Data Type

In DL PEL-HIL, we recommend using three data types: boolean, uint16, and single.

Data Type	Size	Range	Description
boolean	1 bit	1/0	Represents logical states such as on/off or yes/no. Used for flags and condition checks.
uint16	16-bit	[0, 65535]	Efficient for counters, IDs, and hardware register values. Saves memory for positive integers.
single	32-bit	$\pm 1.18 \times 10^{-38}$ to $\pm 3.4 \times 10^{38}$	Used for numeric data with decimals. Suitable for sensor values and control algorithms.

Notes:

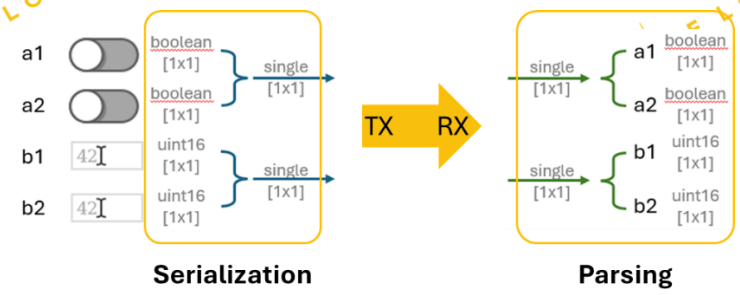
💡 *Some DL built-in blocks specify a data type and therefore require matching input/output signals for consistency.*

💡 *Use single type for precision; use boolean and uint16 for speed and memory efficiency.*

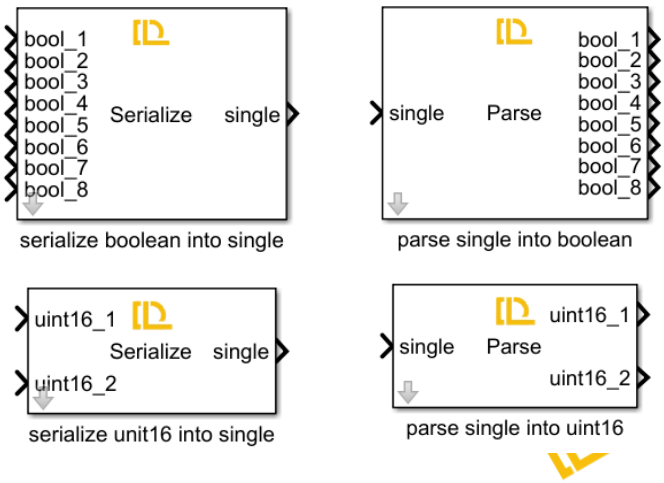
Data Serialization & Parsing

Single type is unified in DL PEL-HIL RX/TX; multiple **booleans** and **uint16s** are serialized into singles before TX.

After RX, **Single** in DL PEL-HIL RX/TX is parsed back into **booleans** and **uint16s** to restore data.



Serialization and Parsing Process



Blocks for Serialization and Parsing

Signal Dimensions

In the context of communication, a signal can take various forms:

- a single data element,
- a combination of multiple single-element signals, or
- a set of signals that each include historical (past) data.

As a starting point, let's first explore the different dimensions that signals can have.

Size	Example	Terminology	Note
[1x1]	5	Scalar	A single value
[2x1]	$\begin{bmatrix}1.1\\2.1\end{bmatrix}$	Column vector	A vector with 2 rows and 1 column
[1x3]	$\begin{bmatrix}1.1 & 1.2 & 1.3\end{bmatrix}$	Row vector	A vector with 1 row and 3 columns
[2x3]	$\begin{bmatrix}1.1 & 1.2 & 1.3\\2.1 & 2.2 & 2.3\end{bmatrix}$	Matrix	A matrix with 2 rows and 3 columns

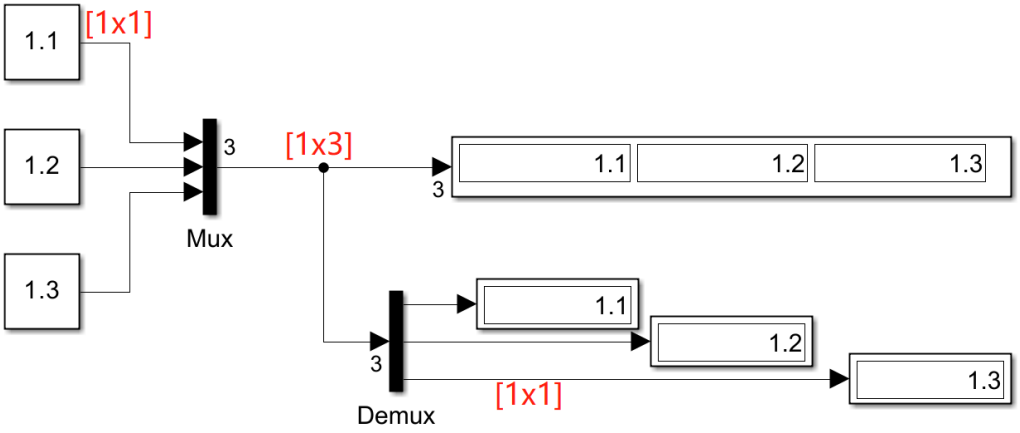
In DL PEL-HIL, we use

- **Scalar** to represent a single data element,
- **Row Vector** to represent a combination of multiple single-element signals,
- **Matrix** to represent a set of signals that each include historical (past) data.

