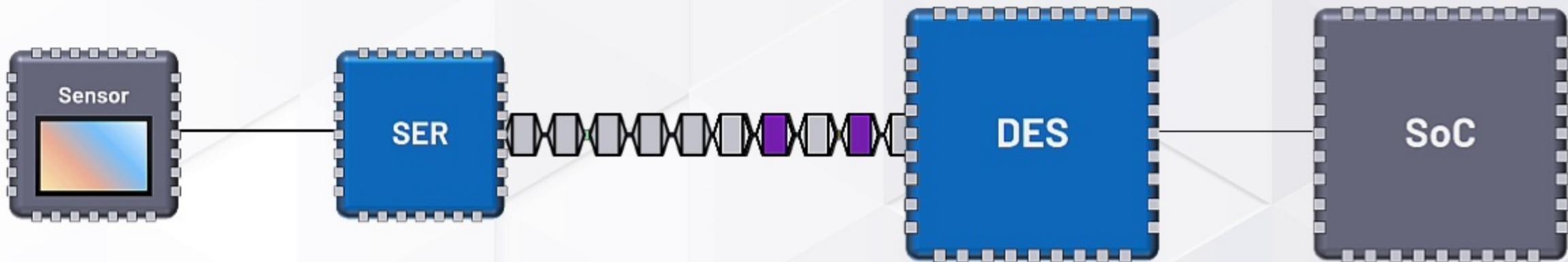
This is where GMSL comes in to allow long distance GPIO state



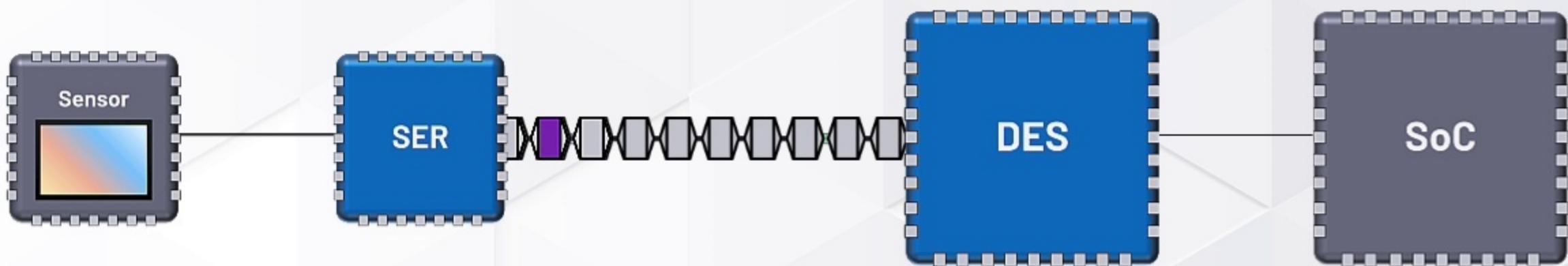
transmission through its packetized link.



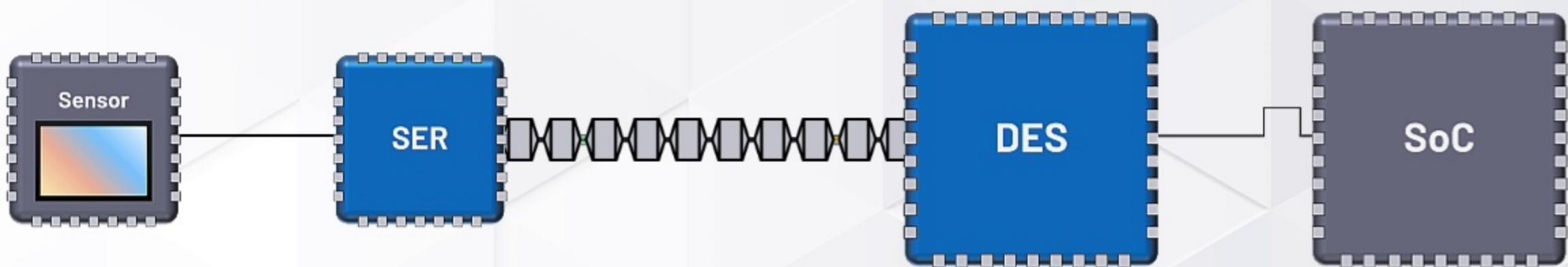
Since the GMSL link is a packetized one, it allows for



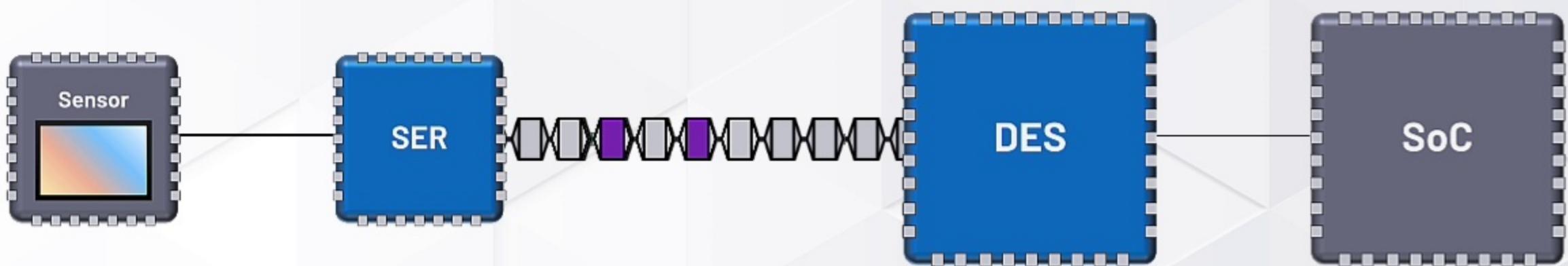
diverse applications.



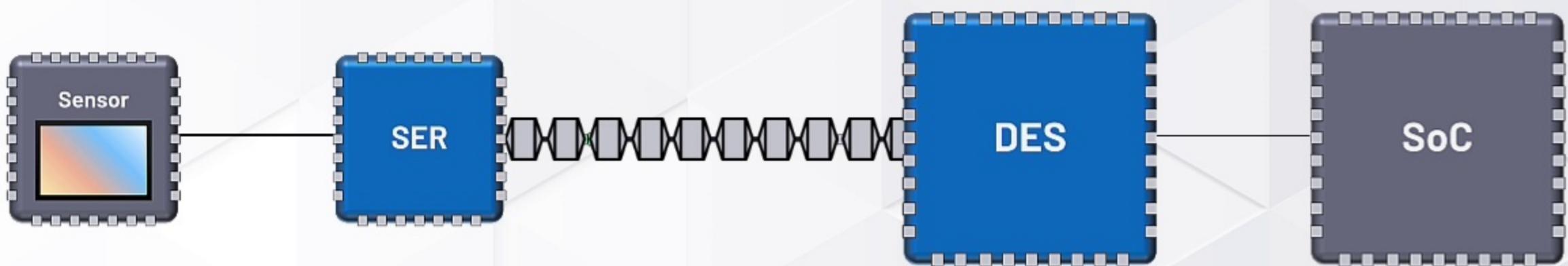
These packets allow GPIO state transmission at the same time as



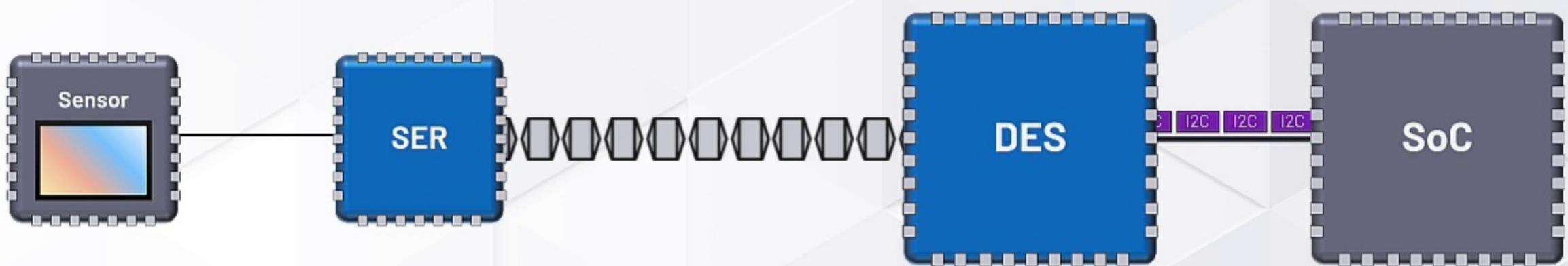
video and other data within the link.



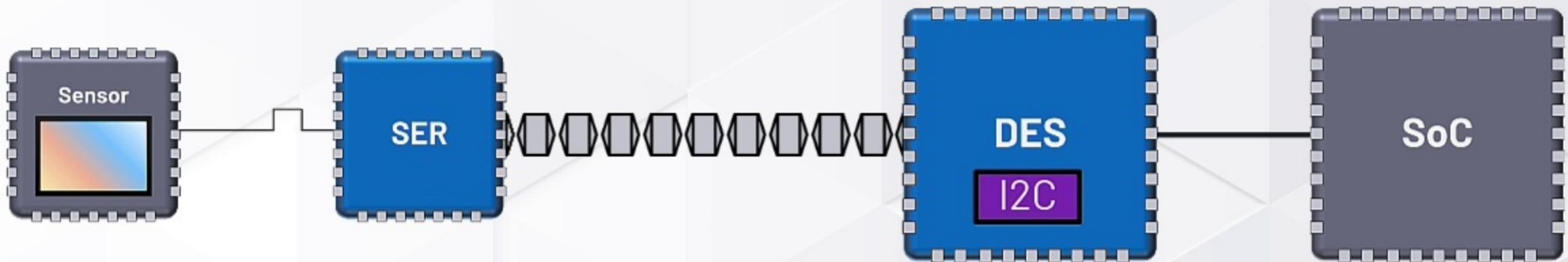
GMSL certis can transmit GPIO states that are received on the



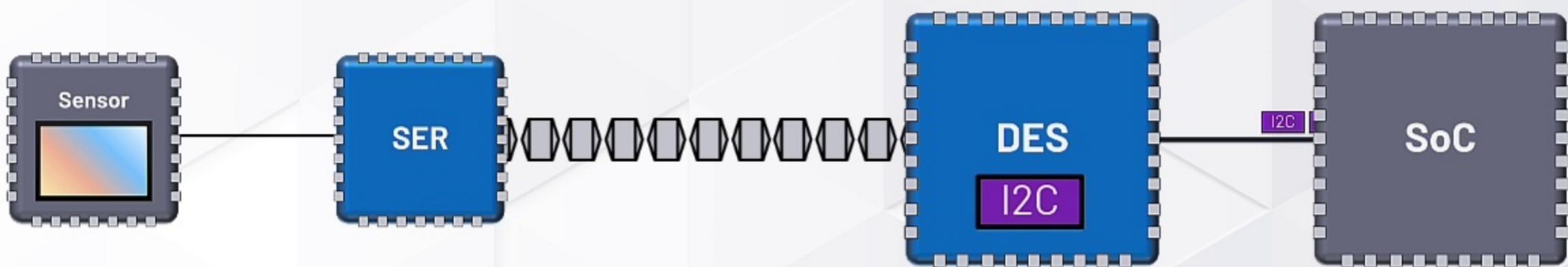
pins from one side to the other.



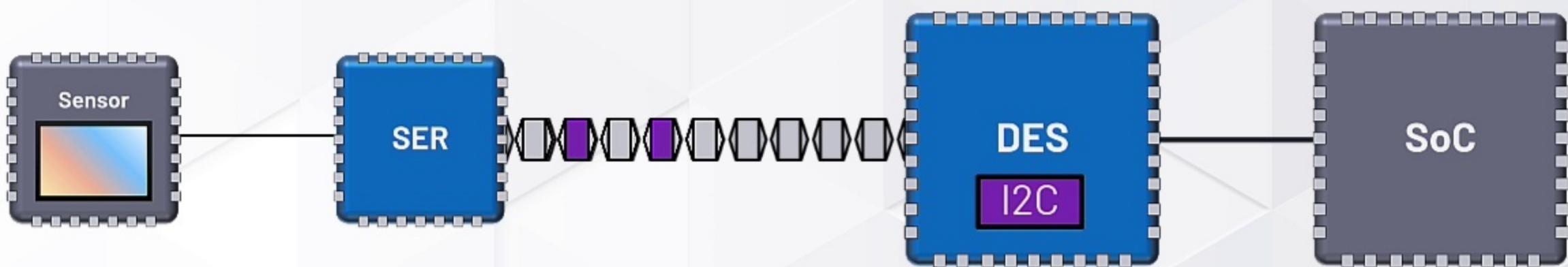
The GMSL certis is also capable of toggling the GPIO statement



program via registers.



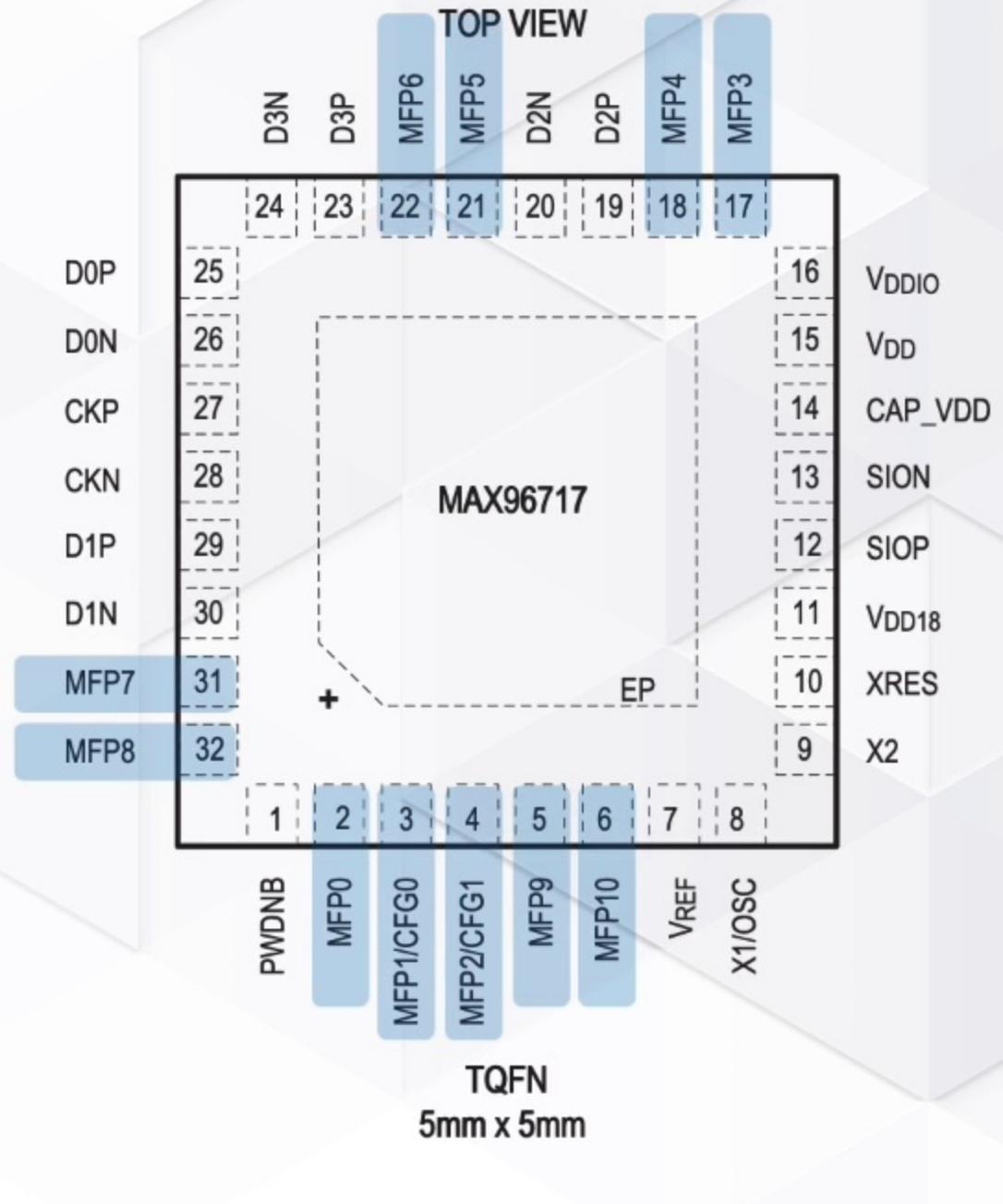
A microcontroller can send an I<sup>2</sup>C command over the GMSL link



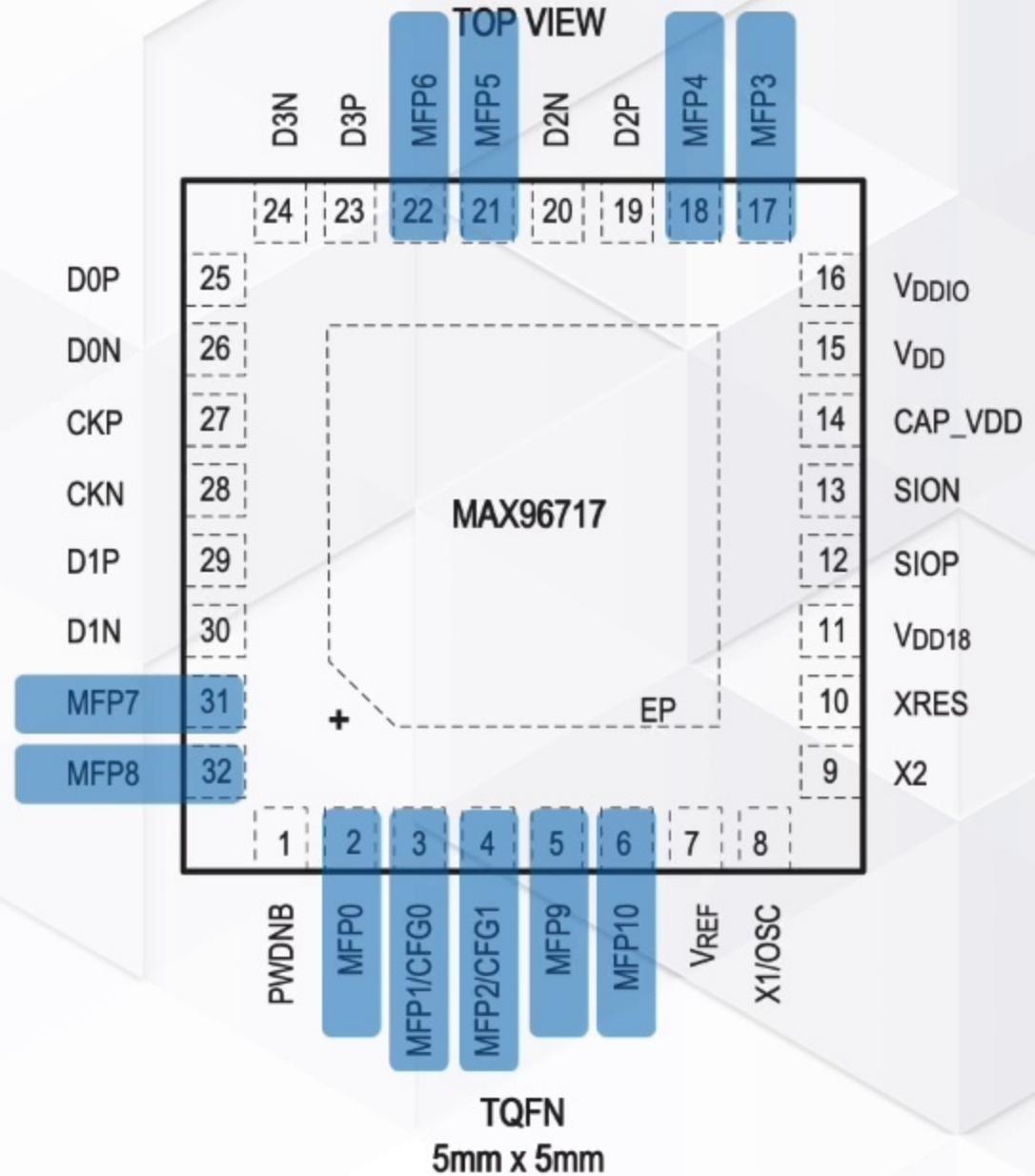
to remotely control the GPIO on the other end of the link.



First let us look inside the devices.



The GPIO feature exists on all GMSL devices and you can access



this feature through the multi function pins or MFP's.



The main consideration is to determine the direction of the

**GPIO 0**

BIT	7	6	5	4	3	2	1	0
Field	RES_CFG	RSVD	TX_COMP_EN	GPIO_OUT	GPIO_IN	GPIO_RX_EN	GPIO_TX_EN	GPIO_OUT_DIS
Reset	0b1	0b0	0b0	0b1	0b1	0b0	0b0	0b1
Access Type	Write, Read		Write, Read	Write, Read	Read Only	Write, Read	Write, Read	Write, Read



BITFIELD	BITS	DESCRIPTION	DECODE
RES_CFG	7	Resistor pullup/pulldown strength	0b0: 40kΩ 0b1: 1MΩ
TX_COMP_E_N	5	Jitter minimization compensation enable	0b0: Jitter compensation disabled 0b1: Jitter compensation enabled
GPIO_OUT	4	GPIO pin output drive value when GPIO_RX_EN = 0. This can be used to drive a value out on an MFP.	0b0: This GPIO pin output is driven to 0 0b1: This GPIO pin output is driven to 1
GPIO_IN	3	GPIO pin local MFP input level	0b0: This GPIO pin value is 0 0b1: This GPIO pin value is 1
GPIO_RX_E_N	2	GPIO out source control. Set to 1 to receive GPIO value from GMSL link. Set GPIO_OUT_DIS to 0 to output the received value on the local MFP.	0b0: This GPIO source disabled for GMSL reception 0b1: This GPIO source enabled for GMSL reception
GPIO_TX_E_N	1	GPIO Tx source control	0b0: This GPIO source disabled for GMSL transmission 0b1: This GPIO source enabled for GMSL transmission
GPIO_OUT_DIS	0	Disable GPIO output driver	0b0: Output driver enabled 0b1: Output driver disabled

GPIO signal transmission.

