IEEE Transactions on Power Delivery, Volume 3, No. 3, July 1988

PROGRAMS FOR THE STUDY OF HVDC SYSTEMS

D.G. Chapman, Member J.B. Davies
Manitoba HVDC Research Centre
Winnipeg, Manitoba

F.L. Alvarado, Senior Member R.H. Lasseter, Member
University of Wisconsin
Madison, Wisconsin

S. Lefebvre, Member P.C.S. Krishnayya, Senior Member
IREQ
Varennes, Quebec

J. Reeve, Fellow
John Reeve Consultants, Ltd.
Waterloo, Ontario

N.J. Balu, Senior Member
EPRI
Palo Alto, California

Abstract - HVdc technology has reached a point where multiterminal HVdc systems are being planned and installed. Planning and operating studies for systems of this type will become increasingly important but most of the tools available for their study are somewhat restricted in capability. This paper reports on load flow and stability programs which provide the capability for modeling essentially any HVdc system in service or being planned, without compromise in the models used.

INTRODUCTION

A recent research project was concerned with the development of a set of load flow and transient stability programs incorporating new, flexible HVdc network and control models.

There are several load flow/transient stability program combinations available commercially. Many, but not all, provide a limited, fixed representation of HVdc systems. The more powerful programs permit flexibility in dc network and/or dc control representations.

The programs described in this paper provide full flexibility in the representation of HVdc networks, permitting arbitrary connection of lines and of HVdc converters and extending to the representation of bipolar systems, of systems involving remote grounds or metallic return connections and of interconnected converters with independently specified characteristics.

This same philosophy of full flexibility has been applied to the steady state control characteristics and to the control models available for stability studies. This is necessary since control models are not standardized.

THE LOAD FLOW

In the load flow program it is possible to define HVdc systems in a manner analogous to ac system models. This allows the representation of HVdc systems which might involve both series and shunt connected converters or arbitrary transmission network configurations. The program can represent more than one independent HVdc system being restricted only by the total number of converters and lines represented in all dc networks - a limit established by program parameters.

The basic building blocks for network modeling are the converter and the line. A converter is a device having two HVdc connections and one ac connection, at the primary terminal of the converter transformer. The converter model contains a converter transformer representation. The dc connections are to dc buses while the ac connection is to the ac commutating bus. The converter description is shown in Figure 1.

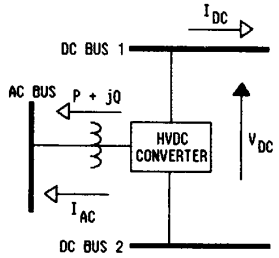

Figure 1: Converter Configuration

The HVdc line model also has two dc connections, again to dc buses.

HVdc systems are built up by describing the connections of the converters and lines through the use of appropriate dc bus names. The converter always has the voltage of the first dc connection positive with respect to the voltage of the second dc connection. One bus name, GROUND, is reserved for the zero voltage bus and this bus must appear at least once in each dc network.

The chosen approach permits the modeling of bipolar systems in a very simple manner. The positive pole is set up with converters having GROUND as the second dc connection while the negative pole is established through converters having GROUND as the first dc connection. Series groups or taps are established when neither dc connection is GROUND. A metallic neutral return connection may be represented by a dc line. Figure 2 represents a fictional but feasible HVdc system representation.

For full flexibility in the network topology, circuit identifiers are available to permit connection

87 SM 549-9 A paper recommended and approved by the IEEE Transmission and Distribution Committee of the IEEE Power Engineering Society for presentation at the IEEE/PES 1987 Summer Meeting, San Francisco, California, July 12 - 17, 1987. Manuscript submitted August 27, 1986; made available for printing April 17, 1987.

0885-8977/88/0700-1182$01.00©1988 IEEE

of, for example, parallel converters on the same ac bus and parallel dc lines.

Lastly, open lines and/or converters may be represented although the need for these elements is minimal, at least for load flow solutions.

