StreamDevice: waveform Records

Normal Operation

With waveform records, the format converter is applied to each element. Between the elements, a separator is printed or expected as specified by the `Separator` variable in the protocol. When parsing input, a space as the first character of the `Separator` matches any number of any whitespace characters.

During input, a maximum of `NELM` elements is read and `NORD` is updated accordingly. Parsing of elements stops when the separator does not match, conversion fails, or the end of the input is reached. A minimum of one element must be available.

During output, the first `NORD` elements are written.

The format data type must be convertible to or from the type specified in the `FTVL` field. The variable *x[i]* stands for one element of the written or read value.

DOUBLE format (e.g. `%f`):

Output:*x[i]*=`double(VAL[i])`
`FTVL` can be `"DOUBLE"`, `"FLOAT"`, `"LONG"`, `"ULONG"`, `"SHORT"`, `"USHORT"`, `"CHAR"`, `"UCHAR"`, or `"ENUM"` (which is treated as `"USHORT"`).
Input: `VAL[i]=FTVL(`*x[i]*`)`
`FTVL` must be `"FLOAT"` or `"DOUBLE"`

LONG or ENUM format (e.g. `%i` or `%{`):

Output: *x[i]*=`long(VAL[i])`
`FTVL` can be `"LONG"`, `"ULONG"`, `"SHORT"`, `"USHORT"`, `"CHAR"`, `"UCHAR"`, or `"ENUM"` (which is treated as `"USHORT"`).
Signed values are sign-extended to long, unsigned values are zero-extended to long before converting them.
Input: `VAL[i]=FTVL(`*x[i]*`)`
`FTVL` can be `"DOUBLE"`, `"FLOAT"`, `"LONG"`, `"ULONG"`, `"SHORT"`, `"USHORT"`, `"CHAR"`, `"UCHAR"`, or `"ENUM"` (which is treated as `"USHORT"`).
The value is truncated to the least significant bytes if `FTVL` has a smaller data size than `long`.

STRING format (e.g. `%s`):

If `FTVL=="STRING"`:

Output: *x[i]*=`VAL[i]`
Input: `VAL[i]=`*x[i]*
Note that this is an array of strings, not an array of characters.

If `FTVL=="CHAR"` or `FTVL="UCHAR"`:

In this case, the complete waveform is treated as a large single string of size `NORD`. No separators are printed or expected.
Output: *x*=`range(VAL,0,NORD)`
The first `NORD` characters are printed, which might be less than `NELM`.
Input: `VAL=`*x*`, NORD=length(`*x*`)`
A maximum of `NELM-1` characters can be read. `NORD` is updated to the index of the first of the trailing zeros. Usually, this is the same as the string length.

Other values of `FTVL` are not allowed for this format.

Initialization

During initialization, the `@init` handler is executed, if present. All format converters work like in normal operation.

---

aai aao ai ao bi bo mbbi mbbo mbbiDirect mbboDirect longin longout stringin stringout calcout scalcout

Dirk Zimoch, 2005