**GPIO 0**

BIT	7	6	5	4	3	2	1	0
Field	RES_CFG	RSVD	TX_COMP_EN	GPIO_OUT	GPIO_IN	GPIO_RX_EN	GPIO_TX_EN	GPIO_OUT_DIS
Reset	0b1	0b0	0b0	0b1	0b1	0b0	0b0	0b1
Access Type	Write, Read		Write, Read	Write, Read	Read Only	Write, Read	Write, Read	Write, Read



BITFIELD	BITS	DESCRIPTION	DECODE
RES_CFG	7	Resistor pullup/pulldown strength	0b0: 40kΩ 0b1: 1MΩ
TX_COMP_E_N	5	Jitter minimization compensation enable	0b0: Jitter compensation disabled 0b1: Jitter compensation enabled
GPIO_OUT	4	GPIO pin output drive value when GPIO_RX_EN = 0. This can be used to drive a value out on an MFP.	0b0: This GPIO pin output is driven to 0 0b1: This GPIO pin output is driven to 1
GPIO_IN	3	GPIO pin local MFP input level	0b0: This GPIO pin value is 0 0b1: This GPIO pin value is 1
GPIO_RX_E_N	2	GPIO out source control. Set to 1 to receive GPIO value from GMSL link. Set GPIO_OUT_DIS to 0 to output the received value on the local MFP.	0b0: This GPIO source disabled for GMSL reception 0b1: This GPIO source enabled for GMSL reception
GPIO_TX_E_N	1	GPIO Tx source control	0b0: This GPIO source disabled for GMSL transmission 0b1: This GPIO source enabled for GMSL transmission
GPIO_OUT_DIS	0	Disable GPIO output driver	0b0: Output driver enabled 0b1: Output driver disabled

Every GPIO can be configured as an input, output or input slash

**GPIO 0**

BIT	7	6	5	4	3	2	1	0
Field	RES_CFG	RSVD	TX_COMP_EN	GPIO_OUT	GPIO_IN	GPIO_RX_EN	GPIO_TX_EN	GPIO_OUT_DIS
Reset	0b1	0b0	0b0	0b1	0b1	0b0	0b0	0b1
Access Type	Write, Read		Write, Read	Write, Read	Read Only	Write, Read	Write, Read	Write, Read



BITFIELD	BITS	DESCRIPTION	DECODE
RES_CFG	7	Resistor pullup/pulldown strength	0b0: 40kΩ 0b1: 1MΩ
TX_COMP_E_N	5	Jitter minimization compensation enable	0b0: Jitter compensation disabled 0b1: Jitter compensation enabled
GPIO_OUT	4	GPIO pin output drive value when GPIO_RX_EN = 0. This can be used to drive a value out on an MFP.	0b0: This GPIO pin output is driven to 0 0b1: This GPIO pin output is driven to 1
GPIO_IN	3	GPIO pin local MFP input level	0b0: This GPIO pin value is 0 0b1: This GPIO pin value is 1
GPIO_RX_E_N	2	GPIO out source control. Set to 1 to receive GPIO value from GMSL link. Set GPIO_OUT_DIS to 0 to output the received value on the local MFP.	0b0: This GPIO source disabled for GMSL reception 0b1: This GPIO source enabled for GMSL reception
GPIO_TX_E_N	1	GPIO Tx source control	0b0: This GPIO source disabled for GMSL transmission 0b1: This GPIO source enabled for GMSL transmission
GPIO_OUT_DIS	0	Disable GPIO output driver	0b0: Output driver enabled 0b1: Output driver disabled

output.

**GPIO 0**

BIT	7	6	5	4	3	2	1	0
Field	RES_CFG	RSVD	TX_COMP_EN	GPIO_OUT	GPIO_IN	GPIO_RX_EN	GPIO_TX_EN	GPIO_OUT_DIS
Reset	0b1	0b0	0b0	0b1	0b1	0b0	0b0	0b1
Access Type	Write, Read		Write, Read	Write, Read	Read Only	Write, Read	Write, Read	Write, Read



BITFIELD	BITS	DESCRIPTION	DECODE
RES_CFG	7	Resistor pullup/pulldown strength	0b0: 40kΩ 0b1: 1MΩ
TX_COMP_E_N	5	Jitter minimization compensation enable	0b0: Jitter compensation disabled 0b1: Jitter compensation enabled
GPIO_OUT	4	GPIO pin output drive value when GPIO_RX_EN = 0. This can be used to drive a value out on an MFP.	0b0: This GPIO pin output is driven to 0 0b1: This GPIO pin output is driven to 1
GPIO_IN	3	GPIO pin local MFP input level	0b0: This GPIO pin value is 0 0b1: This GPIO pin value is 1
GPIO_RX_E_N	2	GPIO out source control. Set to 1 to receive GPIO value from GMSL link. Set GPIO_OUT_DIS to 0 to output the received value on the local MFP.	0b0: This GPIO source disabled for GMSL reception 0b1: This GPIO source enabled for GMSL reception
GPIO_TX_E_N	1	GPIO Tx source control	0b0: This GPIO source disabled for GMSL transmission 0b1: This GPIO source enabled for GMSL transmission
GPIO_OUT_DIS	0	Disable GPIO output driver	0b0: Output driver enabled 0b1: Output driver disabled

However the GPIO cannot be configured to be input and

**GPIO 0**

BIT	7	6	5	4	3	2	1	0
Field	RES_CFG	RSVD	TX_COMP_EN	GPIO_OUT	GPIO_IN	GPIO_RX_EN	GPIO_TX_EN	GPIO_OUT_DIS
Reset	0b1	0b0	0b0	0b1	0b1	0b0	0b0	0b1
Access Type	Write, Read		Write, Read	Write, Read	Read Only	Write, Read	Write, Read	Write, Read



BITFIELD	BITS	DESCRIPTION	DECODE
RES_CFG	7	Resistor pullup/pulldown strength	0b0: 40kΩ 0b1: 1MΩ
TX_COMP_E_N	5	Jitter minimization compensation enable	0b0: Jitter compensation disabled 0b1: Jitter compensation enabled
GPIO_OUT	4	GPIO pin output drive value when GPIO_RX_EN = 0. This can be used to drive a value out on an MFP.	0b0: This GPIO pin output is driven to 0 0b1: This GPIO pin output is driven to 1
GPIO_IN	3	GPIO pin local MFP input level	0b0: This GPIO pin value is 0 0b1: This GPIO pin value is 1
GPIO_RX_E_N	2	GPIO out source control. Set to 1 to receive GPIO value from GMSL link. Set GPIO_OUT_DIS to 0 to output the received value on the local MFP.	0b0: This GPIO source disabled for GMSL reception 0b1: This GPIO source enabled for GMSL reception
GPIO_TX_E_N	1	GPIO Tx source control	0b0: This GPIO source disabled for GMSL transmission 0b1: This GPIO source enabled for GMSL transmission
GPIO_OUT_DIS	0	Disable GPIO output driver	0b0: Output driver enabled 0b1: Output driver disabled

output at the same time.

**GPIO 0**

BIT	7	6	5	4	3	2	1	0
Field	RES_CFG	RSVD	TX_COMP_EN	GPIO_OUT	GPIO_IN	GPIO_RX_EN	GPIO_TX_EN	GPIO_OUT_DIS
Reset	0b1	0b0	0b0	0b1	0b1	0b0	0b0	0b1
Access Type	Write, Read		Write, Read	Write, Read	Read Only	Write, Read	Write, Read	Write, Read



BITFIELD	BITS	DESCRIPTION	DECODE
RES_CFG	7	Resistor pullup/pulldown strength	0b0: 40kΩ 0b1: 1MΩ
TX_COMP_E_N	5	Jitter minimization compensation enable	0b0: Jitter compensation disabled 0b1: Jitter compensation enabled
GPIO_OUT	4	GPIO pin output drive value when GPIO_RX_EN = 0. This can be used to drive a value out on an MFP.	0b0: This GPIO pin output is driven to 0 0b1: This GPIO pin output is driven to 1
GPIO_IN	3	GPIO pin local MFP input level	0b0: This GPIO pin value is 0 0b1: This GPIO pin value is 1
GPIO_RX_E_N	2	GPIO out source control. Set to 1 to receive GPIO value from GMSL link. Set GPIO_OUT_DIS to 0 to output the received value on the local MFP.	0b0: This GPIO source disabled for GMSL reception 0b1: This GPIO source enabled for GMSL reception
GPIO_TX_E_N	1	GPIO Tx source control	0b0: This GPIO source disabled for GMSL transmission 0b1: This GPIO source enabled for GMSL transmission
GPIO_OUT_DIS	0	Disable GPIO output driver	0b0: Output driver enabled 0b1: Output driver disabled

The GPIO signal direction must be determined to configure the

**GPIO 0**

BIT	7	6	5	4	3	2	1	0
Field	RES_CFG	RSVD	TX_COMP_EN	GPIO_OUT	GPIO_IN	GPIO_RX_EN	GPIO_TX_EN	GPIO_OUT_DIS
Reset	0b1	0b0	0b0	0b1	0b1	0b0	0b0	0b1
Access Type	Write, Read		Write, Read	Write, Read	Read Only	Write, Read	Write, Read	Write, Read



BITFIELD	BITS	DESCRIPTION	DECODE
RES_CFG	7	Resistor pullup/pulldown strength	0b0: 40kΩ 0b1: 1MΩ
TX_COMP_E_N	5	Jitter minimization compensation enable	0b0: Jitter compensation disabled 0b1: Jitter compensation enabled
GPIO_OUT	4	GPIO pin output drive value when GPIO_RX_EN = 0. This can be used to drive a value out on an MFP.	0b0: This GPIO pin output is driven to 0 0b1: This GPIO pin output is driven to 1
GPIO_IN	3	GPIO pin local MFP input level	0b0: This GPIO pin value is 0 0b1: This GPIO pin value is 1
GPIO_RX_E_N	2	GPIO out source control. Set to 1 to receive GPIO value from GMSL link. Set GPIO_OUT_DIS to 0 to output the received value on the local MFP.	0b0: This GPIO source disabled for GMSL reception 0b1: This GPIO source enabled for GMSL reception
GPIO_TX_E_N	1	GPIO Tx source control	0b0: This GPIO source disabled for GMSL transmission 0b1: This GPIO source enabled for GMSL transmission
GPIO_OUT_DIS	0	Disable GPIO output driver	0b0: Output driver enabled 0b1: Output driver disabled

GPIO pins as an INPUT on the serializer, an output on the

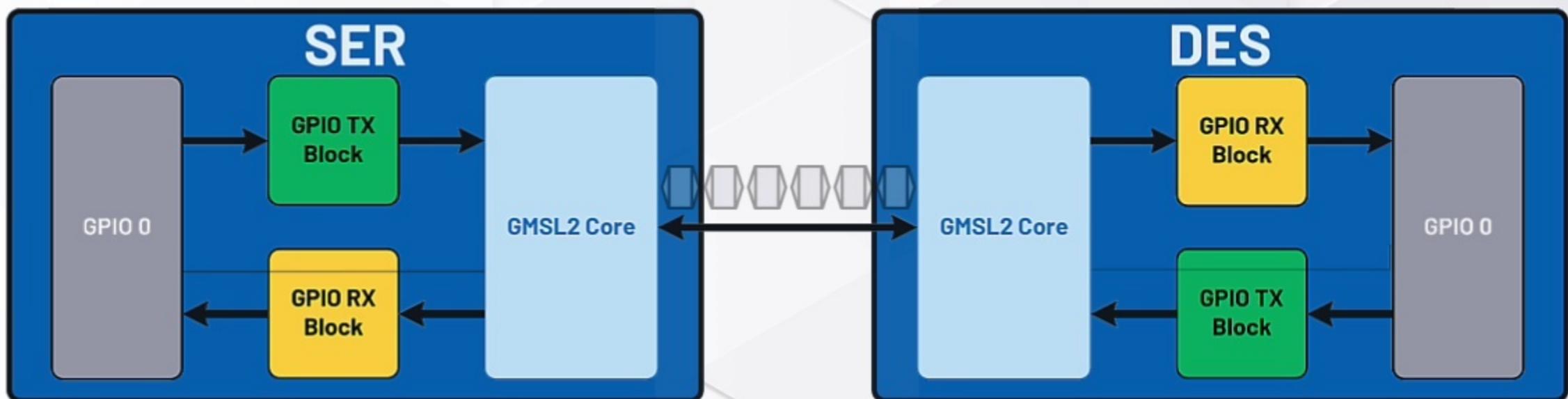
**GPIO 0**

BIT	7	6	5	4	3	2	1	0
Field	RES_CFG	RSVD	TX_COMP_EN	GPIO_OUT	GPIO_IN	GPIO_RX_EN	GPIO_TX_EN	GPIO_OUT_DIS
Reset	0b1	0b0	0b0	0b1	0b1	0b0	0b0	0b1
Access Type	Write, Read		Write, Read	Write, Read	Read Only	Write, Read	Write, Read	Write, Read



BITFIELD	BITS	DESCRIPTION	DECODE
RES_CFG	7	Resistor pullup/pulldown strength	0b0: 40kΩ 0b1: 1MΩ
TX_COMP_E_N	5	Jitter minimization compensation enable	0b0: Jitter compensation disabled 0b1: Jitter compensation enabled
GPIO_OUT	4	GPIO pin output drive value when GPIO_RX_EN = 0. This can be used to drive a value out on an MFP.	0b0: This GPIO pin output is driven to 0 0b1: This GPIO pin output is driven to 1
GPIO_IN	3	GPIO pin local MFP input level	0b0: This GPIO pin value is 0 0b1: This GPIO pin value is 1
GPIO_RX_E_N	2	GPIO out source control. Set to 1 to receive GPIO value from GMSL link. Set GPIO_OUT_DIS to 0 to output the received value on the local MFP.	0b0: This GPIO source disabled for GMSL reception 0b1: This GPIO source enabled for GMSL reception
GPIO_TX_E_N	1	GPIO Tx source control	0b0: This GPIO source disabled for GMSL transmission 0b1: This GPIO source enabled for GMSL transmission
GPIO_OUT_DIS	0	Disable GPIO output driver	0b0: Output driver enabled 0b1: Output driver disabled

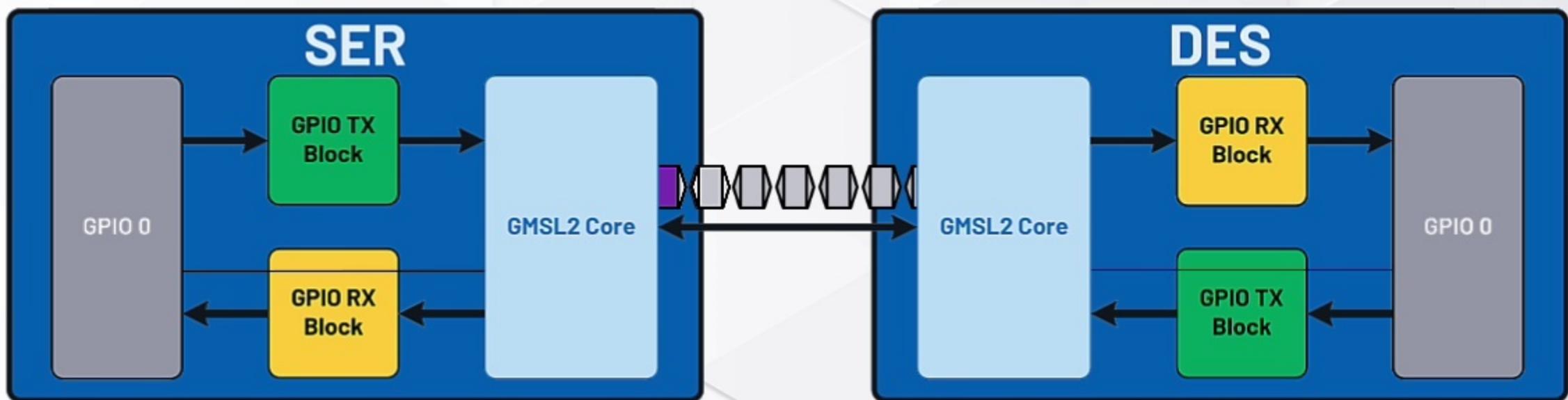
deserializer, or vice versa.



RX\_EN = 1 or Receive Enabled

TX\_EN = 1 or Transmit Enabled

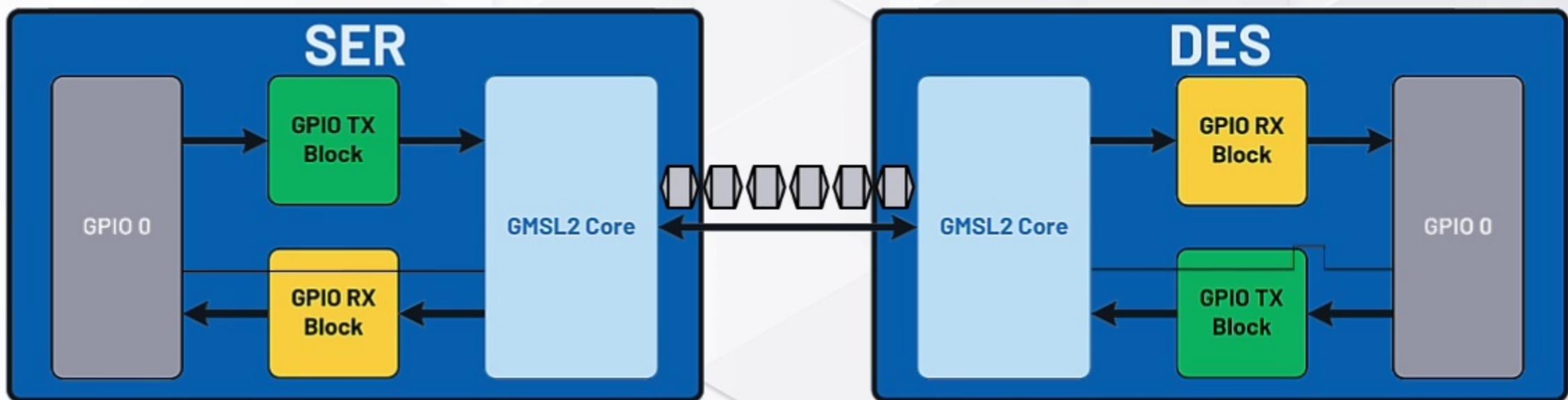
At this point it is good to clarify the transmit TX or



RX\_EN = 1 or Receive Enabled

TX\_EN = 1 or Transmit Enabled

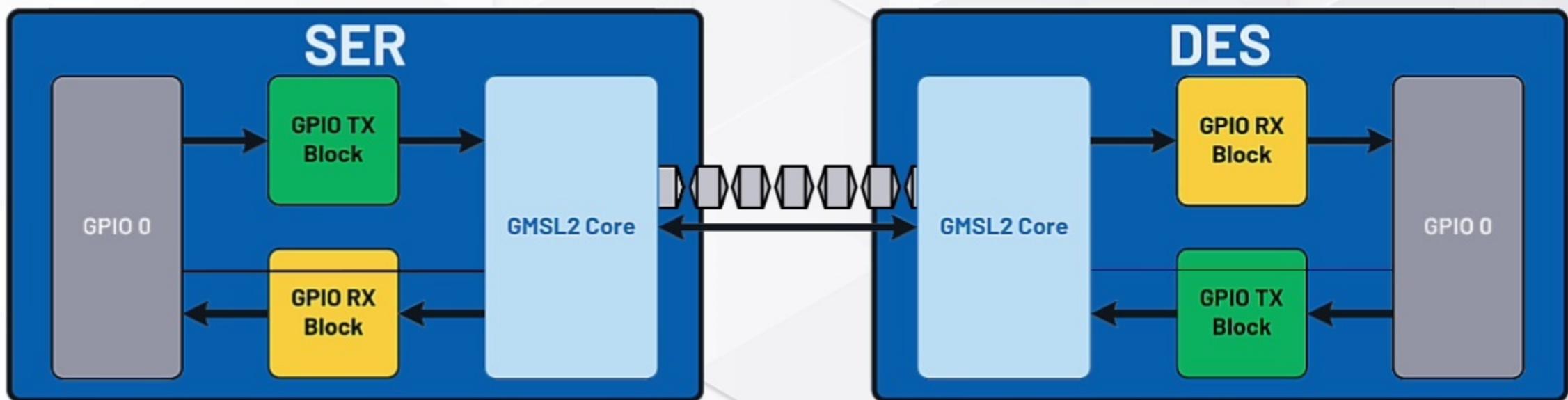
receive RX nomenclature that will be used with the GPI that



RX\_EN = 1 or Receive Enabled

TX\_EN = 1 or Transmit Enabled

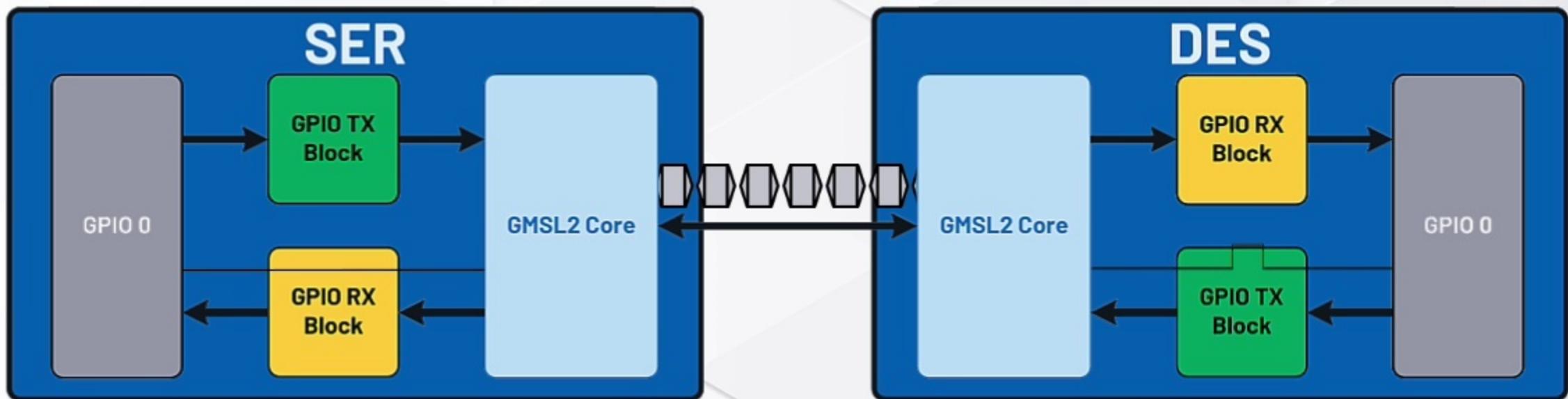
opens at configuration.



