

Switching Converter Analysis Tool

SCAT K.492
For Windows

Copyright (C) by *M.Nakahara*, 23, Apr, 2010

1. SCAT K492 の概要	15
2. 回路の作成.....	19
2. 1 K492 の起動と終了 (File:Exit)	19
2. 2 回路データの読み込み (File:Open)	19
2. 3 回路データの保存.....	20
(1) 回路データの保存 (File:Save)	20
(2) スコープデータの保存 (File:Save Data:WVF, FFT, FRA, CMP)	20
(3) 部品データの保存 (File:Save)	20
2. 4 回路の作成 (File:New)	21
2. 5 図形の整形と変更.....	22
(1) 図形を選択する (Sselect All)	22
(2) 図形を移動する	22
(3) 図形の大きさをかえる (Zoom in, Zoom out)	22
(4) 図形を回転させる (Rotate)	22
(5) 図形を反転させる (MirrorH, MirrorV)	23
(6) 図形を消去する (Delete)	23
(7) 消去した図形をもとに戻す (Undelete)	23
(8) 図形を複製する (Copy)	23
(9) 図形のシンボルを表示する (Symbol)	23
(10) 素子の値を表示する (Value, Value Mode, Value Unit)	24
(11) 図形の色を変える (Color)	24
(12) 図形をグループ化する (Group)	24
(13) 図形のグループ化を解除する (Ungroup)	24
(14) 素子極性の表示, 非表示を行う (Polarity)	25

(15)	図形を活性化する (Activate)	25
(16)	図形を不活性化する (Deactivate)	25
(17)	図形の線の太さを変える (Line Width)	25
(18)	文字のフォントを変える (Font)	25
(19)	文字の大きさを変える (Font Size)	25
(20)	素子をカスタム化する (Customize)	26
(21)	デフォルト値を変える (View:Option:Default, View:Initialize)	26
(22)	図形全体を定位置に移動する (View:Home)	27
(23)	図形全体を拡大する (View:Zoom in)	27
(24)	図形全体を縮小する (View:Zoom out)	27
(25)	表示色を元の色にもどす (View:Refresh)	27
(26)	ノードの表示と消去 (View:Node:Create, Delete)	27
(27)	画面のクリップボードへのコピー (Window:Copy)	27
2. 6	カスタム回路	28
(1)	カスタム回路の作成	28
(2)	カスタム回路の変更	30
(3)	カスタム回路におけるシンボル	31
(4)	カスタム回路データファイル	31
3.	素子モデルとパラメータの設定	33
3. 1	素子モデル	34
(1)	インダクタ (LCR:Inductor, Inductor-R)	34
(2)	可飽和インダクタ (LCR:Sat Inductor)	35
(3)	キャパシタ (LCR:Capacitor, Capacitor-RL)	35
(4)	抵抗 (LCR:Resistor)	36
(5)	直流電圧源 (Source:DC Voltage)	36
(6)	交流電圧源 (Source:AC Voltage)	36
(7)	交流三相電圧源 (Source:3Phase Voltage)	37

(8) プログラマブル電圧源 (Source:Prg Voltage)	37
(9) テーブル電圧源 (Source:Tbl Voltage)	38
(1 0) 直流電流源 (Source:DC Current)	39
(1 1) 交流電流源 (Source:AC Current)	39
(1 2) スイッチ (Switch:Switch)	39
(1 3) PWMスイッチ (Switch:PWM Switch)	40
(1 4) NチャンネルMOSFET (Switch:Nch MOSFET)	40
(1 5) PチャンネルMOSFET (Switch:Pch MOSFET)	41
(1 6) NPNトランジスタ (Switch:NPN Transistor)	41
(1 7) PNPトランジスタ (Switch:PNP Transistor)	42
(1 8) ダイオード (Diode:Diode)	42
(1 9) ダイオードブリッジ (Diode:Diode Bridge)	43
(2 0) ツェナーダイオード (Diode:Zener)	43
(2 1) サイリスタ (Diode:Thyristor)	43
(2 2) ショントレギュレータ (Diode:Shunt Regulator)	44
(2 3) フォトカプラ (Diode:Photo Coupler)	44
(2 4) 変圧器 (Transformer:Winding, Winding-R, Core, Make)	45
(2 5) 三相変圧器 (Transformer:3Phase Y-Y, 3Phase Y-D, 3Phase D-D)	46
(2 6) センサー (Sensor:V Sensor, I Sensor, VI Sensor, IV Sensor)	46
(2 7) パルス発生器 (Generator:Pulse)	47
(2 8) のこぎり波発生器 (Generator:Saw Tooth)	47
(2 9) 波形発生器 (Generator:Function)	48
(3 0) 周波数特性信号源 (Generator:AC Sweep)	49
(3 1) パルス変調器 (Modulator:PWM)	50
(3 2) 周波数変調器 (Modulator:SFM)	50

(33)	素子パラメータ変調器 (Modulator:DPM)	51
(34)	同期制御器 (Modulator:SYC)	51
(35)	プログラマブル制御器 (Modulator:PRC)	51
(36)	スイッチング周波数発生器 (Modulator:FRD)	52
(37)	オペアンプ (OpAmp:OpAmp)	53
(38)	理想オペアンプ (OpAmp:Ideal OpAmp)	53
(39)	積分器 (OpAmp:Integrator)	53
(40)	微分器 (OpAmp:Differentiator)	54
(41)	ボルテージフォロワ (OpAmp:V Follower)	54
(42)	リミッタ (OpAmp:Limiter)	55
(43)	コンパレータ (Comparator:Comparator)	55
(44)	ラッチ (Comparator:Latch)	55
(45)	単安定マルチバイブレータ (Comparator:Vibrator)	56
(46)	トリガ (Comparator:Trigger)	56
(47)	算術演算器 (Operator:Arithmetic)	57
(48)	関数演算器 (Operator:Function)	58
(49)	プログラマブル演算器 (Operator:Program)	59
(50)	テーブル演算器 (Operator:Table)	59
(51)	三相二相変換器 (Operator:DQC)	61
(52)	二相三相変換器 (Operator:DQI)	62
(53)	正相逆相変換器 (Operator:DQP)	63
(54)	論理演算器 (Logic:Logic)	63
(55)	ブール演算器 (Logic:Bool)	64
(56)	D S P (DSP: DSP)	65
(57)	D S P 初期ブロック (DSP: Initial)	65
(58)	D S P 入力ブロック (DSP: Input)	65
(59)	D S P 出力ブロック (DSP: Output)	65
(60)	D S P 検出ブロック (DSP: Detector)	66

(6 1) DSP機能ブロック (DSP: Block1, Block2, Block3, Block4)	66
(6 2) 接続線 (Connection:Wire, Half Wire)	66
(6 3) 自由接続線 (Connection:Free Wire)	67
(6 4) ジャンパー線 (Connection:Jumper)	67
(6 5) 遠ジャンパー線 (Connection:Far Jumper)	67
(6 6) グランド端子 (Connection:Ground)	68
(6 7) グランド端子2 (Connection:Ground2)	68
(6 8) オープン素子 (Connection:Open)	68
(6 9) ターミネータ (Connection:Terminator)	68
(7 0) カスタム端子 (Cmt Terminal)	69
(7 1) テキスト (Others:Text)	69
3. 2 素子パラメータの設定 (Parameter:Device)	69
3. 3 スイッチング周波数の設定 (Parameter:Frequency)	70
3. 4 出力変数の設定 (Parameter:Output)	70
(1) シンボル	71
(2) タイプ	71
(3) モード	71
3. 5 解析条件の設定 (Parameter:Condition)	73
(1) Nta	73
(2) Ntw	73
(3) Fta	73
(4) Compensate	74
4. 表示ウィンドウ.....	75
4. 1 メインウィンドウ (Window:Analysis, Window:Topology)	75
4. 2 状態ウィンドウ (Window:Window:State)	76
4. 3 W V F ウィンドウ, コントロールパネル (Window:Window:WVF)	78

(1) 出力変数リスト (Outputs)	79
(2) 出力変数データリスト	80
(3) X軸変数データリスト (X-Axis)	80
(4) DC/AC ボタン (DC/AC)	80
(5) Positive/Negative ボタン (P/N)	81
(6) Widen ボタン (Wdn)	81
(7) Narrow ボタン (Nrw)	81
(8) X軸 Adjust ボタン (AdX)	81
(9) Y軸 Adjust ボタン (AdY)	82
(10) F F T解析ボタン (FFT)	82
(11) CMP ウィンドウボタン (CMP)	82
(12) Display ボタン (Dsp)	82
(13) 高調波解析ボタン (HA)	82
(14) 波形演算ボタン (+, -, *, /)	83
(15) 縦スクロールバー	83
(16) マーカー表示ボタン (Mkr)	83
(17) マーカーセットボタン (Set)	83
(18) ピークボタン (Pk)	84
(19) ズームボタン (Zm)	84
(20) 横スクロールバー	84
(21) 表示線オプション (View:Option:Scope)	85
4.4 F F T ウィンドウ, コントロールパネル (Window:Window:FFT)	85
(1) 出力変数リスト (Output)	86
(2) 出力変数データリスト	86
(3) 周波数データリスト (Frequency)	87
(4) X軸 Widen ボタン (WdX)	87
(5) X軸 Narrow ボタン (NrX)	87

(6) Y軸 Widen ボタン (WdY)	87
(7) Y軸 Narrow ボタン (NrY)	87
(8) 振幅位相切り替えボタン(Amp/Phs)	87
(9) X軸 Adjust ボタン (AdX)	87
(10) Y軸 Adjust ボタン (AdY)	88
(11) 縦スクロールバー	88
(12) マーカー表示ボタン (Mkr)	88
(13) マーカーセットボタン (Set)	88
(14) ピークボタン (Pk)	89
(15) ズームボタン (Zm)	89
(16) 横スクロールバー	89
(17) 表示線オプション(Veiw:Option:Scope)	89
4.5 F R A ウィンドウ, コントロールパネル	
(Window:Window:FRA)	89
(1) 出力変数リスト (Output)	90
(2) 出力変数データリスト	90
(3) 周波数データリスト (Frequency)	90
(4) Y軸 Widen ボタン (Wdn)	91
(5) Y軸 Narrow ボタン (NrW)	91
(6) X軸 Adjust ボタン (AdX)	91
(7) Y軸 Adjust ボタン (AdY)	91
(8) 利得余裕ボタン (GaM)	91
(9) 位相余裕ボタン (PhM)	91
(10) 縦スクロールバー	92
(11) マーカー表示ボタン (Mkr)	92
(12) マーカーセットボタン (Set)	92
(13) ピークボタン (Pk)	92
(14) ズームボタン (Zm)	92

(15) 横スクロールバー	93
(16) 表示線オプション(Veiw:Option:Scope)	93
4.6 CMP ウィンドウ, コントロールパネル	
(Window:Window:CMP)	93
(1) 出力変数リスト (Output)	94
(2) 出力変数データリスト	94
(3) X軸変数データリスト (X-Axis)	94
(4) DC/AC ボタン (DC/AC)	94
(5) Positive/Negative ボタン (P/N)	95
(6) Widen ボタン (Wdn)	95
(7) Narrow ボタン (Nrw)	95
(8) X軸 Adjust ボタン (AdX)	95
(9) Y軸 Adjust ボタン (AdY)	95
(10) 出力調整ボタン(Fit)	96
(11) 変数名変更ボタン(Ren)	96
(12) Display ボタン (Dsp)	96
(13) 消去ボタン(Del)	96
(14) 縦スクロールバー	96
(15) マーカー表示ボタン (Mkr)	96
(16) マーカーセットボタン (Set)	97
(17) ピークボタン (Pk)	97
(18) ズームボタン (Zm)	97
(19) 横スクロールバー	98
(20) 表示線オプション(Veiw:Option:Scope)	98
4.7 カスタムウィンドウ	98
(1) カスタム回路の読み込み (File:Read)	98
(2) カスタム回路の保存 (File:Save)	98
(3) シンボル変更 (Function:Symbol)	98

(4) クリップボードへのコピー (Function:Copy)	98
(5) 表示設定 (Function:Initialize View)	98
4. 8 DSPウィンドウ	99
(1) ブロック図の読み込み (Read)	99
(2) ブロック図の保存 (File:Save)	99
(3) シンボル変更 (File:Symbol)	99
(4) 全体プログラムの表示 (Function:Program)	99
(5) ブロックの実行順序 (Function:Block Order)	99
(6) DSPアクティブ化 (Function:Active)	99
(7) クリップボードへのコピー (Function:Copy)	99
(8) 表示設定 (Function:Initialize View)	99
4. 9 ネットリスト (Window:Information:Net List)	99
4. 10 メモ (Window:Information:Memo)	100
4. 11 バージョン表示 (Window:Information>About)	100
4. 12 ウィンドウのクリップボードへのコピー (Window:Copy)	100
4. 13 プリント出力 (File:Print)	100
5. シミュレーション機能	102
5. 1 過渡応答解析 (Analysis:Transient)	102
5. 2 波形解析 (Analysis:Waveform)	102
5. 3 ステップ応答解析 (Analysis:Step:Transient, Waveform)	103
5. 4 定常解析 (Analysis:Steady)	104
(1) DC-DCコンバータの定常解析	104
(2) その他のコンバータの定常解析	104
5. 5 スweep解析 (Analysis:Sweep:Sweep, Transient, Waveform, Steady)	104
5. 6 入力電流の高調波解析	106
(1) 整数倍高調波のみの高調波解析	106
(2) 次数間高調波を含む高調波解析	107

5. 7	継続解析, 再解析, 解析中止 (Analysis:Continue, Parameter:Update, Analysis:Quit)	108
5. 8	周波数特性解析 (Analysis:Frequency:Fast, Slow)	109
	(1) 一巡周波数特性	109
	(2) 部分回路周波数特性	112
	(3) オープンループ周波数特性	112
	(4) 周波数特性解析の注意点	113
5. 9	スクリプト解析 (Analysis:Script, Window:Script:New, Edit)	114
6.	制御方法	116
6. 1	制御素子	116
	(1) のこぎり波発生器 (Saw Tooth)	116
	(2) コンパレータ素子 (Comparator)	116
	(3) オペアンプ素子 (OpAmp)	116
	(4) パルス発生器 (Pulse)	116
	(5) PWMスイッチ (PWM Switch)	116
	(6) パルス変調器 (PWM)	117
	(7) 周波数変調器 (SFM)	117
	(8) 素子パラメータ変調器 (DPM)	117
	(9) 同期制御器 (SYC)	117
	(10) 演算素子 (Operator, Logic)	117
	(11) プログラマブル素子	118
	(12) DSPとDSPブロック素子	118
6. 2	低周波近似	118
6. 3	一般的なPWM制御	118
6. 4	PWM素子を用いたPWM制御	119
	(1) PWM時比率制御	119
	(2) PWM時刻制御	122

(3) 同期PWM制御 (デッドタイム制御)	122
6. 5 スイッチング周波数制御	124
(1) スイッチング周波数制御	124
(2) スイッチング周期制御	125
(3) 時比率一定制御	126
(4) 時比率一定制御と同期制御	126
6. 6 ピーク電流制御	127
(1) ピーク電流制御	127
(2) PWMピーク電流制御	128
6. 7 素子パラメータ制御	128
6. 8 プログラム制御	129
(1) プログラマブル演算器 (Operator:Program)	129
(2) プログラマブル電圧源 (Source:Prg Voltage)	129
(3) プログラマブル制御器 (Modulator:PRC)	130
6. 9 DSP制御	130
(1) メイン回路の作成	131
(2) ブロック図の作成	131
(3) ブロックプログラムの読み込み	132
(4) ブロック図の保存と読み込み	135
(5) 出力変数の設定	136
(6) 全体プログラムの確認	136
(7) ブロックプログラムの作成法	136
6. 10 太陽電池最大電力点追跡 (MPPT) 制御	139
(1) 太陽電池のモデル化	140
(2) コンバータ回路の作成	141
(3) MPPT制御部の作成	142
(4) 太陽電池特性やMPPTアルゴリズムの変更	143
6. 11 制御順序	143

7. アルゴリズムの概要.....	145
7. 1 トポロジー条件.....	145
7. 2 解析アルゴリズム.....	145
(1) 区分的線形性.....	145
(2) 状態解析アルゴリズム.....	146
(3) トポロジー解析.....	147
(4) 過渡応答解析アルゴリズム.....	147
(5) 定常解析アルゴリズム.....	147
7. 3 解析誤差.....	148
8. エラーメッセージ.....	149
8. 1 解析エラーメッセージ.....	149
8. 2 入出力エラーメッセージ.....	149
8. 3 警告メッセージ.....	150
付録 A 高調波規制の概略.....	152
1. クラス.....	152
2. 周波数スペクトラム規制値.....	152
3. グループ.....	154
付録 B プログラマブル素子言語文法.....	155
付録 C スクリプト言語文法.....	159
付録 D SCATLIB ライブラリ.....	166
1. ファイル構成.....	166
2. DSP機能プログラム.....	166
(1) ADD_2.BLK.....	166
(2) ADD_3.BLK.....	166
(3) ADD_4.BLK.....	166
(4) DIF_AMP.BLK.....	166
(5) GET_PARAM.BLK.....	166
(6) INPUT.BLK.....	167