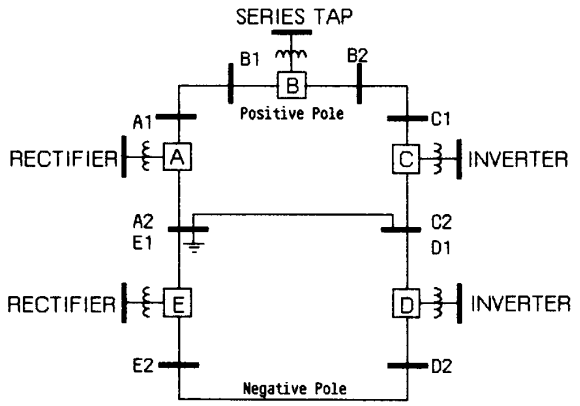

Figure 2: Example HVdc System

Once a network is established it is then necessary to establish an operating point for the system. In the load flow solution being discussed there is essentially full flexibility in the description of the operating point. For each converter two operating setpoints, chosen from among dc voltage, dc current, converter real power, converter reactive power, converter firing order alpha, converter extinction angle gamma and converter transformer tap position must be specified. In practice, the combinations of voltage and current, voltage and power, and current and power are not permitted and the use of both alpha and gamma at the same converter is discouraged. In most HVdc load flow solutions there will be at least one converter set with fixed voltage and one with fixed current (or power).

The solution may be based on the converter operating points alone, or may include constraints, in the form of firing and extinction angle limits, transformer tap limits and current limits. The transformer taps can be constrained to within maximum and minimum limits and to discrete tap positions.

A provision within the program permits the user to invoke a user written FORTRAN routine (known as a Central Scheduler) to apply elaborate voltage, current and other constraints. This powerful feature is required, in part, because of the impossibility of describing a general algorithm to apply such constraints. These routines can have essentially any dc or ac network quantity as an input and can affect any normal control point in the ac and dc systems.

Some general algorithms are possible for shunt connected HVdc systems, and a current limiting algorithm has been supplied. Similarly, for two terminal systems, a voltage compounding algorithm has been provided.

The load flow solution recognizes the contribution made to area interchange over the HVdc networks but does not alter power flow on the dc system to ensure that the area interchange limits are met. A Central Scheduler routine could, however, be written to perform this function.

The dc solution is performed sequentially with the ac solution. After each ac iteration, the dc system is solved to convergence. Two sets of dc convergence criteria are provided, a less precise set being used until the ac system has converged to within 10 times the final ac tolerance. Central Scheduler action is inhibited until the dc solution has reached the final tolerance stage.

The dc solution chosen is based on the method described by Reeve, et al [1], with some modifications to suit the overall formulation. The dc equations used are shown in Appendix A. At present these equations form the basis for a Newton solution using a full matrix representation of the Jacobian. Future work will involve the use of sparse matrix techniques for the solution. The Jacobian is non-symmetric with respect to structure. The solution involves both the converter and the network equations. The Jacobian structure is shown in Figure 3.

	α	γ	Ia	a	Ø	Qa	Pa	Vd	Id	VD	ID	
F_α	X	X		X				X				m
F_γ	X	X		X					X			m
F_I			X	X					X			m
F_V				X	X			X				m
F_Q					X	X		X	X			m
F_P							X	X	X			m
F_v								$X^{(1)}$		A		m
F_i									$X^{(2)}$		I	N
F_G										−Y	I	N

(1) This is I if no V_d is specified

(2) This is A^T if no I_d is specified

Figure 3: Load Flow Jacobian

In Figure 3, m and N are the number of converters and nodes, respectively. The row and column headings are as defined in Appendix A. X refers to the presence of a non-zero term, Y is the system admittance matrix and A is the incidence matrix.

Convergence and execution speed are the two main issues concerning the load flow solution. Convergence of the dc and converter equations alone, with fixed ac voltages, is by itself very good because of the choice of Newton's method. However, the presence of an ac system (particularly a weak one) makes the ac voltage highly sensitive to variations on the dc converters. This can easily lead to a failure of convergence of the entire process. The failure to converge is almost invariably the result of the ac system attempting to solve for an unreasonable dc condition. Convergence is greatly improved (at a slight expense in computation time) by insuring that a reasonable dc solution information is attained prior to the first dc solution attempt.