RX\_EN = 1 or Receive Enabled

TX\_EN = 1 or Transmit Enabled

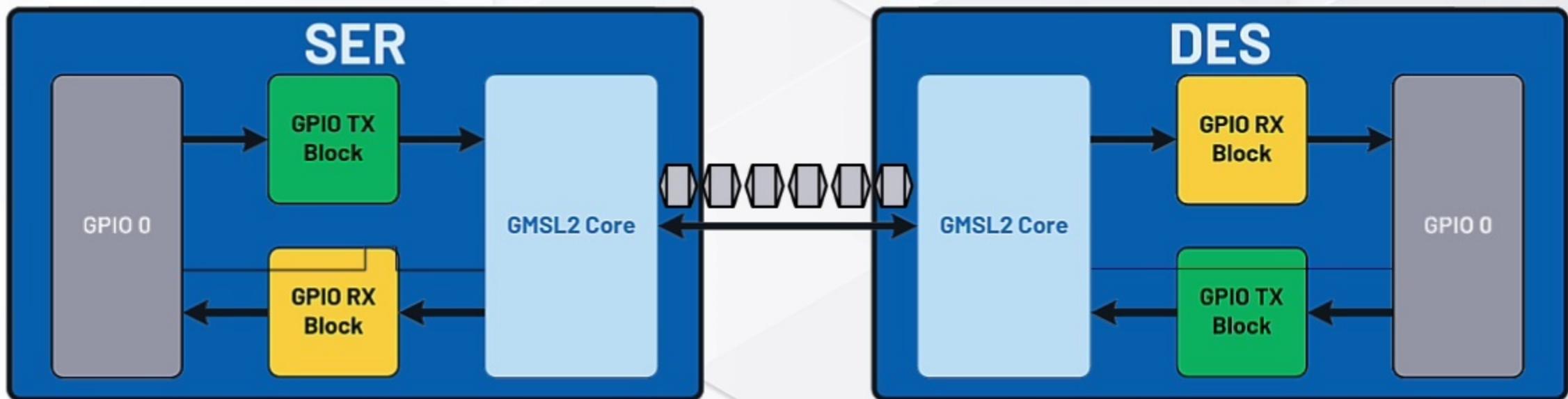
The GPIO pin enabled as TX will transmit its received logic



RX\_EN = 1 or Receive Enabled

TX\_EN = 1 or Transmit Enabled

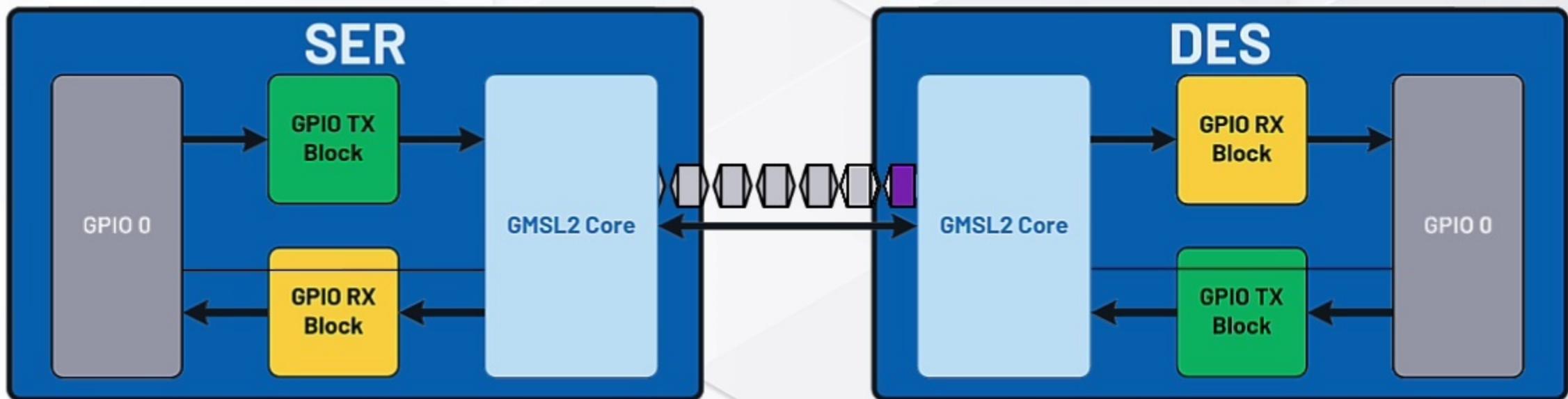
input pin state to the GMSL link.



RX\_EN = 1 or Receive Enabled

TX\_EN = 1 or Transmit Enabled

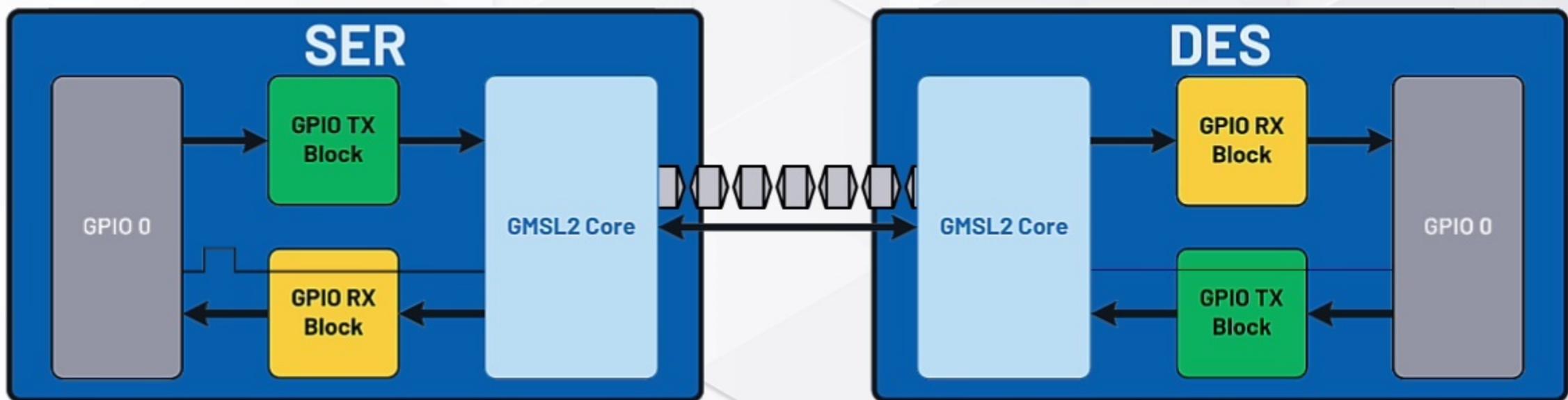
Conversely, the GPIO pin enabled as RX will receive the remote



RX\_EN = 1 or Receive Enabled

TX\_EN = 1 or Transmit Enabled

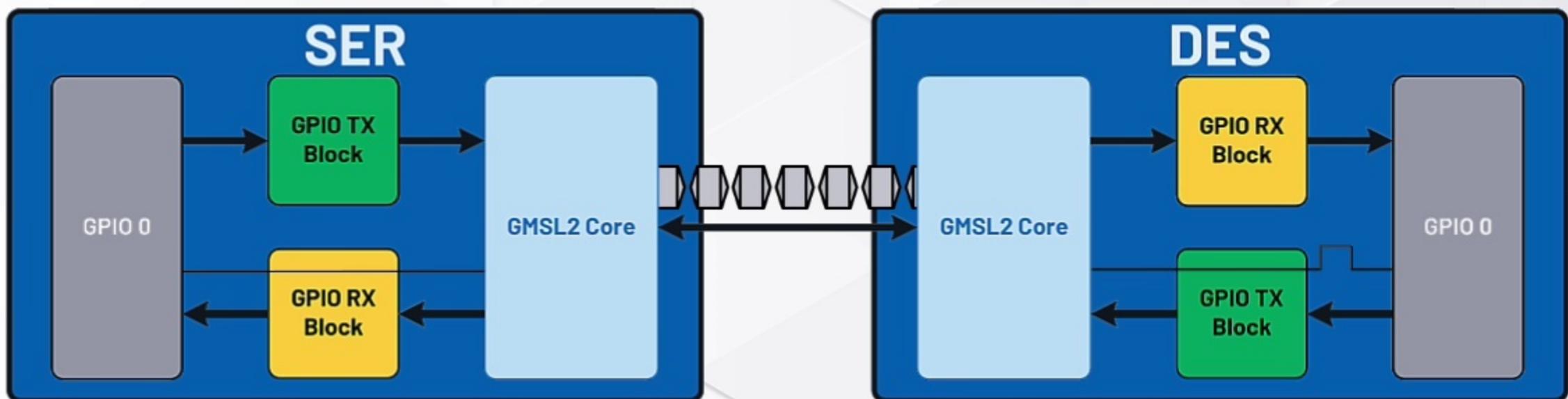
logic state from the GMSL link and transmit that state out to



RX\_EN = 1 or Receive Enabled

TX\_EN = 1 or Transmit Enabled

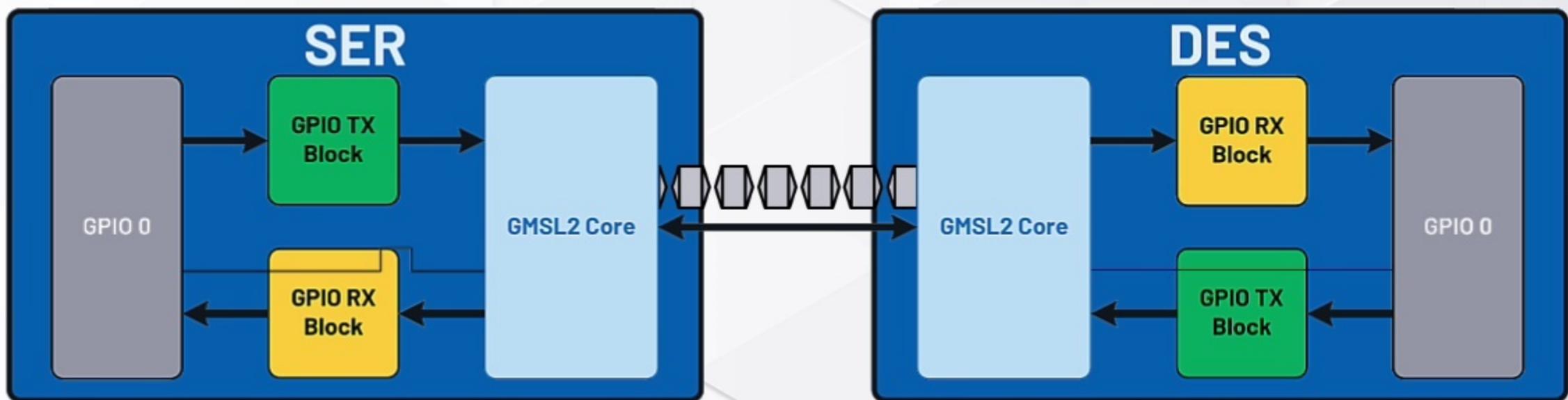
its local pin.



RX\_EN = 1 or Receive Enabled

TX\_EN = 1 or Transmit Enabled

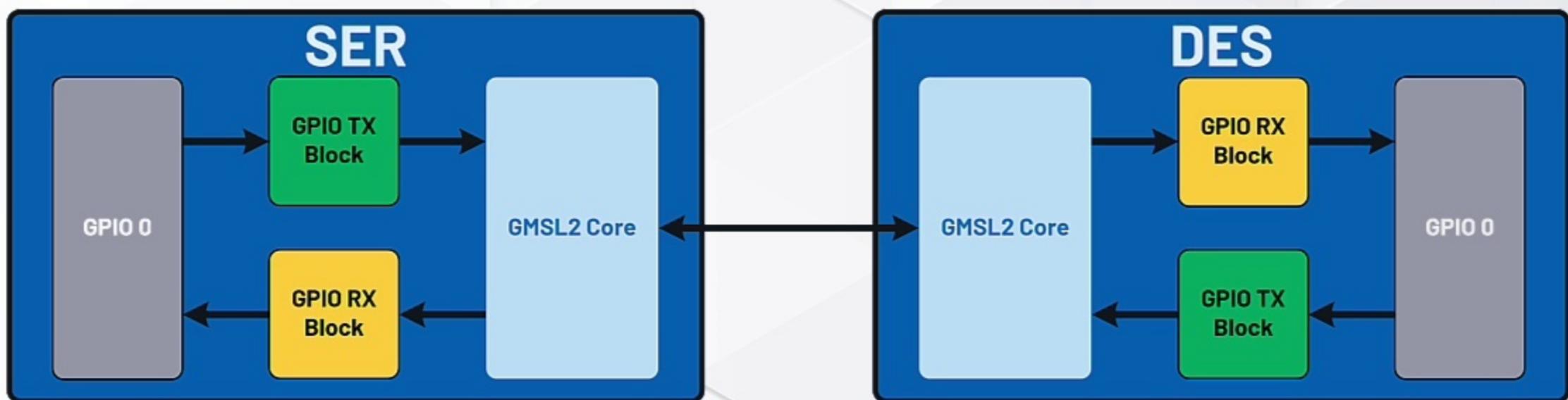
Remember that the transmit TX and receive RX is in reference



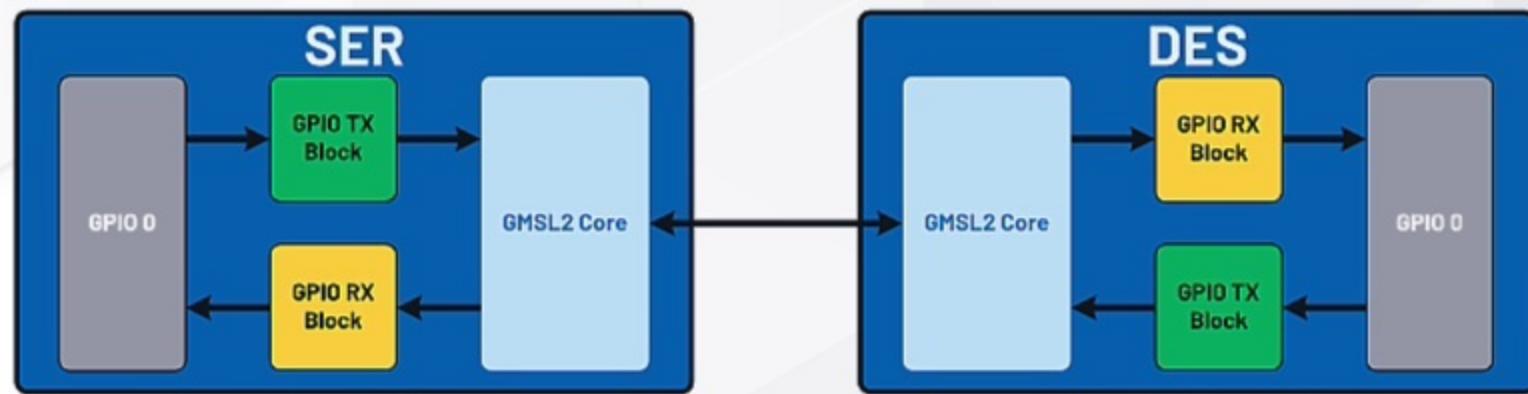
RX\_EN = 1 or Receive Enabled

TX\_EN = 1 or Transmit Enabled

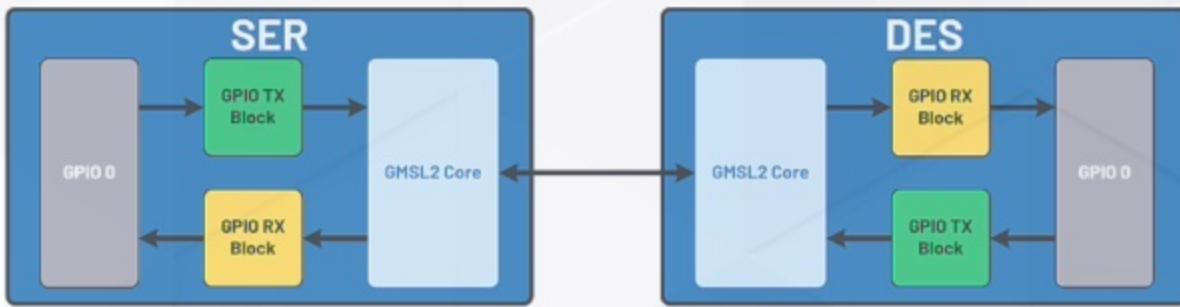
to the internal GMSL link.



Aside from the direction of the GPIO signal, there are a

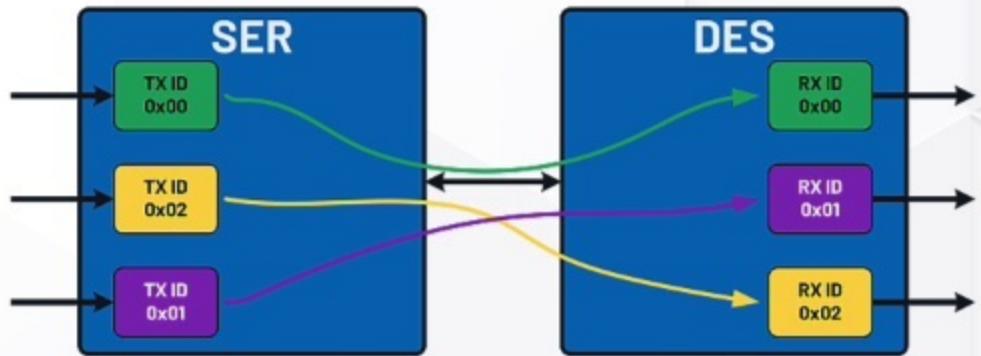


handful of settings used to set up the GPIO's based on the

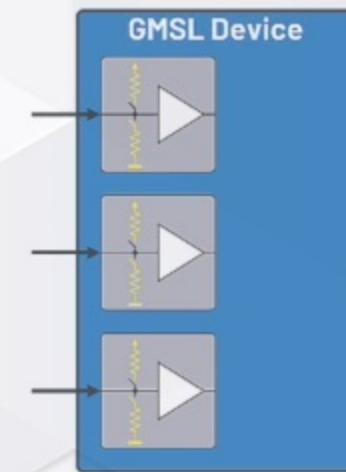


required need, such as the transmit and receive ID, pull

## Receive and Transmit ID Mapping

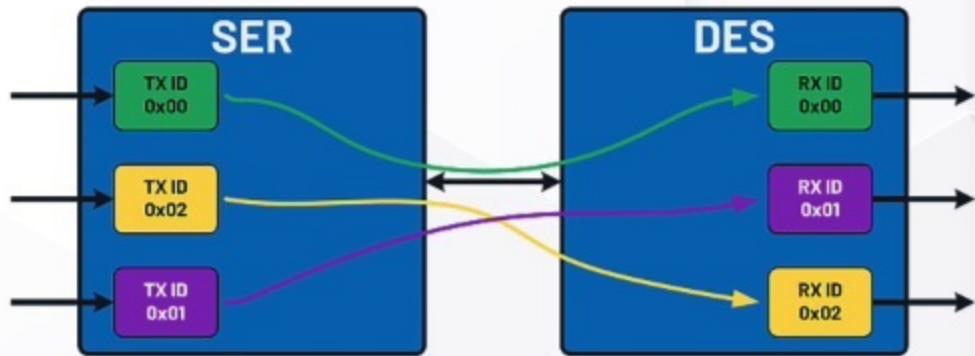


## Internal Pull-Up and Pull-Down Options

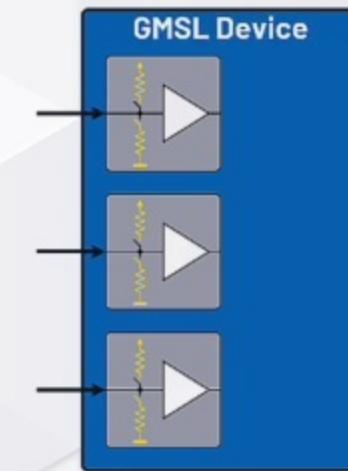


down, pull up configuration, jitter reduction, and status

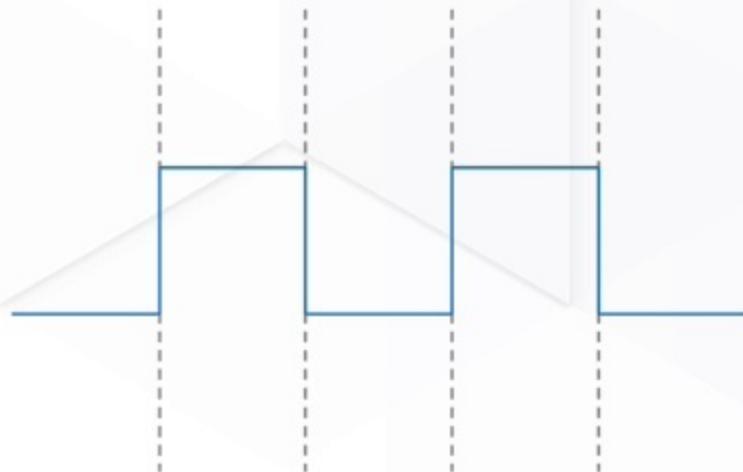
## Receive and Transmit ID Mapping



## Internal Pull-Up and Pull-Down Options



## Jitter Reduction

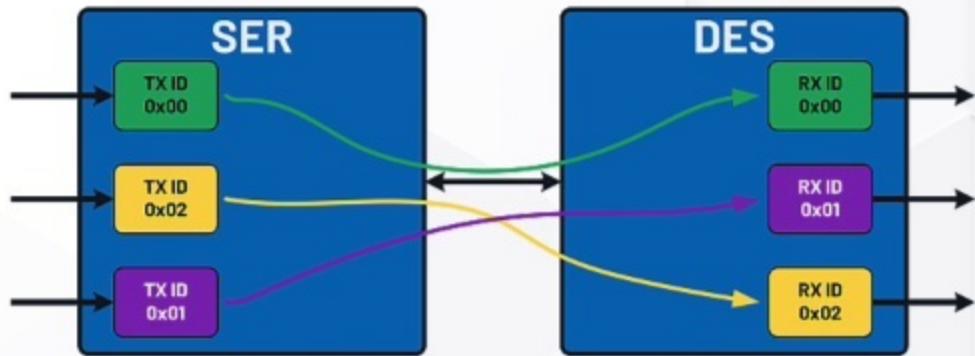


## Status Reporting

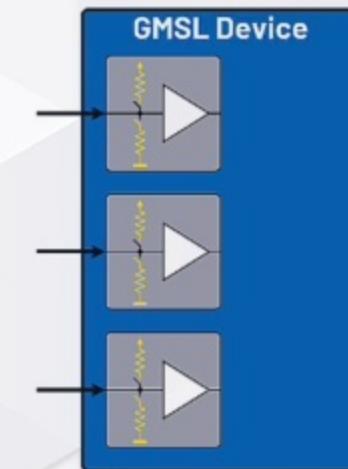
**GPIO Input Detected [✓]**

reporting.