(7)	LIMIT. BLK	167
(8)	SET_PARAM. BLK	167
(9)	SET_PARAM_2. BLK	167
(1 0)	PID_AMP. BLK	167
(1 1)	SENSE_I. BLK	167
(1 2)	SENSE_V. BLK	167
(1 3)	ZERO_POLE. BLK	167
3 .	太陽電池D S P機能プログラム	168
(1)	SOLAR_CELL_DIODE_I. BLK	168
(2)	SOLAR_CELL_MATH_I. BLK	168
(3)	MPPT_GRADIENT_POWER. BLK	168
4 .	太陽電池カスタム回路	168
(1)	solarcell-diode-i. cvt	168
(2)	solarcell-math-i. cvt	168
(3)	mppt-gradient-power. cvt	168
5 .	太陽電池サンプル回路	168
(1)	SolarCell_Boost. cvt	168
(2)	SolarCell_Buck. cvt	168

1. SCAT K492 の概要

ようこそ SCAT K492へ

K492はWindows上で動作する主にスイッチング電源を対象にした回路解析シミュレータであり、以下のような優れた特長があります。

(1) パソコン環境で実行可能なシミュレータ

K492の実行環境はパソコンレベルで十分OKです。プラットフォームはパソコンの世界的標準OSであるWindowsです。

(2) 高速安定解析アルゴリズム

K492のアルゴリズムはスイッチング電源用に最適化されており、全くのオリジナルのアルゴリズムを採用しています。このため、実行速度は極めて高速であり、SPICEなどの汎用シミュレータに比較して二桁以上の改善がなされています。もう、計算時間で苦勞することはありません。さらに、シミュレーションの安定性も抜群であり、収束性の問題も解決しています。スイッチング電源の試行錯誤のシミュレーションも十分可能です。

(3) 自動刻み時間アルゴリズム

SCATシミュレータでは、回路シミュレーションで最も重要なパラメータである刻み時間を自動的に最適設定します。ユーザーは、刻み時間などシミュレーション特有のパラメータに煩わされることなく、気軽にシミュレーションできます。

(4) 多彩なスコープウィンドウ

K492では、波形や特性表示用に、実に4枚ものスコープウィンドウを搭載しており、多様な波形、特性観測を行うことができます。また、その操作もマウ

スクリックで自由に行うことができ、このような機能を持つのはSCATシミュレータだけです。

(5) スイッチング動作解析

K492では、オン・オフのスイッチング動作を詳細に調べる機能を豊富に搭載しています。従来の状態ウィンドウに加え、回路図やスコープウィンドウも連動して、状態解析をサポートします。これも他のシミュレータには見られないSCATシミュレータならではの機能です。

(6) オブジェクト指向

K492の最大の特長は、オブジェクト指向プログラミングにより実現されており、無駄のないコンパクトなソフトウェアであることです。さらに、ユーザーからのフィードバックを基に改良拡張を迅速に実現しており、SCATシミュレータは、いまこの瞬間にも進化しています。

それでは、K492のシミュレーション機能を紹介します。

(1) 過渡応答解析

過渡応答解析はK492が最も得意とする解析です。電圧、電流の応答波形はもちろん、それらの一周期での最大値や最小値の軌跡も簡単に表示できます。また、波形解析や継続解析と組み合わせれば過渡応答の任意の時点での波形が観測できます。

(2) 波形解析

K492は、強力なオシロスコープを内蔵しています。これを用いて実験感覚で各部の電流、電圧波形を観測できます。また、回路パラメータの異なる波形も比較できる専用のスコープウィンドウを用意しています。

(3) 定常解析

定常解析は従来の汎用シミュレータが最も苦手とする解析です。K492はこの定常解析を他のシミュレータにない画期的なアルゴリズムで高速に実行します。しかも、スイッチやダイオードのオン・オフ状態も完全に決定します。

(4) スイープ解析

スイープ解析とは、あるパラメータをある範囲で変化させたときの特性解析であり、負荷特性などがその典型的な例です。K492は負荷特性に限らず、ほとんどのパラメータに対するスイープ解析が可能であり、しかも非常に高速です。今までのシミュレータが数時間かかっていたのが数分で終了します。また、そのメニュー項目も4つあり、様々なスイープ解析が行えます。

(5) 制御時解析

制御回路を含む場合の前述の高速解析が可能です。従来のシミュレータでは、制御時の解析は非常に困難ですが、K492は、PWM制御、スイッチング周波数制御などの多彩な制御時の解析が簡単に行えます。

(6) 交流入力解析，三相入力解析，インバータ解析

K492では、交流入力時の解析が可能であり、リップル解析や力率解析の強力な手段となります。三相回路、インバータ回路、アクティブフィルタ回路など、あらゆる回路の解析ができます。

(7) F F Tおよび高調波解析

高調波規制が厳しくなっており、力率改善特性解析が重要となっています。K492は強力なF F Tウィンドウを装備しているので、任意の時点でF F T解析が実行できます。さらに、高調波規制のクラス分類、入力電流の力率、歪み率の計算など、高調波規制に関する高度な機能を搭載しています。

(8) 周波数特性解析

安定性を判断するには、周波数特性解析は不可欠です。K492は周波数特性が簡単に測定できます。一巡特性はもちろんのこと、部分回路の周波数特性もオリジナル回路を変更することなく測定可能です。この周波数特性解析には、二種類の解析法を準備しており、多様な回路に対応しています。

(9) デジタル制御解析

K492の最大の特長は、スマートグリッド時代を迎えて、今後主流となるDSPを利用したデジタル制御スイッチング電源が非常に容易に解析できることです。高価なシミュレーションツールにも勝るとも劣らないDSPプログラムの作成、検証機能が搭載されています。

(10) 太陽電池解析

代表的な新エネルギーとして注目される太陽電池もK492の持つ

K492をインストールするには、以下のファイルを適切なディレクトリにコピーするだけです。

K492PRx. EXE (xはバージョン番号) : K492本体

<CVTSMPL> : 回路のサンプルデータファイル

2. 回路の作成

本章では、回路の作成方法について説明します。本マニュアルでは、マウスの左ボタンでクリックすることを単に「クリック」と呼び、右ボタンでクリックすることを「右クリック」と呼ぶことにします。また、ダブルクリックとドラッグについても同様とします。また、メニューコマンドを、File:Exit のように表記します。これは、File メニューの Exit コマンドを表しています。他のコマンドも同様です。

2. 1 K492 の起動と終了 (File:Exit)

K492 を起動します。アプリケーションの起動のしかたは Windows のマニュアルを参照してください。K492 を起動すると、まず始めに、オープニングメッセージとともにメインウィンドウが現れます。

K492を終了するには、File:Exitをクリックします。このとき、回路データが変更されていれば、その保存をうながしてきますので、保存する場合はYes (はい) ボタン、しない場合にはNo (いいえ) ボタン、もとに戻る場合にはCancel (キャンセル) ボタンをクリックします。回路データの保存については、別項を参照してください。

2. 2 回路データの読み込み (File:Open)

作成した回路データを読み込むには、File:Open をクリックします。すると、ファイルダイアログボックスが現れます。そこで、Windows の標準的な方法で目的ファイルを指定します。するとデータが読み込まれ、回路図がメインウィンドウに表示されます。

メインウィンドウが解析モードであれば拡張子 CVT を持った回路データファイルと SHP を持った図形データファイルが読み込まれ、解析できる状態になります。メインウィンドウが図形モードであれば、拡張子 TPL または SHP を持った部品データを読み込み、図形作成状態になります。メインウィンドウについて

は、別項を参照してください。

2. 3 回路データの保存

(1) 回路データの保存 (File:Save)

回路データを保存するには、File:Saveをクリックします。すると、ファイルダイアログボックスが現れるので、保存するファイルの名前を設定します。メインウィンドウが解析モードであれば、拡張子は CVTでなければなりません。拡張子を省略すると自動的にこの拡張子が付けられます。デフォルトは現在のファイル名であり、このまま保存すると上書きされ、古いデータが新しいデータに更新されます。保存するファイル名と同じファイル名をもつファイルが既に存在すれば上書きするかどうかの確認のメッセージが表示されます。

ファイル名には日本語は使用しないでください。SCATでは日本語のファイル名はサポートされていません。

(2) スコープデータの保存 (File:Save Data:WVF, FFT, FRA, CMP)

後述する、WVF, FFT, FRA, CMPの各スコープウィンドウの表示波形データはファイルに保存できます。データファイルの拡張子は、それぞれ、WVF, FFT, FRA, CMPです。これらの波形データのファイルは、テキスト形式であり各データはタブで区切られているため、他の表計算やグラフ作成のアプリケーションソフトで容易に読むことができます。

(3) 部品データの保存 (File:Save)

メインウィンドウが図形モードであれば、部品としてデータの保存が可能です。保存方法は、解析モードにおける保存と同じです。拡張子は TPLでなければなりません。メインウィンドウに表示されている図形全てが部品として保存され、TPLファイルと同時にPRTファイルが作成されます。素子の各パラメータ値は、全て設定する必要はありません。部品として保存されたファイルは、メインウィンドウが図形モードであれば読み込むことができます。読み込まれた

部品は、既にメインウィンドウに表示されている回路に追加されます。よく使用する回路構成は部品として保存しておけば回路作成の効率化が図れます。

2. 4 回路の作成 (File:New)

メインウィンドウには、解析モードと図形モードの二つのモードがあります。回路を作成したり変更したりするには、図形モードにする必要があります。解析モードから図形モードへ移るには、Window:Topology をクリックします。もし、メインウィンドウのタイトルバーに TOPOLOGY と表示されていれば、すでに図形モードになっています。K492 の起動直後では、解析モードになっており、タイトルには ANALYSIS と表示されています。

新しい回路を作成するには、図形モードにおいて File:New をクリックします。するとすでに作成されている回路はすべて削除されます。削除された回路は Undelete で復元することができます。

図形モードにおいて、図形のないところで右クリックするとマウスのカーソル位置に素子メニューが現れます。ここで、必要な素子をクリックすると、画面上に指定した素子の図形が現れます。図形はドラッグすることにより移動でき、画面上の任意の位置に置くことができます。同じ要領で、つぎつぎに素子を表示し結合していきます。このとき、図形をクリックするとその四隅にハンドルと呼ばれる小さな四角形が現れ、その図形が選択されたことを表します。図形が選択された状態で右クリックすると機能メニューが現れ、さらに右クリックするとサブ機能メニューが現れます。このメニューを利用して図形の回転 (Rotate) や反転 (MirrorH, MirrorV) を行い、適当な形にします。図形の選択を解除するには、図形のないところでクリックします。また、図形をドラッグして他の図形の端子に近づけ、マウスのボタンを放すと自動接続機能により端子どうしが自動的に接続されます。このとき、接続点には自動的にノードが表示されます。通常素子は以上の操作で接続していきます。接続線の表示には、スペースキーを押しながらドラッグするか、素子メニューの Wire を使用します。また、スペースキーを押しながら、右クリックするとグランド素子 (Ground) が

表示されます。各素子については、別項を参照してください。

2. 5 図形の整形と変更

図形の整形と変更に関して以下の機能があります。

(1) 図形を選択する (Sselect All)

図形をクリックすると四隅または両端にハンドルが現れ選択状態になります。このとき、Ctrlキーを押しながら図形をクリックすると複数の図形が選択できます。また、素子メニューのSselect Allを利用すると、全図形が選択できます。また、各素子をクリックするかわりに、素子のない場所でドラッグし、表示された選択ボックスで素子を囲むことによっても選択できます。このとき、Altキーを押しながら実行すると文字を含まない素子のみを選択でき、Ctrlキーを押しながら実行すると素子を含まない文字のみを選択できます。

(2) 図形を移動する

図形をクリックして選択し、そのままドラッグすれば図形を移動できます。素子が移動すると、そのシンボルや値があればシンボルや値も移動します。

(3) 図形の大きさをかえる (Zoom in, Zoom out)

図形を選択しハンドルを表示します。その後、Altキーを押しながら、ハンドルのひとつをドラッグすれば大きさを変えることができます。ただし、Wire, Half Wire, Free Wire, CoreはAltキーを押す必要はありません。また、後述のサブ機能メニューのZoom in, Zoom outでも大きさを変更できます。

(4) 図形を回転させる (Rotate)

図形を回転させるには、図形を選択した後、右クリックします。すると、機能メニューが現れるので、Rotateをクリックします。後述の機能についても同様に行います。ただし、シンボルやコメントなど文字に対しては、Rotateは実

行できません。

(5) 図形を反転させる (MirrorH, MirrorV)

図形を横反転させるには、機能メニューのMirrorH、縦反転させるにはMirrorVを利用します。ただし、文字に対しては、MirrorH, MirrorVは実行できません。

(6) 図形を消去する (Delete)

図形を消去するには、機能メニューのDeleteを利用します。消去された図形は、TopologyモードであればUndeleteでもとにもどすことができます。また、素子が消去されるとそのシンボルや値も同時に消去され、シンボルや値のみを消去することはできません。シンボルと値の表示や非表示を行うには機能メニューのSymbolとValueを利用します。

(7) 消去した図形をもとに戻す (Undelete)

消去した図形をもとにもどすには、素子メニューのUndeleteを利用します。このコマンドのみ、機能メニューではなく素子メニューにあります。

(8) 図形を複製する (Copy)

図形を複製するには、機能メニューのCopyを利用します。このとき、素子にシンボルや値があればシンボルや値もコピーされます。シンボルのみのコピーや値のみのコピーはできません。

(9) 図形のシンボルを表示する (Symbol)

素子のシンボルを表示するには、機能メニューのSymbolを利用します。ONにすると、図形の近くにそのシンボルが現れ、OFFにすると消えます。シンボルも一つの図形として扱われます。デフォルトでは、ONになっています。ただし、変圧器の巻線とコアは、必ず、変圧器を構成した後にONしなければなりません。

(10) 素子の値を表示する (Value, Value Mode, Value Unit)

素子の値を表示するには、機能メニューのValueを利用します。ONにすると、値が表示可能であれば図形の近くにその値が現れ、OFFにすると消えます。値も一つの図形として扱われます。デフォルトでは、OFFになっています。ただし、変圧器の巻線は、必ず、変圧器を構成した後にONしなければなりません。また、機能メニュー表示時にさらに右クリックするとサブ機能メニューが現れます。この中のValue Modeは、表示方法をかえるためのものであり、'1m'のように、指数記号(k, m, uなど)を付加して表示するか、指数記号なしで'0.001'のように表示するか選択できます。さらに、Value UnitはONであれば単位を表示し、OFFであれば単位を表示しません。

(11) 図形の色を変える (Color)

図形の色を変えるには、図形を選択し、機能メニューのColorで設定する色をクリックします。

(12) 図形をグループ化する (Group)

図形をグループ化するには、複数の図形をひとつの図形にまとめることを言います。これを行うには、まず、グループ化する素子を選択し、次に、機能メニューのGroupを実行します。グループ化された図形はひとつの図形とみなされ、他の機能がそのまま適用できます。ただし、シンボルや値のみをグループ化することはできません。

(13) 図形のグループ化を解除する (Ungroup)

図形のグループ化を解除するには、グループ化された図形を選択して、機能メニューのUngroupを利用します。

(14) 素子極性の表示, 非表示を行う (Polarity)

素子の極性を表示するには、サブ機能メニューのPolarityを利用します。ONにすると、必要であれば素子の+電圧端子に丸印“・”が表示され、OFFにすると消えます。

(15) 図形を活性化する (Activate)

図形を活性化するには、サブ機能メニューのActivateを利用します。活性化された図形は回路素子として認識されます。

(16) 図形を不活性化する (Deactivate)

図形を不活性化するには、サブ機能メニューのDeactivateを利用します。図形が不活性化されると、その図形は、単なる図形となり回路素子として認識されません。

(17) 図形の線の太さを変える (Line Width)

図形の線の太さを変えるには、図形を選択し、サブ機能メニューのLine Widthで線幅を入力します。

(18) 文字のフォントを変える (Font)

文字のフォントを変えるには、文字を選択し、サブ機能メニューのFontで設定するフォントをクリックします。

(19) 文字の大きさを変える (Font Size)

文字の大きさを変えるには、文字を選択し、サブ機能メニューのFont Sizeで文字の大きさを入力します。

(20) 素子をカスタム化する (Customize)

素子をカスタム化します。カスタム化された素子は、カスタム回路を作成または読み込むことができるようになります。カスタム回路については、カスタム回路の項を参照してください。

(21) デフォルト値を変える (View:Option:Default, View:Initialize)

図形表示に関するデフォルト値を変えるには、View:Option:Defaultを利用します。以下のパラメータのデフォルト値が設定できます。

- [1] Device Width : 図形の幅
- [2] Device Height : 図形の高さ
- [3] Device Color : 図形の色
- [4] Line Color : 接続線の色
- [5] Line Width : 線幅
- [6] Font : 文字のフォント
- [7] Font Color : 文字の色
- [8] Font Size : 文字の大きさ
- [9] Program Font Size : プログラム文字の大きさ
- [10] Selection Handle Size : 素子選択ハンドルの大きさ
- [11] Symbol : 素子表示時のシンボル同時表示・非表示
- [12] Value : 素子表示時の値同時表示・非表示
- [13] Value Mode : 値の表示モード
- [14] Value Unit : 値の単位の表示・非表示
- [15] Highlight Color : ハイライト色
- [16] OFF1 Color : オフ状態の第一表示色
- [17] OFF2 Color : オフ状態の第二表示色
- [18] Limit Color : 出力変数の制限値を超えた素子の表示色
- [19] Node Color : 接続点の色
- [20] Node Size : 接続点の大きさ

また、View:Initializeを用いると、これらの設定値で図形全体が再表示されます。

(2 2) 図形全体を定位置に移動する (View:Home)