The dc load flow solution information is provided in dc only solution reports and is incorporated into the ac solution report. For difficult cases a full convergence report is available.

To simplify the output reports, it is possible to

group converters and lines and to obtain reports of, for example, the total real and reactive power flow into all converters in a group (useful if the group comprises converters connected to a single ac bus).

The new load flow program has been written in a highly modular fashion, with the dc solution taking ac bus voltage information in and returning real and reactive power flows at the converter terminals. Thus, while the code has been implemented in an EPRI program it is expected to be quite portable to other programs.

STABILITY PROGRAM

Given the highly flexible approach to HVDC modeling in the load flow program it was desirable to continue this approach in the stability program.

However, in a stability program speed of execution and freedom from convergence difficulties are very important. These objectives are achieved by:

- Limiting the direct modes of operation of converters to four: V_d, I_d, α and γ. Each of these four modes result in a linear Thevenin or Norton converter model for the dc solution.
- Representing the dc system as constant I - constant ϕ (power factor) current injections during the ac solution. This also makes the ac solution free from convengence problems. The constant I - constant ϕ model is a reasonable approximation to all four control modes, although far from perfect. Appendix F of [2] justifies this choice of model in more detail.

The stability program takes the network configuration and data from a load flow history case and augments this representation through the provision of three line models. Any dc line may be represented by its resistance, its resistance and inductance, or by a π equivalent incorporating resistance, inductance and shunt capacitance. The line model is adequate for transients slower than 1 ms and it neglects line coupling. While capacitance in dc line models is desirable for such studies as those considering dc breaker performance numerical stability is not assured.

The problem of numerical oscillations has been handled through the use of a "damped trapezoidal" integration process [3], that removes high frequency numerical oscillations while permitting the model response to approximate the response of the modeled system. The integration method permits the user to specify the amount of damping used and it permits a longer time step than might normally be used.

The integration technique is incorporated into the solution of the inductance response, the capacitance response and the control system response, with different damping factors available for each integration process.

User Defined Controls

The need for flexibility in the description of control models has been recognized by others [4]. The new stability program provides a very complete and flexible model development system known as User Defined Controls (UDC) where the dc control models are fully described by input data alone. The UDC system handles the ordering, initialization and solution of arbitrary control models.

The UDC system permits a program user to enter control system descriptions which suit the needs of a specific study, match known control system information and permits the study of advanced control concepts.

The UDC system consists of a set of building blocks, listed in Appendix B, such as sources, measurement blocks for ac and dc system quantities, signal processing, dynamic transfer functions, logic functions and converter control blocks. As a minimum each converter must be provided with an INTERFace block which controls the operation of that specific converter.

Each converter can have the dc voltage, current, alpha or gamma controlled by the UDC system. Additionally, the converter can be placed into a BLOCKed mode (current in the converter is zero) or a BYPASSed or COMFAILed mode (voltage on the converter is zero, current passes through a bypass device).

The source for the value of the controlled variable may come from a model as simple as a constant block or from a fully detailed controller model. Through the use of logic functions, the source and nature of that signal may be varied during the stability run.

In the development of the UDC system means were included to simplify the modeling process:

(a) A free format input processor was developed which permits the user to define the control blocks in a simple and readable format. Comments may be imbedded in the data. The ordering of the data for each block is important but the layout may be modified to permit the user to read the data easily. For example, the parameters for the general Laplace transfer function description (an SBLOCK) may be written on two lines, to permit the numerator terms to be placed over the denominator terms, for a natural transfer function description.

Several of the blocks may have multiple inputs. The UDC system, for blocks such as those of type MULTIP, permits an unlimited number of inputs.

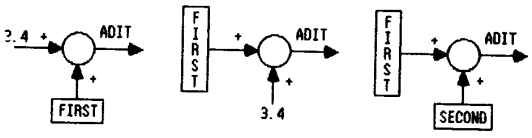

Figure 4: Representations of Addition of Two Signals

In control systems, control parameters such as time constants, are fixed values. The UDC system requires real numbers for such parameters. However, in control block diagrams, constants might be added to variables. The UDC system, in the case of block input descriptions and several other descriptions, will accept real numbers or the names of other blocks as legitimate inputs. Thus, the block ADIT, adding 3.4 to the output of block FIRST, could be described as:

```
ADIT, BINARY, PLUS, P, 3.4, FIRST:
```

or

```
ADIT, BINARY, PLUS, P, FIRST, 3.4:
```

or another block SECOND, having the value of 3.4 could be defined and then

```
ADIT, BINARY, PLUS, P, FIRST, SECOND:
```

could be used. In these examples P is an output print enabling flag.