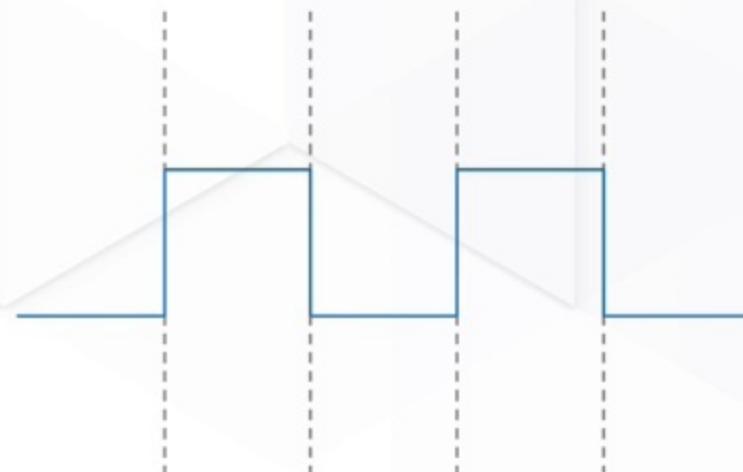
## Receive and Transmit ID Mapping



## Internal Pull-Up and Pull-Down Options



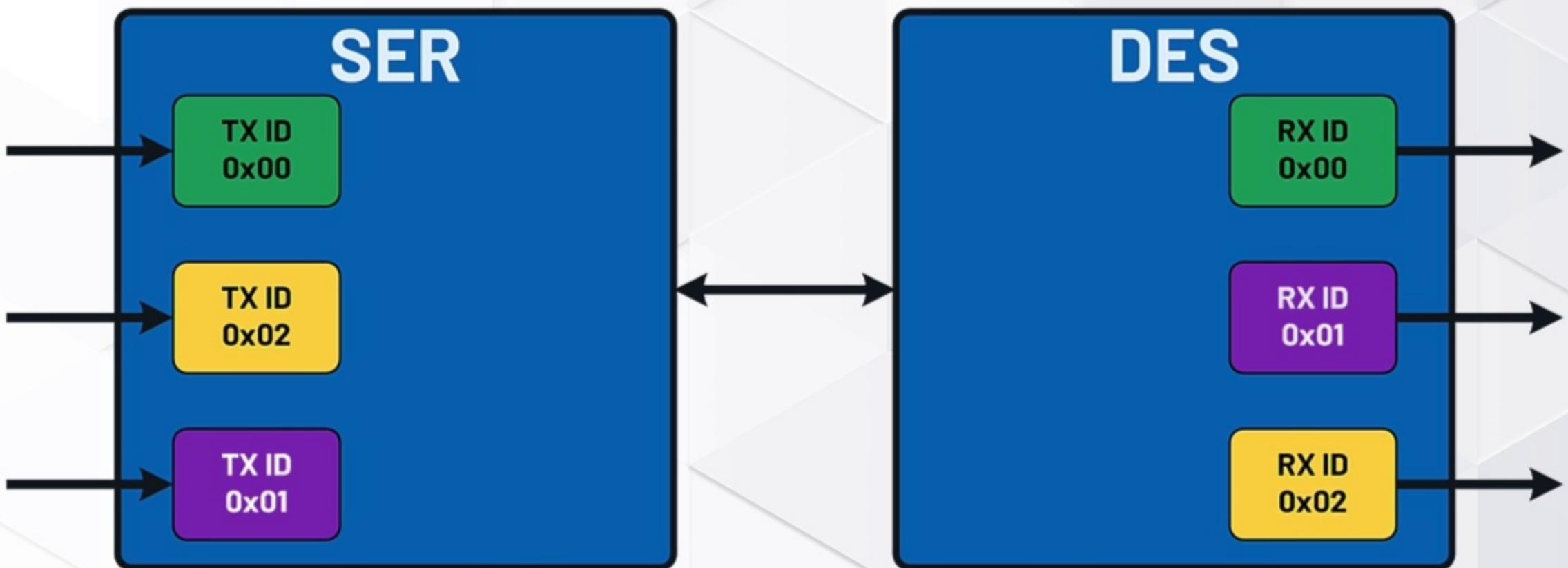
Jitter Reduction



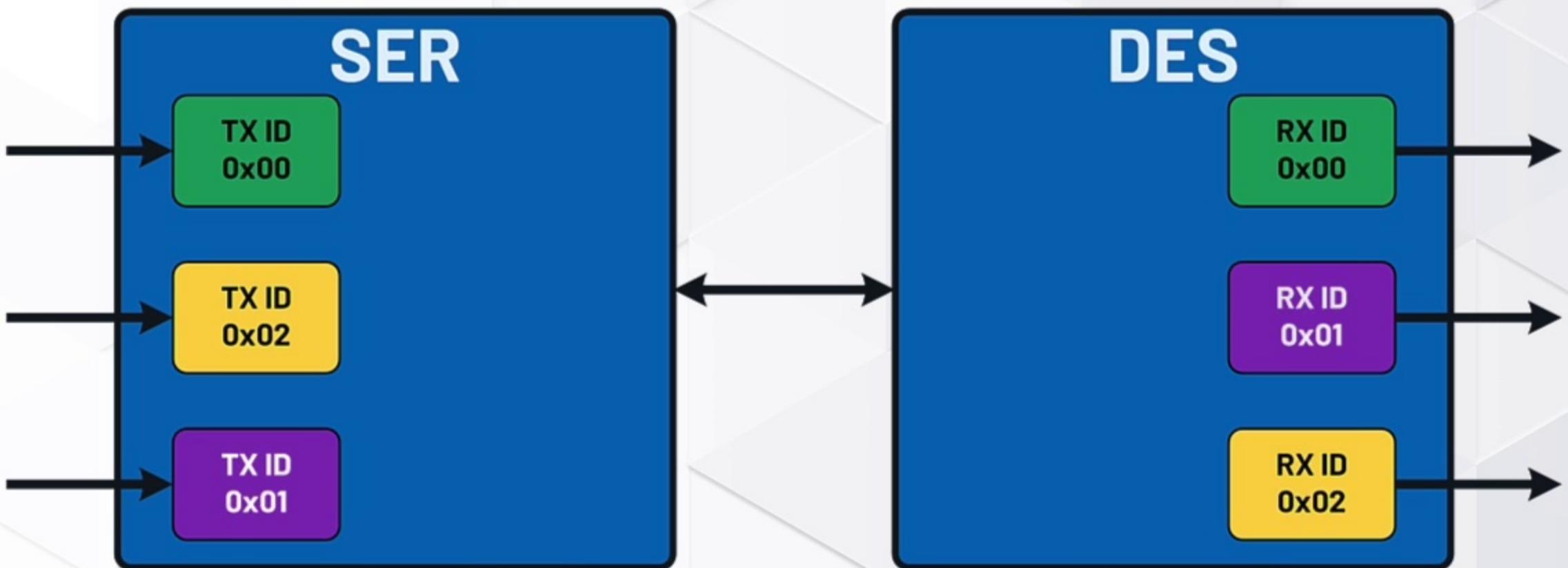
Status Reporting

**GPIO Input Detected [✓]**

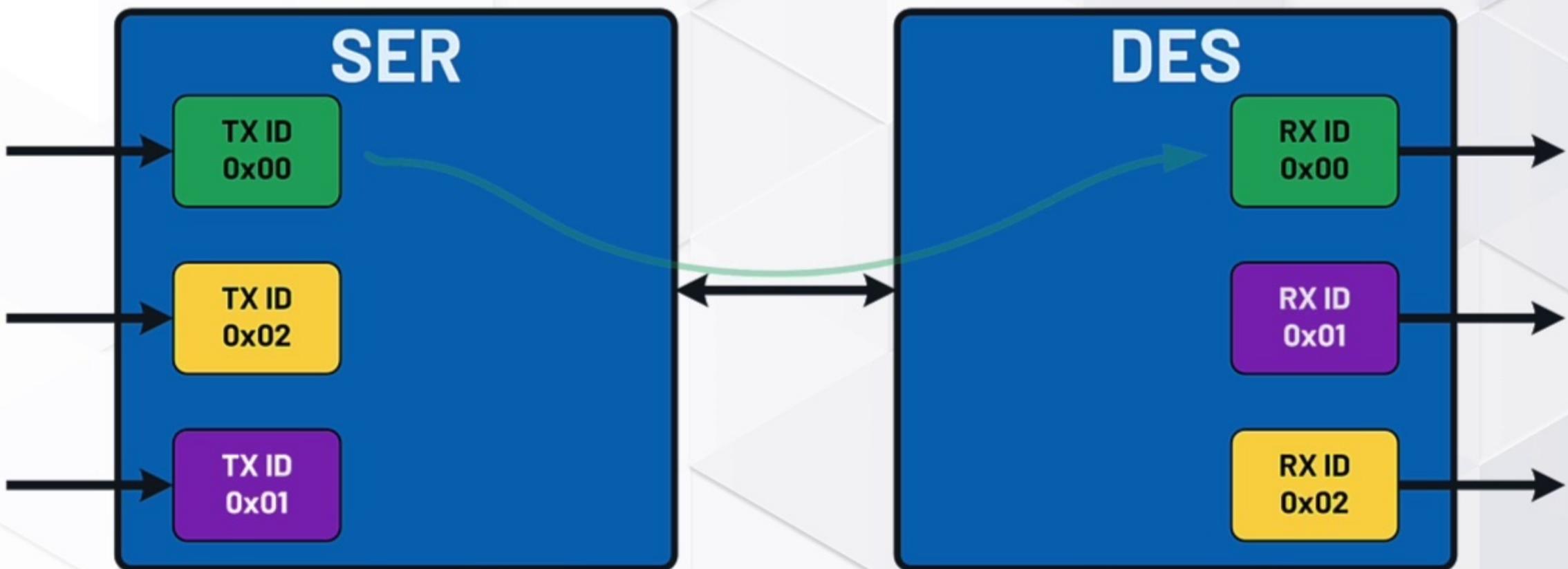
We'll review the operation of each of these features next.



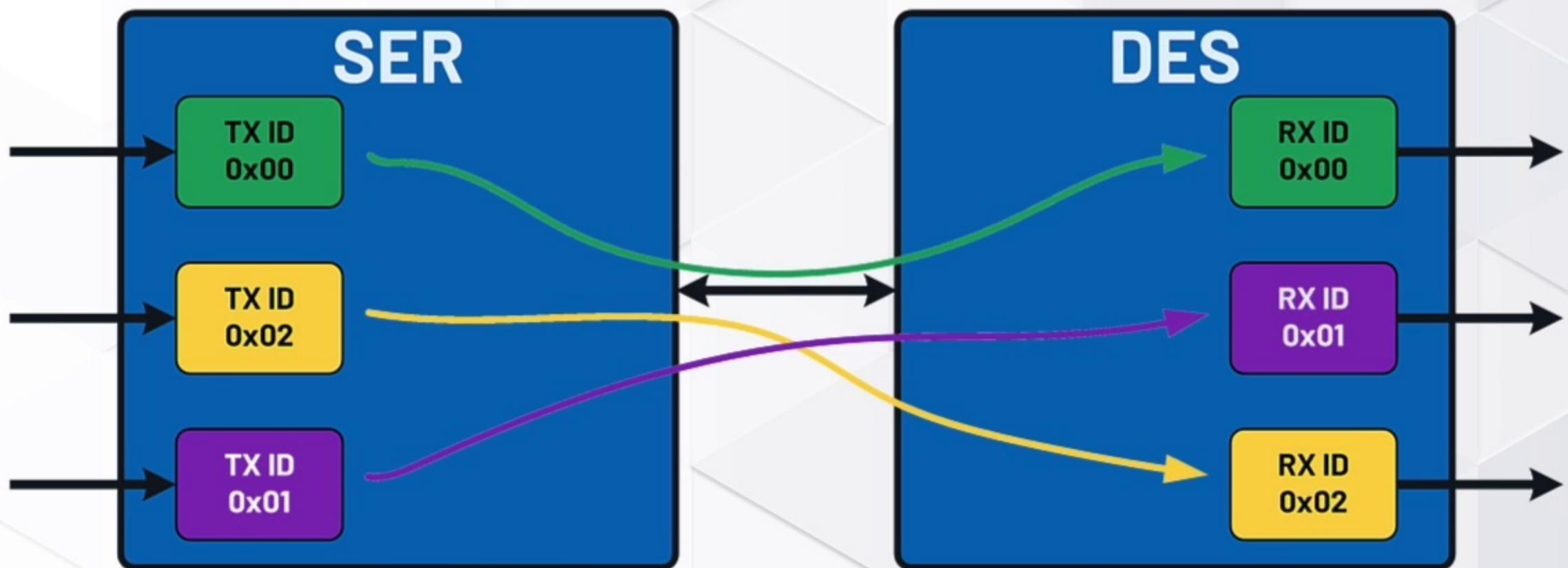
The GPIO capability is also very flexible within the GMSL



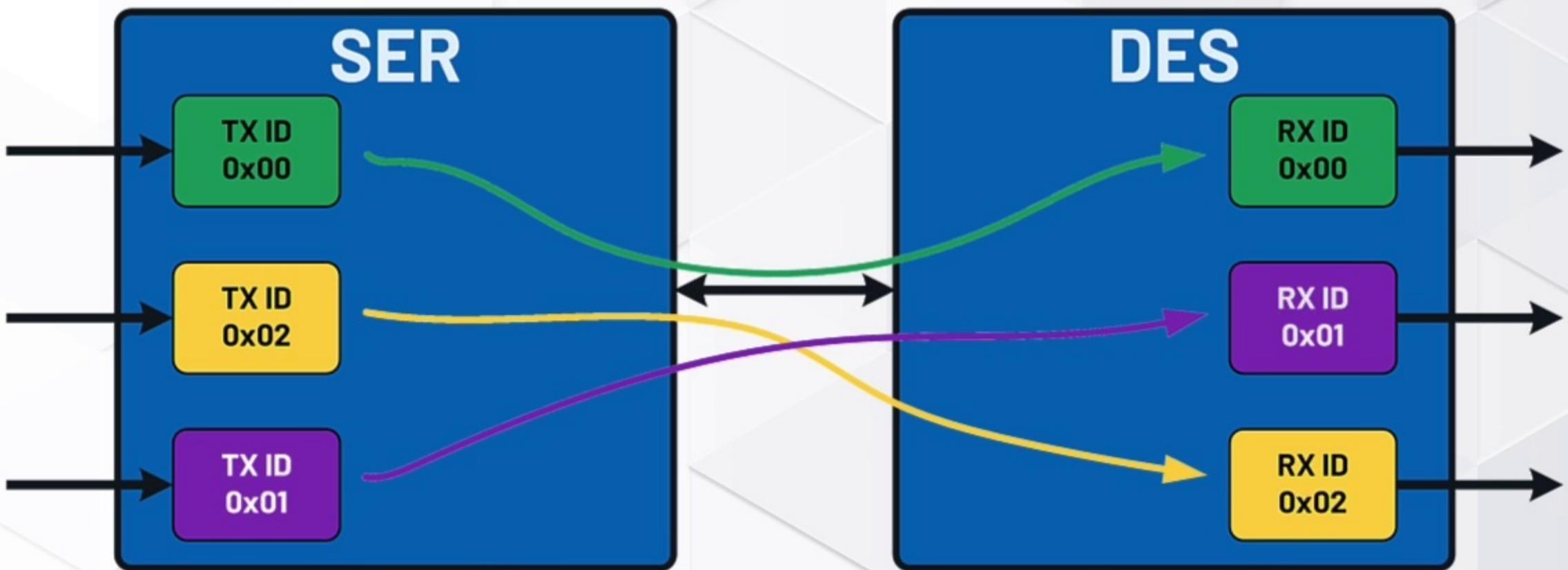
devices.



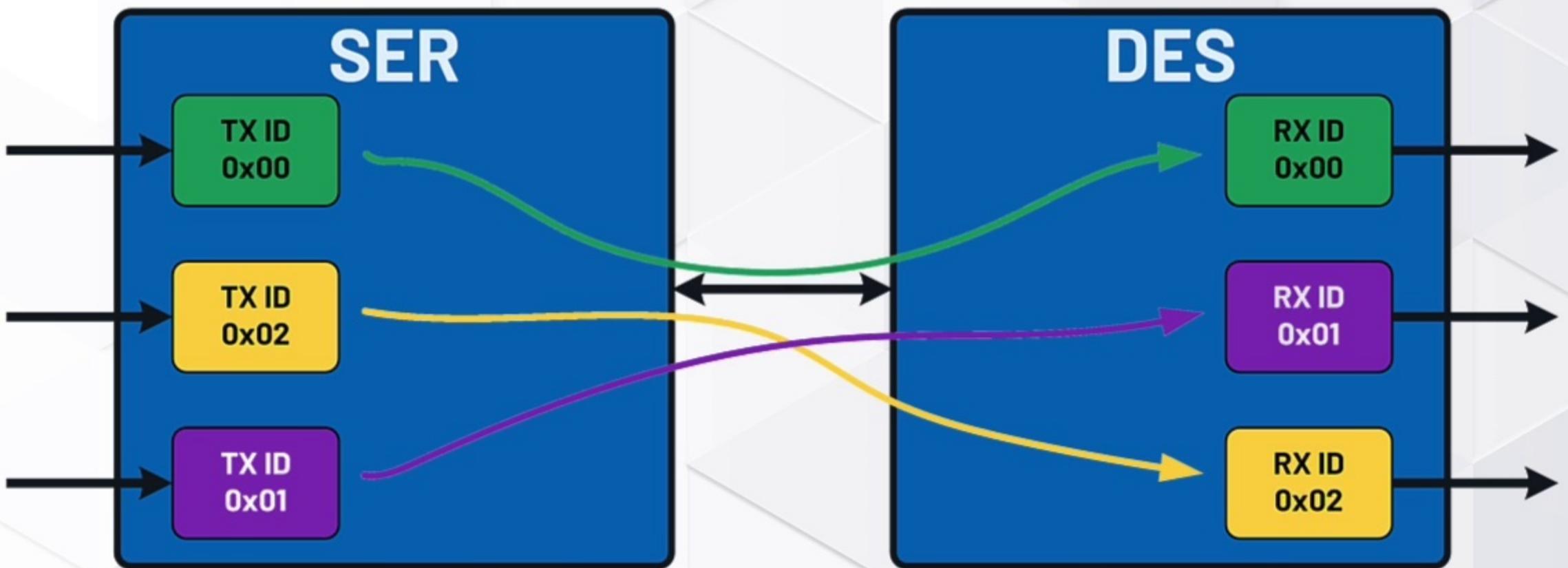
Each GPIO on one side of the GMSL link can be mapped to any



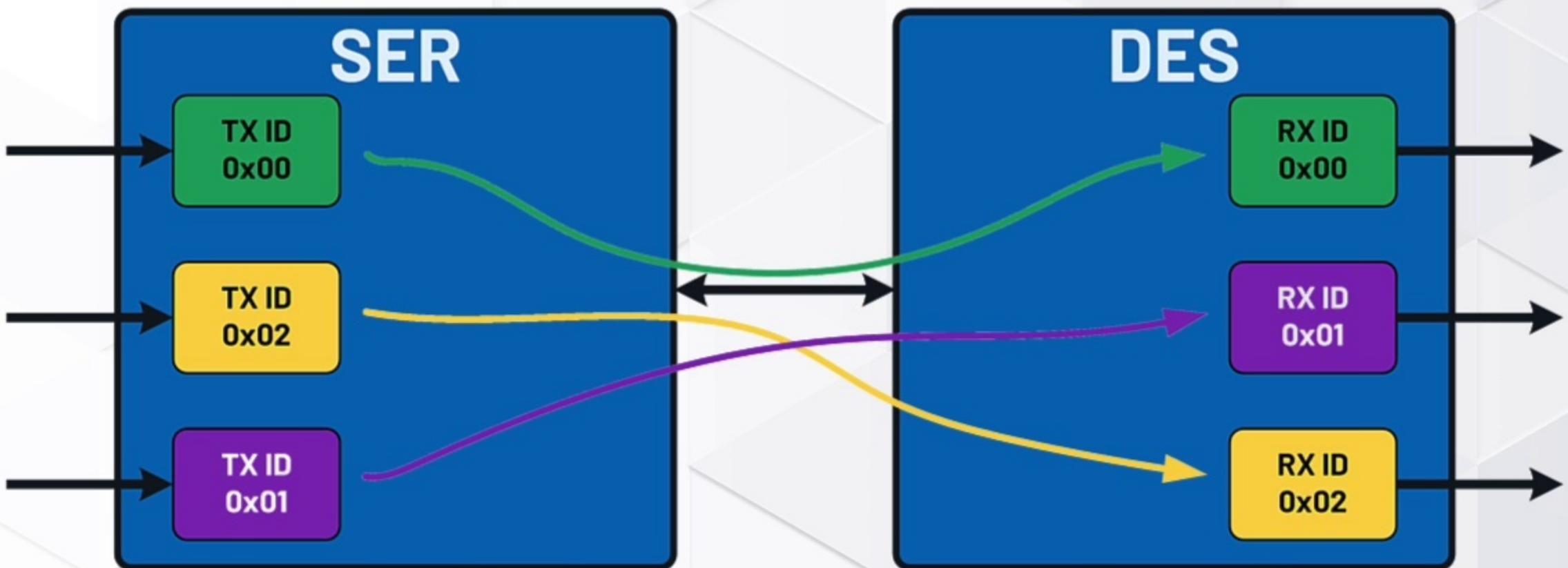
GPIO on the remote side using a 5 bit transmit receive ID that



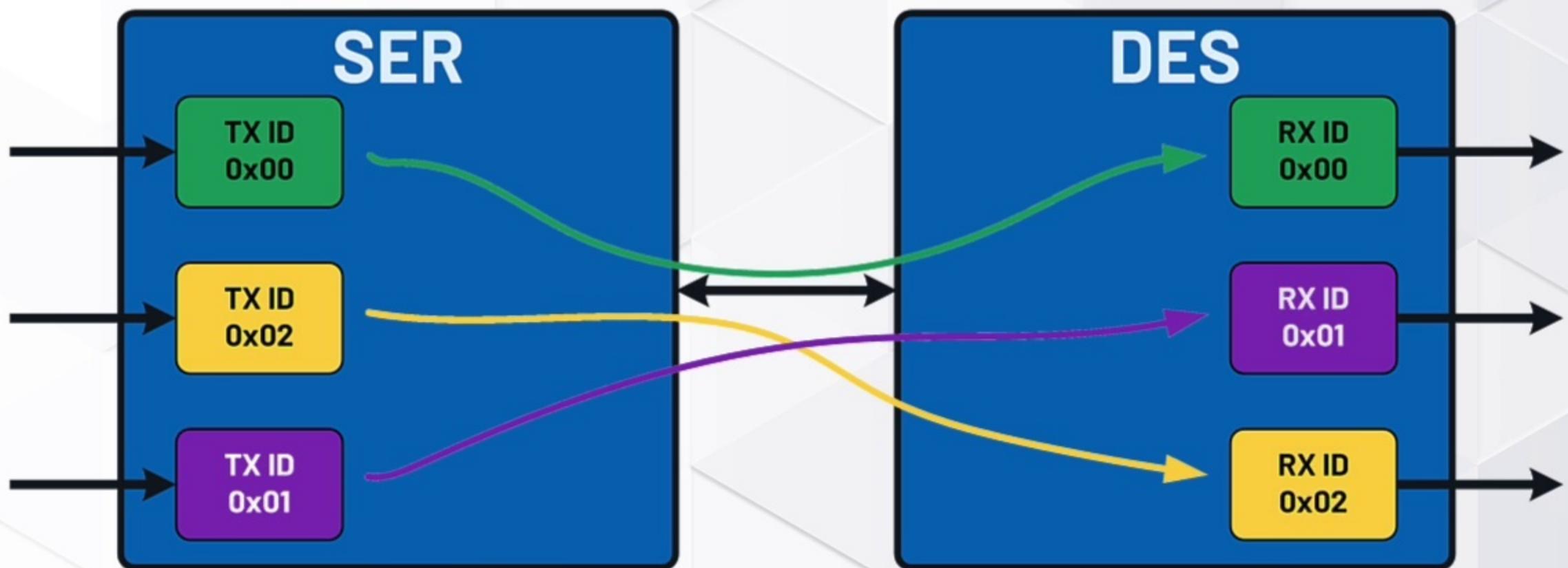
is configurable for up to 32 unique GPIO mappings to map GPIO



pins together through the GMSL link.



The transmitting GPIO needs to have the same transmit ID as the



receive ID that the receiving GPIO has been configured for.

