axis_data_to_axis_string.v

AUTHORS

JAY CONVERTINO

DATES

2022/09/19

INFORMATION

Brief

Parse raw binary data into ASCII string output.

License MIT

axis_data_to_axis_string

```
module axis_data_to_axis_string #(
parameter
DELIMITER
 =
";",
parameter
TERMINATION
 =
"\n",
parameter
SBUS_WIDTH
 =
1,
parameter
```

```
USER_WIDTH
 =
4,
parameter
DEST_WIDTH
 =
4,
parameter
PREFIX_LEN
 =
1,
parameter
DATA_PREFIX
 =
"#",
parameter
DEST_PREFIX
 =
"&",
parameter
USER_PREFIX
 =
"*"
) ( input aclk, input arstn, input [(SBUS_WIDTH*8)-1:0] s_axis_tdata, input
```

Parse raw binary data into ASCII string output.

Parameters

DELIMITER parameter	break value between multple strings
TERMINATION parameter	termination value of full string from serial port, byte only. (\n = 0A \r = 0D).
SBUS_WIDTH parameter	bus width of slave (data) input
USER_WIDTH parameter	user width of slave bus, only in 4 bit nibbles, and at least 4 bits.
DEST_WIDTH parameter	dest width of slave bus, only in 4 bit nibbles, and at least 4 bits.
PREFIX_LEN parameter	length of following prefix strings.
DATA_PREFIX parameter	prefix for data hex strings
DEST_PREFIX parameter	prefix for destination hex strings
USER_PREFIX parameter	prefix for user hex strings

Ports

aclk	Clock for AXIS
arstn	Negative reset for AXIS
s_axis_tdata	Input data
s_axis_tvalid	When set active high the input data is valid
s_axis_tuser	User data to convert.
s_axis_tdest	Destination data to convert
s_axis_tready	When active high the device is ready for input data.
m_axis_tdata	Output data

m_axis_tvalid	When active high the output data is valid
m_axis_tready	When set active high the output device is ready for data.

VARIABLES

s_axis_tready

```
assign s_axis_tready = (
                                                    (counter == 0) ? 1 &

arstn
:
0
)
```

ready if count is zero, this is a FWFT so no worries in pumping out data.

m_axis_tdata

```
assign m_axis_tdata = char_buffer[STRING_LEN*8-1 -:8]
```

output whatever is in the character buffer.

m_axis_tvalid

```
assign m_axis_tvalid = (
                                                    counter > 0 ?

1
:
0
)
```

Counter greater than 0? Valid output is available.