View:Homeを用いると、図形全体が画面左上に移動します。

(2 3) 図形全体を拡大する (View:Zoom in)

図形全体を拡大するには、View:Zoom inを利用します。

(2 4) 図形全体を縮小する (View:Zoom out)

図形全体を縮小するには、View:Zoom outを利用します。

(2 5) 表示色を元の色にもどす (View:Refresh)

さまざまな機能によって変更された表示色を元の色に戻すには、View:Refreshを利用します。メインウィンドウと状態ウィンドウに機能します。

(2 6) ノードの表示と消去 (View:Node:Create, Delete)

全ての回路の接続点にノードを表示するには、View:Node:Create Nodeを利用します。反対に全てのノードを消去するにはView:Node>Deleteを利用します。

(2 7) 画面のクリップボードへのコピー (Window:Copy)

表示されている内容をクリップボードへコピーするには、Window:Copyを利用します。コピーできる内容は以下のとおりです。

[1] Main : メインウィンドウ

[2] State : 状態ウィンドウ

[3] WVF : W V F ウィンドウ

[4] FFT : F F T ウィンドウ

[5] FRA : F R A ウィンドウ

2. 6 カスタム回路

(1) カスタム回路の作成

カスタム回路とは、回路を一つの素子として表現するものです。非常に複雑な回路や同じような回路部分が複数ある場合に非常に便利です。K. 492では、このカスタム回路を簡単に作成変更ができます。カスタム回路の作成手順は次のとおりです。

[1] カスタム素子の作成

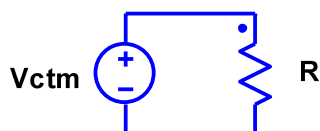
[2] カスタム回路の作成

以下にそれぞれの手順について説明します。

[1] カスタム素子の作成

カスタム素子とは、カスタム回路の表示用の素子で、カスタム回路の入出力端子の数と同じ端子数をもつ素子であれば、どの素子にしてもかまいません。

簡単な例として、下図のような回路をTopologyモードで作成します。

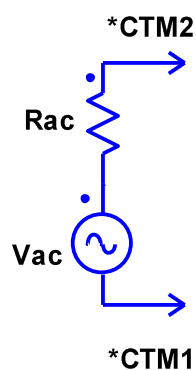


この図で、Vctmは、Prg Voltageを選択し、シンボルをVctmに変更したものです。カスタム回路の表示素子であるカスタム素子としてこのVctmを使用することになりますが、カスタム素子の選択は自由で、端子数が同じならば、他の素子を選択してもかまいません。現時点では、VctmはPrg Voltageですが、つぎに、これをカスタム化しカスタム素子に変換します。Vctmをクリックして選択し、右クリックを繰り返してサブ機能メニューを表示させます。そして、CustomizeをクリックすればVctmはカスタム化され正式のカスタム素子となります。一度カスタム化すると元の素子に戻すことはできません。

[2] カスタム回路の作成

次に、カスタム回路を作成します。Vctmを右クリックすると新たにカスタムウィンドウが表示されます。このカスタムウィンドウは基本的に解析機能がないこと以外はメインウィンドウと同じです。そこで、通常の回路を作成する方法と同じ方法で、カスタム化する回路を作成します。この回路をカスタム回路といいます。

次に、入力または出力用の端子にカスタム端子(Cmt Terminal)を接続します。カスタム端子には自動的にシンボルが"*CMT1"などのように設定され、数字が端子番号となります。このシンボルは変更できません。このカスタム端子を、前の手順 [1] で作成したカスタム素子の端子番号に合わせて、下図のように、接続します。たとえば、カスタム端子*CMT1をカスタム素子の端子番号1に対応するように接続します。素子に定義されている端子番号については、第3章を参照してください。カスタム回路のカスタム端子とカスタム素子の端子は1対1に対応していなければならない、同じ端子 (Wireなどの接続線を含む) に複数のカスタム端子を接続することはできません。



カスタム回路

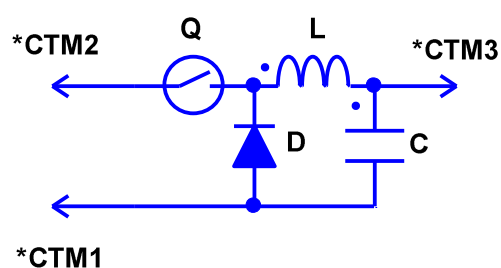


カスタム素子

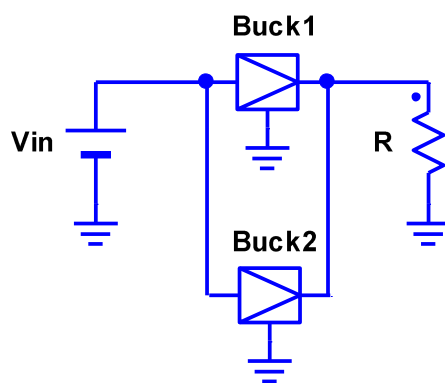
つぎに、Menu:Saveコマンドを使用しカスタム回路に適切なファイル名をつけて保存すればカスタム回路の作成は完了です。このとき、回路データはCVTファイルとして保存されます。また、カスタム回路を新規に作成する代わりに、すでに作成したCVTファイルをMenu:Readコマンドで読み込んでもかまいません。

これでカスタム回路作成の手順は全て完了です。右上のWindows標準のクローズボタンをクリックしてカスタムウィンドウを閉じ、Analysisモードに移行すれば解析可能となります。カスタムウィンドウのその他の機能については、第4章を参照してください。

次の例は、降圧形コンバータを2台並列にした応用例です。カスタム素子には、Operator:Functionを使用しており、これら二つのカスタム素子で、同じカスタム回路ファイルを読み込んで使用し並列化しています。



カスタム回路（降圧形コンバータ）



メイン回路

カスタム回路は、ネスト可能です。すなわち、カスタム回路内にカスタム回路を含むことができます。

(2) カスタム回路の変更

すでに作成したカスタム回路のパラメータや回路構成を変更する場合は、カスタム素子を右クリックして、カスタムウィンドウを表示させます。あとは、

メインウィンドウの方法と同じです。ただし、Analysis, Topologyモードの変更は、全てのカスタムウィンドウを閉じた状態で、メインメニューで行います。Analysisモードでは、パラメータの値が変更でき、Topologyモードでは、パラメータと回路構成の両方が変更可能です。

(3) カスタム回路におけるシンボル

カスタム回路内の素子には、二つのシンボルがあります。カスタム回路内で定義される局所シンボルとメイン回路の回路全体で定義される大域シンボルです。大域シンボルは以下の規則によって、局所シンボルから生成されます。

大域シンボル = カスタム素子大域シンボル. 局所シンボル

たとえば、前図において、Buck1内のQの大域シンボルはBuck1.Qとなります。

もし、カスタム回路がネストされていれば、対応するカスタム素子シンボルがピリオドで接続されます。カスタムウィンドウ内での素子シンボルは必ず局所シンボルとなります。また、出力変数は必ず大域シンボルとなりメインウィンドウで指定されます。

(4) カスタム回路データファイル

カスタム素子に読み込まれているカスタム回路のデータファイルは、メインメニューでメイン回路を保存するとき、次のファイル名に変更されて自動的に保存されます。ただし、すでに変更されていればそのままのファイル名で保存されます。

カスタム回路のファイル名 = C TM+メイン回路ファイル名+カスタム素子の
大域シンボル+カスタム回路作成時のファイル名

各部分の接続には、アンダーバー' _ ' が使用されます。たとえば、最初に作成したカスタム回路のファイル名が"Buck"、それを読み込んでいるカスタム素子の
大域シンボルが"Buck1"、メイン回路のファイル名が"Paralell"であれば、

新しいカスタム回路データファイル名は、“CTM_Parallel_Buck1_Buck”となります。したがって、カスタム素子ごとにデータファイルが生成されるので、個別にパラメータ変更や回路構成が自由にできます。また、カスタム回路作成時に保存したファイルは変更されませんので、何度でも再利用可能です。

3. 素子モデルとパラメータの設定

本章では、K492で利用できる素子のモデルとそのパラメータの設定方法について説明します。K492における数値パラメータは、時比率などの少数の例外を除いて数値とmやkなどの指数記号の組み合わせで表示され、単位はMK S A単位系の基本単位です。数値入力時には、数値のみでも数値と指数記号の組み合わせでもどちらでも入力できます。たとえば、インダクタンスの場合で、1.5mHを表す場合、表示時には1.5mと表示され、入力時には、小数表現 0.0015 または 指数表現 1.5e-3 のように数値のみでも、1.5mのように数値と指数記号の組み合わせでも入力できます。

指数表現の形式は、

(仮数部) e (符号) (指数部)

であり、例えば

$$0.0047 = 4.7e-3$$

$$95200 = 9.52e4$$

となります。入力する場合、符号が正の場合は「+」を省略できますが、負の場合は必ず「-」をつけください。また、使用できる指数記号は以下のとおりです。

p 1.0e-12

n 1.0e-9

u 1.0e-6

m 1.0e-3

k 1.0e+3

M 1.0e+6

G 1.0e+9

T 1.0e+12

特に、1.0e-6を表す指数記号は、uであることを注意してください。

3. 1 素子モデル

各素子のパラメータを設定するには、全ての選択素子を解除し、目的の素子にカーソルを移動して右クリックします。すると、その素子のパラメータダイアログボックスが現れますので、各パラメータを設定します。

また、いくつかの素子には出力変数表示用の内部シンボルが存在し、全体として“素子シンボル . 内部シンボル”のように素子シンボルと内部シンボルの間にピリオド“.”をいれて表します。たとえば、過飽和インダクタでは、インダクタを表す1個の内部シンボル“L”が存在します。したがって、可飽和インダクタのシンボルとして“SL”が設定されている場合、全体のシンボルは“SL.L”のように表示されます。この表示法は出力変数の設定に使用されます。

(1) インダクタ (LCR:Inductor, Inductor-R)



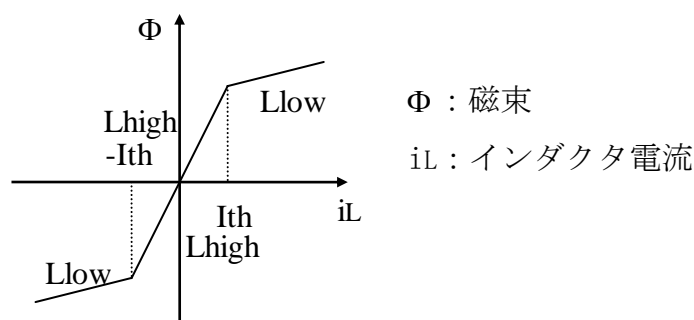
パラメータとして、Inductorには、シンボル(Symbol)、インダクタンス(Value)、電流初期値(Iini)があり、さらに、Inductor-Rには、等価直列抵抗(ESR)があります。シンボルは、各素子に対して重複のない一意なシンボルにする必要があります。また、文字数には制限はありませんが、表示の関係上、半角英数字で1～4文字を推奨します。電流初期値とは、解析開始時におけるインダクタ電流の値です。ただし、灰色表示の場合は与える必要はありません。特別な値がなければ、0を設定します。また、Inductor-Rの等価直列抵抗は、0に設定すると短絡とみなされます。Inductorには、内部シンボルはなく、Inductor-Rの内部シンボルは素子電圧“V”と素子電流“I”です。

端子についている番号は、カスタム化する場合に参照される端子番号です。詳しくはカスタム回路の説明を参照してください。

(2) 可飽和インダクタ (LCR:Sat Inductor)



下図のような特性をもつインダクタであり、パラメータとして、シンボル (Symbol), 飽和時インダクタンス (Llow), 非飽和時インダクタンス (Lhigh), しきい電流 (Ith), 電流初期値 (Iini)があります。下図のように、流れる電流の絶対値が、Ith 以下のとき Lhigh のインダクタンス、Ith より大きいとき Llow のインダクタンスとなります。内部シンボルは、インダクタ“L”です。



(3) キャパシタ (LCR:Capacitor, Capacitor-RL)



パラメータとして、Capacitorには、シンボル (Symbol), キャパシタンス (Value), 電圧初期値 (Vini)があり、さらに、Capacitor-RLには、等価直列抵抗 (ESR), 等価直列インダクタンス (ESL), 等価直列インダクタンスの電流初期値 (Iini)があります。Capacitor-RLの等価直列抵抗と等価直列インダクタンスは0に設定すると短絡とみなされます。等価直列インダクタンスが0の場合、その初期値は無視されます。Capacitorの内部シンボルはなく、Capacitor-RLの内部シンボルは素子電圧“V”と素子電流“I”です。等価直列インダクタンスは、

あまり小さな正の値を設定すると、計算時間と計算精度が悪くなるので、通常は0に設定してください。

(4) 抵抗 (LCR:Resistor)



パラメータとして、シンボル(Symbol), 抵抗値(Value)があります。

(5) 直流電圧源 (Source:DC Voltage)



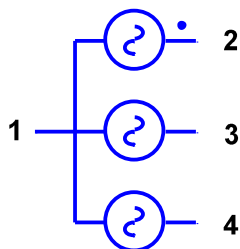
パラメータとして、シンボル(Symbol), 電圧値(Value)があります。

(6) 交流電圧源 (Source:AC Voltage)



交流電圧を発生させる素子であり、パラメータとして、シンボル(Symbol), 振幅(Vrms), 周波数(Freq), 初期位相(Phini), モード(Mode)があります。振幅は実効値で与え、初期位相は度で与えます。また、モードにより、正弦波(Sin), 半波(Half), 全波(Full)が選択できます。この電圧源はサンプルホールド動作をします。各スイッチングの初めに電圧がサンプル設定され、そのスイッチング期間では電圧はホールドされ一定となります。したがって、周波数は、スイッチング周波数に比べて十分低い必要があります(100分の1以下推奨)。内部シンボルは、出力電圧“V”です。

(7) 交流三相電圧源 (Source:3Phase Voltage)



交流三相電圧を発生させる素子であり、パラメータとして、シンボル (Symbol), 振幅 (Vrms), 周波数 (Freq), 初期位相 (Phini), モード (Mode) があります。振幅と周波数は3個の交流電圧源に対して共通であり、振幅は実効値で与えます。初期位相は一番上の電圧源に対して度で与え、モードがNormalの場合、初期位相のずれは上の方から順に0度, 240度, 120度となります。モードが、Reverseの場合は、上の方から、0度, 120度, 240度となります。この電圧源は、サンプルホールド動作をします。各スイッチングの初めに電圧がサンプル設定され、そのスイッチング期間では電圧はホールドされ一定となります。したがって、周波数は、スイッチング周波数に比べて十分低い必要があります (100分の1以下推奨)。内部シンボルは、3個の交流電源 "V1", "V2", "V3" (上から順) です。

(8) プログラマブル電圧源 (Source:Prg Voltage)



この電圧源は、プログラムにより任意の電圧を発生させることができます。この電圧源は、サンプルホールド動作をします。各スイッチングの初めに電圧がサンプル設定され、そのスイッチング期間では電圧はホールドされ一定となります。詳細は、制御の項目を参照してください。パラメータとしてシンボル (Symbol) があります。内部シンボルは、出力電圧 "V" です。

(9) テーブル電圧源 (Source:Tbl Voltage)



この電圧源は、ユーザーが定義したテーブルにしがって電圧を発生させます。パラメータには、(Symbol), テーブルファイル(Table File)があります。この電圧源を使用する場合、まず、テーブルファイルを作成します。適当なエディタ(メモ帳など)を起動し、カンマでくぎって、時間, その時間に対する出力値を書いていきます。たとえば、

1.0e-3, 5.0

2.0e-3, 10.0

3.0e-3, 15.0

のように、書いていきます。次に、このデータをテキストファイルとして保存します。このファイルの拡張子は SRC に設定します。そこで、テーブル電圧源の Table ボタンをクリックすると、ファイルダイアログボックスが現れるので、テーブルファイルを指定します。これで、準備完了です。設定したテーブルファイルの内容は、List ボタンをクリックすることにより、メモ帳で確認変更することができます。ただし、変更した場合はメモ帳で保存し、再度 Table ボタンで読み込んでください。時間が、テーブルファイルに定義されている時間の中間値ときは線形補間を行います。たとえば、上の例で、実際の時間が、1.5e-3 の場合、出力電圧は、7.5 となります。また、時間が、定義されている範囲をこえた場合は、出力電圧は最終値になります。たとえば、上の例で、時間が 4.0e-3 の場合、出力電圧は 15.0 となります。この電圧源は、サンプルホールド動作をします。各スイッチングの初めに電圧がサンプル設定され、そのスイッチング期間では電圧はホールドされ一定となります。内部シンボルは、出力電圧“V”です。

(1 0) 直流電流源 (Source:DC Current)



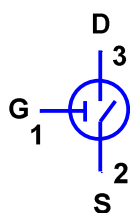
パラメータとして、シンボル(Symbol), 電流値(Value)があります。

(1 1) 交流電流源(Source:AC Current)



交流電流を発生させる素子であり、パラメータとして、シンボル(Symbol), 振幅(Vrms), 周波数(Freq), 初期位相(Phini), モード(Mode)があります。振幅は実効値で与え、初期位相は度で与えます。また、モードにより、正弦波(Sin), 半波(Half), 全波(Full)が選択できます。この電流源はサンプルホールド動作をします。各スイッチングの初めに電流がサンプル設定され、そのスイッチング期間では電流はホールドされ一定となります。したがって、周波数は、スイッチング周波数に比べて十分低い必要があります(100分の1以下推奨)。内部シンボルは、出力電流“I”です。