GPIO 0								
BIT	7	6	5	4	3	2	1	0
Field	PULL_UPDN_SEL[1:0]	OUT_TYPE	GPIO_TX_ID[4:0]					
Reset	0b10	0b1	0b00000					
Access Type	Write, Read	Write, Read	Write, Read					
BITFIELD	BITS	DESCRIPTION			DECODE			
PULL_UPDN_SEL	7:6	Buffer pullup/pulldown configuration			0b00: None 0b01: Pullup 0b10: Pulldown 0b11: Reserved			
OUT_TYPE	5	Driver type selection			0b0: Open-drain 0b1: Push-pull			
GPIO_TX_ID	4:0	GPIO ID for pin while transmitting			0bXXXXX: This GPIO transmit ID			

BIT	7	6	5	4	3	2	1	0
Field	OVR_RES_CFG	RSVD	-	GPIO_RX_ID[4:0]				
Reset	0b0	0b1	-	0b00000				
Access Type	Write, Read		-	Write, Read				
BITFIELD	BITS	DESCRIPTION			DECODE			
OVR_RES_C FG	7	Override non-GPIO port function IO setting. When set, RES_CFG, PULL_UPDN_SEL, OUT_TYPE, and GPIO_OUT_DIS are effective when pin is configured as non-GPIO. When cleared, non-GPIO pin function determines IO type.			0b0: Non-GPIO function determines IO type when alternate function is selected 0b1: RES_CFG, PULL_UPDN_SEL, OUT_TYPE, and GPIO_OUT_DIS determine IO type for non-GPIO configuration			
GPIO_RX_ID	4:0	GPIO ID for pin while receiving			0bXXXXX: This GPIO receive ID			



The transmit and receive IDs are by default matched up to their

GPIO 0								
BIT	7	6	5	4	3	2	1	0
Field	PULL_UPDN_SEL[1:0]	OUT_TYPE	GPIO_TX_ID[4:0]					
Reset	0b10	0b1	0b00000					
Access Type	Write, Read	Write, Read	Write, Read					
BITFIELD	BITS	DESCRIPTION		DECODE				
PULL_UPDN_SEL	7:6	Buffer pullup/pulldown configuration		0b00: None 0b01: Pullup 0b10: Pulldown 0b11: Reserved				
OUT_TYPE	5	Driver type selection		0b0: Open-drain 0b1: Push-pull				
GPIO_TX_ID	4:0	GPIO ID for pin while transmitting		0bXXXXX: This GPIO transmit ID				

BIT	7	6	5	4	3	2	1	0
Field	OVR_RES_CFG	RSVD	-	GPIO_RX_ID[4:0]				
Reset	0b0	0b1	-	0b00000				
Access Type	Write, Read		-	Write, Read				
BITFIELD	BITS	DESCRIPTION		DECODE				
OVR_RES_C FG	7	Override non-GPIO port function IO setting.		When set, RES_CFG, PULL_UPDN_SEL, OUT_TYPE, and GPIO_OUT_DIS are effective when pin is configured as non-GPIO. When cleared, non-GPIO pin function determines IO type.				
GPIO_RX_ID	4:0	GPIO ID for pin while receiving		0bXXXXX: This GPIO receive ID				



GPIO numbers.

GPIO 0								
BIT	7	6	5	4	3	2	1	0
Field	PULL_UPDN_SEL[1:0]	OUT_TYPE	GPIO_TX_ID[4:0]					
Reset	0b10	0b1	0b00000					
Access Type	Write, Read	Write, Read	Write, Read					
BITFIELD	BITS	DESCRIPTION		DECODE				
PULL_UPDN_SEL	7:6	Buffer pullup/pulldown configuration		0b00: None 0b01: Pullup 0b10: Pulldown 0b11: Reserved				
OUT_TYPE	5	Driver type selection		0b0: Open-drain 0b1: Push-pull				
GPIO_TX_ID	4:0	GPIO ID for pin while transmitting		0bXXXXX: This GPIO transmit ID				

BIT	7	6	5	4	3	2	1	0
Field	OVR_RES_CFG	RSVD	-	GPIO_RX_ID[4:0]				
Reset	0b0	0b1	-	0b00000				
Access Type	Write, Read		-	Write, Read				
BITFIELD	BITS	DESCRIPTION		DECODE				
OVR_RES_C FG	7	Override non-GPIO port function IO setting.		When set, RES_CFG, PULL_UPDN_SEL, OUT_TYPE, and GPIO_OUT_DIS are effective when pin is configured as non-GPIO. When cleared, non-GPIO pin function determines IO type.				
GPIO_RX_ID	4:0	GPIO ID for pin while receiving		0bXXXXX: This GPIO receive ID				



So GPIO 0 will have a receive and transmit ID of hex 0 on both

GPIO 0								
BIT	7	6	5	4	3	2	1	0
Field	PULL_UPDN_SEL[1:0]	OUT_TYPE	GPIO_TX_ID[4:0]					
Reset	0b10	0b1	0b00000					
Access Type	Write, Read	Write, Read	Write, Read					
BITFIELD	BITS	DESCRIPTION		DECODE				
PULL_UPDN_SEL	7:6	Buffer pullup/pulldown configuration		0b00: None 0b01: Pullup 0b10: Pulldown 0b11: Reserved				
OUT_TYPE	5	Driver type selection		0b0: Open-drain 0b1: Push-pull				
GPIO_TX_ID	4:0	GPIO ID for pin while transmitting		0bXXXXX: This GPIO transmit ID				

BIT	7	6	5	4	3	2	1	0
Field	OVR_RES_CFG	RSVD	-	GPIO_RX_ID[4:0]				
Reset	0b0	0b1	-	0b00000				
Access Type	Write, Read		-	Write, Read				
BITFIELD	BITS	DESCRIPTION		DECODE				
OVR_RES_C FG	7	Override non-GPIO port function IO setting.		When set, RES_CFG, PULL_UPDN_SEL, OUT_TYPE, and GPIO_OUT_DIS are effective when pin is configured as non-GPIO. When cleared, non-GPIO pin function determines IO type.				
GPIO_RX_ID	4:0	GPIO ID for pin while receiving		0bXXXXX: This GPIO receive ID				



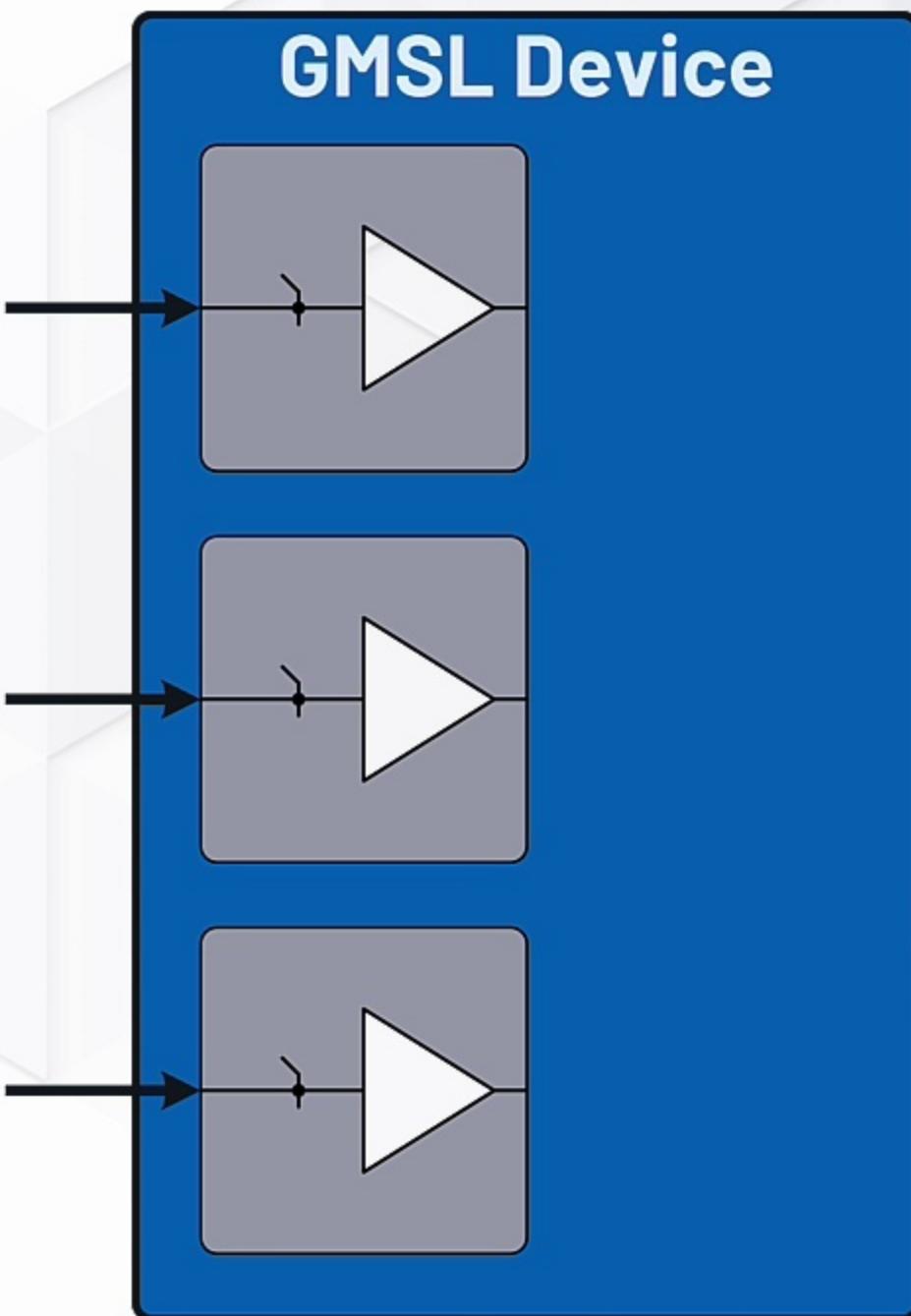
devices and they match up by default.

GPIO 0								
BIT	7	6	5	4	3	2	1	0
Field	PULL_UPDN_SEL[1:0]	OUT_TYPE	GPIO_TX_ID[4:0]					
Reset	0b10	0b1	0b00000					
Access Type	Write, Read	Write, Read	Write, Read					
BITFIELD	BITS	DESCRIPTION		DECODE				
PULL_UPDN_SEL	7:6	Buffer pullup/pulldown configuration		0b00: None 0b01: Pullup 0b10: Pulldown 0b11: Reserved				
OUT_TYPE	5	Driver type selection		0b0: Open-drain 0b1: Push-pull				
GPIO_TX_ID	4:0	GPIO ID for pin while transmitting		0bXXXXX: This GPIO transmit ID				

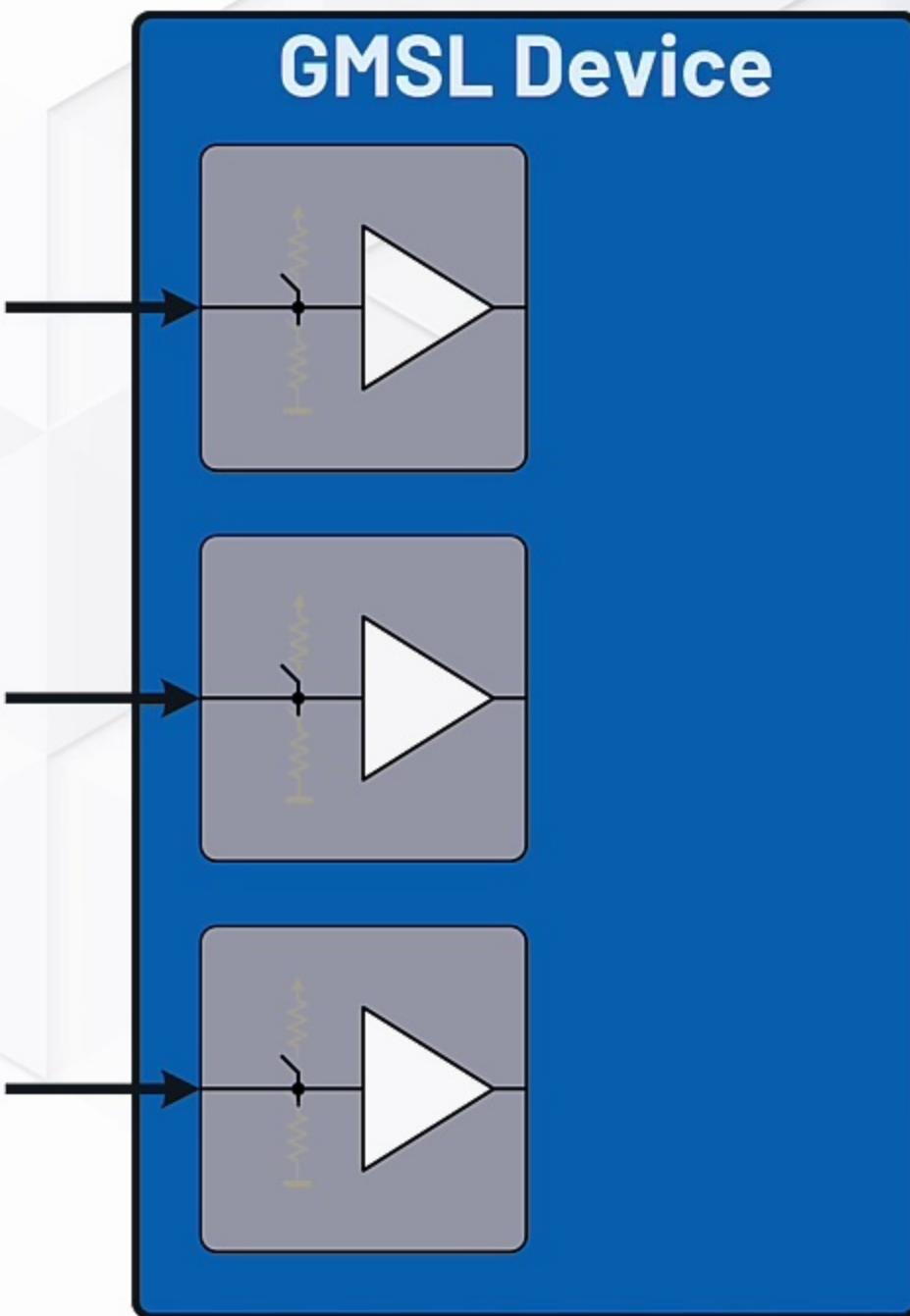
BIT	7	6	5	4	3	2	1	0
Field	OVR_RES_CFG	RSVD	-	GPIO_RX_ID[4:0]				
Reset	0b0	0b1	-	0b00000				
Access Type	Write, Read		-	Write, Read				
BITFIELD	BITS	DESCRIPTION		DECODE				
OVR_RES_C FG	7	Override non-GPIO port function IO setting.		When set, RES_CFG, PULL_UPDN_SEL, OUT_TYPE, and GPIO_OUT_DIS are effective when pin is configured as non-GPIO. When cleared, non-GPIO pin function determines IO type.				
GPIO_RX_ID	4:0	GPIO ID for pin while receiving		0bXXXXX: This GPIO receive ID				



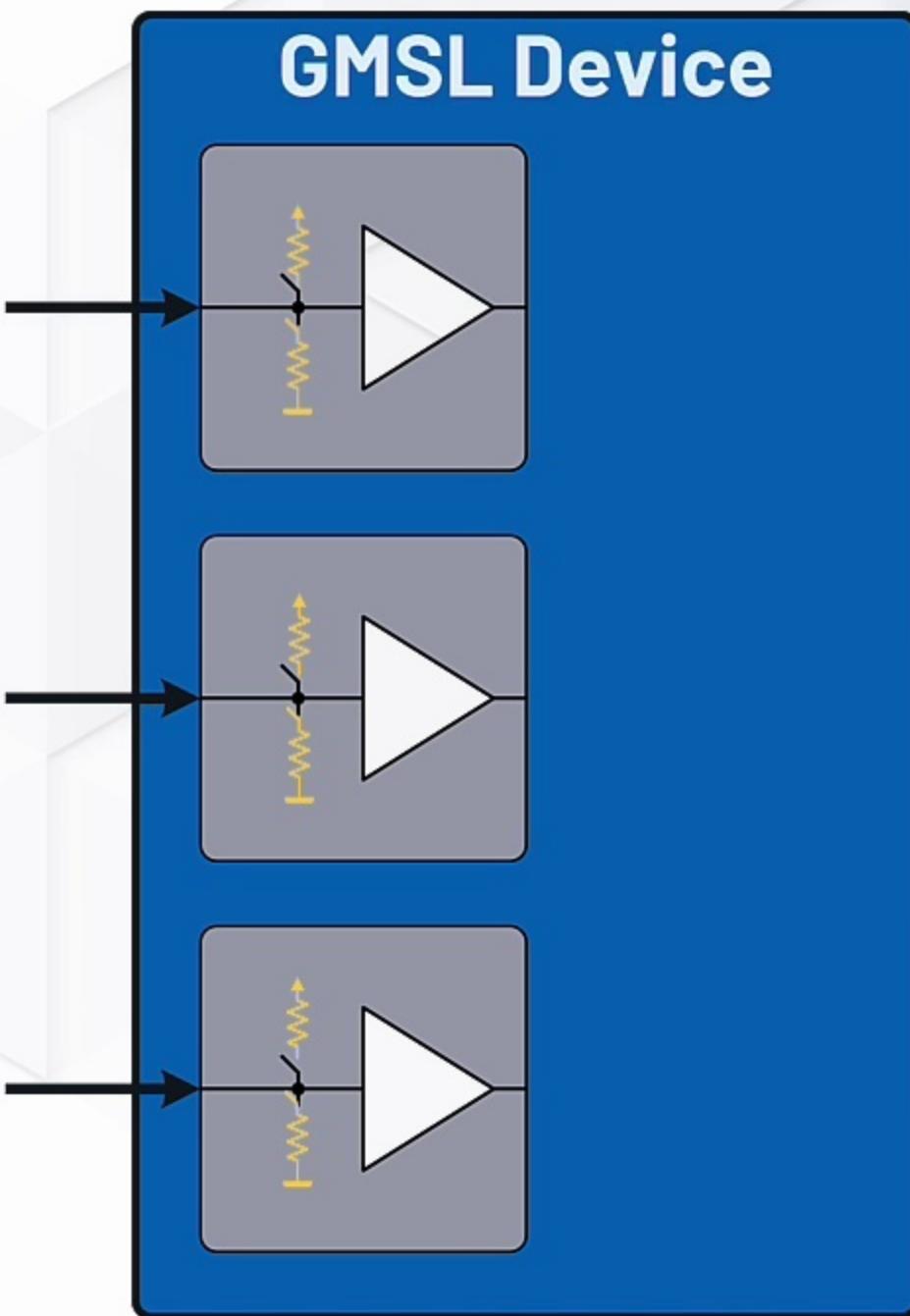
If the IDs do not match the GPIO will not propagate through.



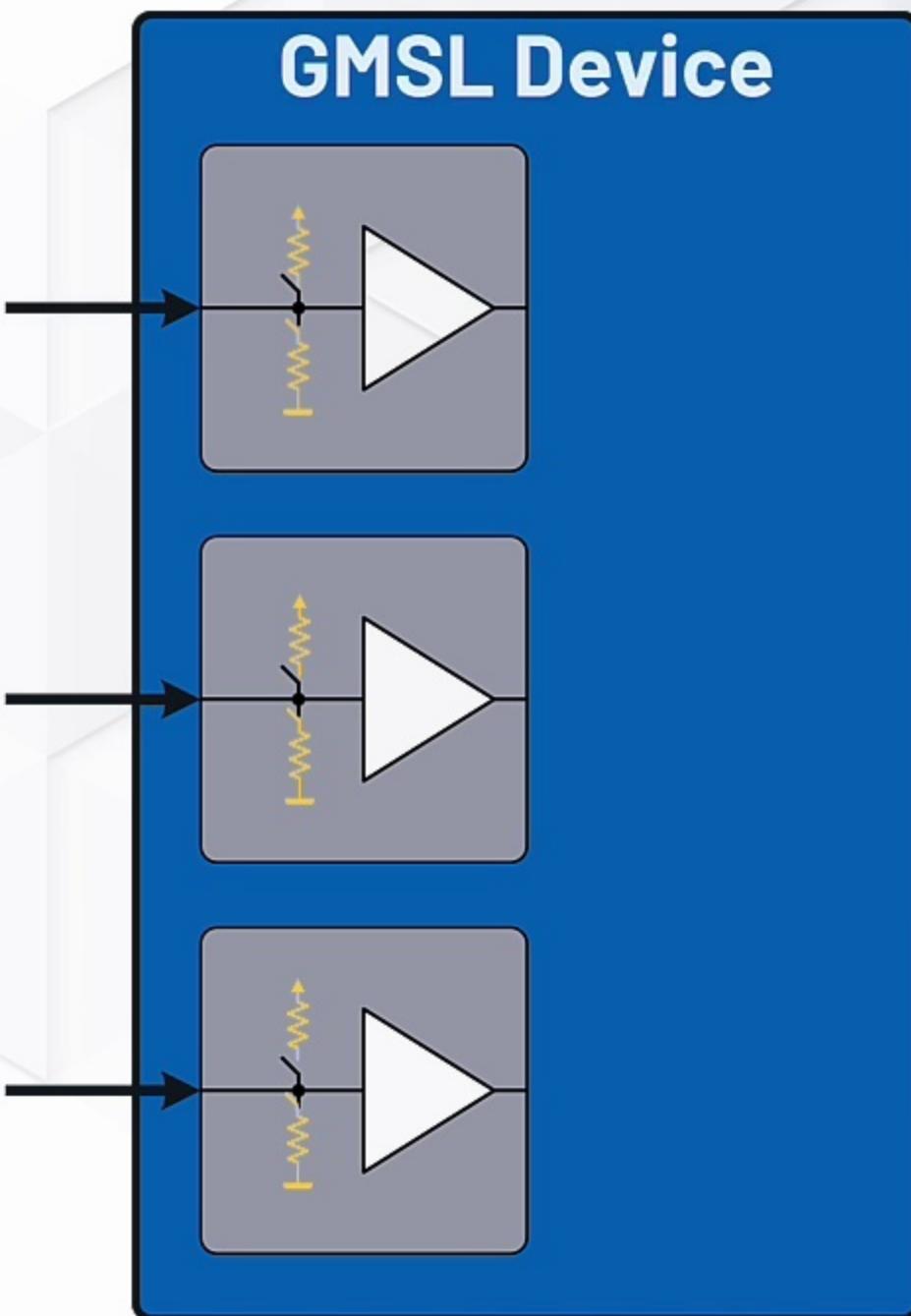
The pins configured as GPIO pins on the GMSL device also



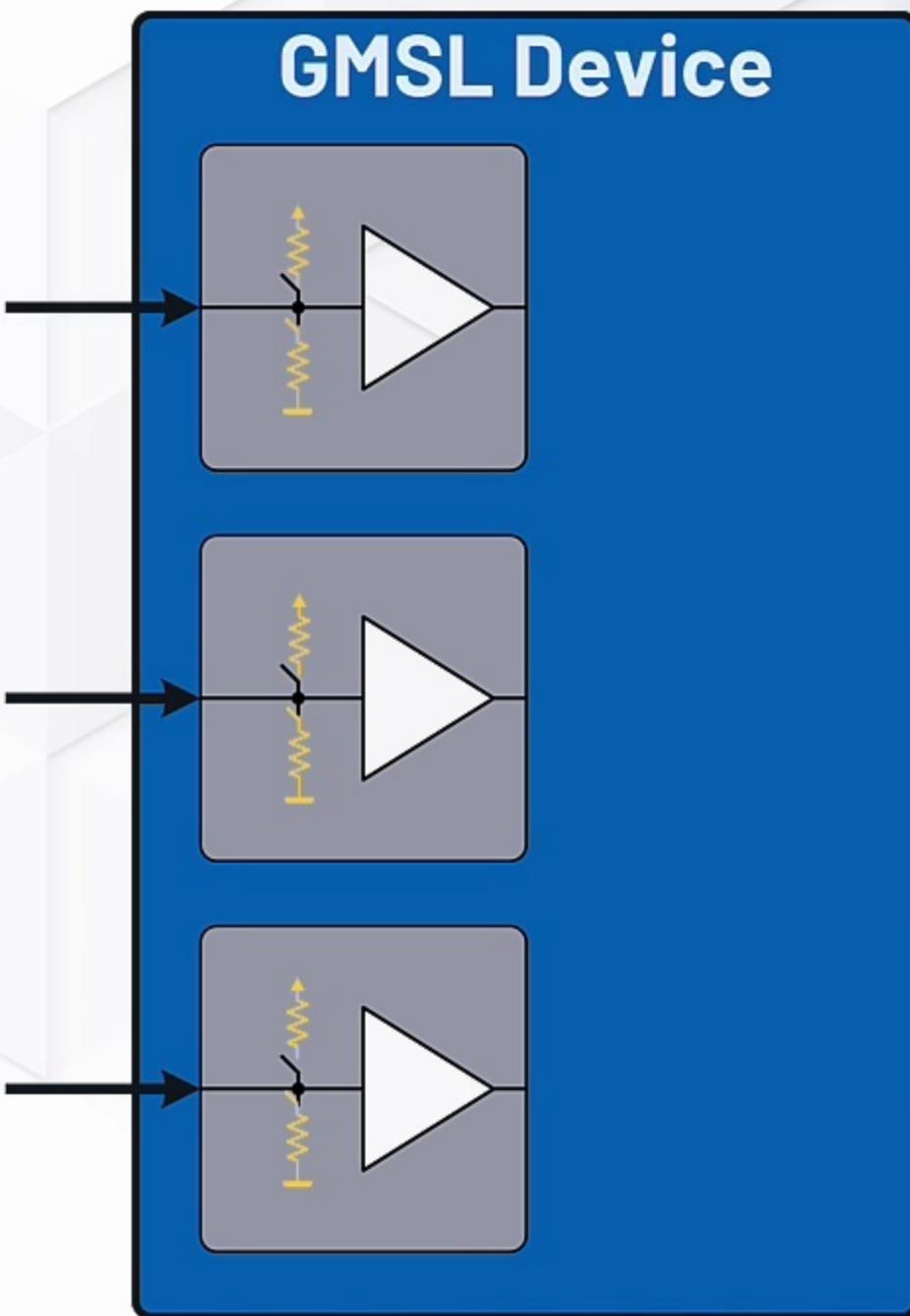
have configurable and optional pull up and pull down resistors.



