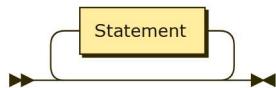
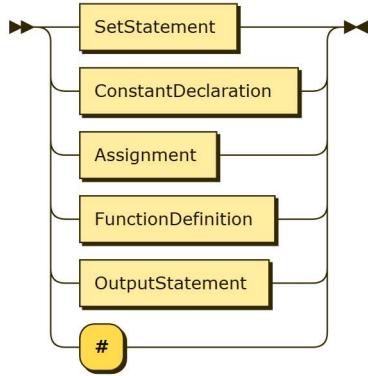


Program:

Program ::= Statement*

no references

Statement:

```

Statement
::= SetStatement
| ConstantDeclaration
| Assignment
| FunctionDefinition
| OutputStatement
| '#'

```

referenced by:

- [FunctionDefinition](#)
- [Program](#)

SetStatement:

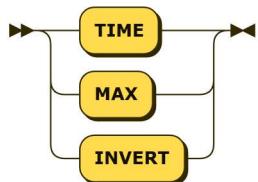
```

SetStatement
::= 'SET' SetParameter Value

```

referenced by:

- [Statement](#)

SetParameter:

```

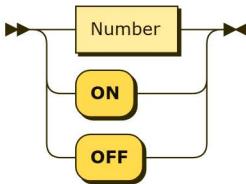
SetParameter
::= 'TIME'
| 'MAX'
| 'INVERT'

```

referenced by:

- [SetStatement](#)

Value:



Value ::= Number
| 'ON'
| 'OFF'

referenced by:

- [SetStatement](#)

ConstantDeclaration:

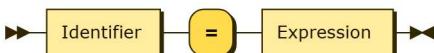


ConstantDeclaration ::= 'CONSTANT' Identifier '=' Expression

referenced by:

- [Statement](#)

Assignment:



Assignment ::= Identifier '=' Expression

referenced by:

- [Statement](#)

OutputStatement:

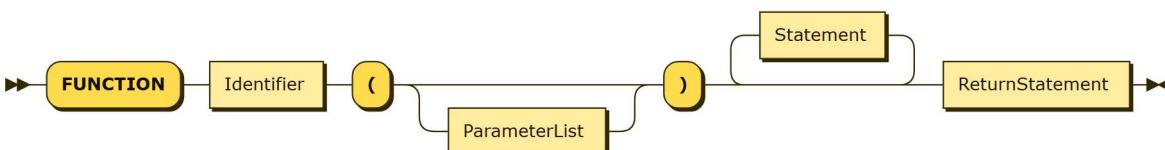


OutputStatement ::= 'OUTPUT' '(' ArgumentList ')'

referenced by:

- [Statement](#)

FunctionDefinition:



FunctionDefinition ::= 'FUNCTION' Identifier '(' ParameterList? ')' Statement* ReturnStatement

referenced by:

- [Statement](#)

ReturnStatement:

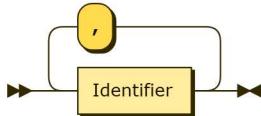


```
ReturnStatement
  ::= 'RETURN' '(' Identifier ')'
```

referenced by:

- [FunctionDefinition](#)

ParameterList:

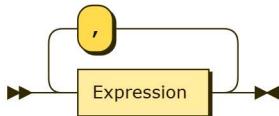


```
ParameterList
  ::= Identifier ( ',' Identifier )*
```

referenced by:

- [FunctionDefinition](#)

ArgumentList:

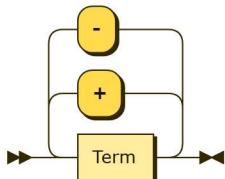


```
ArgumentList
  ::= Expression ( ',' Expression )*
```

referenced by:

- [FunctionCall](#)
- [OutputStatement](#)

Expression:

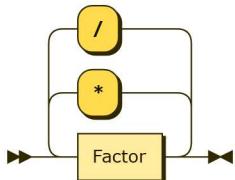


```
Expression
  ::= Term ( ( '+' | '-' ) Term )*
```

referenced by:

- [ArgumentList](#)
- [Assignment](#)
- [ConstantDeclaration](#)
- [Factor](#)

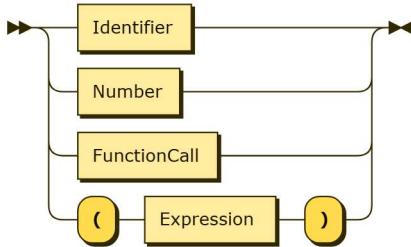
Term:



```
Term      ::= Factor ( ( '*' | '/' ) Factor )*
```

referenced by:

- [Expression](#)

Factor:

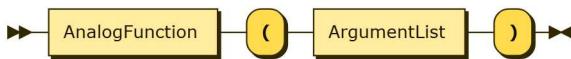
```

Factor ::= Identifier
| Number
| FunctionCall
| '(' Expression ')'

```

referenced by:

- [Term](#)

FunctionCall:

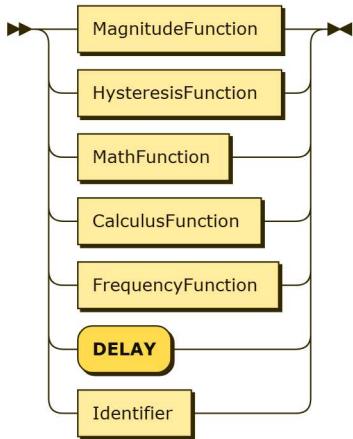
```

FunctionCall
::= AnalogFunction '(' ArgumentList ')'

```

referenced by:

- [Factor](#)

AnalogFunction:

```

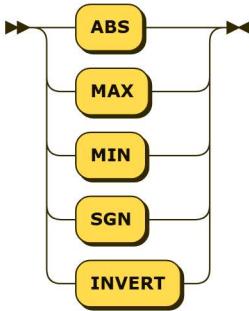
AnalogFunction
::= MagnitudeFunction
| HysteresisFunction
| MathFunction
| CalculusFunction
| FrequencyFunction
| 'DELAY'
| Identifier

```

referenced by:

- [FunctionCall](#)

MagnitudeFunction:

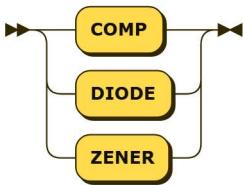


```
MagnitudeFunction
  ::= 'ABS'
  | 'MAX'
  | 'MIN'
  | 'SGN'
  | 'INVERT'
```

referenced by:

- [AnalogFunction](#)

HysteresisFunction:

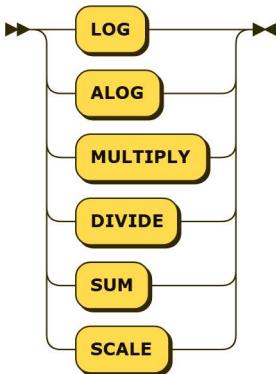


```
HysteresisFunction
  ::= 'COMP'
  | 'DIODE'
  | 'ZENER'
```

referenced by:

- [AnalogFunction](#)

MathFunction:

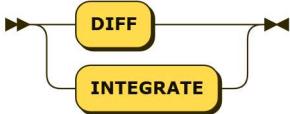


```
MathFunction
  ::= 'LOG'
  | 'ALOG'
  | 'MULTIPLY'
  | 'DIVIDE'
  | 'SUM'
  | 'SCALE'
```

referenced by:

- [AnalogFunction](#)

CalculusFunction:

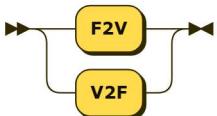


```
CalculusFunction
 ::= 'DIFF'
 | 'INTEGRATE'
```

referenced by:

- [AnalogFunction](#)

FrequencyFunction:

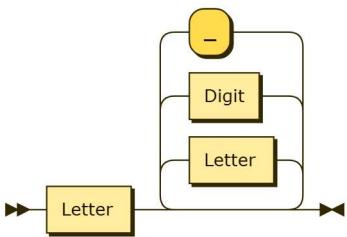


```
FrequencyFunction
 ::= 'F2V'
 | 'V2F'
```

referenced by:

- [AnalogFunction](#)

Identifier:

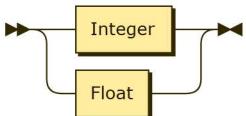


```
Identifier
 ::= Letter ( Letter | Digit | '_' )*
```

referenced by:

- [AnalogFunction](#)
- [Assignment](#)
- [ConstantDeclaration](#)
- [Factor](#)
- [FunctionDefinition](#)
- [ParameterList](#)
- [ReturnStatement](#)

Number:

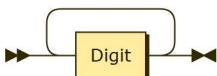


```
Number ::= Integer
 | Float
```

referenced by:

- [Factor](#)
- [Value](#)

Integer:

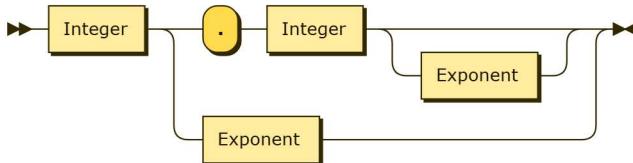


```
Integer ::= Digit+
```

referenced by:

- [Exponent](#)
- [Float](#)
- [Number](#)

Float:

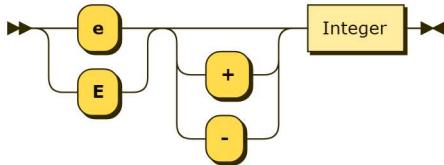


```
Float ::= Integer ('.' Integer Exponent? | Exponent)
```

referenced by:

- [Number](#)

Exponent:



```
Exponent ::= ('e' | 'E')? (+ | -)? Integer
```

referenced by:

- [Float](#)

Digit:

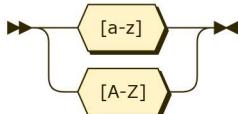


```
Digit ::= [0-9]
```

referenced by:

- [Identifier](#)
- [Integer](#)

Letter:



```
Letter ::= [a-zA-Z]
```

referenced by:

- [Identifier](#)