Design: Captures attributes of an LWC hardware implementation. - **name** *(string)*: A unique identifier for the design. It can consist of English letters, digits, dashes, and underscores and must start with a letter. - **description** *(string)*: A short description of the design. - **author** *(string or array of string)*: Author or list of authors and developers who have contributed to this implementation. - **url** *(string)*: Uniform Resource Locator pointing to a webpage or source repository associated with this design or its author(s). - **license** *(string or array of string)*: License or licenses covering this design's use and distribution. Use SPDX (ISO/IEC 5962:2021) short identifiers when applicable. *Examples:* `"SHL-2.1"` - **version** *(string)*: Design version. *Examples:* `"0.0.1"` - **rtl**: Details of the synthesizable RTL design. - **sources** *(array of string)*: Non-empty list of HDL source file paths in correct compilation order. All paths must be relative to the location of the configuration file. Path separator is `/` (slash) on all platforms. Paths are case-sensitive and should not contain whitespaces. HDL language is inferred from the file extension. - **includes** *(array of string)*: Non-empty list of HDL include file paths, such as Verilog headers. Include files are not directly compiled but need to be present for elaboration of design. Order is arbitrary. All paths must be relative to the location of the configuration file. Path separator is `/` (slash) on all platforms. Paths are case-sensitive and should not contain whitespaces. *Default:* `[]` - **top** *(string)*: Name of top-level RTL entity/module. *Default:* `LWC` - **clock**: Top level RTL clock signal. Only a single clock is supported by LWC API. - **port** *(string)*: Name of the top-level RTL clock input. *Default:* `clk` - **reset**: Top level reset signal. Only a single reset is supported by LWC API. - **port** *(string)*: Name of top-level RTL reset input. *Default:* `reset` - **active*high*** **(boolean)*: Polarity of the reset signal. Active-high (positive) if true, otherwise active-low. _Default:* `true` - **asynchronous** *(boolean)*: Whether reset is asynchronous with respect to `rtl.clock`. *Default:* `false` - **parameters**: Top-level design parameters or generics specified as a key-value map. The default value of each parameter is overridden by synthesis tool, simulator, testbench, or wrapper. For the best tool compatibility, we only support integer and string values. *Examples:* `{"G_NUM_SHARES": 2, "G_BACKDOOR": 0}` - **language**: Information about Hardware Description/ Design Language(s). - **vhdl**: Common VHDL features supported by all VHDL source files. VHDL files must have a `.vhd` or `.vhdl` extension. - **version** *(string)*: VHDL language standard. *Supported values:* `1993`, `2000`, `2002`, `2008` *Default:* `1993` - **synopsys** *(boolean)*: Use of non-standard Synopsys packages which were placed in the IEEE namespace, e.g. 'std*logic*arith'. Dependence on such packages is *strongly* discouraged. *Default:* `false` - **verilog**: Common Verilog (pre-SystemVerilog) language features supported by all Verilog source files. Verilog files must have a `.v` extension. - **version** *(string)*: Verilog language standard. *Supported values:* `1995`, `2001` *Default:* `2001` - **systemverilog**: SystemVerilog (IEEE 1800-2005 and onwards) language features supported by all SystemVerilog source files. SystemVerilog files must have a `.sv` extension. - **version** *(string)*: SystemVerilog language standard. *Supported values:* `2005`, `2009` *Default:* `2009` - **tb**: Details of test-bench used for verification of top-level design. [Optional]. - **sources** *(array of string)*: Source files used only for verification. Should *not* contain any of the files included in 'rtl.sources'. - **includes** *(array of string)*: HDL include file paths. *Default:* `[]` - **top** *(string)*: Name of top-level test entity or module. - **parameters**: Testbench parameter or generics specified as a key-value map. The default value of each parameter is overridden by the simulator. For the best tool compatibility, we only support integer and string values. *Examples:* `{"G_TEST_MODE": 0}` - **language**: Information about HDL or programming languages used in the testbench. - **vhdl**: Common VHDL features supported by all VHDL source files. VHDL files must have a `.vhd` or `.vhdl` extension. - **version** *(string)*: VHDL language standard. *Supported values:* `1993`, `2000`, `2002`, `2008` *Default:* `1993` - **synopsys** *(boolean)*: Use of non-standard Synopsys packages which were placed in the IEEE namespace, e.g. 