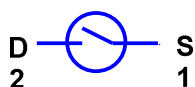
(1 2) スイッチ (Switch:Switch)



スイッチは、理想FETとして表され、ゲート(G), ソース(S), ドレイン(D)の3端子があります。パラメータとして、シンボル(Symbol), オン抵抗(Ron), オフ抵抗(Roff), しきい電圧(Vth), ヒステリシス電圧(Vhys)があります。ゲート・ソース間はオープンであり、ゲートを+側、ソースを-側とした電圧が

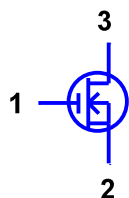
$V_{th}+V_{hys}/2$ より高いとオンになり、ドレイン・ソース間がオン抵抗になります。逆にゲート・ソース間電圧が $V_{th}-V_{hys}/2$ より低いとオフになり、ドレイン・ソース間がオフ抵抗となります。内部シンボルは、ソース・ドレイン間抵抗（ドレインが電圧+側，ソースが電圧-側）"R"，ゲート・ソース間電圧" V_s "です。

(1 3) PWMスイッチ (Switch:PWM Switch)



PWMスイッチは、パルス発生器とスイッチを合成した素子であり、ソース(S)，ドレイン(D)の2端子があります。パラメータとして、シンボル(Symbol)，オン抵抗(R_{on})，オフ抵抗(R_{off})，初期状態(Initial State)，状態変化時刻(Transition)，モード(Mode)があります。PWMスイッチは、スイッチングの各周期において、初期状態から始まり、状態変化時刻で設定された時刻でオン・オフの状態変化を生じます。オン状態ではオン抵抗となり、オフ状態ではオフ抵抗となります。状態変化時刻の意味および設定方法は、パルス発生器と同じです。内部シンボルは、ドレイン・ソース間抵抗（ドレインが電圧+側，ソースが電圧-側）"R"です。

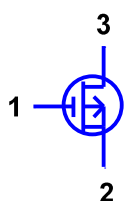
(1 4) Nチャンネル MOSFET (Switch:Nch MOSFET)



NチャンネルMOSFETで、パラメータとして、シンボル(Symbol)，ゲート・ソース間しきい電圧(V_{th})，順方向伝達アドミタンス(Y_{fs})，ドレイン・ソース間オン抵抗($R_{ds(on)}$)があります。内部シンボルは、ゲート・ソース間電圧

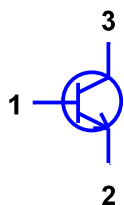
“ V_{gs} ”, ドレイン・ソース間電圧“ V_{ds} ”, ドレイン電流“ I_d ”です。このMOSFETは、増幅動作が必要な場合に使用します。オン・オフのスイッチング動作だけが必要な場合は、スイッチまたはPWMスイッチを使用してください。

(15) PチャンネルMOSFET (Switch:Pch MOSFET)



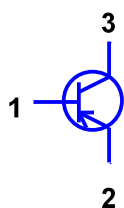
PチャンネルMOSFETであり、内容はNチャンネルMOSFETと同じです。特に、ゲート・ソース間しきい電圧(V_{th})は、負の値にする必要はなく、正の値で設定します。

(16) NPNトランジスタ (Switch:NPN Transistor)



NPNトランジスタです。パラメータとして、シンボル(Symbol), エミッタ接地Hパラメータ (H_{ie} , H_{fe}), ベース・エミッタ間電圧(V_{be}), コレクタ・エミッタ間飽和電圧($V_{ce(sat)}$), ベース・エミッタ間飽和抵抗($R_{ce(sat)}$)があります。内部シンボルは、ベース・エミッタ間電圧“ V_{be} ”, コレクタ・エミッタ間電圧“ V_{ce} ”, ベース電流“ I_b ”, コレクタ電流“ I_c ”です。このトランジスタは、増幅動作が必要な場合に使用します。オン・オフのスイッチング動作だけが必要な場合は、スイッチまたはPWMスイッチを使用してください。

(17) PNPトランジスタ (Switch:PNP Transistor)



PNPトランジスタです。内容はNPNトランジスタと同じです。

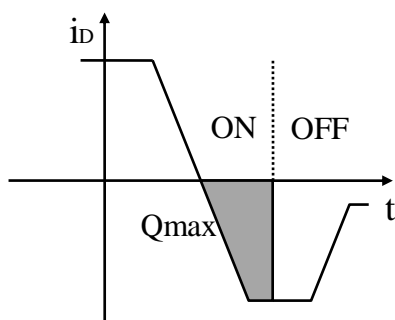
(18) ダイオード (Diode:Diode)



パラメータとして、シンボル(Symbol), オン抵抗(R_{on}), オフ抵抗(R_{off}), オン電圧(V_{on}), 最大蓄積電荷(Q_{max})があります。ダイオードは、抵抗と直流オン電圧との直列回路で表され、アノード・カソード間のバイアス電圧が、オン電圧より高いとオン状態となり、ダイオードは、オン抵抗とオン電圧の直列回路となります。逆に、バイアス電圧がオン電圧より低いとオフ状態となり、ダイオードは、オフ抵抗とオン電圧の直列回路となります。

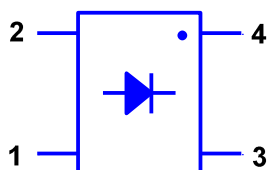
また、最大蓄積電荷 Q_{max} は、ダイオードの逆方向回復特性を解析するためのパラメータであり、逆方向電流の積分値の最大値を表しています。逆方向回復特性を解析するには、 Q_{max} を正の値に設定します。

ダイオードが逆バイアスされた時点で Q_{max} が正の値の場合、下図のようにオン状態が持続し、逆方向電流の積分値が Q_{max} になるまで逆方向電流が流れず。そして、その後オフ状態となります。



内部シンボルは、順方向電圧“V”，順方向電流“I”です。

(19) ダイオードブリッジ (Diode:Diode Bridge)



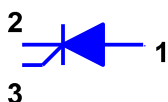
4個のダイオードからなるダイオードブリッジです。パラメータとして、シンボル(Symbol)，オン抵抗(R_{on})，オフ抵抗(R_{off})，オン電圧(V_{on})があります。これらのパラメータは4個のダイオードに対して共通です。内部シンボルは4個のダイオードの電流“ I_n ”，電圧“ V_n ”(n=1, 2, 3, 4)および入力電圧“ V_i ”と出力電圧“ V_o ”です。

(20) ツェナーダイオード(Diode:Zener)



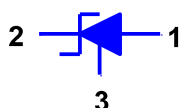
パラメータとして、シンボル(Symbol)，オン抵抗(R_{on})，オフ抵抗(R_{off})，オン電圧(V_{on})，ツェナー電圧(V_{zn})，ツェナー抵抗(R_{zn})があります。ツェナー抵抗とは、逆バイアス導通時の抵抗のことです。内部シンボルは、ツェナー電圧“V”です。

(21) サイリスタ(Diode:Thyristor)



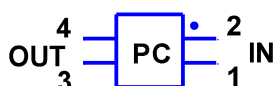
パラメータとして、シンボル(Symbol), オン抵抗(Ron), オフ抵抗(Roff), オン電圧(Von), トリガ電圧(Vtrg)があります。サイリスタは、アノード・カソード間が順バイアスのとき、トリガを+側、カソードを一側とした電圧が、トリガ電圧より高ければ、トリガされオンします。一旦トリガされた後は通常のダイオードと同じ働きをし、アノード・カソード間が逆バイアスされるとリセットされます。内部シンボルは、順方向電圧“V”, 順方向電流“I”, トリガ・カソード間電圧“Vt”です。

(2 2) シヤントレギュレータ(Diode:Shunt Regulator)



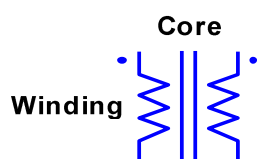
パラメータとして、シンボル(Symbol), ゲイン(Gain), 最小電圧(Vmin), 最大電圧(Vmax), 基準電圧(Vref)があります。内部シンボルは、アノード・基準電圧“Vr”, 出力電圧“Vo”です。

(2 3) フォトカプラ(Diode:Photo Coupler)



上図 IN が入力端子、OUT が出力端子で、パラメータとしてシンボル(Symbol), 入力ダイオードのオン抵抗(Ron), オフ抵抗(Roff), オン電圧(Von), 電流ゲイン(Gain)があります。入力側と出力側は絶縁されており、入力電流が電流ゲイン倍されて出力電流となります。

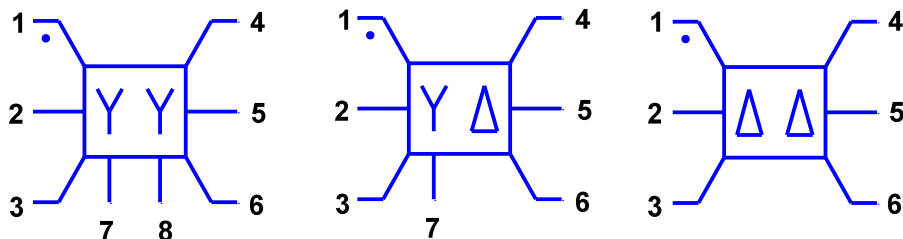
(24) 変圧器 (Transformer: Winding, Winding-R, Core, Make)



変圧器を構成するには、まず、素子メニューを利用して巻線 (Winding, Winding-R) を表示し接続します。巻線の数はい任意です。次に、これらの巻線用のコア (Core) を表示し、他の図形と交わらないように適当に配置します。そこで、素子メニューで Make をクリックすると、これまで表示した巻線とコアはグループ化されひとつの変圧器ができあがります。一度構成した変圧器は一つの素子とみなされ、Ungroupで分解することはできません。このようにして、変圧器をいくつも構成することができます。また、図形の色は、Makeをクリックして変圧器が構成される前では赤色、後ではデフォルト色になります。巻線とコアは必ずMakeにより変圧器として構成しなければなりません。

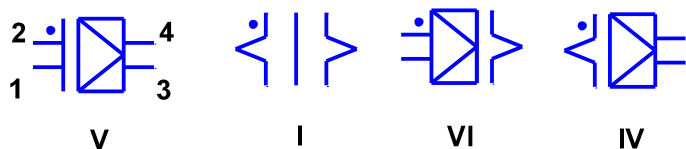
Windingには、巻線のシンボル (Symbol)、巻数比 (Value) のパラメータがあり、さらに、Winding-Rには、等価直列抵抗 (ESR) があり、素子電圧と素子電流の内部シンボル "V", "I" があります。等価直列抵抗は、0 に設定すると短絡とみなされ、この場合、素子電流の内部シンボル "I" はなくなります。コアについては、シンボル (Symbol) があります。変圧器は理想変圧器なので、励磁インダクタンスやもれインダクタンスなどは外付けする必要があります。また、変圧器構成後は、コアのシンボルが変圧器のシンボルとなり、巻線のシンボルは "変圧器のシンボル . 巻線のシンボル" で表されます。たとえば、変圧器のシンボルを "T"、巻線のシンボルを "n1" とすれば、変圧器構成後の巻線のシンボルは "T.n1" となります。変圧器はカスタム化できず、機能メニューの Symbol と Value は変圧器の構成後に ON 可能となります。

(2 5) 三相変圧器 (Transformer:3Phase Y-Y, 3Phase Y-D, 3Phase D-D)



三相変圧器には、Y-Y 型 (3Phase Y-Y)、Y-Δ 型 (3Phase Y-D)、Δ-Δ (3Phase D-D) があり、パラメータとして、シンボル (Symbol), 一次巻数 (N_{wp}), 二次巻数 (N_{ws}), 一次巻線抵抗 (R_{wp}), 二次巻線抵抗 (R_{ws}) があります。Y 巻線の下向きの端子は中性端子です。

(2 6) センサー (Sensor:V Sensor, I Sensor, VI Sensor, IV Sensor)

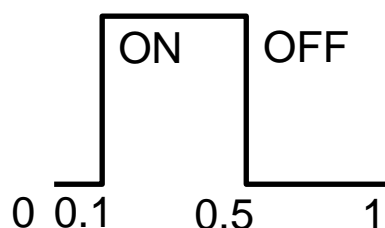


センサーには、電圧入力・電圧出力の電圧センサー (V Sensor)、電流入力・電流出力の電流センサー (I Sensor)、電圧入力・電流出力の電圧電流センサー (VI Sensor)、電流入力・電圧出力の電流電圧センサー (IV Sensor) があり、パラメータとして、シンボル (Symbol), ゲイン (Gain) があります。入力がゲイン倍されて出力となります。また、入力と出力は絶縁されており、入力・出力インピーダンスによる回路への影響はありません。内部シンボルは、入力電圧 " V_i ", 出力電圧 " V_o " があり、入力電流, 出力電流に対する内部シンボルはありません。

(27) パルス発生器 (Generator:Pulse)



矩形波のパルス電圧を発生させます。パラメータとして、シンボル(Symbol), オン電圧 (Von), オフ電圧 (Voff), 初期状態 (Initial State), 状態変化時刻 (Transition), モード (Mode)があります。パルス発生器は、スイッチングの各周期において、初期状態から始まり、状態変化時刻で設定された時刻でオン・オフの状態変化を生じます。オン状態では、端子電圧がオン電圧（上図の左端子が電圧+側、右端子が電圧-側）となり、オフ状態ではオフ電圧となります。たとえば、下図のように、初期状態がオフ状態から始まり、時比率 0.1 でオン状態に変化し、時比率 0.5 で再びオフ状態にもどる場合、Initial State を OFF、Mode を Ratio に設定します。次に、Transition ボックスで値を 0.1 と入力しリターンキーを押して改行します。同様に 0.5 を入力します。このとき、Mode が Ratio の場合は、状態変化する時刻をスイッチング周期で割った比率を入力し、Mode が Time の場合は、時刻そのものを入力します。これを、状態変化が起こる回数（ただし、周期の始めと終わりの状態変化は除く。下図では 0.1 と 0.5 の2回）だけ繰り返します。このようにして、一周期中に状態変化を何回でも設定できます。内部シンボルは、出力電圧“V”です。



(28) のこぎり波発生器 (Generator:Saw Tooth)



のこぎり波を発生する素子です。パラメータとして、シンボル(Symbol), 振

幅 (Vamp), 折れ曲がり時間率 (Df1d), モード (Mode) があります。この素子の周期は Parameter:Frequency の設定周波数または F R D 素子の周波数で決定されます。Df1d で波形が上昇から下降に変化する折れ曲がり時間が決まり、モードが Ratio のときは周期に対する割合を、Time のときは時間を設定します。内部シンボルは、出力電圧 "V" です。

(29) 波形発生器 (Generator:Function)



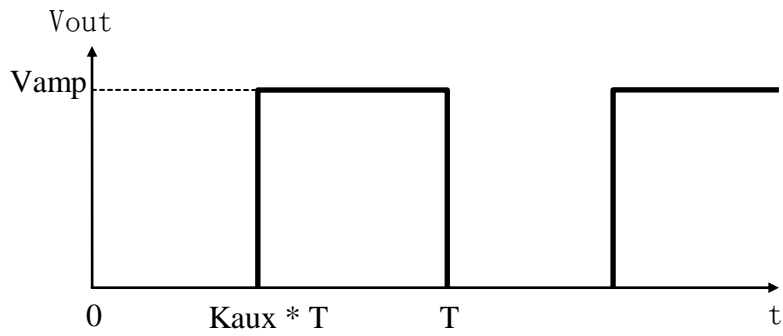
波形発生器は、モードにより下図のような矩形波 (Square), 直線波 (Linear), 指数波 (Exponential) の低周波周期波形を発生させる線形電圧源であり、パラメータとして、シンボル (Symbol), 振幅 (Vamp), 周波数 (Freq), 初期位相時間 (Tini), 補助定数 (Kaux), モード (Mode) があります。Tini が 0 の場合、最初の 1 周期 ($0 < t < T = 1 / \text{Freq}$) において以下で定義される電圧 V_{out} を繰り返し発生させます。

$$\begin{aligned} \text{矩形波: } V_{out} &= 0 \quad (0 < t / T < Kaux) \\ &= Vamp \quad (Kaux < t / T < 1) \end{aligned}$$

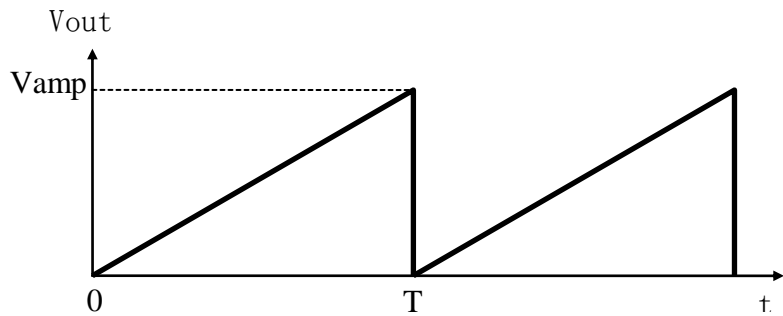
$$\text{直線波: } V_{out} = Vamp * t / T$$

$$\text{指数波: } V_{out} = Vamp * \exp(Kaux * t / T)$$

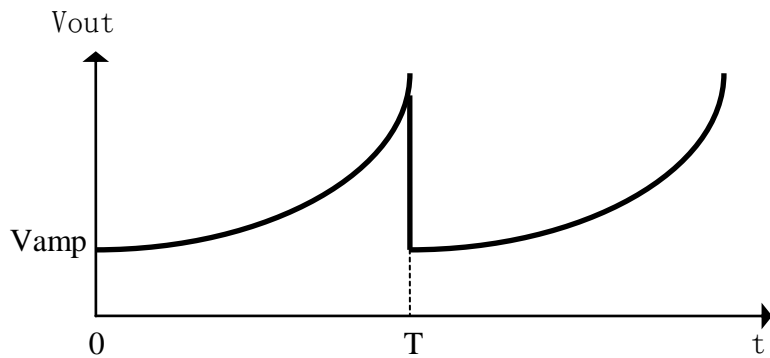
Tini が 0 でない場合は、Tini だけ時刻が進みます ($t \rightarrow t + Tini$)。この波形発生器は、サンプルホールド動作をします。各スイッチングの初めに電圧がサンプル設定され、そのスイッチング期間では電圧はホールドされ一定となります。したがって、周波数は、スイッチング周波数より十分低い必要があります (100 分の 1 以下推奨)。内部シンボルは、出力電圧 "V" です。