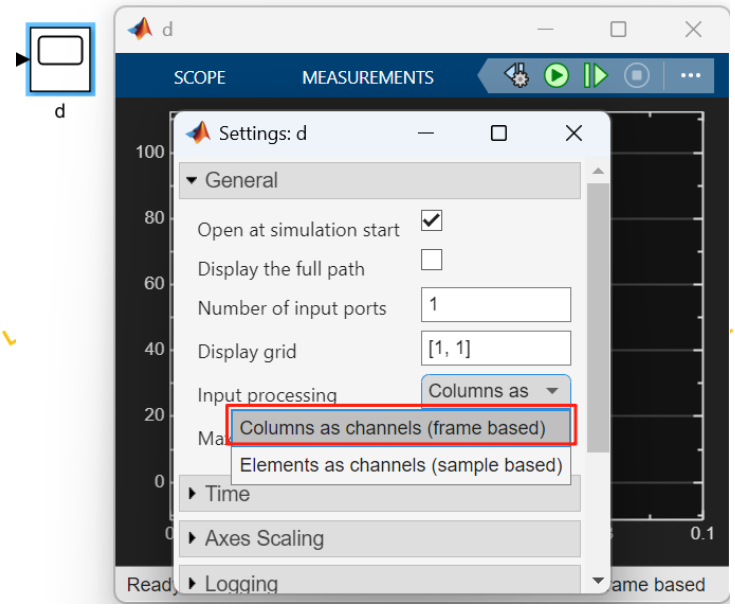
Row Vector Construction & Extraction

Use mux to combine same-type **scalar** elements into a **row vector**.

Use demux to extract **scalar** elements from a **row vector**.

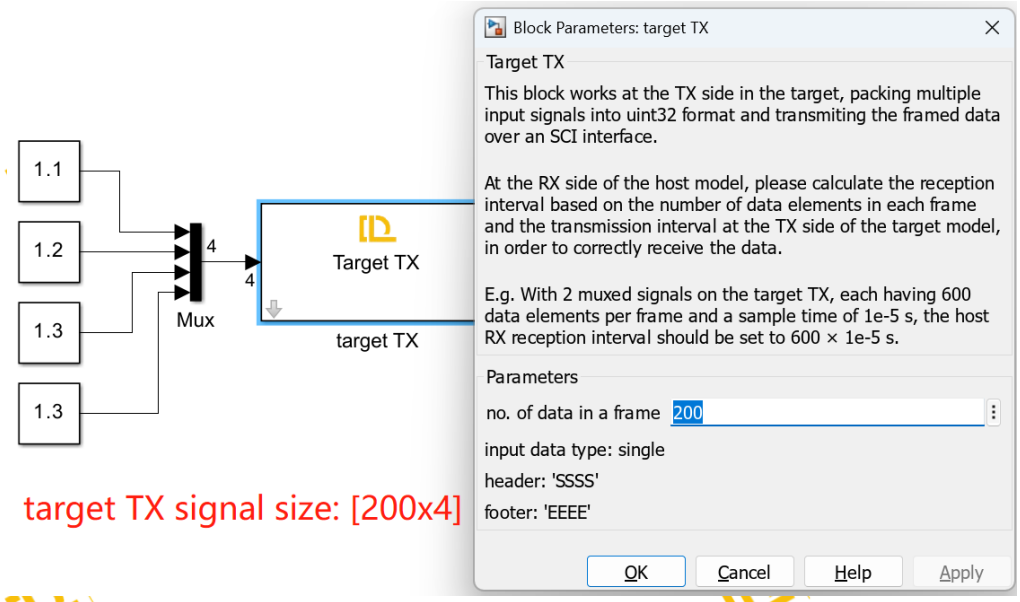


To observe the waveform of a row vector with an oscilloscope, process the input as **columns as channels**.



Matrix Construction & Extraction

Matrix creation occurs only before target model's TX: **no. of columns** defined by mux, **no. of rows** defined by target TX's 'no. of data in a frame'.



Column extraction occurs only after host model's RX, done by split matrix.