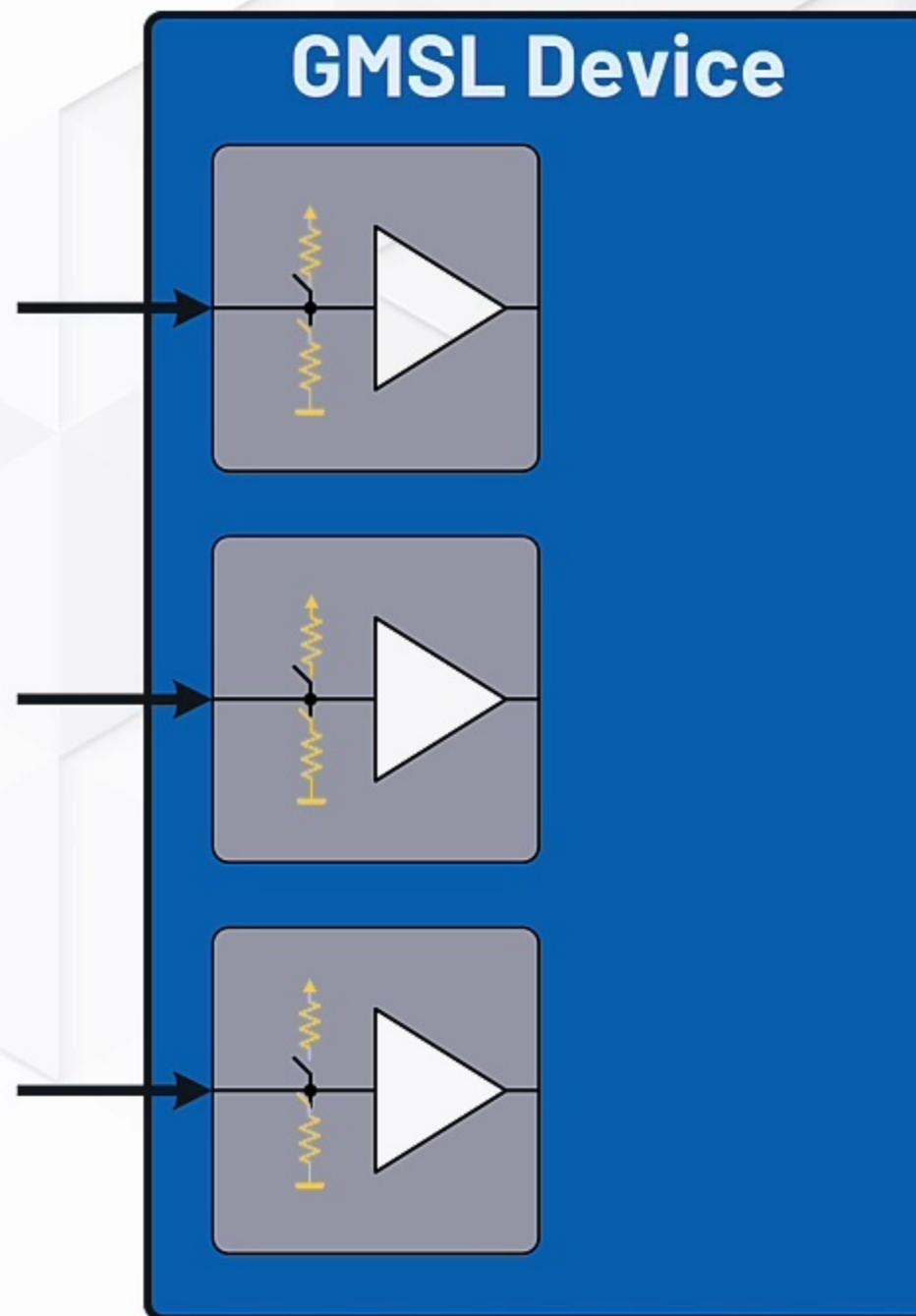
The output driver type is also selectable as open drain or push



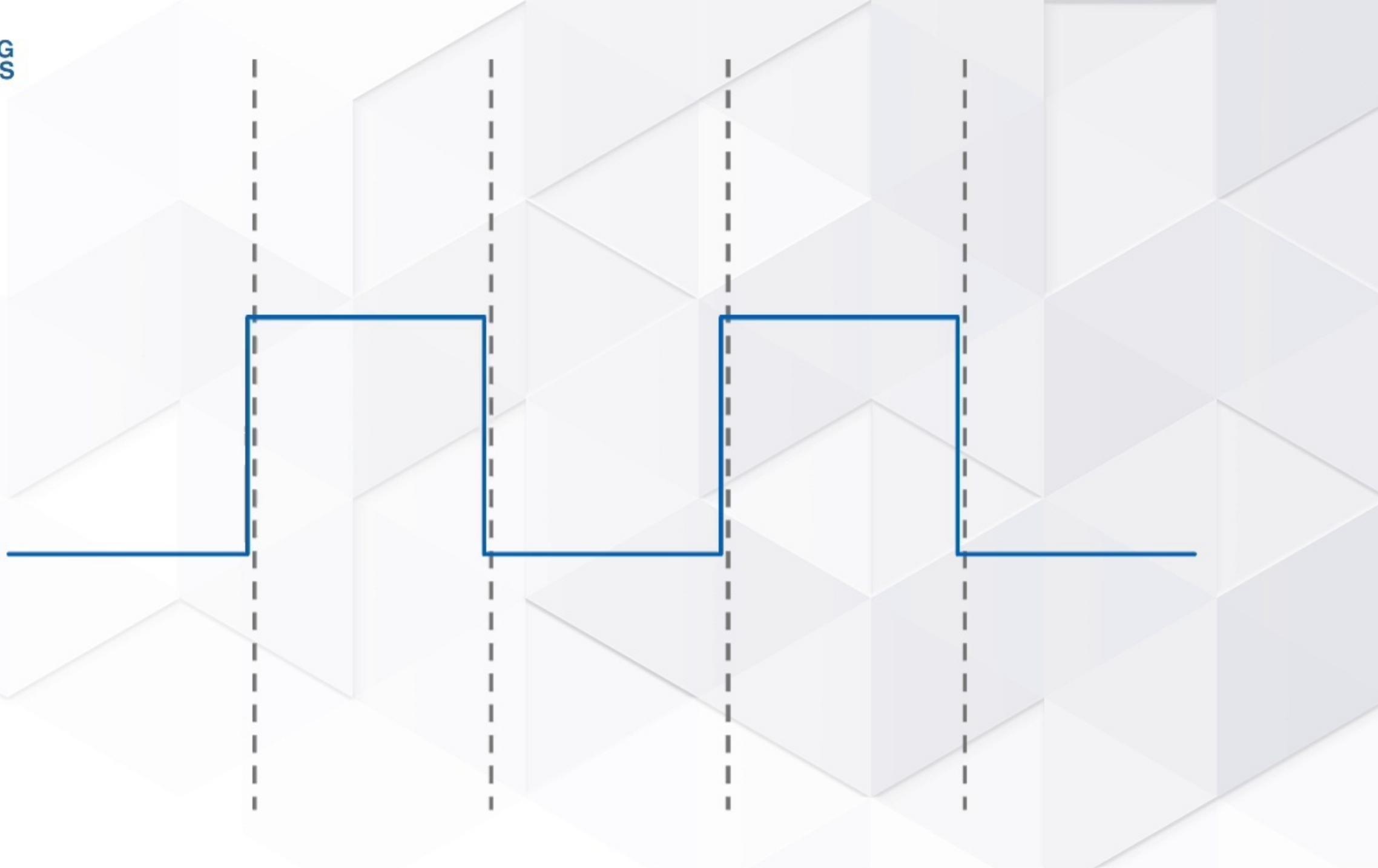
pull circuits and their edge rates can be adjusted.



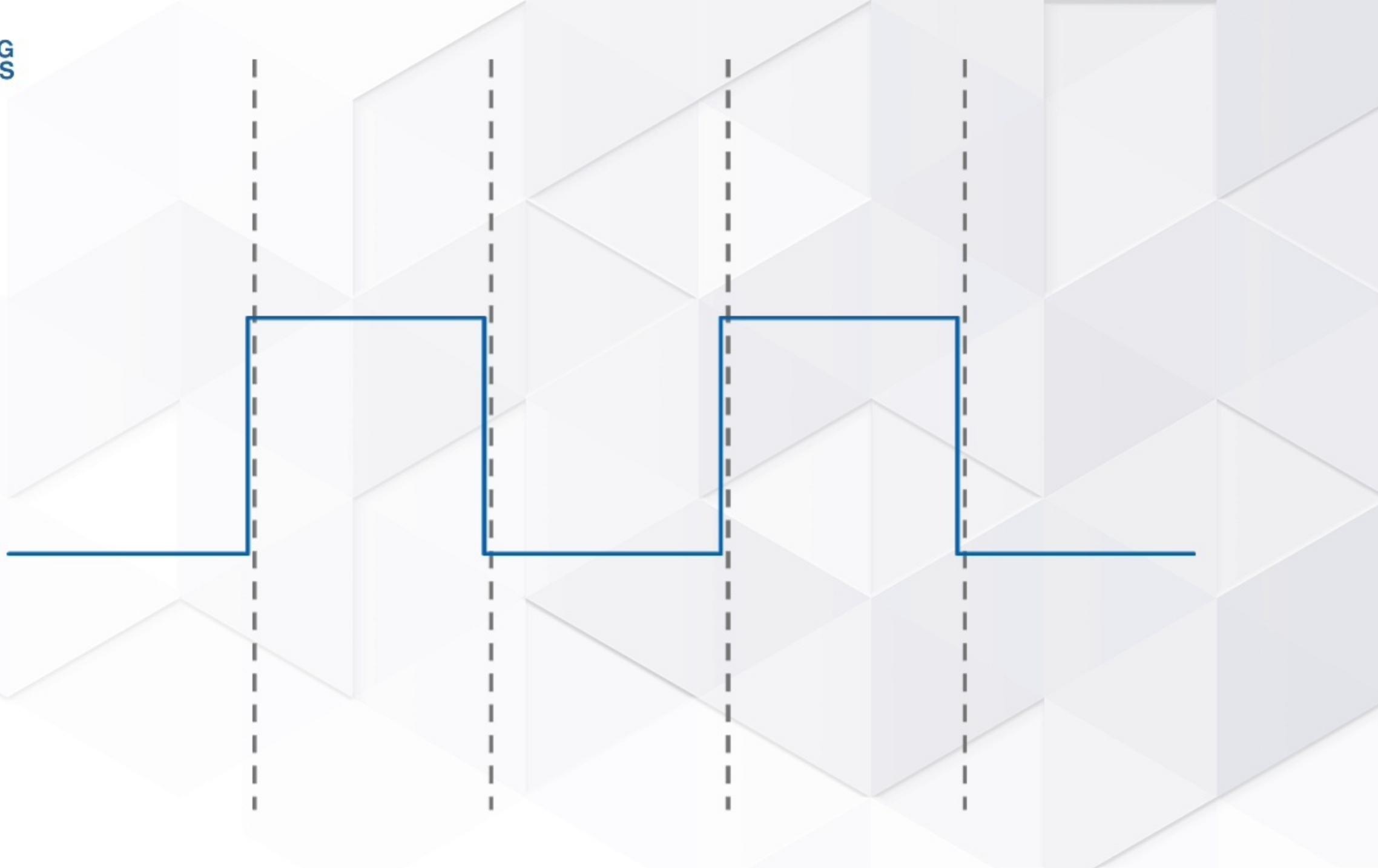
It is recommended to use lower value external pull up and pull



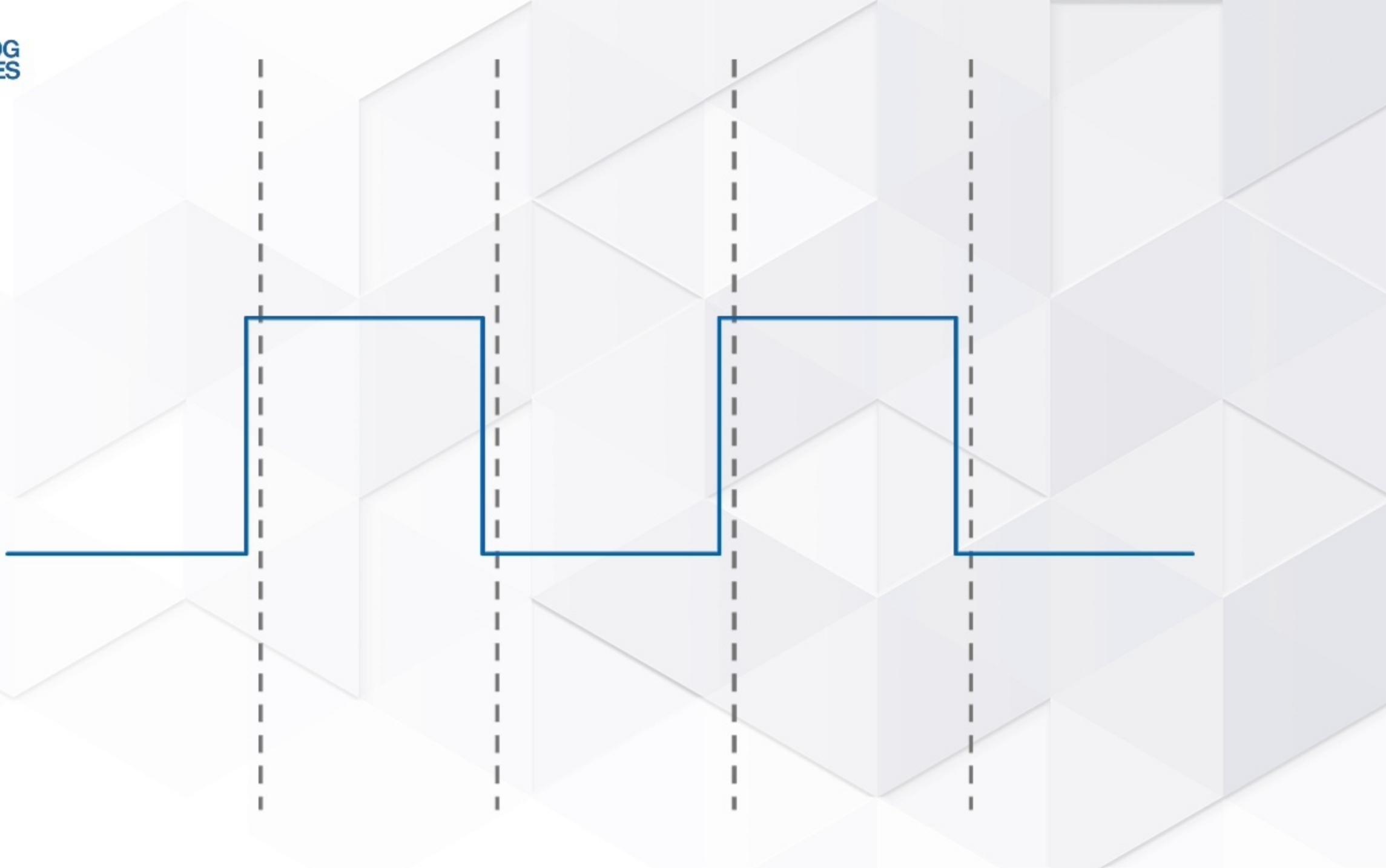
down resistors for stronger pull up and pull down.



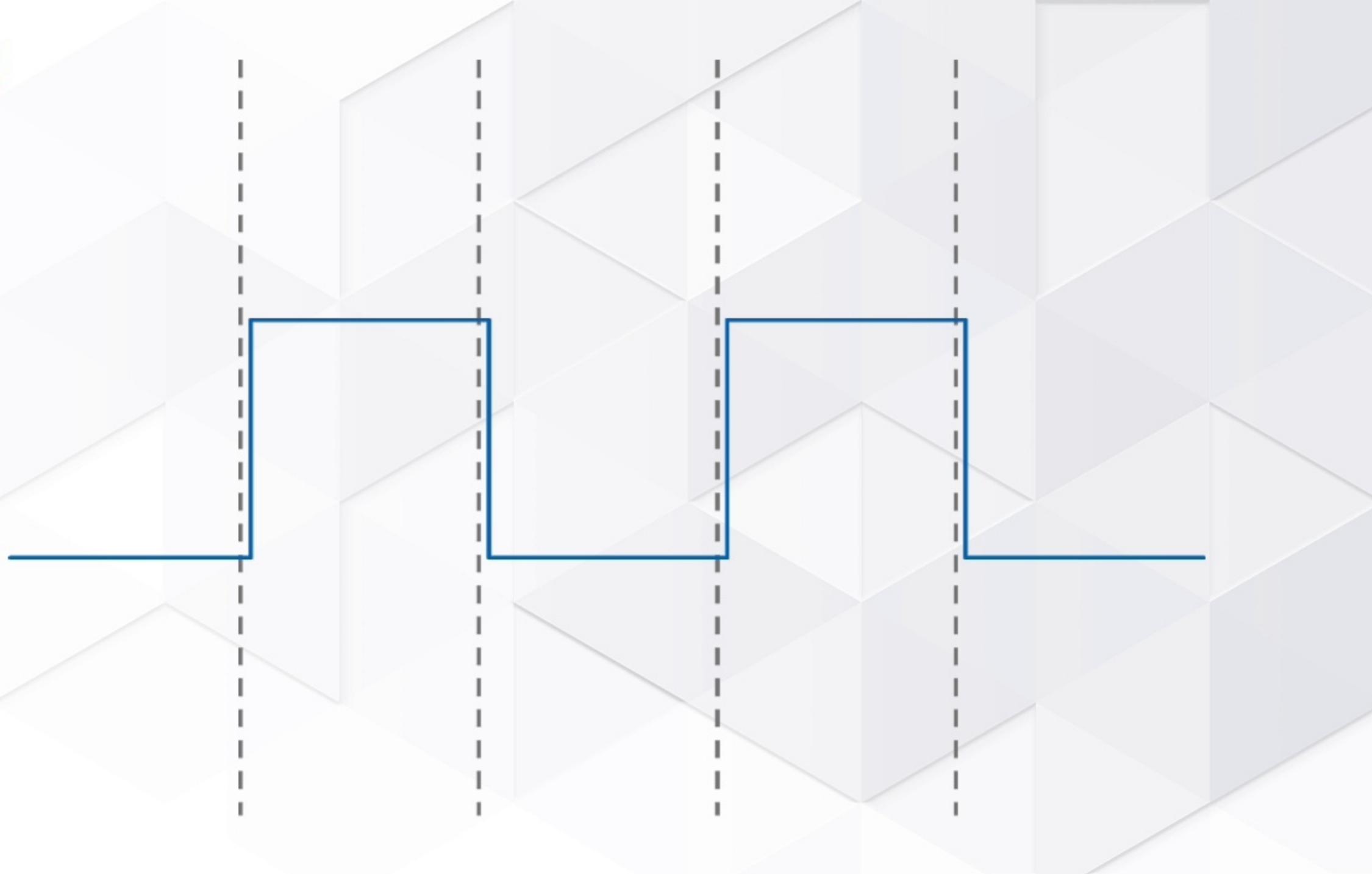
In certain applications jitter on the GPI of a signal could be



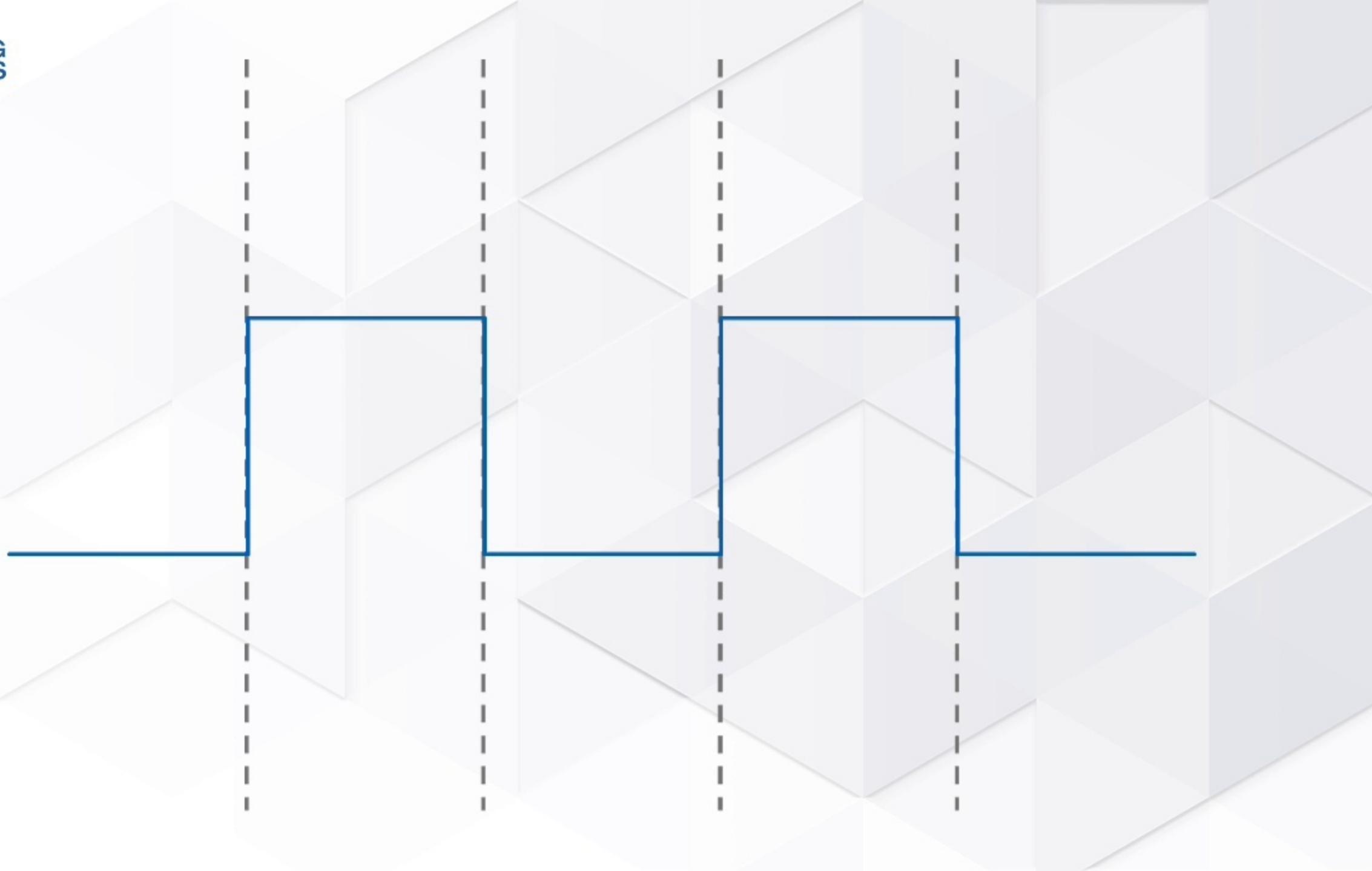
a concern for stricter timing requirements.



