```
<!-- Root element -->
<!ELEMENT RULE_SET (CPU, TABLE*, TOKEN*, RULE+)>


<!-- CPU construction -->
<!ELEMENT CPU (REGISTER+)>
<!ELEMENT REGISTER (INDEX+)>
<!ELEMENT INDEX (EMPTY)>
<!ATTLIST REGISTER size CDATA #REQUIRED>


<!-- 'size' - It is implied that the size will be set to the value of REGISTER if not given
-->
<!-- 'bit' - Is the offset from the beginning of the register. If not given value is set to
zero -->
<!ATTLIST INDEX
        name CDATA #REQUIRED
        size CDATA #IMPLIED
        bit CDATA #IMPLIED>


<!-- Generic lookup table -->
<!ELEMENT TABLE (FUNCTION, KEY+)>
<!ATTLIST TABLE name CDATA #REQUIRED>


<!ELEMENT FUNCTION (ARGUMENT+)>
<!ELEMENT FUNCTION name CDATA #REQUIRED>


<!ELEMENT ARGUMENT (EMPTY)>
<!ATTLIST ARGUMENT name CDATA #REQUIRED>


<!ELEMENT KEY (VALUE)>
<!ATTLIST KEY name CDATA #REQUIRED>


<!ELEMENT VALUE (KEY|TARGET+)>
<!ATTLIST VALUE val CDATA #REQUIRED>


<!ELEMENT TARGET (EMPTY)>
<!ATTLIST TARGET name CDATA #REQUIRED>

<!-- Token - user defined keyword for grammar -->
<!ELEMENT TOKEN (BINARY_DATA)>
<!ATTLIST TOKEN name CDATA #REQUIRED>

<!-- Data for a token - represents a need to parse N number of bits -->
<!ELEMENT BINARY_DATA (BINARY_ITEM+)>
<!ATTLIST BINARY_DATA size CDATA #REQUIRED>


<!-- ITEM -->
<!ELEMENT BINARY_ITEM (EMPTY)>
<!ATTLIST name CDATA #REQUIRED
        shift CDATA #IMPLIED
        mask CDATA #REQUIRED>

<!-- Rule -->
<!ELEMENT RULE (GRAMMAR, OUTPUT?)>
<!ATTLIST RULE name CDATA #REQUIRED>

<!-- What to look for in the input for this rule -->
<!ELEMENT GRAMMAR (EMPTY)>

<!-- What to do once the target input is found -->
<!ELEMENT OUTPUT (EMPTY)>
```