Signal Measurement API Documentation

Each function measures a specific characteristic of a signal relative to a reference. If the hardware for the measurement is unavailable, the function returns an expected default value plus a random error spread, simulating measurement noise.

Common Parameters

- **signal** (`str`): The signal name or identifier; default is `'VCC'`.
- **reference** (`str`): The reference node for the measurement; default is `'GND'`.
- **expected_value** (`int|float|dict|None`): The expected or fallback value returned if hardware is not available. It can be a number or complex type (e.g., dict for spectrum).
- **error_spread** (`float`): The maximum variability to add as noise when returning the fallback value. Default is `0.0`. For integer counts, error_spread is typically zero.

API Functions

SIGNAL_PHASE_MEASURE

- **template** (`API`): The signal name or identifier; default is `SIGNAL_PHASE_MEASURE(signal: str = 'VCC', reference: str = 'GND', expected_value: float = 0.0, error_spread=0.0)`. Measure the phase difference between the signal and reference. Returns: Phase in degrees or radians (float).

SIGNAL_PEAK_TO_PEAK_MEASURE

Measure the peak-to-peak amplitude of the signal waveform. Returns: Peak-to-peak voltage (float).

SIGNAL_RMS_MEASURE

Measure the Root Mean Square (RMS) value of the signal. Returns: RMS voltage (float).

SIGNAL_MEAN_MEASURE

Measure the mean (average) value of the signal across a measurement window. Returns: Mean voltage (float).

SIGNAL_SPECTRUM_MEASURE

Calculate the frequency spectrum of the signal. Returns: A spectrum representation; usually a dictionary or array of frequency bins and magnitudes.

SIGNAL_MIN_MEASURE

Measure the minimum voltage level of the signal during the measurement period. Returns: Minimum voltage (float).

SIGNAL_MAX_MEASURE

Measure the maximum voltage level of the signal during the measurement period. Returns: Maximum voltage (float).

SIGNAL_HIGH_MEASURE

Measure the high-level voltage of the signal. Returns: High-level voltage (float).

SIGNAL_LOW_MEASURE

Measure the low-level voltage of the signal. Returns: Low-level voltage (float).

SIGNAL_AMPLITUDE_MEASURE

Measure the amplitude of the signal (peak relative to base). Returns: Amplitude voltage (float).

SIGNAL_POSITIVE_PULSE_COUNT_MEASURE

Count the number of positive pulses detected on the signal line. Returns: Number of positive pulses (int).

SIGNAL_NEGATIVE_PULSE_COUNT_MEASURE

Count the number of negative pulses detected on the signal line. Returns: Number of negative pulses (int).

SIGNAL_RAISING_EDGE_COUNT_MEASURE

Count the rising edges detected on the signal. Returns: Number of rising edges (int).

SIGNAL_FALLING_EDGE_COUNT_MEASURE

Count the falling edges detected on the signal. Returns: Number of falling edges (int).

Usage Notes

- All measurements require that the appropriate hardware callback is registered under the respective measurement key (e.g., `'signal_peak_to_peak_measure'`) inside `g.hardware_callbacks`.
- Returned values are appended to `g.output` with a dictionary noting the measurement type, signals, and obtained values for traceability.
- If hardware is unavailable or the measurement fails, the function returns `expected_value ± error_spread` for numeric metrics, or `expected_value` as-is for complex types like spectrum.
- The APIs are designed for flexibility, enabling substitution between real hardware measurement and software simulation/mock.