Figure 4 shows these various combinations.

(b) Control systems models must be initialized if the dynamic run is to proceed without an initial transient. The initialization is preceded by an ordering process to ensure that the control system is solved correctly with a minimum of time delays.

The solution process chosen is sequential, implying that integration is performed when a block involving integration is encountered in the solution. Integration is not performed simultaneously for all integrators. However, within each block of type SBLOCK integration is performed simultaneously.

Sequential solution of the blocks provides for handling of limits within the models without need for iteration (as would be required with implicit integration).

Solution of the output of each block proceeds from known values to unknown values. If a loop occurs in the control model, such that the input to a block depends on the output value of the same block, the feedback value will be the value calculated at the previous time step.

An ordering process [5] is provided to ensure that the solution will proceed from known values to unknown values. Algebraic loops are recognized, as described above, and the program has the capability to detect the blocks which make up the loop (even nested loops and loops which intersect) and to order the blocks correctly.

The initialization process takes place after the ordering of the blocks has been completed. When a loop is encountered the program iterates the block variables towards steady state. The algorithm used is simple, the feedback value being calculated as the average of the output of the block providing the feedback and the previous feedback value. The iteration stops when the correction is less than a user defined tolerance.

An integrator provides a break in the forward path for control signals during initialization, the output bearing no relationship to the input. Consequently, it should be apparent that initialization beyond integrators would require the user to provide additional information to the control system structure. It was also recognized that non-linearities and general complexities of control systems might prevent a user from being able to predefine reference settings, whether algebraically or numerically, within the control system. An automatic means to establish integrator output values and reference settings has been provided through a concept known as subsystems.

A subsystem is a portion of a control system model which starts at either a reference setting or an integrator and feeds to either an INTERFace block or a BREAK block where the signal value is known.

The loop tracing logic of the program, during the ordering process, finds and records all blocks between the start and end of a subsystem. Then, during initialization, when a subsystem is encountered, the value of the output of the first block of the subsystem is modified, using a linear interpolation scheme, to make the value of the last block of the subsystem equal to the desired value (within a tolerance). The process is similar to the initialization process for a loop.

Once the control system model is initialized the identities of the loops and subsystems have no further relevance and these identities are deleted.

To date the ordering and initialization algorithms have performed well, permitting initialization of the control models to match load flow steady-state data. Some enhancement to the ordering may be warranted. Occasionally, if the ordering of blocks in the input data is inappropriate, a loop or subsystem will be found but the first selected block in the loop or subsystem will inhibit the ordering process (it will be a block inside the loop, for the loop case). Specifically, blocks feeding into a loop may be inadvertently so defined in the input data order as to prevent proper loop tracing. For example, a block providing a limit within the loop might be selected as the start of the loop, rather than the correct starting block.

Ordering takes into account blocks within the signal path as well as blocks, such as the limits mentioned above, affecting the signal path.

(c) If it were mandatory to use the full stability program for debugging and testing a particular UDC model, the costs of model development would be high. To circumvent this problem a separate program containing the UDC and HVdc stability solution routines has been established. This "standalone" program can be initialized from a history file, if HVdc systems are being established, or from a "pseudo-history" if general control systems are being developed. The "pseudo-history" provides initial values for blocks which otherwise (in a stability run) require data from the ac and dc systems and also provide ac commutating voltages for the converter models.

After the control system is checked for correct data and structure, it can be exercised using information either from the dc solution or from curves describing the output of FROMAC and FROMDC blocks as functions of time. In any standalone solution ac bus voltages are held constant. Thus, the dc control systems can be tested on a infinitely strong ac system. This permits preliminary testing of the controls without any additional influence of a weak ac system.