(a) 矩形波



(b) 直線波



(c) 指数波

(30) 周波数特性信号源 (Generator:AC Sweep)



周波数特性を解析するための交流信号電圧源です。パラメータとして、シンボル(Symbol), 最小周波数(Fmin), 最大周波数(Fmax), 周波数分割数(Ndiv), 過渡応答時間(Tan1), 交流信号振幅(Vac)があり、オプションとして PWM-SFM 解

析(PWM-SFM)があります。内部シンボルは出力電圧“V”です。詳細は、周波数特性解析の項目を参照してください。

(3 1) パルス変調器 (Modulator:PWM)



PWM制御を行う変調器です。パラメータとして、シンボル(Symbol), 変換ゲイン(Gain), 基準時比率 (Dref), 最小時比率(Dmin), 最大時比率(Dmax), 変調信号(Modulated), モード(Mode)があります。また、オプションとして、アクティブ(Active)があります。詳しくは、制御方法の項目を参照してください。内部シンボルは、入力電圧 (上図の上端子が電圧+側, 下端子が電圧-側) “Vi”です。

(3 2) 周波数変調器 (Modulator:SFM)



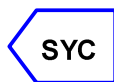
スイッチング周波数制御を行う変調器です。パラメータとして、シンボル(Symbol), 変換ゲイン(Gain), 基準周波数(Fref), 最小周波数(Fmin), 最大周波数(Fmax), 変調信号(Modulated)があります。また、オプションとして、アクティブ(Active)があります。詳しくは、制御方法の項目を参照してください。内部シンボルは、入力電圧 (上図の上端子が電圧+側, 下端子が電圧-側) “Vi”です。

(33) 素子パラメータ変調器(Modulator:DPM)



素子パラメータ変調器は、入力電圧にしたがって、素子のパラメータを変調する変調器です。パラメータとして、シンボル(Symbol)、変換ゲイン(Gain)、基準パラメータ値(Xref)、最小パラメータ値(Xmin)、最大パラメータ値(Xmax)、変調信号(Modulated)、パラメータ(Parameters)があります。また、オプションとして、アクティブ(Active)があります。詳しくは、制御方法の項目を参照してください。内部シンボルは、入力電圧（上図の上端子が電圧+側、下端子が電圧-側）“Vi”です。

(34) 同期制御器 (Modulator:SYC)



同期制御を行う制御器です。パラメータとして、シンボル(Symbol)、基準タイプ(Base Type)、モード(Mode)、基準状態変化時刻(Transition Base)、同期制御信号(Synchronized)があります。また、オプションとして、アクティブ(Active)があります。接続端子はありません。詳しくは、制御方法の項目を参照してください。

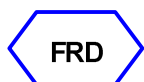
(35) プログラマブル制御器 (Modulator:PRC)



プログラムより任意の制御を行う制御器です。パラメータとしてシンボル(Symbol)があります。また、オプションとして、アクティブ(Active)がありま

す。詳細は、制御方法の項目を参照してください。

(36) スイッチング周波数発生器(Modulator:FRD)



複数の独立したスイッチング周波数の設定を行う素子です。以下のパラメータを設定してください。

(1) 新しいスイッチング周波数を Freq に設定する。

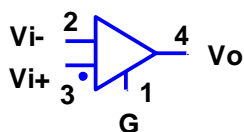
(2) メインスイッチング周波数との位相差時間を Tini に設定する。

メインスイッチング周波数とは、メインメニューで設定するこれまでのスイッチング周波数である。

(3) この新しいスイッチング周波数で動作させる素子名を Device に設定する。入力では素子名ごとに改行する。対象となる素子は PWMSwitch, Pulse, Saw Tooth, PWM, SFM, DPM, SYC, PRC, Latch, Trigger, DSP の 11 種類である。

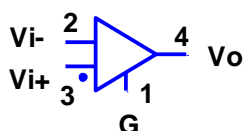
設定できるスイッチング周波数の数には制限がありません。Transient, Waveform 解析時の Cycle 数はメインスイッチング周期で定義され、Transient の出力変数の Mode もメインスイッチング周期で定義されます。さらに、素子のサンプルホールド動作もメインスイッチング周期で行われます。また、FRD を使用する場合の制限として、Transient, Waveform 解析時の PWM, SFM の制御に対する低周波近似補正が行われず、スイッチング周期終了時の完全なサンプリング制御となります。

(37) オペアンプ (OpAmp:OpAmp)



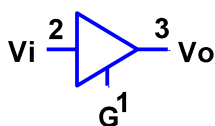
通常のオペアンプです。パラメータとして、シンボル (Symbol), ゲイン (Gain), 最小出力電圧 (V_{min}), 最大出力電圧 (V_{max}) があります。入力の極性は V_{i-} が負で V_{i+} が正であり、入力差電圧 $V_i (= V_{i+} - V_{i-})$ がゲイン倍されて出力電圧 V_o となります。内部シンボルは出力電圧 " V_o " です。

(38) 理想オペアンプ (OpAmp:Ideal OpAmp)



理想オペアンプです。二つの入力と一つの出力およびグラウンドの4端子があります。オペアンプは、入力差電圧 $V_i (= V_{i+} - V_{i-})$ が0になるように出力電圧 V_o を調整します。したがって、必ず帰還して使用し、コンパレータとしての機能はありません。また、出力電圧の制限もありません。グラウンド端子 (G) は必ず回路のグラウンドに相当する部分に接続します。パラメータは、シンボル (Symbol) のみです。内部シンボルは、出力電圧 " V_o " です。

(39) 積分器 (OpAmp:Integrator)



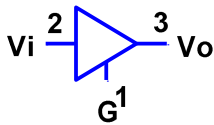
積分器には、シンボル (Symbol), ゲイン (Gain), 制限周波数 (F_{lim}) のパラメータがあり、次の伝達関数を持ちます。

$$\frac{V_o}{V_i} = -Gain \frac{\omega}{s + \omega}$$

$$\omega = 2\pi F_{lim}$$

内部シンボルは、入力電圧"Vi", 出力電圧"Vo"です。

(40) 微分器 (OpAmp:Differentiator)



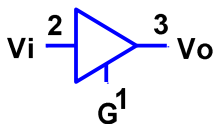
微分器には、シンボル(Symbol), ゲイン(Gain), 制限周波数(Flim)のパラメータがあり、次の伝達関数を持ちます。

$$\frac{V_o}{V_i} = -Gain \frac{s}{s + \omega}$$

$$\omega = 2\pi F_{lim}$$

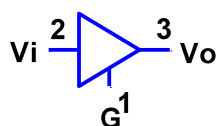
内部シンボルは、入力電圧"Vi", 出力電圧"Vo"です。

(41) ボルテージフォロワ(OpAmp:V Follower)



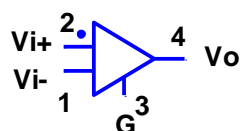
利得が1倍、入力インピーダンスは無限大、出力インピーダンスが零で、主にインピーダンス変換に利用する素子です。パラメータとしてシンボル(Symbol)があり、内部シンボルは、出力電圧"Vo"です。

(4 2) リミッタ (OpAmp:Limiter)



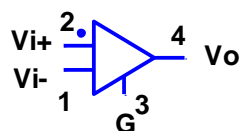
電圧を制限する素子です。パラメータとしてシンボル (Symbol), 最小電圧 (V_{min}), 最大電圧 (V_{max}) があります。内部シンボルは、入力電圧 " V_i ", 出力電圧 " V_o " です。

(4 3) コンパレータ (Comparator:Comparator)



理想コンパレータであり、シンボル (Symbol), オン電圧 (V_{on}), オフ電圧 (V_{off}), ヒステリシス電圧 (V_{hys}) のパラメータがあります。入力差電圧 $V_i (= V_{i+} - V_{i-})$ が $V_{hys}/2$ より大きい場合、出力電圧 V_o が V_{on} となり、 $-V_{hys}/2$ より小さい場合、 V_o が V_{off} となります。内部シンボルは、入力差電圧 " V_i ", 出力電圧 " V_o " です。

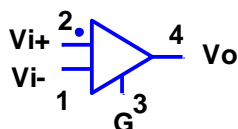
(4 4) ラッチ (Comparator:Latch)



入力トリガにより、出力電圧を保持する素子であり、シンボル (Symbol), オン電圧 (V_{on}), オフ電圧 (V_{off}) のパラメータがあります。ラッチはスイッチング周期の始めでは、必ずオフ状態になり、出力電圧 V_o は V_{off} となっています。そこで、周期内で入力差電圧 $V_i (= V_{i+} - V_{i-})$ が負から正に変化したときラッチは

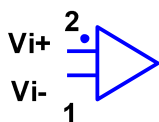
オフ状態からオン状態に変化し、 V_o が V_{off} から V_{on} となります。そして、その後 V_o は V_{on} を保持します。ラッチがオフ状態で、 V_i が正から負に変化した場合はオン状態にはなりません。次のスイッチング周期に入るとラッチは自動的にオフ状態となります。この素子は、主にピーク電流制御を実現するための素子であり、その利用法は、制御方法の項目を参照してください。オプションのアクティブ(Active)をオフにすると、ラッチは常にオフ状態となりオン状態へは変化しません。内部シンボルは、入力差電圧" V_i "、出力電圧" V_o "です。

(4 5) 単安定マルチバイブレータ(Comparator:Vibrator)



入力トリガ電圧により、矩形パルスが発生させる素子であり、シンボル(Symbol)、オン電圧(V_{on})、オフ電圧(V_{off})、パルス時間(T_{on})のパラメータがあります。通常状態では出力電圧は V_{off} であり、入力差電圧 $V_i (= V_{i+} - V_{i-})$ が負から正に変化したときトリガされ、電圧 V_{on} 、時間幅 T_{on} のパルスが発生します。ただし、メイン周期の開始時刻や V_i が正から負に変化したときはトリガされません。内部シンボルは、入力差電圧" V_i "と出力電圧" V_o "です。

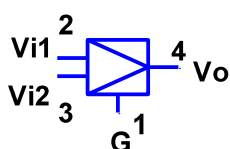
(4 6) トリガ(Comparator:Trigger)



入力トリガ電圧により、スイッチングを更新する素子であり、シンボル(Symbol)、最大時刻(T_{max})のパラメータがあります。入力差電圧 $V_i (= V_{i+} - V_{i-})$ が負のときオフ状態、正のときオン状態となります。スイッチング周期の始まりから T_{max} までに V_i が負から正に変化したときトリガは動作し、メイン

ウィンドウ主回路のメイン周波数とは無関係にスイッチングが更新され、次のスイッチング周期へ移行します。Viが正から負に変化したときは、トリガはオン状態からオフ状態に変化しますが、スイッチング周期の更新は行われません。また、Tmax までにViが負から正に変化しない場合は、Tmaxでスイッチングが更新され、この場合のスイッチング周期はTmaxとなります。オプションのアクティブ(Active)をオフにすると、トリガは常にオフ状態となり機能しません。この場合、主回路に設定されているスイッチング周波数による通常動作を行います。内部シンボルは、入力差電圧“Vi”です。

(47) 算術演算器 (Operator:Arithmetic)



算術演算器は、モードにより二つの入力電圧を加算(Add)、減算(Sub)、乗算(Mul)、除算(Div)、最大値(Max)、最小値(Min)を行う演算器であり、シンボル(Symbol)、ゲイン(Gain)、最小出力電圧(Vmin)、最大出力電圧(Vmax)、モード(Mode)があります。出力電圧Voは、

$$V_o = \text{Gain} * (V_{i1} \circ V_{i2})$$

$$V_o < V_{\min} \rightarrow V_o = V_{\min}$$

$$V_o > V_{\max} \rightarrow V_o = V_{\max}$$

Gain : ゲイン

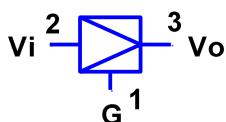
○ : 演算

で表されます。ただし、Maxは入力の大きい値、Minは小さい値を求めます。

この演算器は、平均サンプルホールド動作をします。各スイッチングの最後で入力電圧Vi1, Vi2の平均値がサンプルされ演算が行われます。結果の出力電圧Voは次のスイッチング周期の間ホールドされます。したがって、この演算器は変動周波数の高い信号には追従できず、入力電圧の変動周波数はスイッチン

グ周波数の10分の1以下が望ましいでしょう。内部シンボルは、入力電圧1 "Vi1", 入力電圧2 "Vi2", 出力電圧"Vo"です。

(48) 関数演算器 (Operator:Function)



関数演算器は、モードにより各種関数値を発生させる演算器であり、パラメータには、シンボル(Symbol), ゲイン(Gain), 最小出力電圧(Vmin), 最大出力電圧(Vmax), モード(Mode)があります。出力電圧 Vo は

$$V_o = \text{Gain} * F(V_i)$$

$$V_o < V_{\min} \rightarrow V_o = V_{\min}$$

$$V_o > V_{\max} \rightarrow V_o = V_{\max}$$

Gain : ゲイン

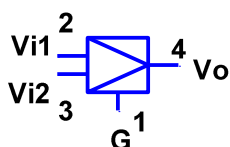
で表され、関数Fには以下の関数があります。

SIN	s i n 関数
COS	c o s 関数
TAN	t a n 関数
SIN-1	s i n 逆関数
COS-1	c o s 逆関数
TAN-1	t a n 逆関数
SQRT	平方根関数
EXP	e x p 指数関数
LOG	l o g 関数
ABS	絶対値関数 ($F(x) = x $)
AMP	比例関数 ($F(x) = x$)

この演算器は、平均サンプルホールド動作をします。各スイッチングの最後で

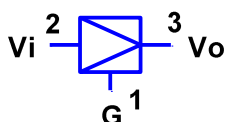
入力電圧 V_i の平均値がサンプルされ演算が行われます。結果の出力電圧 V_o は次のスイッチング周期の間ホールドされます。したがって、変動周波数の高い信号には追従できず、変動周波数はスイッチング周波数の10分の1以下が望ましいでしょう。内部シンボルは、入力電圧“ V_i ”，出力電圧“ V_o ”です。

(49) プログラマブル演算器 (Operator:Program)



プログラムにより V_{i1} と V_{i2} に対して任意の演算を行う演算器です。パラメータとして、シンボル(Symbol)，最小出力電圧(V_{min})，最大出力電圧(V_{max})があります。詳細は、制御方法の項目を参照してください。この演算器は、平均サンプルホールド動作をします。各スイッチング周期の最後で入力電圧 V_{i1} ， V_{i2} の平均値がサンプルされ演算が行われます。結果の出力電圧 V_o は次のスイッチング周期の間ホールドされます。したがって、この演算器は変動周波数の高い信号には追従できず、変動周波数はスイッチング周波数の10分の1以下が望ましいでしょう。内部シンボルは、入力電圧1“ V_{i1} ”，入力電圧2“ V_{i2} ”，出力電圧“ V_o ”です。

(50) テーブル演算器 (Operator:Table)



この演算器は、ユーザーが定義したテーブルにしがって演算を行い、パラメータには、(Symbol)，テーブルファイル(Table File)があります。この演算器を使用する場合、まず、テーブルファイルを作成します。適当なエディタ (メ

モ帳など) を起動し、カンマでくぎって、入力値, その入力値に対する出力値を書いていきます。たとえば、

1.0, 5.0

2.0, 10.0

3.0, 15.0

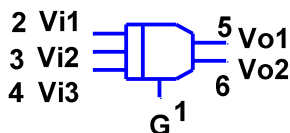
のように、書いていきます。次に、このデータをテキストファイルとして保存します。このファイルの拡張子は TBL に設定します。そこで、テーブル演算器の Table ボタンをクリックすると、ファイルダイアログボックスが現れるので、テーブルファイルを指定します。これで、準備完了です。設定したテーブルファイルの内容は、List ボタンをクリックすることにより、メモ帳で確認変更することができます。ただし、変更した場合はメモ帳で保存し、再度 Table ボタンで読み込んでください。この演算器は、テーブルファイルに定義された入出力関係 F にしたがって、

$$V_o = F(V_i)$$

の演算を行います。このとき、入力値が、テーブルファイルに定義されている入力値の中間値ときは線形補間を行います。たとえば、上の例で、実際の入力値 V_i が、1.5 の場合、出力電圧 V_o は、7.5 となります。また、入力値が、定義されている範囲をこえた場合は、定義されている最小値または最大値に対する出力値を出力します。たとえば、上の例で、入力値が 0.5 の場合、出力値は入力値 1.0 に対する出力値 5.0 となり、入力値が 4.0 の場合、出力値は入力値 3.0 に対する出力値 15.0 となります。