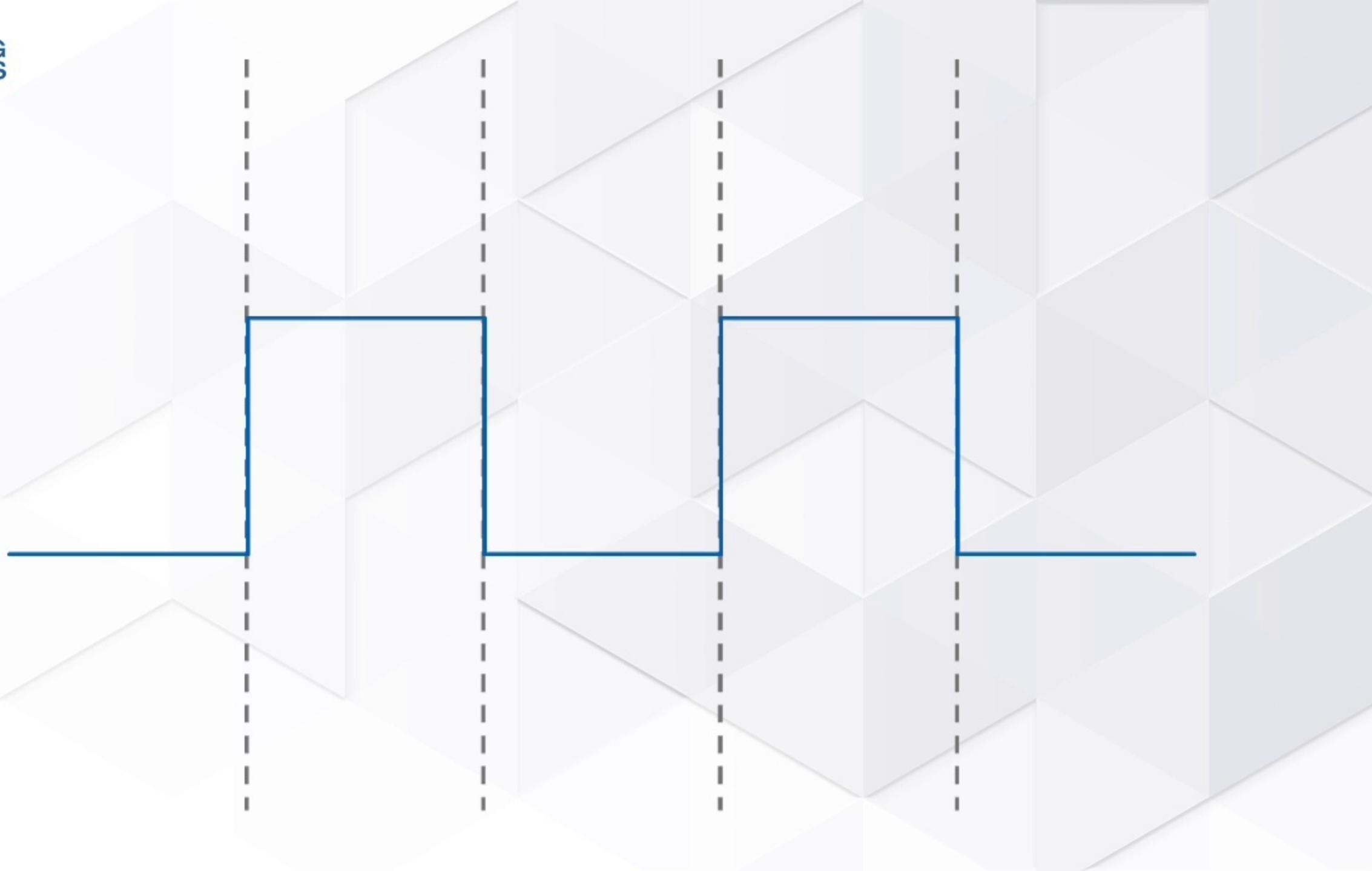
GMSL devices offer a delayed compensation mode which helps



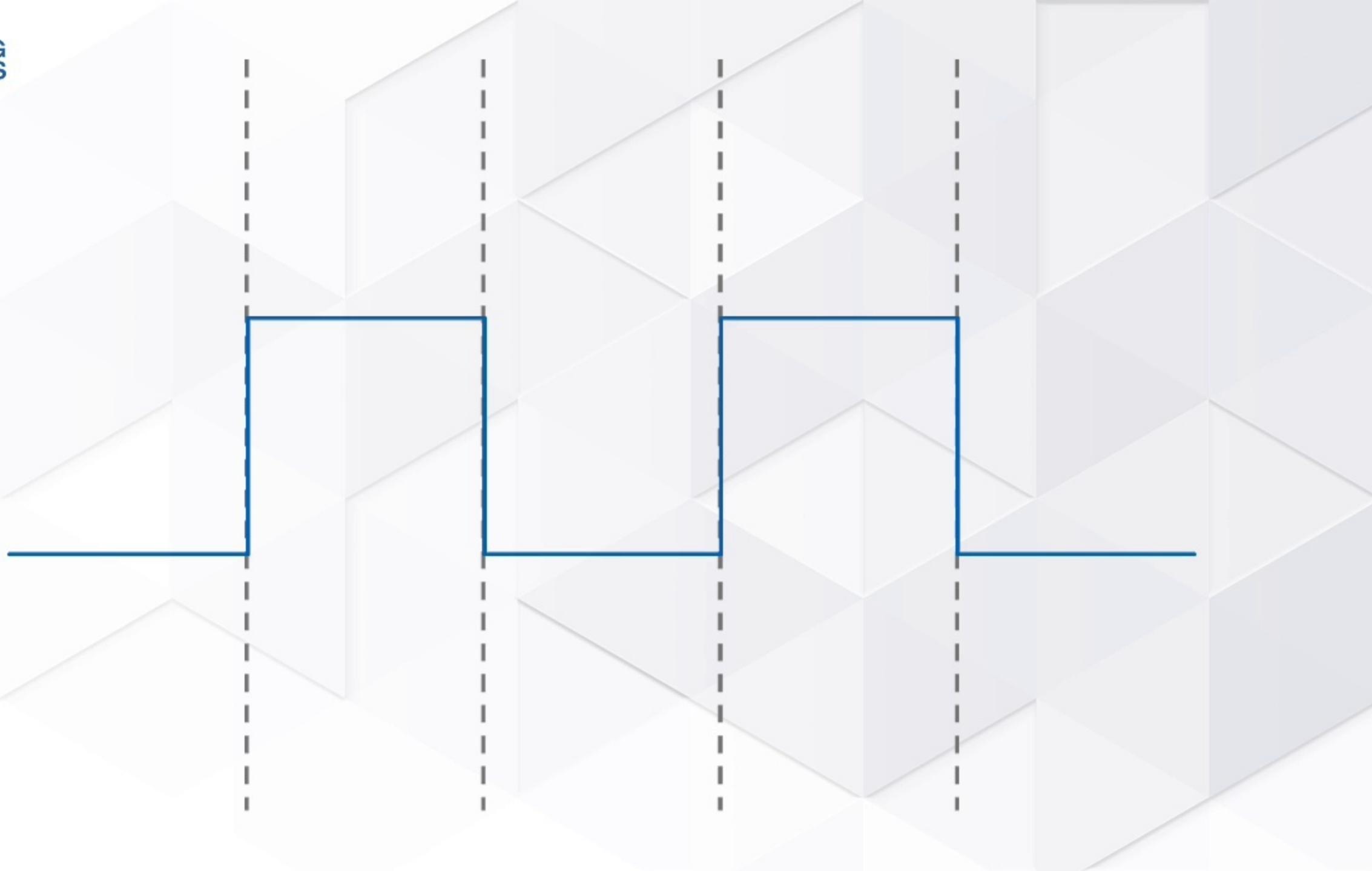
minimize jitter by adding a fixed delay to the GPIO



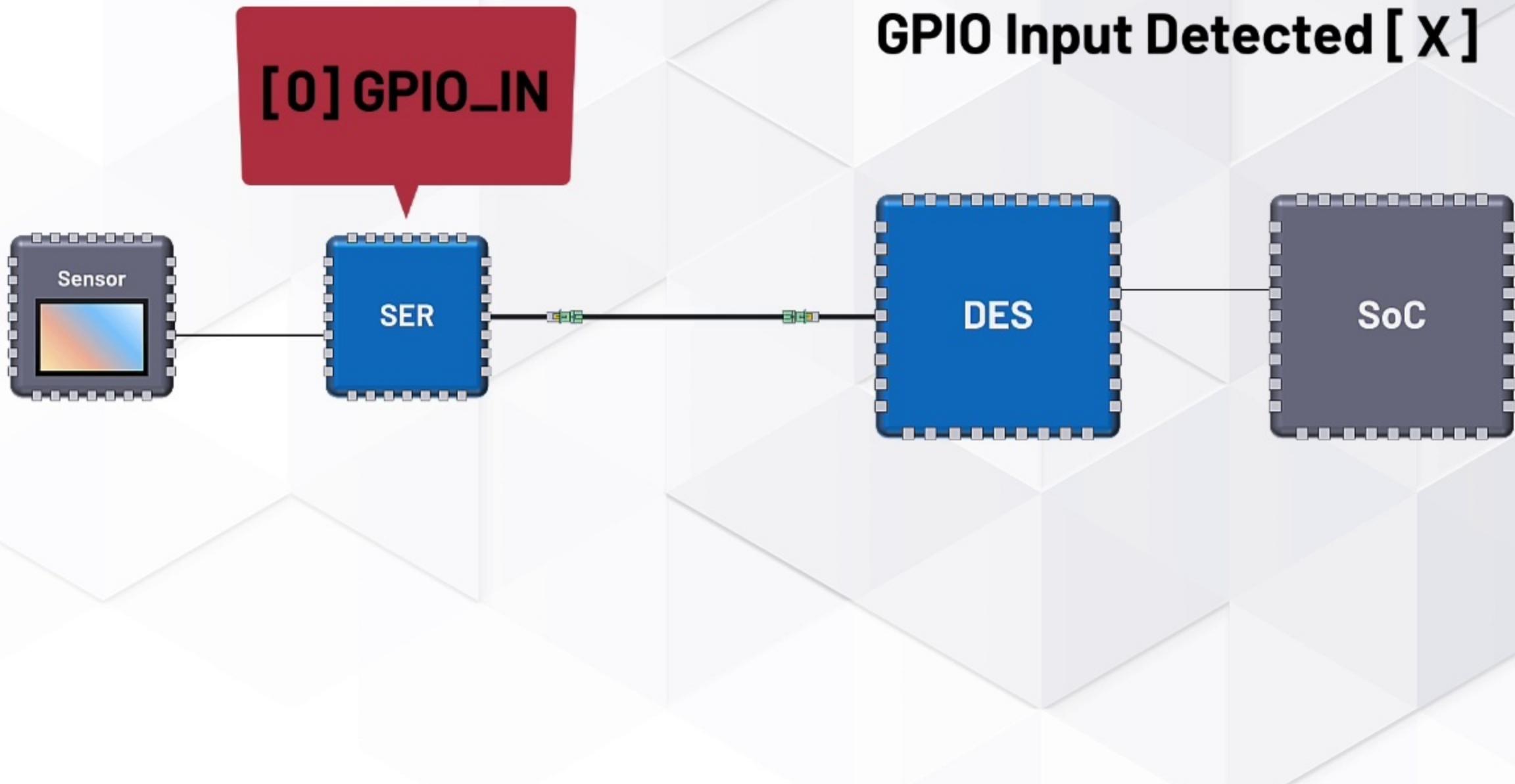
transmission.



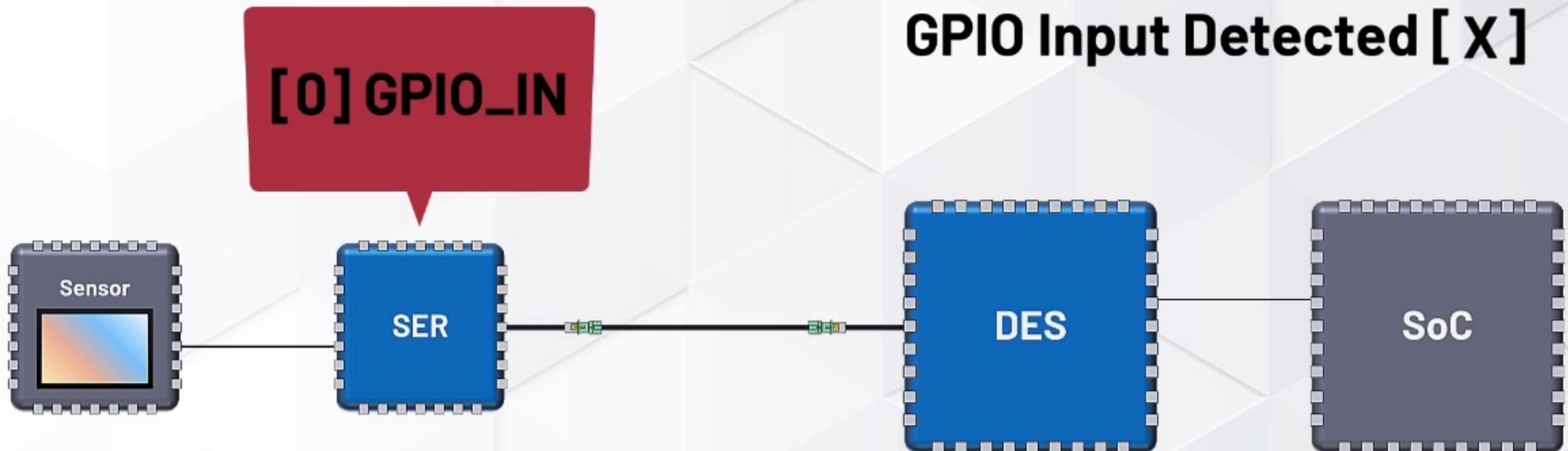
The delay is programmable in the order of a few microseconds with



the benefit of significant jitter reduction on the signal.

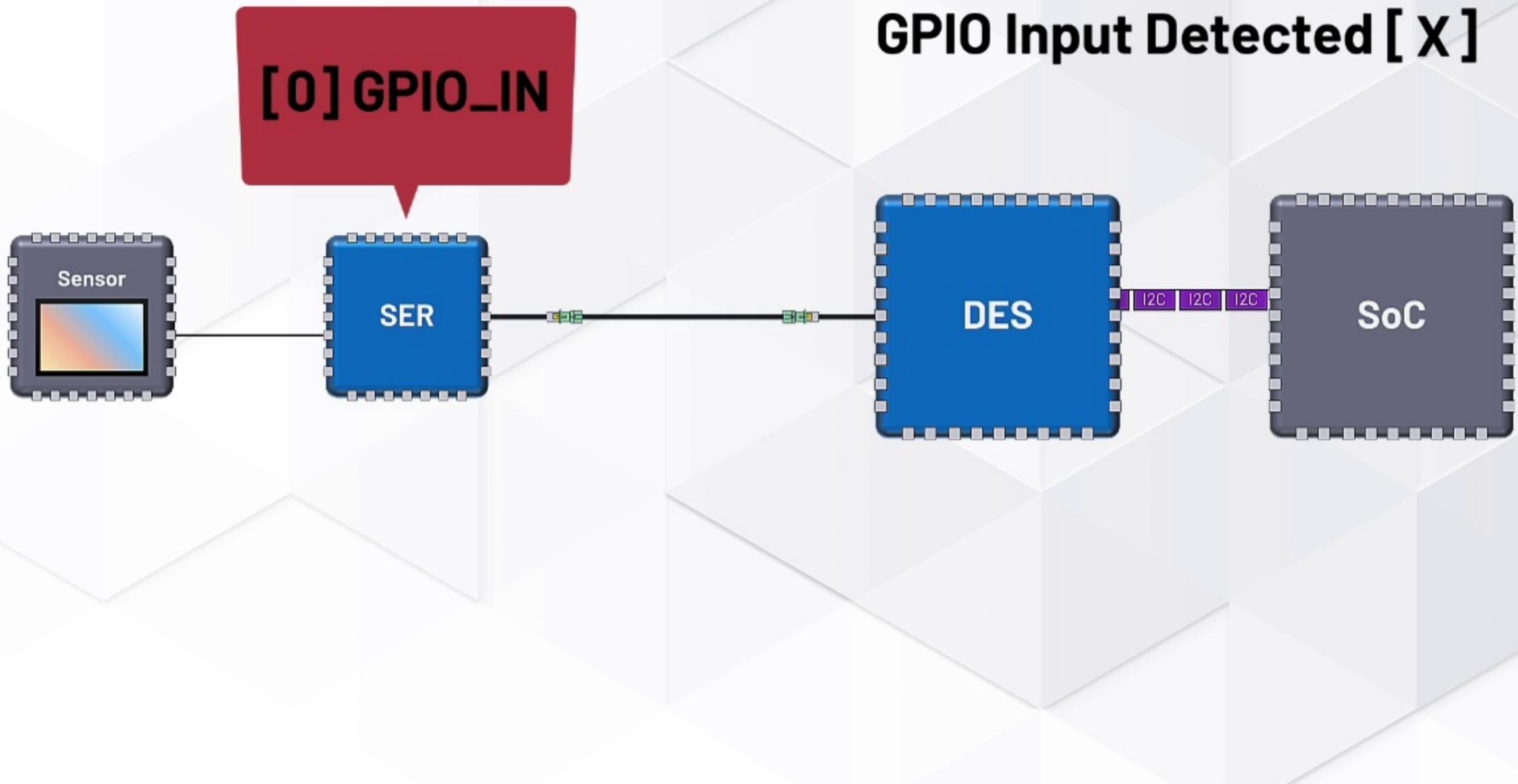


GMSL devices can also report the GPIO pin logic state by pulling

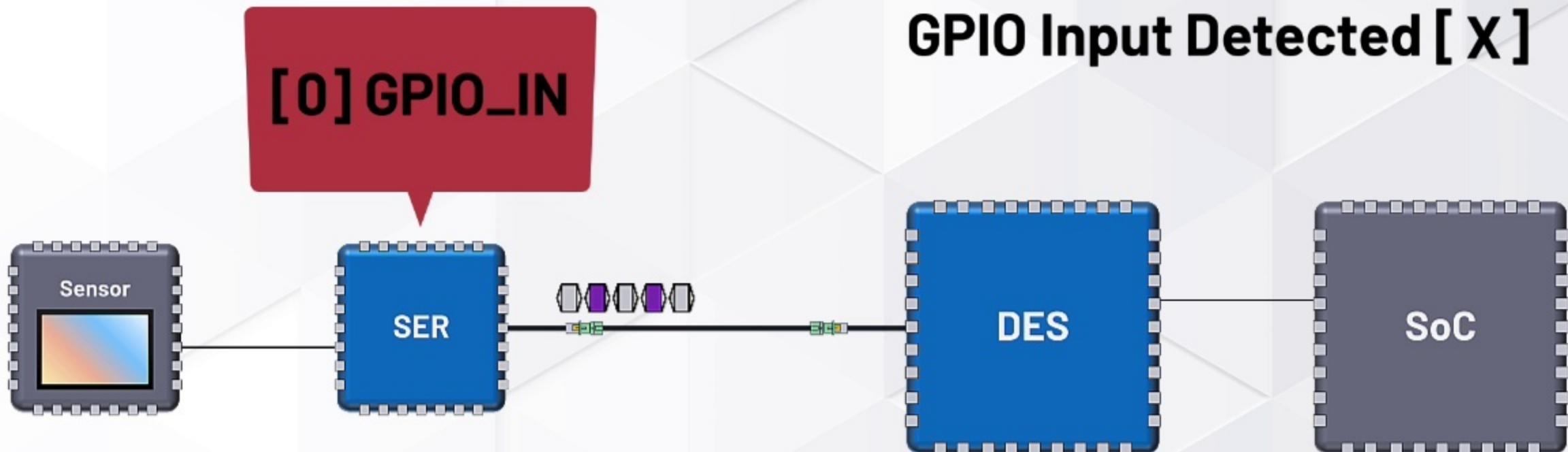


**GPIO Input Detected [ X ]**

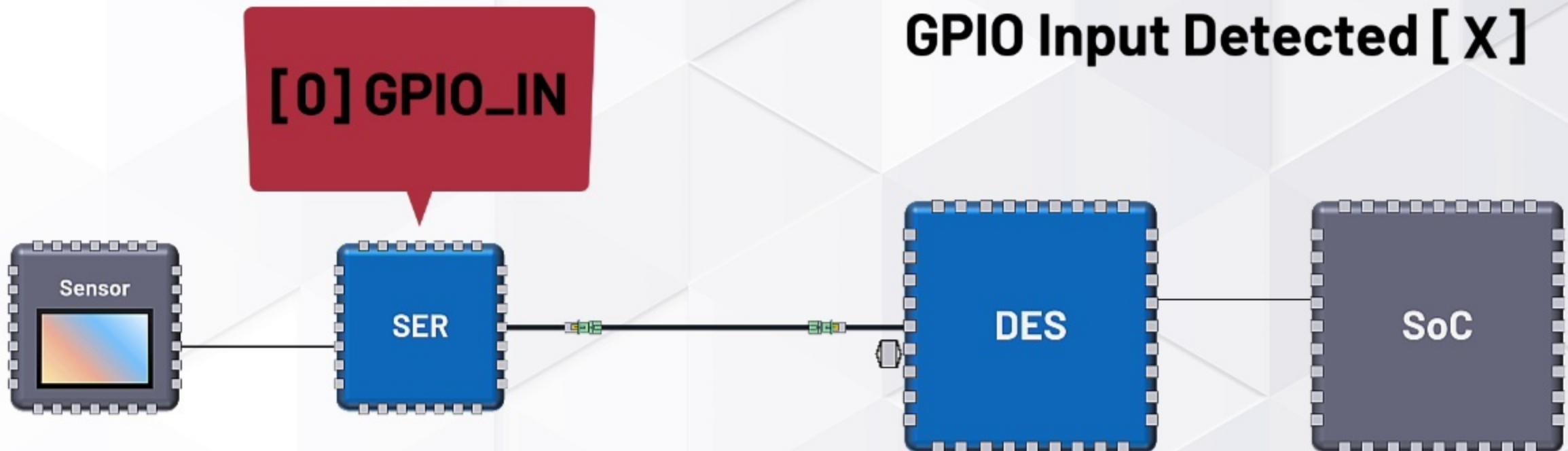
the GPIO in bit first.