The HVdc Network Solution

The dc network solution in the stability program is a direct solution for the network voltages given the network admittance matrix and the network current injectors. The converters have one of the four forms shown in Figure 5, where the current injection expression is a per unit expression. In the voltage controlled converter case, a transformation [6], shown in Figure 6 is applied to remove a node while adding new current injections. Based on the integration method, inductive and capacitive line elements are replaced by Norton equivalents. Thus, at each dc solution, there is an equivalent resistive network and a set of current sources, permitting solution for the node voltages. After solution the node voltages at the nodes removed for voltage sources are calculated and, then, the voltage, current α, γ, and real and reactive power flows for each converter are calculated.

The UDC solution, the dc network solution and the ac solution all proceed at their own specified time steps. When the dc network solution is performed, a running average of the real power injections is calculated within the ac time step. At the start of an ac solution this average power and the latest calculated reactive power are used to calculate the injections into the ac network. These injections are current injections. Within the ac time step the magnitude of the injections and the phase relationship of these injections to the ac voltages are maintained.

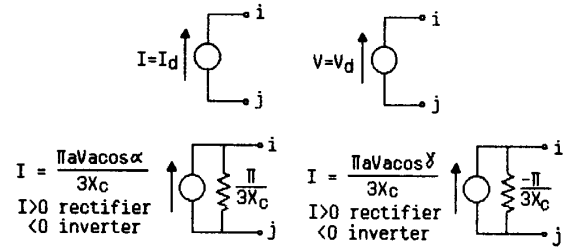

Figure 5: Converter Equivalents

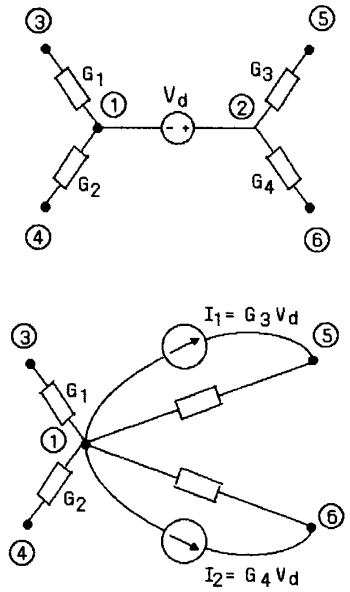

Figure 6: Transformation of Voltage Source

This model of the dc system during the ac solution represents a good approximation to the behavior of the dc system.

APPLICATIONS

To date the programs have been extensively tested by the developers, on a 4-terminal dc system in a 20 bus ac system and on a 1200 bus utility system incorporating both a 2 terminal bipolar dc system and a 4 terminal bipolar dc system.

The program has also been tested extensively by two test utilities [7], utilizing their own system data. The programs have proved to be useful in determining HVdc system ratings, control (steady state and transient) requirements, ac system compensation requirements and the general operating philosophy.

CONCLUSIONS

This paper has described a powerful and flexible approach to the modeling of HVdc networks and controls. The flexibility exists in the description of the network configuration, in the load flow operating point specification and in the description of controls for stability studies and provides the user with the ability to model dc systems to the extent of available information. Compromises resulting from the use of predefined models are avoided. The programs described have been used, successfully, in complex dc system studies.

ACKNOWLEDGEMENTS

The work described in this paper was supported in full by EPRI under RP1964-2. The authors wish to acknowledge assistance provided in the project by the advisors, E. Galachenski, G. Rogers, R. Lee, D. Smith, T. Hatcher, J. Luini and the assistance at the contractors of R. Burton, V. Burtnyk, D. Fletcher, A. Ogale, D. Allan, C. Hasselfield, D. Woodford, A. Heise and J. McNichol.

REFERENCES

1) J. Reeve, G. Fahmy, B. Stott, "Versatile Load Flow Method for Multiterminal HVdc Systems", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-96, No. 3, May/June 1977, pp 925 - 933.

2) "Methodology for Integration of Multiterminal HVDC Links in Large AC Systems - Phase II: Advanced Concepts", EPRI Final Report EL4365, 1986.

3) F.L. Alvarado, R.H. Lasseter, J.J. Sanchez, "Testing of Trapezoidal Integration with Damping for the Solution of Power Transient Problems", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-102, No. 12, December 1983, pp 3783 - 3790.