この演算器は、平均サンプルホールド動作をします。各スイッチング周期の最後で入力電圧 V_i の平均値がサンプルされ演算が行われます。結果の出力電圧 V_o は次のスイッチング周期の間ホールドされます。したがって、変動周波数の高い信号には追従できず、変動周波数はスイッチング周波数の 10 分の 1 以下が望ましいでしょう。内部シンボルは、入力電圧 " V_i ", 出力電圧 " V_o " です。

(5 1) 三相二相変換器 (Operator:DQC)



三相二相変換器には、パラメータとして、シンボル(Symbol), 周波数(Freq), 初期位相(Phini), モード(Mode)があり、以下のように三相入力を二相出力に変換します。

$$Vo1 = (\sqrt{2}/\sqrt{3}) \{Vi1 \cdot \cos(\omega t + \theta) + Vi2 \cdot \cos(\omega t - 2\pi/3 + \theta) + Vi3 \cdot \cos(\omega t + 2\pi/3 + \theta)\}$$

$$Vo2 = (\sqrt{2}/\sqrt{3}) \{Vi1 \cdot \sin(\omega t + \theta) + Vi2 \cdot \sin(\omega t - 2\pi/3 + \theta) + Vi3 \cdot \sin(\omega t + 2\pi/3 + \theta)\}$$

(Mode:Normal の場合)

$$Vo1 = (\sqrt{2}/\sqrt{3}) \{Vi1 \cdot \cos(\omega t + \theta) + Vi2 \cdot \cos(\omega t + 2\pi/3 + \theta) + Vi3 \cdot \cos(\omega t - 2\pi/3 + \theta)\}$$

$$Vo2 = (\sqrt{2}/\sqrt{3}) \{Vi1 \cdot \sin(\omega t + \theta) + Vi2 \cdot \sin(\omega t + 2\pi/3 + \theta) + Vi3 \cdot \sin(\omega t - 2\pi/3 + \theta)\}$$

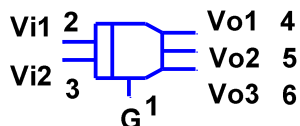
(Mode:Reverse の場合)

$$\omega = 2\pi f$$

f : 周波数, θ : 初期位相

この演算器は、平均サンプルホールド動作をします。各スイッチングの最後で入力電圧 Vi1, Vi2, Vi3 の平均値がサンプルされ演算が行われます。結果の出力電圧 Vo1, Vo2 は次のスイッチング周期の間ホールドされます。したがって、この演算器は変動周波数の高い信号には追従できず、入力電圧の変動周波数はスイッチング周波数の100分の1以下が望ましいでしょう。内部シンボルは、入力電圧1 "Vi1", 入力電圧2 "Vi2", 入力電圧"Vi3", 出力電圧1 "Vo1", 出力電圧2 "Vo2"です。

(5 2) 二相三相変換器 (Operator:DQI)



二相三相変換器には、パラメータとして、シンボル(Symbol), 周波数(Freq), 初期位相(Phini), モード(Mode)があり、以下のように二相入力を三相出力に変換します。

$$Vo1 = (\sqrt{2}/\sqrt{3}) \{Vi1 * \cos(\omega t + \theta) + Vi2 * \sin(\omega t + \theta)\}$$

$$Vo2 = (\sqrt{2}/\sqrt{3}) \{Vi1 * \cos(\omega t - 2\pi/3 + \theta) + Vi2 * \sin(\omega t - 2\pi/3 + \theta)\}$$

$$Vo3 = (\sqrt{2}/\sqrt{3}) \{Vi1 * \cos(\omega t + 2\pi/3 + \theta) + Vi2 * \sin(\omega t + 2\pi/3 + \theta)\}$$

(Mode:Normal の場合)

$$Vo1 = (\sqrt{2}/\sqrt{3}) \{Vi1 * \cos(\omega t + \theta) + Vi2 * \sin(\omega t + \theta)\}$$

$$Vo2 = (\sqrt{2}/\sqrt{3}) \{Vi1 * \cos(\omega t + 2\pi/3 + \theta) + Vi2 * \sin(\omega t + 2\pi/3 + \theta)\}$$

$$Vo3 = (\sqrt{2}/\sqrt{3}) \{Vi1 * \cos(\omega t - 2\pi/3 + \theta) + Vi2 * \sin(\omega t - 2\pi/3 + \theta)\}$$

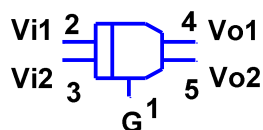
(Mode:Reverse の場合)

$$\omega = 2\pi f$$

f : 周波数, θ : 初期位相

この演算器は、平均サンプルホールド動作をします。各スイッチングの最後で入力電圧 Vi1, Vi2 の平均値がサンプルされ演算が行われます。結果の出力電圧 Vo1, Vo2, Vo3 は次のスイッチング周期の間ホールドされます。したがって、この演算器は変動周波数の高い信号には追従できず、入力電圧の変動周波数はスイッチング周波数の100分の1以下が望ましいでしょう。内部シンボルは、入力電圧1 "Vi1", 入力電圧2 "Vi2", 出力電圧1 "Vo1", 出力電圧2 "Vo2", 出力電圧3 "Vo3"です。

(5 3) 正相逆相変換器 (Operator:DQP)



正相逆相変換器には、パラメータとして、シンボル(Symbol)、周波数(Freq)、初期位相(Phini)があり、以下のように正相逆相を相互変換します。

$$Vo1=Vi1*\cos(2\omega t+\theta)+Vi2*\sin(2\omega t+\theta)$$

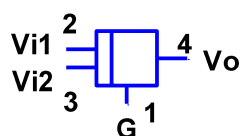
$$Vo2=Vi1*\sin(2\omega t+\theta)-Vi2*\cos(2\omega t+\theta)$$

$$\omega = 2 \pi f$$

f : 周波数, θ : 初期位相

この演算器は、平均サンプルホールド動作をします。各スイッチングの最後で入力電圧 Vi1, Vi2 の平均値がサンプルされ演算が行われます。結果の出力電圧 Vo1, Vo2 は次のスイッチング周期の間ホールドされます。したがって、この演算器は変動周波数の高い信号には追従できず、入力電圧の変動周波数はスイッチング周波数の 100 分の 1 以下が望ましいでしょう。内部シンボルは、入力電圧 1 "Vi1", 入力電圧 2 "Vi2", 出力電圧 1 "Vo1", 出力電圧 2 "Vo2" です。

(5 4) 論理演算器 (Logic:Logic)



論理演算器は、モード(Mode)により 2 入力の論理積(AND)、論理和(OR)、排他的論理和(XOR)、RS フリップフロップ(RS)を行う演算器であり、シンボル(Symbol)、しきい電圧(Vth)、オン電圧(Von)、オフ電圧(Voff)、モード(Mode)のパラメータがあり、オプションとして否定(NOT)があります。入力電圧 Vi1, Vi2 が Vth より大きいとその入力値は論理値 1 とみなされ、小さいと論理値 0

とみなされて演算されます。出力電圧 V_o は、演算結果により

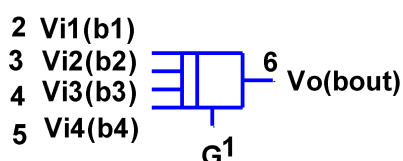
$$V_o = V_{on} \quad (\text{論理値 1 の場合})$$

$$= V_{off} \quad (\text{論理値 0 の場合})$$

で表されます。RS フリップフロップモードでは、禁止入力である V_{i1} , V_{i2} がともに論理値 1 の場合は、出力は変化しません。この素子はサンプルホールド動作ではありません。

また、NOT チェックボックスをチェックすることにより、出力の論理を否定することができます。たとえば、AND 回路は NOT をチェックすると NAND 回路になります。内部シンボルは、入力電圧 1 " V_{i1} ", 入力電圧 2 " V_{i2} ", 出力電圧 " V_o " です。

(5 5) ブール演算器 (Logic:Bool)



ブール演算器は、真理値表により 4 入力の論理演算を行う演算器であり、シンボル (Symbol), しきい電圧 (V_{th}), オン電圧 (V_{on}), オフ電圧 (V_{off}) のパラメータがあり、オプションとして否定 (NOT) があります。入力電圧 V_{i1} , V_{i2} , V_{i3} , V_{i4} が V_{th} より大きいとその入力値は論理値 1 とみなされ、小さいと論理値 0 とみなされて演算されます。出力電圧 V_o は、設定された真理値表にしたがって演算され、その演算結果により

$$V_o = V_{on} \quad (\text{論理値 1 の場合})$$

$$= V_{off} \quad (\text{論理値 0 の場合})$$

で表されます。この素子はサンプルホールド動作ではありません。

また、NOT チェックボックスをチェックすることにより、出力の論理を否定することができます。内部シンボルは、入力電圧 1 " V_{i1} ", 入力電圧 2 " V_{i2} ", 入力電圧 3 " V_{i3} ", 入力電圧 4 " V_{i4} ", 出力電圧 " V_o " です。

(56) DSP (DSP: DSP)



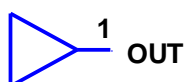
DSPの基本動作はプログラマブル制御器（PRC）と同じですが、プログラムがブロック図で作成できるので、プログラム作成経験のない技術者でも、非常に簡単にDSPプログラムが作成できます。詳細は、デジタル制御の項を参照してください。

(57) DSP初期ブロック(DSP: Initial)



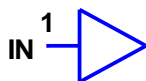
DSPプログラムの初期化用ブロックです。このブロックのプログラムはDSPプログラムの最初に実行されます。変数の初期化などに使用します。

(58) DSP入力ブロック(DSP: Input)



DSPプログラムの入力用ブロックです。主回路から検出した信号や、定数を出力させるプログラムなどを記述します。入力端子はなく、出力端子 OUT があります。

(59) DSP出力ブロック(DSP: Output)



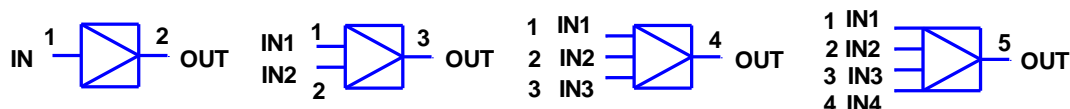
DSPプログラムの出力用ブロックです。DSPプログラムで生成した信号を入力で受取り、主回路に伝達するプログラムなどを記述します。出力端子はなく、入力端子 IN があります。

(60) DSP 検出ブロック (DSP: Detector)



DSPプログラムの信号検出用ブロックです。このブロックを他のブロックの出力端子に接続すると、その接続点信号は、メインメニューの Parameter:Output で出力変数として設定できます。DSPプログラムの内部動作の確認に使用します。

(61) DSP 機能ブロック (DSP: Block1, Block2, Block3, Block4)



DSPプログラムの機能用ブロックです。単純な機能を表すプログラムを記述します。1入力、2入力、3入力、4入力のブロックがあります。各端子名は図のとおりです。

(62) 接続線 (Connection: Wire, Half Wire)

Wireは、接続線を表します。図形全体が導体であり、任意の部分に接続することができますが、Wireを交差させた場合、その交差点は接続とはみなされません。Wireは、ハンドルをドラッグすることにより水平または垂直方向に伸縮

することができます。また、スペースキーを押しながらドラッグすることにより任意の場所に描くことができます。また、Half Wireを選択すると半分の長さのWireが得られます。

(6 3) 自由接続線 (Connection:Free Wire)

Free Wireは、Wireと違って伸縮する方向は自由であること以外は、Wireと同じです。

(6 4) ジャンパー線 (Connection:Jumper)



Jumperは、両端のみ導体で他の部分が絶縁体の線です。接続線をまたぐことを明示するときに使用します。

(6 5) 遠ジャンパー線 (Connection:Far Jumper)



Far Jumper は遠く離れた端子を接続します。複数のこの素子に対し、同じシンボルを設定するとそれらの端子は接続され、シンボルが異なる場合には接続されません。

(66) グランド端子 (Connection:Ground)



Groundは、グランド端子であり、すべてのグランド端子は接続されているとみなされます。スペースキーを押しながら右クリックすることでも表示できます。

(67) グランド端子 2 (Connection:Ground2)



Ground2は、第2のグランド端子であり、すべてのグランド端子2は接続されているとみなされます。Ground2 は Ground とは別のグランドを定義します。

(68) オープン素子 (Connection:Open)



Openは、絶縁素子であり、どこに接続しても回路には影響しません。端子間の電圧を測定するときに使用します。パラメータとして、シンボル(Symbol)があります。

(69) ターミネータ (Connection:Terminator)



ターミネータは、使用しない接続端子を終端するための素子です。

(70) カスタム端子 (Cmt Terminal)

*CTM1



カスタム回路の入出力端子を決める素子です。

(71) テキスト (Others:Text)

テキストボックスに入力することによりテキストを表示できます。テキストも図形として扱われます。

3. 2 素子パラメータの設定 (Parameter:Device)

回路図を作成した直後、各素子のパラメータを設定するには、Parameter:Deviceが便利です。このコマンドを実行すると、各素子のパラメータダイアログボックスを順次表示します。このとき、OKボタンをクリックすると次の素子のパラメータダイアログボックスを表示し、Cancelボタンをクリックすると終了します。ただし、キャンセルした場合、これまでに設定した素子のパラメータ値は有効でありキャンセルできません。

素子パラメータは、テンプレート化できます。テンプレート化すると、別な回路の同じ素子に対して、共通のパラメータ設定が可能となります。その使用方法は、以下のとおりです。

(1) テンプレートの作成

まず、一度、テンプレート化する適当なパラメータ値を素子に設定します。半導体であれば、カタログ値を利用すれば良いでしょう。次に、その素子を、Ctrlキーを押しながら右クリックします。すると、テンプレート作成ダイアログボックスが表示されますので、Name入力ボックスに、適当なテンプレート名を入力します。たとえば、半導体名などを入力します。次に、Makeボタンをクリックすると、Listボックスに、テンプレート名が表示され、そのテンプレート名によって、素子の現在のパラメータ値がK. 492のデータベースに登録されま

す。Showボタンで、Listボックスのハイライト表示されているテンプレートの内容を表示または変更でき、Deleteボタンで、削除することができます。また、登録されているテンプレートは、K. 492の終了時にK. 492と同じディレクトリにある“SCAT.SDB”というファイルに保存され、次の実行時に自動的に読み込まれます。

(2) テンプレートによるパラメータ設定

(1) で作成したテンプレートを使用するには、素子を、Shiftキーを押しながら右クリックします。すると、テンプレート設定ダイアログボックスが表示されるので、Listボックスの設定したいテンプレートをクリックして選択し、OKボタンを押せば、素子のパラメータ値がテンプレート値に設定されます。ただし、素子シンボルは変更されません。Showボタンでテンプレート内容を表示できますが、変更はできません。変更するには、(1) のテンプレート作成ダイアログボックスを使用します。

3. 3 スイッチング周波数の設定 (Parameter:Frequency)

スイッチング周波数の設定は、Parameter:Frequencyを利用します。このコマンドを実行すると周波数ダイアログボックスが表示されるので、Frequency入力ボックスで、スイッチング周波数を入力します。このスイッチング周波数は、Transient解析においては、波形のサンプリング周波数と同じ意味を持ちます。

3. 4 出力変数の設定 (Parameter:Output)

出力変数とは解析の出力を指定する変数であり、素子の電圧、電流が設定できます。ただし、次の変数はK492 で出力変数として設定できません。

[1] 電圧源の電流

[2] 損失のない変圧器巻線の電流

これらの電流を観測するには、これらの素子に直列に寄生損失などの素子を付加し、その電流を観測します。また、電圧に関しては、基本的に素子の電圧しか観測できず、複数の素子間の電圧は直接観測できません。このような場合、

オープン素子を使用します。まず、回路図作成のときに、観測したい端子間にオープン素子を付加し、このオープン素子の電圧を出力変数に設定します。

出力変数を設定するには次の三つのパラメータを指定する必要があります。

(1) シンボル

出力変数に設定できるのは素子の電圧か電流です。その素子シンボルを指定します。このとき、インダクタや抵抗などの単純な素子は、素子シンボル自体を指定します。スイッチやダイオードなどの内部シンボルを有する素子は内部シンボルを指定します。

(2) タイプ

電圧か電流を次のように指定します。

[1] V : 電圧

[2] I : 電流

(3) モード

モードを設定することにより、どの点を一周期の代表点とするかを指定します。たとえば、過渡応答解析を考えましょう。このとき、非常に多くの周期分の計算が行われますが、これを後述するWVFウィンドウに表示する場合、各周期の詳しい波形をそのまま表示すると非常に見難くなるし、また、波形データのメモリ消費も多くなります。そこで、K492 では、これらの問題を回避するため、各周期に代表点を一つ設けてそれを表示するようになっています。代表点として指定できるのは次の通りです。

[1] Ave : 平均値

[2] Max : 最大値

[3] Min : 最小値

[4] Ini : 初期値

[5] Fin : 最終値

[6] Pkv : ピーク値

ここで、ピーク値 Pkv とは、最大値と最小値のうち絶対値の大きい方の値を意味します。このモードは、Analysis:Transient など過渡解析に関係した解析全てに使用されます。

また、オプションとして出力変数の制限値を設定できます。Limit ボタンをクリックし、最小値と最大値を設定します。シミュレーション時にこの制限値の範囲を超えると出力変数の素子の色が、View:Option:Default の Limit Color で設定した色に変わります。

