

AudioScience

**IEEE P1722.1**  
**Control/Mixer/Matrix**  
**Descriptors**

Eliot Blennerhassett, AudioScience Inc

# Introduction

- The generic control, mixer and matrix share common definition of parameter values and parameter value specifications
- Define these separately from generic control
- In the details of the definitions there is a lot of regularity, that makes it easier to understand the definitions by building them up from basic data types to complete definitions

# Data types

- Data types are common to (at least) Control, Mixer, Matrix descriptors. They appear both in the descriptors, and in the GET/SET PDUs for the described objects .
- So define separately from the Control descriptor
- Basic numeric types INT8, UINT8, INT16, UINT16, INT32, UINT32, INT64, UINT64, FLOAT, DOUBLE
- Non-numeric: UTF-8, SAMPLERATE?, DESCRIPTOR\_ID, STRING\_INDEX
- Structured: BODE\_PLOT\_POINT, GPTP\_TIME, SMPTE\_TIME

# Valid values specifications

- Two main ways to define the valid values for a single element.
  1. LINEAR: [min,max,step] applies to numeric types
  2. ENUMERATED : (count, [value, value,...])

(Other ways to define validity are implicit, or a special way determined by the type)

# Value Specification Applicability

- Two ways of applying a validity specification to an array of values.
- COMMON : One spec applies to all elements
- ARRAY : An array of specifications, 1:1 with values.

# Parameter Value Specs.

- Combine data type, validity spec, applicability to get overall details type, for example
- COMMON\_LINEAR\_UINT8
- ARRAY\_LINEAR\_UINT8
- COMMON\_ENUMERATED\_SAMPLERATE
- ARRAY\_ENUMERATED\_TYPE possible, but not currently supported. Limited usefulness because of PDU size limit?

# Value spec common fields

- The numeric specs (LINEAR, ENUMERATED) share common fields: units, default\_value, string, so use a common layout
  - Units (16 bit units)
  - String (16 bit string index)
  - default\_value ( $W * 8$  bits)
  - (min,max,step) or (count, (val,val,...))
-

# Spec doesn't need Current value

- Current parameter value specifications include `current_value`
- However, the `current_value` is always (?) invalid for Mixer and Matrix, and for Control when `unknown_value` flag is set.
- The protocol must still work with the `current_value` removed from the specs, so start from this, and then allow `current_value` to be appended as an optimization.



# Separate value spec from values

Offset (octets)	Length (octets)	Name	Description
0	D	Many fields	Depending on descriptor type
	0.125	R (1 bit field)	Flag; 1 : read-only, 0: read/write
	0.125	U (1 bit field)	Flag; 1: value not present, 0: value present
	1.75	parameter_details_type (14 bit field)	Determines layout of parameter_value_details and parameter_values
	2	number_of_values	Number of elements in parameter_values_array
X	P	parameter_details	Parameter value specification (See section #.#)
X+P	V	parameter_values	Parameter current values. Only present if unknown_value flag==0

# Benefits

- Clear separation of value metadata from values
- Simple device can use static descriptors (current values always retrieved separately)
- Maximum value count is increased slightly (omitting current values leaves more room for specs)
- Small mixer and matrix can include current values
- Layout of Control current values part is shared between descriptor and GET/SET PDU
- Avoidance of conditional language e.g. “The values field only conveys the current value and not the max, min, default and step”

# Minor points

- Is there a better name for BODE\_PLOT? Each array element is a tuple of (magnitude, phase, frequency). Call these what? Phasor? Fourier\_component? ??? The array of values is a ? Frequency\_response?
- It **is** possible to specify some metadata about UTF-8 values: maximum\_octet\_count, maximum\_characters

END