4) R. Proulx, A. Valette, J-P. Gingras, D. Soulier, "User-Oriented Simulation of HVdc Control in a Transient Stability Program", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-104, No. 7, July 1985. pp 1609 - 1616.

5) D.G. Chapman, A.J. Heise, F.L. Alvarado, N.J. Balu, "Ordering and Intialization Algorithms for Control System Models", submitted to 1987 IEEE PES Winter Power Meeting.

6) F.L. Alvarado, "Formation of Y-Node Using the Primitive Y-Node Concept", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-101, No. 12, December 1982, pp 4563-4571.

7) D.G. Chapman, J.B. Davies, E.M. Gulachenski, S. Doe, J.R. McNichol, N. Balu, "Test Experience with Multiterminal HVDC Load Flow and Stability Programs," Paper 87-WM139-9 presented at the 1987 IEEE PES Winter Power Meeting.

APPENDIX A

DC EQUATIONS

The following symbols are used in this appendix and Figure 3.

m - number of converters
N - number of dc buses
V_d - converter voltage
V_D - node voltage
I_d - converter current
I_D - network current injection
V_a - ac bus voltage
I_a - ac current at primary of converter transformer
n - number of series bridges in a converter group
α - firing angle
γ - extinction angle
ϕ - phase angle between V_a, I_a

X_c - commutating reactance
P_a - converter real power
Q_a - converter reactive power
σ - sign term, +1 for inverter -1 for rectifier

Converter Equations

The converter variables are related according to the following equations, expressed in actual values:

Relationship between dc voltage, ac voltage and power factor:

$$F_V = V_d - (3\sqrt{6}/\pi)\, n\, a\, V_a \cos\phi = 0$$

Relationship between dc voltage, ac voltage and firing angles:

$$F\alpha = V_d + \sigma\,(3\sqrt{6}/(2\pi))\, n\, a\, V_a(\cos\alpha - \cos\gamma) = 0$$

Relationship between dc current, voltage and firing angles:

$$F_\gamma = I_d + \sigma\,(\sqrt{3}/(\sqrt{2}X_c))n\, a\, V_a(\cos\alpha+\cos\gamma) = 0$$

Active power relationship:

$$F_p = P_a + V_d\, I_d = 0$$

Reactive power relationship:

$$F_q = Q_a - \sigma\, V_d\, I_d \tan\phi = 0$$

Relationship between currents:

$$F_I = I_a + (\sqrt{6}/\pi)\, a\, I_d = 0$$

The Fourier relation of $\sqrt{6}/\pi$, while only exact for rectangular phase currents, has been judged to remain sufficiently accurate for less idealized waveforms.

Thus, 11 converter variables are mutually related by 6 equations which are independent of θ, the voltage phase angle. The number of equations and variables can be reduced if desired, but in the interests of flexibility, it is not recommended to do so.

The AC Network Equations

The ac network can be represented during the dc solution either as a constant voltage source:

$$F_\theta = V_a - V_{a_{given}} = 0$$

Or it can be represented as a Thevenin equivalent source where only the Q-V variation is of interest:

$$F_\theta = Q_a + V_a{}^2 BN + V_a\, V_{th}\,(BN\cos(n-\theta) + GN\sin(n-\theta)) = 0$$

In this equation V_{th}, BN, GN, n and θ are all parameters. Either of these equations can be used with identical final results. The only difference is in the eventual convergence characteristics.

The DC Network Equations

The dc network established additional relations between converter variables. These relations depend on the manner in which the converter is connected to the dc network (series or parallel). Thus, generality requires converter voltages V_d and currents I_d to be considered as branch quantities. The dc network nodal voltages can be designated as V_D and the dc nodal currents can be designated as I_D. The vectors of converter voltages and nodal voltages and converter currents and nodal currents are interrelated according to a branch-node incidence matrix with elements of 1, -1 and 0:

$$F_V - V_d = [A]\, V_D = 0$$

$$F_i = I_D - [A]^t\, I_d = 0$$

where:

a_{ij}=1 if + side of converter i is connected to node j
-1 if - side of converter i is connected to node j
0 otherwise