'std*logic*arith'. Dependence on such packages is *strongly* discouraged. *Default:* `false` - **verilog**: Common Verilog (pre-SystemVerilog) language features supported by all Verilog source files. Verilog files must have a `.v` extension. - **version** *(string)*: Verilog language standard. *Supported values:* `1995`, `2001` *Default:* `2001` - **systemverilog**: SystemVerilog (IEEE 1800-2005 and onwards) language features supported by all SystemVerilog source files. SystemVerilog files must have a `.sv` extension. - **version** *(string)*: SystemVerilog language standard. *Supported values:* `2005`, `2009` *Default:* `2009` - **python** - **version** *(string)* - **framework** *(string)*: *Supported values:* `cocotb` *Default:* `cocotb` - **lwc**: LWC-specific meta-data. - **algorithm** *(string or array of string)*: LWC AEAD/Hash algorithm(s) supported by this design. Should follow SUPERCOP naming conventions and uniquely identify the scheme's variant and version. In case of duplicates, the second instance indicates support for AEAD and Hash algorithms with the same name. *Examples:* `["giftcofb128v1"]`, `["romulusn1v12"]`, `["gimli24v1", "gimli24v1"]` - **input*sequence***: *Order in which different input segment types should be fed to PDI. - **encrypt** *(array of string)*: _Default:* `['npub', 'ad', 'pt', 'tag']` - **decrypt** *(array of string)*: *Default:* `['npub', 'ad', 'ct', 'tag']` - **pt*block*bits** *(integer)*: Algorithm's size of plaintext/ciphertext 'blocks' in bits. This is the number of bits that the algorithm operates upon during its basic operations. Potentially used for the evaluation of some performance metrics. - **ad*block*bits** *(integer)*: Algorithm's size of associated-data 'blocks' in bits. This is the number of bits that the algorithm operates upon during its basic operations. Potentially used for the evaluation of some performance metrics. - **key*bits*** **(integer)*: _Default:* `128` - **npub*bits*** **(integer)*: _Default:* `128` - **tag*bits*** **(integer)*: _Default:* `128` - **hash** - **supported** *(boolean)*: Whether this implementation supports hashing. *Default:* `false` - **digest*bits*** **(integer)*: Size of hash digest (output) in bits. _Default:* `128` - **ports**: Description of LWC ports. - **pdi**: Public Data Input port. - **bit*width*** **(integer)*: Width of each share of PDI data (bits). Width of '*pdi*data' signal would be `pdi.bit_width × pdi.num_shares` bits. *Minimum:* `8` *Maximum:* `32` *Default:* `32` - **num*shares*** **(integer)*: Number of PDI shares. Default is 1 (unprotected). _Default:* `1` - **sdi**: Secret Data Input port. - **bit*width*** **(integer)*: Width of each share of SDI data (bits). Width of `sdi_data`*

signal would be `sdi.bit_width × sdi.num_shares` *bits. _Minimum:* `8` *Maximum:* `32` *Default:* `32` - **num*shares*** *(integer)*: Number of SDI shares. _Minimum:* `1` *Default:* `1` - **rdi**: Random Data Input port. - **bit*width*** *(integer)*: _Minimum:* `0` *Maximum:* `2048` *Default:* `0` - **sca*protection***: *Implemented countermeasures against side-channel attacks. -* ***target*** *(array of string)*: Type of side-channel analysis attack(s) against which this design is assumed to be secure. _Examples:* `["spa", "dpa", "cpa", "timing"]`, `["dpa", "sifa", "dfia"]` - **masking*schemes*** *(array of string)*: Masking scheme(s) applied in this implementation. Could be name/abbreviation of established schemes (e.g., "DOM", "TI") or reference to a publication. _Default:* `[]` *Examples:* `["TI"]`, `["DOM", "https://eprint.iacr.org/2022/000.pdf"]` - **order** *(integer)*: Claimed order of protectcion. 0 means unprotected. *Default:* `0`