The microprocessor sends a read command and waits for the

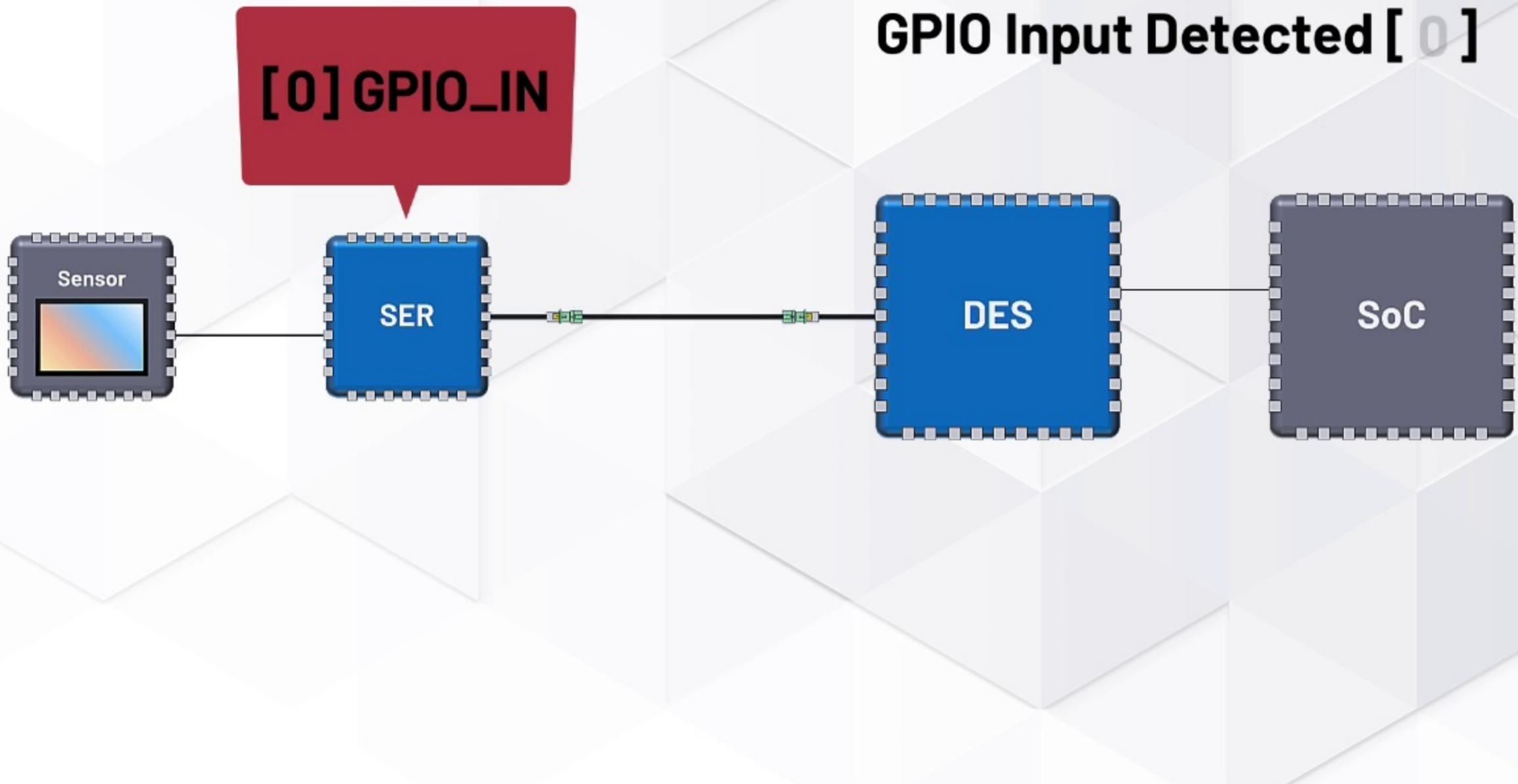


returning packet as the command travels to the remote SerDes



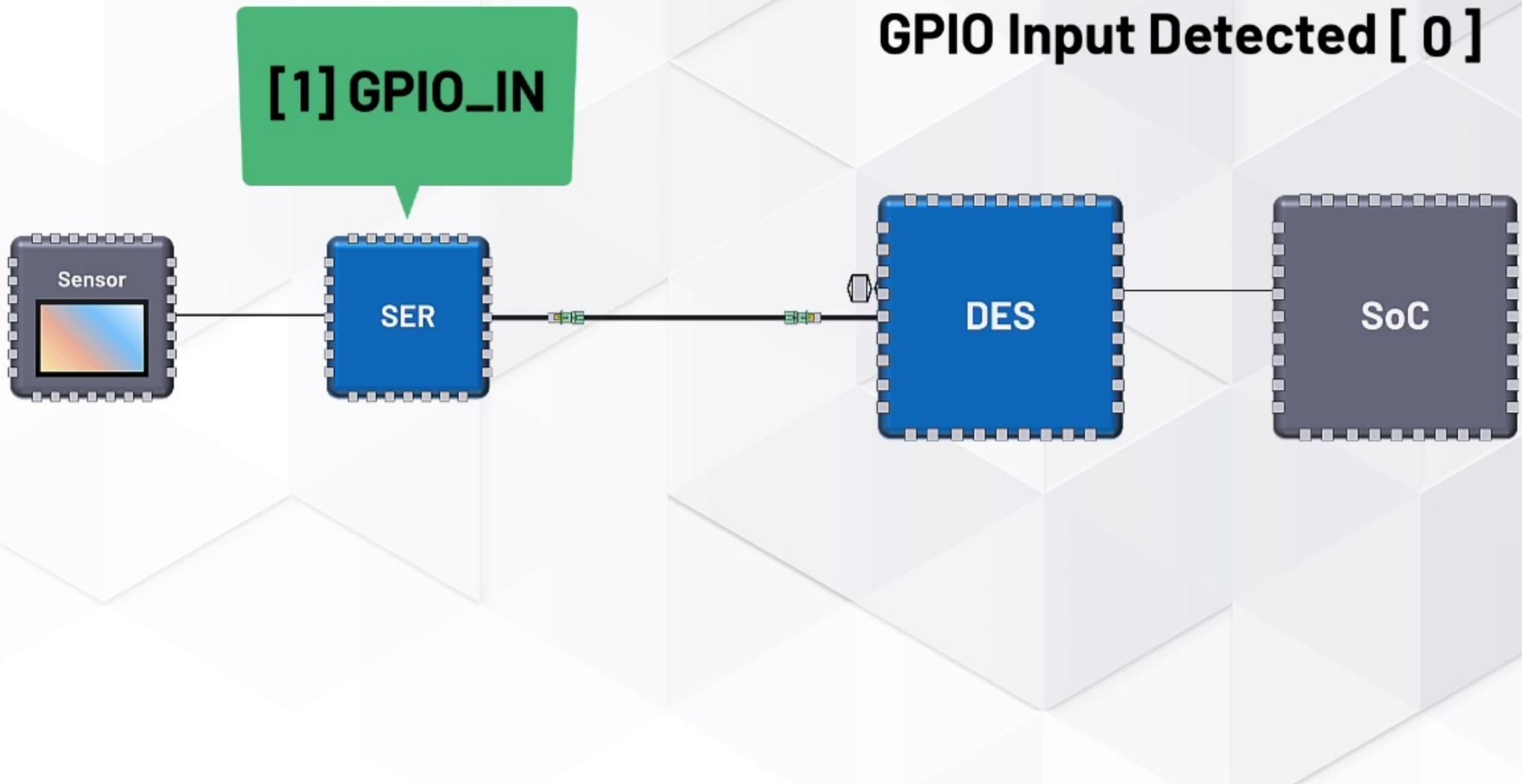
**GPIO Input Detected [ X ]**

device and returns to logic low.



**GPIO Input Detected [ 0 ]**

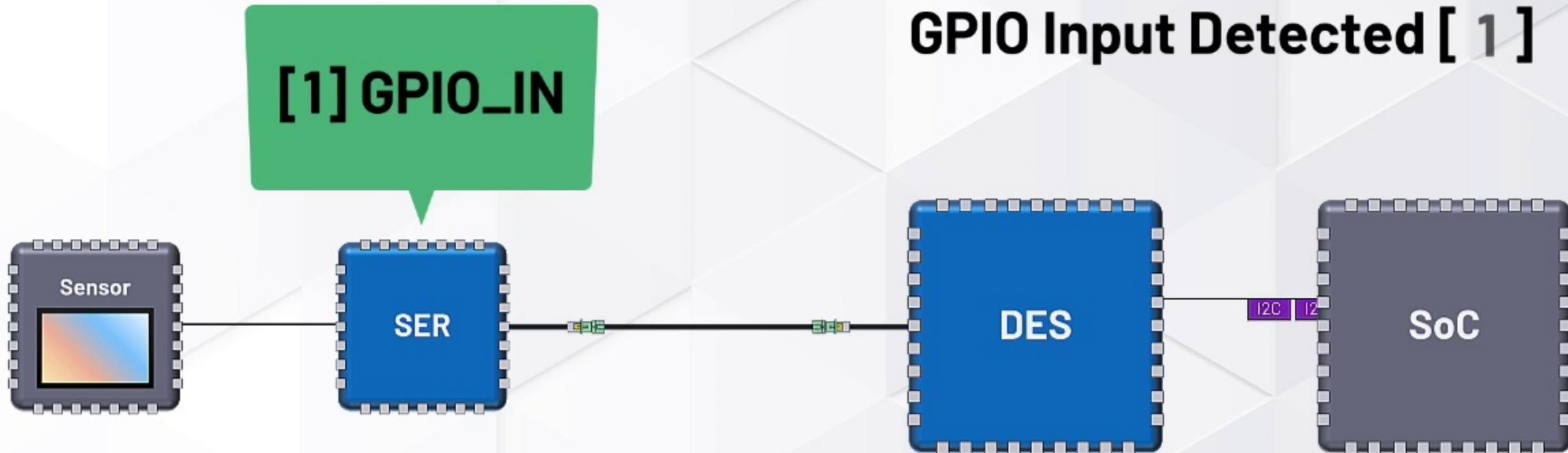
When the input changes and the micro checks the state again, it



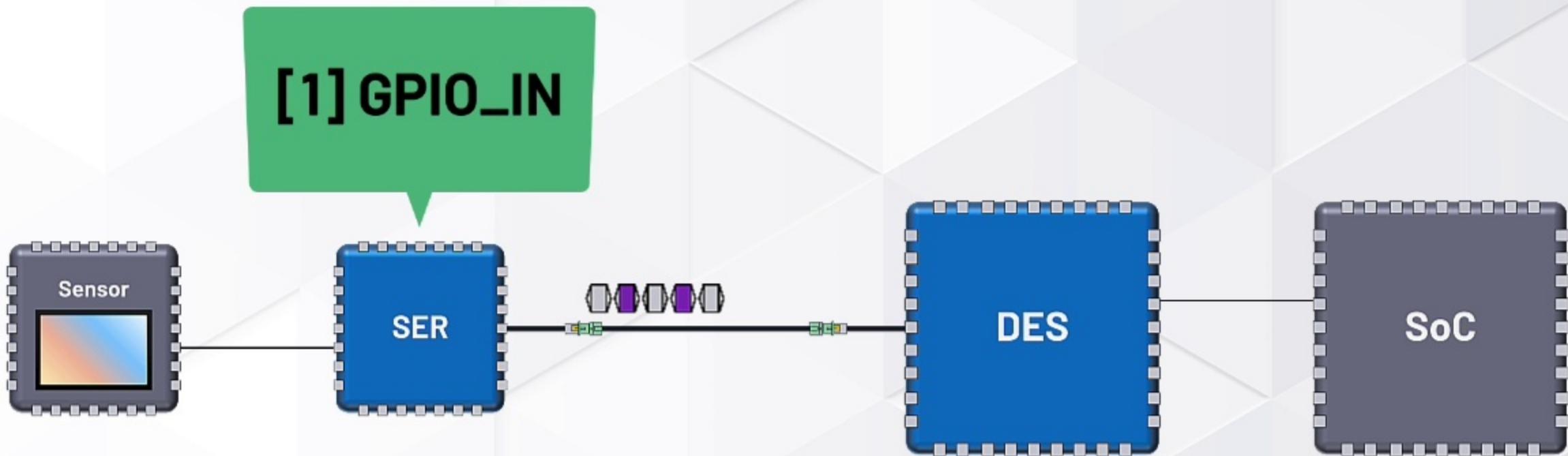
is detected as a logic high.



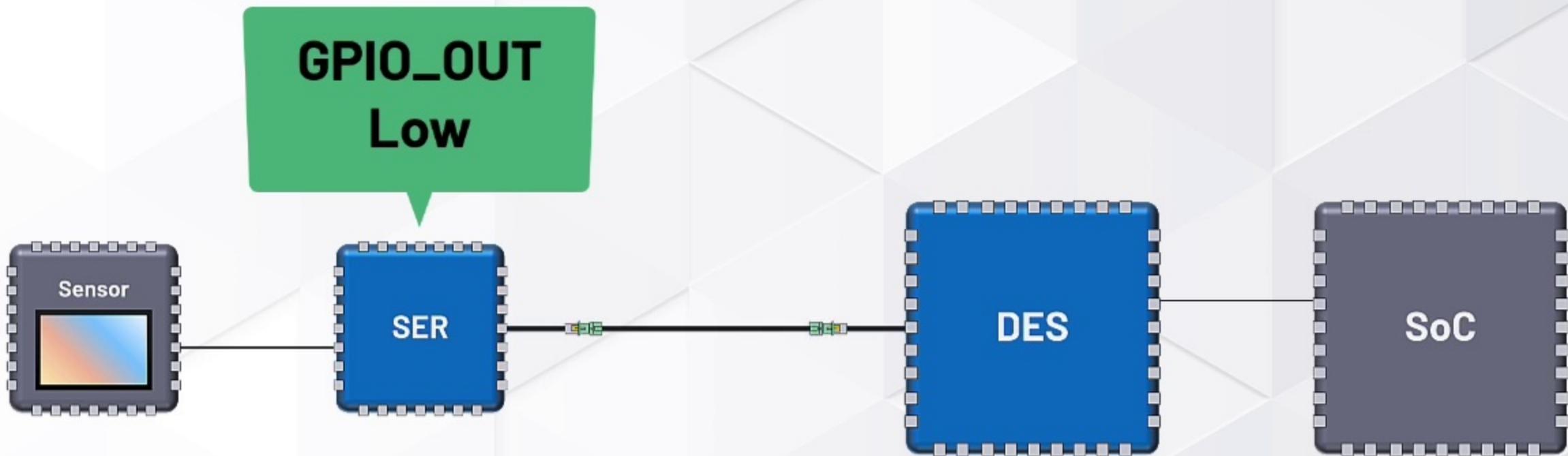
## GPIO Input Detected [ 1 ]



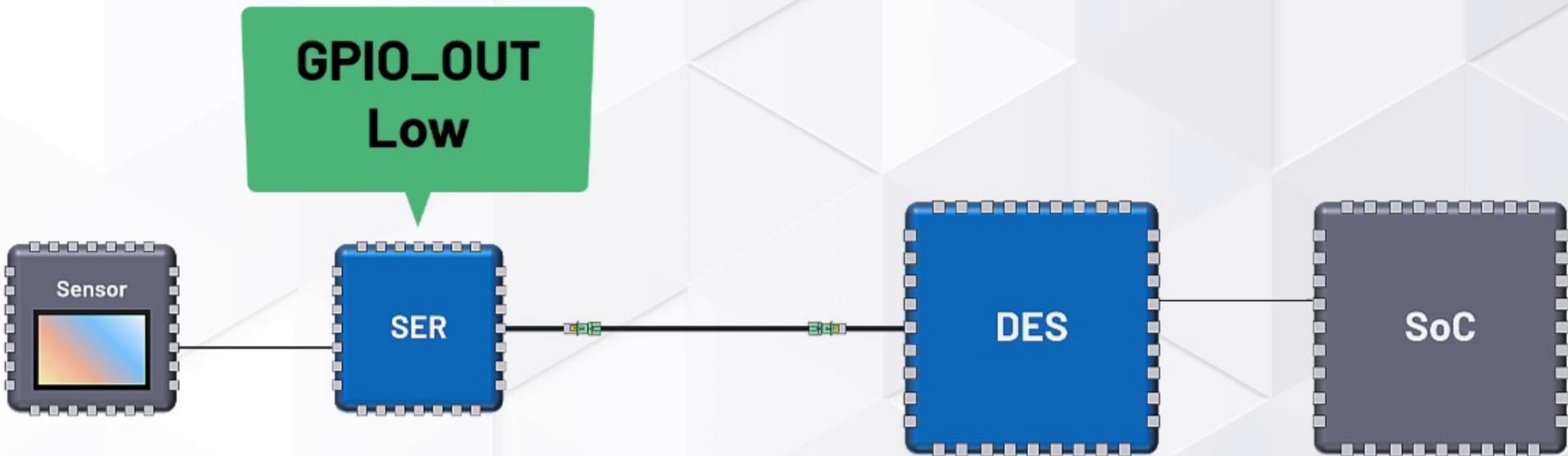
The GPIO's pins can also be manually controlled via writing



to the device registers to pull at logic high or low or to



report back the received state.



This can be helpful when GPIO'S on the peripheral devices are

limited.

Let's summarize what we've covered in this course.

► What is the purp

What is the purpose of GPIO'S?

- ▶ What is the purpose of GPIOs?
  - ▶ GPIOs are simple pins that allow receiving a signal input or transmitting a signal output.

GPIO'S are simple pins that allow receiving a signal input

- ▶ What is the purpose of GPIOs?
  - ▶ GPIOs are simple pins that allow receiving a signal input or transmitting a signal output.

or transmitting a signal output.

- ▶ What is the purpose of GPIOs?
  - ▶ GPIOs are simple pins that allow receiving a signal input or transmitting a signal output.
- ▶ What are the system benefits of GPIOs in a GMSL system?

What are the system benefits of GPIO's in a GMSL system?

- ▶ What is the purpose of GPIOs?
  - ▶ GPIOs are simple pins that allow receiving a signal input or transmitting a signal output.
- ▶ What are the system benefits of GPIOs in a GMSL system?
  - ▶ GPIOs can be used for a variety of low-level commands like reset, enable, synchronization, etc.

GPIO's can be used for a variety of low level commands like

- ▶ What is the purpose of GPIOs?
  - ▶ GPIOs are simple pins that allow receiving a signal input or transmitting a signal output.
- ▶ What are the system benefits of GPIOs in a GMSL system?
  - ▶ GPIOs can be used for a variety of low-level commands like reset, enable, synchronization, etc.

reset, enable synchronization, etcetera.

- ▶ How do context

How do you explain the terminology of GPIO in the

- ▶ How do you explain the terminology of GPIO in the context of a GMSL link?

context of a GMSL link?

- ▶ How do you explain the terminology of GPIO in the context of a GMSL link?
  - ▶ GPIO directions on the GMSL link need to be thought of in terms of the direction that the GPIO signal is traveling.

GPIO directions on the GMSL link need to be thought of in

- ▶ How do you explain the terminology of GPIO in the context of a GMSL link?
  - ▶ GPIO directions on the GMSL link need to be thought of in terms of the direction that the GPIO signal is traveling.

terms of the direction that the GPI of a signal is travelling.

- ▶ How do you explain the terminology of GPIO in the context of a GMSL link?
  - ▶ GPIO directions on the GMSL link need to be thought of in terms of the direction that the GPIO signal is traveling.

For example, if an SoC wants to communicate with an imager, the

- ▶ How do you explain the terminology of GPIO in the context of a GMSL link?
  - ▶ GPIO directions on the GMSL link need to be thought of in terms of the direction that the GPIO signal is traveling.

SoC would output a signal to the deserializer.

- ▶ How do you explain the terminology of GPIO in the context of a GMSL link?
  - ▶ GPIO directions on the GMSL link need to be thought of in terms of the direction that the GPIO signal is traveling.

The deserializer would then transmit the signal across the

- ▶ How do you explain the terminology of GPIO in the context of a GMSL link?
  - ▶ GPIO directions on the GMSL link need to be thought of in terms of the direction that the GPIO signal is traveling.

link for the serializer to receive the signal.

- ▶ How do you explain the terminology of GPIO in the context of a GMSL link?
  - ▶ GPIO directions on the GMSL link need to be thought of in terms of the direction that the GPIO signal is traveling.

The serializer would then output the received signal.

- ▶ How do you explain the terminology of GPIO in the context of a GMSL link?
  - ▶ GPIO directions on the GMSL link need to be thought of in terms of the direction that the GPIO signal is traveling.
- ▶ How do you explain the methods of interfacing with the GPIOs?

How do you explain the methods of interfacing with the GPIO's

- ▶ How do you explain the terminology of GPIO in the context of a GMSL link?
  - ▶ GPIO directions on the GMSL link need to be thought of in terms of the direction that the GPIO signal is traveling.
- ▶ How do you explain the methods of interfacing with the GPIOs?
  - ▶ GPIOs can be received signals going in to an I2C communication or a register or GPIO signals going out of a register.

GPIO's can be received and transmitted as low level signals

- ▶ How do you explain the terminology of GPIO in the context of a GMSL link?
  - ▶ GPIO directions on the GMSL link need to be thought of in terms of the direction that the GPIO signal is traveling.
- ▶ How do you explain the methods of interfacing with the GPIOs?
  - ▶ GPIOs can be received and transmitted as low-level signals going in to and out of the GMSL link. Using I2C communication, GPIO states can be read from a register or GPIO states can be set from a register.

going into and out of the GMSL link using I2C communication.

- ▶ How do you explain the terminology of GPIO in the context of a GMSL link?
  - ▶ GPIO directions on the GMSL link need to be thought of in terms of the direction that the GPIO signal is traveling.
- ▶ How do you explain the methods of interfacing with the GPIOs?
  - ▶ GPIOs can be received and transmitted as low-level signals going in to and out of the GMSL link. Using I2C communication, GPIO states can be read from a register or GPIO states can be set from a register.

GPIO states can be read from a register or GPIO states can be

- ▶ How do you explain the terminology of GPIO in the context of a GMSL link?
  - ▶ GPIO directions on the GMSL link need to be thought of in terms of the direction that the GPIO signal is traveling.
- ▶ How do you explain the methods of interfacing with the GPIOs?
  - ▶ GPIOs can be received and transmitted as low-level signals going in to and out of the GMSL link. Using I2C communication, GPIO states can be read from a register or GPIO states can be set from a register.

set from a register.

- What are the options for GPIO configuratic

What are the options for GPIO configurations?

- ▶ What are the options for GPIO configurations?
  - ▶ GPIOs can be configured with ti IDs for mapping, pull-up and pu can be enabled, and jitter can b sensitive applications.

GPIO's can be configured with transmit and receive lges for

- ▶ What are the options for GPIO configurations?
  - ▶ GPIOs can be configured with transmit and receive IDs for mapping, pull-up and pull-down resistors can be enabled, and jitter can be compensated for sensitive applications.

mapping.

- ▶ What are the options for GPIO configurations?
  - ▶ GPIOs can be configured with transmit and receive IDs for mapping, pull-up and pull-down resistors can be enabled, and jitter can be compensated for sensitive applications.

Pull up and pull down resistors can be enabled and jitter can be

- ▶ What are the options for GPIO configurations?
  - ▶ GPIOs can be configured with transmit and receive IDs for mapping, pull-up and pull-down resistors can be enabled, and jitter can be compensated for sensitive applications.

compensated for sensitive applications with the GMSL link



already in the system.



GPIO's provide a simple but efficient method of low level



signalling over long distances for any general purpose.