次に、出力変数の設定方法について説明します。Parameter:Output をクリックすると出力変数ダイアログボックスが現れます。このとき、WVF ウィンドウが開いていれば自動的に閉じます。現在設定されている出力変数は、Outputs リストボックスに表示されます。

出力変数を追加するには、Parameters リストボックスで追加する出力変数に対するシンボルをクリックして指定し、Type ラジオボタンと Mode ラジオボタンでそれぞれ出力変数のタイプとモードを設定します。その後、Set ボタンをクリックすれば出力変数が追加されます。指定したシンボルによっては、タイプが選択できないことがあります。

設定されている出力変数を削除するには、Outputs リストボックスで削除する出力変数を指定し、その後、Delete ボタンをクリックします。このとき、削除しようとしている出力変数が演算出力変数の演算対象になっている場合、“Cannot delete an operation operand.” という警告メッセージが現れ削除できません。この場合、まず、その演算出力変数を削除し、その後、目的の出力変数を削除します。演算出力変数については以下の説明を参照してください。

加算 '+' ボタンを利用することにより、波形に対する加算ができます。手順は以下の通りです。

まず、Outputs リストボックスより、被加算波形をクリックして選択します。次に、 '+' ボタンをクリックします。このとき、 '+' ボタンの表示が C(Cancel) に変わります。つぎに、加算波形を Outputs リストボックスよりクリックして

選択します。加算できる波形は、被加算波形と同じタイプ（電圧V，電流I）を持つ波形のみです。加算した結果が演算出力変数として、Outputs リストボックスに付け加わります。この演算出力変数の名前は、“%1+2”のように、最初に“%”が付き、その後に加算波形のリスト上の番号と加算記号が組み合わさって付けられます。’+’ ボタンをクリックした段階（表示 C）でもう一度このボタンをクリックすると演算はキャンセルされます。同様に、’-’，’*’，’/’ ボタンを利用することにより減算、乗算、除算ができます。

3. 5 解析条件の設定 (Parameter:Condition)

Parameter メニューの Condition で、刻み時間や低周波近似補正に関する解析パラメータが設定できます。設定できるパラメータは以下のとおりです。

(1) Nta

減衰波形に対する刻み時間を制御します。増加するほど解析精度が良くなりますが、解析時間は長くなります。減衰波形が正しくないとき、1~100 の範囲で増加してください。

(2) Ntw

正弦波形に対する刻み時間を制御します。増加するほど解析精度が良くなりますが、解析時間は長くなります。振動波形が正しくないとき、100~1000 で増加してください。

(3) Fta

刻み時間の繰り返し時間を制御します。増加するほど解析精度が良くなりますが、解析時間は長くなります。パルス波形が正しくないとき、10~20 で増加してください。

これらのパラメータは、通常、変更する必要はありません。特別に精度が必要な場合増加させてください。デフォルト値より減少させるのは推奨できません。

(4) Compensate

Transient, Waveform 解析時の PWM, SFM の制御に対する低周波近似補正を行うかどうかを決めます。チェックすれば行われ、チェックしなければ行われません。通常、アナログ制御の場合はチェックし、デジタル制御の場合はチェックしません。FRD を使用した場合はチェックできません。

4. 表示ウィンドウ

本章では、K492 で使用される表示ウィンドウについて説明します。K492 には、いろいろな情報を表示するために以下のようなウィンドウがあります。

4. 1 メインウィンドウ (Window:Analysis, Window:Topology)

既に説明したとおりメインウィンドウには、解析モード(Analysis)、図形モード(Topology)の二つのモードがあります。解析モードは、解析を行うモード、図形モードは回路の作成変更を行うモードです。これらの、モードを変更するには、Window:Analysis と Window:Topology を利用します。図形モードに入ると、それまで行った解析結果は全てクリアされます。したがって、Window:Topology を実行すると確認メッセージが表示されます。また、図形モードから出るときは、回路接続チェック、パラメータチェック、トポロジー解析などが行われ、正常に終了すれば“Topology is updated.”と表示されます。

解析モードにおいて、回路図表示以外のところで右クリックするとクリック点にフローティングメニューが表示されます。

メインウィンドウのタイトルバーには、以下の情報が表示されます。

- [1] バージョン名
- [2] 回路データのファイル名
- [3] メインスイッチング周波数
- [4] メインウィンドウモードまたは解析名
- [5] 解析時間
- [6] 現在の状態数
- [7] 状態解析繰り返し数／状態解析繰り返し予定数
- [8] 定常解析収束率
- [9] 状態解析失敗総数／状態解析失敗サイクル総数

4. 2 状態ウィンドウ (Window:Window:State)

状態ウィンドウは、スイッチやダイオードなどの状態変化素子の一周期における状態表示を行います。状態ウィンドウを表示させるには、Window:Window:State をクリックします。すると、画面左下の定位置に状態ウィンドウが表示され、メニューにはチェックマークが付き、状態ウィンドウが開いた状態で Window:Window:State を実行すると状態ウィンドウは最小化され、メニューのチェックマークが消えます。さらに、最小化された状態で Window:Window:State を実行すると、以前の位置に状態ウィンドウが開きます。状態ウィンドウを閉じるには、Windows の操作方法に従ってください。

状態ウィンドウには、状態解析の結果が表示されます。このウィンドウはスイッチやダイオードなど状態変化素子のオン(ON)・オフ(OFF)状態表示のほか次のような情報を表示します。

(1) ダイオードの逆回復状態

ダイオードの最大蓄積電荷を正の値に設定した場合、逆方向回復特性が解析され、逆方向回復状態を REC で表示します。

(2) 可飽和インダクタの飽和状態

可飽和インダクタの飽和状態を、正バイアスで飽和した場合には ON+ で、負バイアスで飽和した場合には ON- で表示します。

(3) くり返し回数

最初の行に状態解析くり返し数が表示されます。

(4) 収束率

最初の行に定常解析における収束率が%で表示されます。この値は厳密なものではないので目安程度と考えてください。

(5) 状態解析情報

状態決定に何等かの異常があった場合に、状態番号の左横およびその行の最右端に以下の記号を表示し異常を知らせます。異常がなければ何も表示されません。

[1] S : 状態決定失敗

[2] T : 状態変化時刻決定失敗

これらの詳細は解析アルゴリズム項目を参照してください。

(6) 状態時間

各状態の持続時間が、“Ti”として右端に表示されます。このTiがスイッチング周期の1 / 1000以上の場合は、表示色が View:Option:State の Font Color になり、そうでなければ Shadow Color になります。

(7) 状態変化時刻

状態変化時刻が、“Tt”として右端に表示されます。

Shift キーを押しながら一つの状態をクリックすると、クリックした状態だけがハイライト表示され、Ctrl キーを押しながらクリックするとハイライト表示された状態が追加されます。これらのキーを押しながら右クリックすると全て通常表示に戻ります。ハイライト色は、View:Option:State の Mark Color で設定します。

Alt キーを押しながら状態をクリックすると、メインウィンドウの OFF 状態の素子色が変わりその状態が反映されます。Switch, PWM Switch, Diode, Thyristor の素子色は View:Option:Default の OFF1 Color で設定し、Pulse, Sat Inductor, Comparator, Latch, Trigger, Logic, Bool は OFF2 Color で設定します。さらに、このときWVFウィンドウにスイッチング一周期の波形が表示されている場合には、WVFウィンドウに以下のマーカー表示が行われます。

[1] マーカーモードの場合、マーカーがクリックされた状態の開始時刻に移動する。

[2] ノーマルモードまたはマーカーモードの範囲モードであれば、マーカーがクリックされた状態の開始時刻と終了時刻に移動する。

状態解析において、状態数が制限値(1000)に達すると状態解析を打ち切り、メッセージボックスによって解析を中止するか、継続して次の状態解析に移るかを聞いてきます。

状態ウィンドウは、閉じておいたほうが解析速度は向上します。特に、過渡応答解析や波形解析では、解析が確実に行われていれば閉じておいた方がよいでしょう。

状態ウィンドウのオプションは View:Option:State で設定でき、以下のパラメータがあります。

[1] Font Size : 文字の大きさを設定します。

[2] Font Color : 文字色を設定します。

[3] Shadow Color : 期間が短い状態の文字色を設定します。

[4] Mark Color : ハイライト色を設定します。

[5] Error Trap : 状態解析エラー時中断用フラグを設定します。

このフラグをチェックすると状態解析エラー時に解析が中断します。

4. 3 WVF ウィンドウ, コントロールパネル (Window:Window:WVF)

K492 はオシロスコープをシミュレーションするWVF ウィンドウと呼ばれる強力なウィンドウを持っています。WVF ウィンドウとそのコントロールパネルは最初に自動的に表示されます。閉じた状態でWVF ウィンドウを表示させるには、Window:Window:WVF をクリックします。すると、画面右下の定位置にWVF ウィンドウが表示され、メニューにはチェックマークがつきます。WVF ウィンドウが開いた状態で Window:Window:WVF を実行するとWVF ウィンドウは最小化され、メニューのチェックマークが消えます。さらに、最小化され

た状態で Window:Window:WVF を実行すると、以前の位置にWVF ウィンドウが開きます。WVF ウィンドウを閉じるには、右上にあるクローズボタンをクリックします。

K492 は波形表示用のWVF ウィンドウとその波形を操作するコントロールパネルを別々に持っています。コントロールパネルが消えた状態で、カーソルをWVF ウィンドウ内に移動し、右クリックするとコントロールパネルが現れます。コントロールパネルには次のような機能があります。

(1) 出力変数リスト (Outputs)

出力変数を表示します。反転表示されている出力変数がコントロールパネルの操作の対象になり、これを変更するには、対象とする出力変数をクリックします。Shift キーを押しながら、WVF ウィンドウの波形をクリックしても同様なことができます。このとき、タイトルバーには、操作対象の出力変数名が表示されます。

また、Alt キーを押しながら、これを行うとクリックした出力変数に対応するメインウィンドウの素子色がハイライト色に変わります。出力変数を単にクリックすると元に戻ります。

逆に Alt キーを押しながらメインウィンドウ上の素子をクリックすると、出力変数の中で、クリックした素子に関する出力変数のみをこの出力変数リストに表示することができます。Alt キーを押しながら素子を右クリックすると元に戻ります。

また、Shift キーを押しながら、出力変数をクリックすると、ダイアログボックスが現れ、それにより出力変数リストで選択されている出力変数の範囲および線の太さと色を変更できます。

さらに、Ctrl キーを押しながら、出力変数をクリックすると、その出力変数の制限値が出力変数データリストに表示され、WVF ウィンドウにその制限値ラインが表示されます。出力変数を単にクリックすると元に戻ります。

(2) 出力変数データリスト

出力変数リストのすぐ下に対象出力変数に関する以下のデータを表示します。

- [1] Max : Y軸最大値
- [2] Min : Y軸最小値
- [3] Div : 軸1目盛りの値
- [4] Ave : 平均値
- [5] Rms : 実効値

これらのパラメータのなかで、Max と Min は任意の値に変更できます。これらの値を変更するには、変更する項目をクリックします。すると、Value 入力ボックスが現れるので、新しい値を入力します。また、平均値(Ave)と実効値(Rms)はWV F ウィンドウの表示範囲で計算されます。

(3) X軸変数データリスト (X-Axis)

X軸変数に関する以下のデータを表示します。

- [1] Var : X軸変数名
- [2] Max : X軸最大値
- [3] Min : X軸最小値
- [4] Div : X軸1目盛りの値

Max, Min は任意の値に変更できます。また、X軸変数は、通常、時間（シンボルト）ですが、X軸変数名をクリックすると、出力変数リストの対象出力変数がX軸変数になり、オシロスコープのX-Y機能が実現できます。もとの時間変数に戻すにはX軸変数名をもう一度クリックします。X-Y機能を利用した場合、出力変数データリストの平均値(Ave)と実効値(Rms)は無効です。

(4) DC/AC ボタン (DC/AC)

DC モードと AC モードがあります。DC モードでは、対象出力変数の直流分（平均値）と交流分の和の波形表示を行い、AC モードでは、対象出力変数の直流分を除去して交流分のみの波形表示を行います。デフォルトはDCモードにな

っており、このボタンをクリックすることにより切り替えが行われ、現在のモードがボタンに表示されます。また、Alt キーを押しながらこれを実行すると、全ての出力変数に対して実行されます。

(5) Positive/Negative ボタン (P/N)

Positive モードと Negative モードがあります。Positive モードでは、回路で定義された極性で表示を行い、Negative モードでは、定義された極性と逆極性で表示を行います。デフォルトは Positive モードになっており、このボタンをクリックすることにより切り替えが行われ、現在のモードがボタンに表示されます。また、Alt キーを押しながらこれを実行すると、全ての出力変数に対して実行されます。

(6) Widen ボタン (Wdn)

このボタンをクリックすることにより、対象出力変数に対する Y 軸範囲を広げます。したがって、相対的に波形は小さくなります。また、Alt キーを押しながらこれを実行すると、全ての出力変数に対して実行されます。

(7) Narrow ボタン (Nrw)

このボタンをクリックすることにより、対象出力変数に対する Y 軸範囲を狭めます。したがって、相対的に波形は大きくなります。また、Alt キーを押しながらこれを実行すると、全ての出力変数に対して実行されます。

(8) X 軸 Adjust ボタン (AdX)

WVF ウィンドウのズーム機能などにより X 軸範囲を変更した場合、表示波形がとぎれたりスコープ範囲内に表示しきれなかったりします。このボタンをクリックすると波形が WVF ウィンドウ全体に合わせて表示されます。

(9) Y 軸 Adjust ボタン (AdY)

このボタンをクリックすると表示波形がWVFウィンドウ内に入るようにY軸範囲が調整されます。また、Alt キーを押しながらこれを実行すると、全ての出力変数に対して実行されます。

(1 0) FFT解析ボタン (FFT)

このボタンをクリックすると対象出力変数のFFT解析が実行され、FFTウィンドウとコントロールパネルが表示されます。FFTウィンドウについては、別項を参照してください。

(1 1) CMPウィンドウボタン (CMP)

このボタンをクリックするとCMPウィンドウが開き、対象出力変数波形がCMPウィンドウに登録されます。CMPウィンドウには、WVFウィンドウの波形であれば、任意の波形が登録でき、パラメータの違いによる波形比較などが容易にできます。CMPウィンドウについては、別項を参照してください。

(1 2) Display ボタン (Dsp)

このボタンをクリックすることにより、波形の表示をするか否かを切り替えます。表示をしない場合、出力変数リストの出力変数番号の代わりに“#”が付き表示されます。一時、波形を消したい場合に使用します。また、Alt キーを押しながらこれを実行すると、全ての出力変数に対して実行されます。

(1 3) 高調波解析ボタン (HA)

このボタンをクリックすると高調波解析モードになります。高調波解析については、高調波解析の項目を参照してください。このボタンはトグルになっており、もう一度クリックすると通常モードに戻ります。

(14) 波形演算ボタン (+, -, *, /)

このボタンを利用することにより、波形に対する四則演算ができます。手順は Parameter:Output と同じです。演算出力変数を削除するには WVF ウィンドウの Outputs リスト上で Del キーを押しながらクリックします。

(15) 縦スクロールバー

縦スクロールバーをスクロールすることにより、対象出力変数の零レベルを上下に移動することができます。

(16) マーカー表示ボタン (Mkr)

このボタンをクリックするとマーカーモードになりマーカーが表示されます。このとき、出力変数データリストには、対象出力変数のマーカー現在点の Y 軸値 (Val)、開始点から現在点までの平均値 (Ave)、開始点から現在点までの波形面積を X 軸範囲で割った部分平均値 (Ave (part))、開始点から現在点までの実効値 (Rms) が表示され、X 軸変数データリストには、X 軸値 (Val) とインデックス番号 (Index) が表示されます。さらに、WVF ウィンドウのタイトルには、X 軸値と Y 軸値が表示されます。もう一度クリックすると通常モードに戻ります。また、コントロールパネルが表示されていれば、WVF ウィンドウを右クリックしても、このボタンと同じことができます。

一周期の波形が表示されているとき、Alt キーを押しながら波形をクリックすると、メインウィンドウの OFF 状態の素子色が変わり、クリック点の状態が反映されます。このとき状態ウィンドウが表示されている場合には、状態ウィンドウの対応する状態がハイライト表示されます。これらの表示を元に戻すには、Alt キーを押しながら右クリックします。

(17) マーカーセットボタン (Set)

マーカーモードにおいて、このボタンをクリックするとマーカー範囲モードになり、マーカーがセットされ、ボタン表示が Clr (Clear) となります。もう

一度クリックするとクリアされ、表示が Set となります。マーカー範囲モードで、マーカーを移動すると、出力変数データリストには、セット点から現在点の間の Y 軸値差 (Dif)、平均値 (Ave)、部分平均値 (Ave(part))、実効値 (Rms) が表示され、X 軸変数データリストには X 軸値差 (Dif) とインデックス番号差 (Index) が表示されます。

マーカー範囲モードで一周期の波形が表示されているとき、Alt キーを押しながら波形をクリックすると、クリック点を含む状態の開始点と終了点にマーカーが移動し、上記と同様な表示が行われます。

(18) ピークボタン (Pk)

マーカーモードにおいて、このボタンをクリックするとマーカーが WVF ウィンドウの表示範囲内において、対象出力変数の最大値または最小値に交互に移動します。

(19) ズームボタン (Zm)