The F_i and F_V equations establish 1 equation per converter and 1 equation per node, but introduce 2N additional unknowns. However, the nodal currents I_D and nodal voltages V_D are related according to the conductance matrix of the dc network itself:

$$F_G = I_D - [Y]\, V_D = 0$$

APPENDIX B

UDC BLOCKS

Type	Subtype	
FROMDC	VCONV	Converter dc voltage (kV)
	VCOMM	Converter Vdo (kV)
	ICONV	Converter Current (A)
	ALPHA	Converter α (°)
	GAMMA	Converter γ (°)
	PAC	Converter real power (MW)
	QAC	Converter reactive power (MVAr)
	BRIDGE	Number of series bridges in converter
	XC	Converter commutating reactance (Ω)
	TAP	Converter transformer tap position (pu)
	IMODE	Converter load flow operating mode
	RMODE	Converter operating mode
	COMFAIL	True, if converter is failing commutation
	BLOCK	True, if converter is blocked
	VNODE	DC bus voltage (kV)
	PLINE	DC line real power (MW)
	ILINE	DC line current (A)
FROMAC	VMAG	AC bus voltage magnitude (pu)
	VANG	AC bus voltage angle (°)
	FREQ	AC bus frequency (Hz)
	ILINE	AC line current magnitude (A)
	PLINE	AC line real power flow (MW)
	QLINE	AC line reactive flow (MVAr)
SOURCE	DC	Step function
	RAMP	Ramp function
	SIN	Sinusoidal source
	TIME	Simulation time
	DTAC	AC solution time step
	DTDC	DC solution time step
	DTCC	UDC solution time step
UNARY	SIN	Sine of input angle in degrees
	COS	Cosine of input angle in degrees
	TAN	Tangent of input angle in degrees
	SINH	Hyperbolic sine
	COSH	Hyperbolic cosine
	TANH	Hyperbolic tangent
	ABS	Absolute value
	EXP	e^x, x is input
	LOG	ln x, x is input
	INV	1/x, x is input
	UDNONL	Linear interpolation of arbitrary curve
BINARY	PLUS	$x_1 + x_2$
	MINUS	$x_1 - x_2$

	MULT	$x_1 * x_2$
	DIV	x_1 / x_2
	POWER	$x_1^{x_2}$, $x_1 \geq 0$
MULTIP	SUM	$\sum x_i$
	SUMMER	$\sum K_i.x_i$ K_i constants
	MIN	minimum $\{x_i\}$
	MAX	maximum $\{x_i\}$
LIMIT	STATIC	Constrain input within limits
	DYNAMIC	Saturating lead-lag function
SBLOCK	NORMAL	Polynominal transfer function - up to 10th order.
	LIMIT	Saturating polynominal transfer function - to 10th order.
OTHER	BRIDGE	Change number of series bridges in converter.
	DELAY	Communications delay
	DIGIT	Signal quantizer
	LINEDC	Change dc line parameters when logic true.
	RLC	Set DC line parameters before initialization.
	TAP	Change converter transformer tap when logic true.
	LEVEL	Signal ramps from level 1 to level 2, holds, then resets.
	HYSTER	Hysteresis function
	DEADB1	3 level deadband.
	DEADB2	Deadband with linear output
LOGIC	AND	Logical AND of inputs, $x_1 \ldots x_n$
	COUNTER	Record number of times input goes true.
	EXPRESS	Convert logical expression to logic quantity
	FLIPFLOP	Set - reset flip flop
	NOT	Negate input
	OR	Logical OR of inputs $X_1 \ldots X_n$
	SWITCH	Logic controlled switch, 1 of 2 inputs selected
	TIMER	Logic controlled timer.
	XOR	Logical exclusive OR of inputs $x_1 \ldots x_n$.
SETUP and SUBSYS	VALUE	Fixed value evaluated at initialization
	OFFSET	Signal plus fixed value.
	INTEG	Integrator with gain
	LINTEG	Saturating integrator with gain
	START	(SETUP only) - give value to block atinitialization
	BREAK	(SUBSYS only) - end of subsystem
INTERF		Interface to control converter from UDC system.