マーカー範囲モードで、このボタンをクリックすると、WVF ウィンドウの X 軸範囲がセット点と現在点の範囲に設定されます。これを解除するには、AdX ボタンを利用します。

また、WVF ウィンドウ内でドラッグすると矩形枠が表示され、その矩形枠 X 軸範囲でズームが行われます。Y 軸のズームは行われません。このとき、Alt キーを押しながらこの機能を使うと、表示範囲がスイッチング周期の整数倍となります。このドラッグによるズーム機能は、通常モードでも有効です。

(20) 横スクロールバー

マーカーモードにおいて、この横スクロールバーをスクロールすることにより、マーカーを移動することができます。また、WVF ウィンドウ内でクリックすると、その X 座標値にマーカーが移動します。

(21) 表示線オプション(View:Option:Scope)

View:Option:Scope を利用すると、スコープウィンドウの各表示線と目盛りの大きさと色を変更することができます。この設定は、WVF, FFT, FRA, CMP ウィンドウ共通で、設定できる表示線は以下のとおりです。

[1] Selected Line : 出力変数リストで選択されている波形線

[2] Frame Line : 外枠線

[3] Scale Line : スケール線

[4] Scale Font : 目盛り

[5] Zero Line : 零レベル線

[6] Marker Line : マーカー線

[7] Limit Line : 高調波規制直線 (FFTのみ)

[8] Background : 背景色

対象出力変数の線の幅と色は、通常、前述の出力オプションで設定された幅と色に関係なく、Selected Line で設定された幅と色になります。ただし、Selected Line の幅が0の場合は、その幅は自身の設定幅となり、同様に、Selected Line の色が None の場合は、その色は自身の設定色になります。また、スケール線と零レベル線の幅が1のときのみ、それらは点線となり、1以上の場合は連続線となります。また、背景色は、設定以後に生成されたスコープウィンドウに適用されます。

4.4 FFTウィンドウ, コントロールパネル(Window:Window:FFT)

K492 は、FFT用のスコープウィンドウであるFFTウィンドウとそのコントロールパネルを持っています。このFFTウィンドウにおいてFFT解析ができます。FFTウィンドウを開くには、WVFウィンドウのコントロールパネルのFFT ボタンをクリックします。すると、直ちにWVFウィンドウのコントロールパネルで指定されている対象出力変数 (WVFウィンドウにおいて選択されている波形) に対するFFT解析が瞬時に実行され、各周波数成分の実

効値振幅が棒グラフ状に表示されます。このとき、波形のサンプリングはWV F ウィンドウの表示範囲で行われ、サンプリング数は 1024 点です。したがって、WV F ウィンドウの X 軸範囲の逆数が F F T 解析結果の基本周波数となり、各周波数成分はこの整数倍になります。

F F T ウィンドウが開いた状態で Window:Window:FFT を実行すると F F T ウィンドウは最小化され、メニューのチェックマークが消えます。さらに、最小化された状態で Window:Window:FFT を実行すると、以前の位置に F F T ウィンドウが開きます。F F T ウィンドウを閉じるには、右上のクローズボタンをクリックします。

F F T ウィンドウ内で右クリックするとコントロールパネルが現れます。以下に、コントロールパネルの機能について説明します。

(1) 出力変数リスト (Output)

F F T 解析を行った出力変数を表示します。Shift キーを押しながら、出力変数をクリックすると、ダイアログボックスが現れ、それにより出力変数リストで選択されている出力変数の範囲および線の太さと色を変更できます。

(2) 出力変数データリスト

出力変数リストのすぐ下に出力変数に関する以下のデータを表示します。

- [1] Max : Y 軸最大値
- [2] Min : Y 軸最小値
- [3] Div : Y 軸 1 目盛りの値
- [4] Thd : 全高調波歪率 (%)

これらのパラメータのなかで、Max と Min は任意の値に変更できます。これらの値を変更するには、変更する項目をクリックします。すると、Value 入力ボックスが現れるので、新しい値を入力します。

(3) 周波数データリスト (Frequency)

X軸変数の周波数に関する以下のデータを表示します。

[1] Max : X軸最大値

[2] Min : X軸最小値

[3] Div : X軸1目盛りの値

Max, Min は出力変数と同様に任意の値に変更できます。

(4) X軸 Widen ボタン (WdX)

このボタンをクリックすることにより、X軸範囲を広げます。

(5) X軸 Narrow ボタン (NrX)

このボタンをクリックすることにより、X軸範囲を狭めます。

(6) Y軸 Widen ボタン (WdY)

このボタンをクリックすることにより、Y軸範囲を広げます。

(7) Y軸 Narrow ボタン (NrY)

このボタンをクリックすることにより、Y軸範囲を狭めます。

(8) 振幅位相切り替えボタン(Amp/Phs)

表示内容を振幅 (Amp) または位相 (Phs) に交互に切り替えます。デフォルトは振幅です。K492 の F F T は cos 変換であり、一般式は、

$$v = \sum_{n=0} V_n \cos(n\omega t + \theta_n)$$

で表されます。この式の V_n が振幅、 θ_n が位相を表します。

(9) X軸 Adjust ボタン (AdX)

F F T ウィンドウのズーム機能などにより X 軸範囲を変更した場合、このボ

タンをクリックすると波形が変更前の表示に戻ります。

(10) Y軸 Adjust ボタン (AdY)

このボタンをクリックすると表示波形がFFTウィンドウ内に入るようにY軸範囲が調整されます。

(11) 縦スクロールバー

縦スクロールバーをスクロールすることにより、出力変数の零レベルを上下に移動することができます。

(12) マーカー表示ボタン (Mkr)

このボタンをクリックするとマーカーモードになりマーカーが表示されます。このとき、出力変数データリストには、出力変数のマーカー点のY軸値 (Val) が表示され、振幅表示の場合は基本波に対する相対値 (HR) も百分率で表示されます。周波数データリストには、マーカー点の周波数 (Val) と基本周波数に対する高調波次数 (Index) が表示されます。さらに、FFTウィンドウのタイトルには、X軸値とY軸値が表示されます。このボタンをもう一度クリックすると通常モードに戻ります。通常モードにおいて、コントロールパネルが表示されていれば、FFTウィンドウ内を右クリックしても、このボタンと同じことができます。

(13) マーカーセットボタン (Set)

マーカーモードにおいて、このボタンをクリックするとマーカー範囲モードとなり、表示がClr (Clear) となります。もう一度クリックするとクリアされ、表示がSet となります。マーカー範囲モードで、マーカーを移動すると、出力変数データリストにはセット点から現在点の間のY軸値差 (Dif), 周波数データリストにはX軸値差 (Dif) と高調波次数差 (Index) が表示されます。

(14) ピークボタン (Pk)

マーカーモードにおいて、このボタンをクリックするとマーカーが F F T ウィンドウの表示範囲内において、対象出力変数の最大値または最小値に交互に移動します。

(15) ズームボタン (Zm)

マーカー範囲モードにおいて、このボタンをクリックすると、F F T ウィンドウの周波数範囲がセット点と現在点の範囲に設定されます。これを解除するには、AdX ボタンを利用します。また、F F T ウィンドウ内でドラッグすると矩形枠が表示され、その矩形枠 X 軸範囲でズームが行われます。Y 軸のズームは行われません。このドラッグによるズーム機能は、通常モードでも有効です。

(16) 横スクロールバー

マーカーモードにおいて、この横スクロールバーをスクロールすることにより、マーカーを移動することができます。また、F F T ウィンドウ内でクリックすると、その X 座標値にマーカーが移動します。

(17) 表示線オプション (View:Option:Scope)

View:Option:Scope を利用すると、スコープウィンドウの各表示線と目盛りの大きさと色を変更することができます。この設定は、W V F, F F T, F R A, C M P ウィンドウ共通で、W V F ウィンドウの項目を参照してください。

4.5 F R A ウィンドウ, コントロールパネル (Window:Window:FRA)

K492 は、周波数特性解析用のスコープウィンドウである F R A ウィンドウとそのコントロールパネルを持っています。F R A ウィンドウは、周波数特性解析 Analysis:Freq Response を実行すると自動的に画面に表示されます。F R A ウィンドウが開いた状態で Window:Window:FRA を実行すると F R A ウィンドウ

は最小化され、メニューのチェックマークが消えます。さらに、最小化された状態で Window:Window:FRA を実行すると、以前の位置に F R A ウィンドウが開きます。F R A ウィンドウを閉じるには、右上のクローズボタンをクリックします。

F R A ウィンドウの位相表示は、まれに 3 6 0 度を基準に表示される場合がありますが、これは 0 度基準と同じです。

F R A ウィンドウ内で右クリックするとコントロールパネルが現れます。以下に、コントロールパネルの機能について説明します。

(1) 出力変数リスト (Output)

利得(Gain)と位相(Phase)の出力変数を表示します。Shift キーを押しながら、出力変数をクリックすると、ダイアログボックスが現れ、それにより出力変数リストで選択されている出力変数の範囲および線の太さと色を変更できます。

(2) 出力変数データリスト

出力変数リストのすぐ下に出力変数に関する以下のデータを表示します。

- [1] Max : Y 軸最大値
- [2] Min : Y 軸最小値
- [3] Div : Y 軸 1 目盛りの値

これらのパラメータのなかで、Max と Min は任意の値に変更できます。これらの値を変更するには、変更する項目をクリックします。すると、Value 入力ボックスが現れるので、新しい値を入力します。また、利得はの単位は dB で、位相の単位は度です。

(3) 周波数データリスト (Frequency)

X 軸変数の周波数に関する以下のデータを表示します。

- [1] Max : X 軸最大値
- [2] Min : X 軸最小値

Max, Min は出力変数と同様に任意の値に変更できます。

(4) Y軸 Widen ボタン (Wdn)

このボタンをクリックすることにより、Y軸範囲を広げます。

(5) Y軸 Narrow ボタン (Nrw)

このボタンをクリックすることにより、Y軸範囲を狭めます。

(6) X軸 Adjust ボタン (AdX)

このボタンをクリックすると波形がF R Aウィンドウ全体に合わせて表示されます。

(7) Y軸 Adjust ボタン (AdY)

このボタンをクリックすると表示波形がF R Aウィンドウ内に入るようにY軸範囲が調整されます。

(8) 利得余裕ボタン (GaM)

このボタンをクリックすると、位相0度の位置にマーカーが移動し、その位置の利得を出力変数データリストに表示します。位相が0度に達していなければ“Not defined”と表示されます。一巡伝達特性の利得余裕を求めるときに使用します。

(9) 位相余裕ボタン (PhM)

このボタンをクリックすると、利得0 dB の位置にマーカーが移動し、その位置の位相を出力変数データリストに表示します。利得が0 dB に達していなければ“Not defined”と表示されます。一巡伝達特性の位相余裕を求めるときに使用します。

(10) 縦スクロールバー

縦スクロールバーをスクロールすることにより、出力変数の零レベルを上下に移動することができます。

(11) マーカー表示ボタン (Mkr)

このボタンをクリックするとマーカーモードになりマーカーが表示されます。このとき、出力変数データリストには、出力変数のマーカー点のY軸値が表示され、周波数データリストには、マーカー点の周波数とインデックス番号が表示されます。さらに、FRAウィンドウのタイトルには、X軸値とY軸値が表示されます。このボタンをもう一度クリックすると通常モードに戻ります。通常モードにおいて、コントロールパネルが表示されていれば、FRAウィンドウ内を右クリックしても、このボタンと同じことができます。

(12) マーカーセットボタン (Set)

マーカーモードにおいて、このボタンをクリックするとマーカー範囲モードになり、表示がClr (Clear)となります。もう一度クリックするとクリアされ、表示がSetとなります。マーカー範囲モードで、マーカーを移動すると、セット点から現在点の間のY軸値、X軸値がそれぞれ出力変数データリストと周波数データリストに表示されます。

(13) ピークボタン (Pk)

マーカーモードにおいて、このボタンをクリックするとマーカーがFRAウィンドウの表示範囲内において、対象出力変数の最大値または最小値に交互に移動します。

(14) ズームボタン (Zm)

マーカーモードにおいて、セットボタンによりマーカーがセットされた状態でこのボタンをクリックすると、FRAウィンドウの周波数範囲がセット点

と現在点の範囲に設定されます。これを解除するには、AdX ボタンを利用します。また、F R A ウィンドウ内でドラッグすると矩形枠が表示され、その矩形枠 X 軸範囲でズームが行われます。Y 軸のズームは行われません。このドラッグによるズーム機能は、通常モードでも有効です。

(15) 横スクロールバー

マーカーモードにおいて、この横スクロールバーをスクロールすることにより、マーカーを移動することができます。また、F R A ウィンドウ内でクリックすると、その X 座標値にマーカーが移動します。

(16) 表示線オプション(View:Option:Scope)

View:Option:Scope を利用すると、スコープウィンドウの各表示線と目盛りの大きさと色を変更することができます。この設定は、W V F, F F T, F R A, C M P ウィンドウ共通で、W V F ウィンドウの項目を参照してください。

4. 6 C M P ウィンドウ, コントロールパネル (Window:Window:CM P)

K492 は、波形比較用のスコープウィンドウである C M P ウィンドウとそのコントロールパネルを持っています。C M P ウィンドウは、W V F ウィンドウのコントロールパネルの C M P ボタンをクリックするか、Analysis:Sweep のシミュレーションを実行すると自動的に画面に表示されます。C M P ウィンドウが開いた状態で Window:Window:CM P を実行すると C M P ウィンドウは最小化され、メニューのチェックマークが消えます。さらに、最小化された状態で Window:Window:CM P を実行すると、以前の位置に C M P ウィンドウが開きます。C M P ウィンドウを閉じるには、右上のクローズボタンをクリックします。

C M P ウィンドウ内で右クリックするとコントロールパネルが現れます。以下に、コントロールパネルの機能について説明します。

(1) 出力変数リスト (Output)

出力変数を表示します。Shift キーを押しながら、出力変数をクリックすると、ダイアログボックスが現れ、それにより出力変数リストで選択されている出力変数の範囲および線の太さと色を変更できます。

(2) 出力変数データリスト

出力変数リストのすぐ下に出力変数に関する以下のデータを表示します。

- [1] Max : Y軸最大値
- [2] Min : Y軸最小値
- [3] Div : Y軸1目盛りの値
- [4] Ave : 平均値
- [5] Rms : 実効値

これらのパラメータのなかで、Max と Min は任意の値に変更できます。これらの値を変更するには、変更する項目をクリックします。すると、Value 入力ボックスが現れるので、新しい値を入力します。

(3) X軸変数データリスト (X-Axis)

X軸変数に関する以下のデータを表示します。

- [1] Var : X軸変数名
- [2] Max : X軸最大値
- [3] Min : X軸最小値
- [4] Div : X軸1目盛りの値

Max, Min は任意の値に変更できます。

(4) DC/AC ボタン (DC/AC)

DC モードと AC モードがあります。DC モードでは、対象出力変数の直流分(平均値)と交流分の和の波形表示を行い、AC モードでは、対象出力変数の直流分を除去して交流分のみの波形表示を行います。デフォルトはDCモードにな

っており、このボタンをクリックすることにより切り替えが行われ、現在のモードがボタンに表示されます。また、Alt キーを押しながらこれを実行すると、全ての出力変数に対して実行されます。

(5) Positive/Negative ボタン (P/N)

Positive モードと Negative モードがあります。Positive モードでは、回路で定義された極性で表示を行い、Negative モードでは、定義された極性と逆極性で表示を行います。デフォルトは Positive モードになっており、このボタンをクリックすることにより切り替えが行われ、現在のモードがボタンに表示されます。また、Alt キーを押しながらこれを実行すると、全ての出力変数に対して実行されます。

(6) Widen ボタン (Wdn)

このボタンをクリックすることにより、Y軸範囲を広げます。Alt キーを押しながらこれを実行すると、全ての出力変数に対して実行されます。

(7) Narrow ボタン (Nrw)

このボタンをクリックすることにより、Y軸範囲を狭めます。Alt キーを押しながらこれを実行すると、全ての出力変数に対して実行されます。

(8) X軸 Adjust ボタン (AdX)

このボタンをクリックすると波形がCMP ウィンドウ全体に合わせて表示されます。

(9) Y軸 Adjust ボタン (AdY)

このボタンをクリックすると表示波形がCMP ウィンドウ内に入るようにY軸範囲が調整されます。Alt キーを押しながらこれを実行すると、全ての出力変数に対して実行されます。

(10) 出力調整ボタン(Fit)

このボタンをクリックすると、選択されている出力変数のY軸範囲が、残りの出力変数に対して適用されます。この機能は、Analysis:Sweep 解析で得られた複数の波形を同じY軸範囲で比較したいとき使用します。

(11) 変数名変更ボタン(Ren)

このボタンをクリックすると、選択された出力変数の名前が任意に変更されます。この機能は、Analysis:Sweep 解析で得られた同じ名前を持つ複数の波形に異なる名前をつけるときに使用します。

(12) Display ボタン (Dsp)

このボタンをクリックすることにより、波形の表示をするか否かを切り替えます。表示をしない場合、出力変数リストの出力変数番号の代わりに“#”が付きまます。一時、波形を消したい場合に使用します。また、Alt キーを押しながらこれを実行すると、全ての出力変数に対して実行されます。

(13) 消去ボタン(Del)

このボタンをクリックすると、選択された出力変数が消去できます。このとき確認メッセージが表示されます。出力変数が全て消去されるとCMPウィンドウは、自動的に閉じられます。

(14) 縦スクロールバー

縦スクロールバーをスクロールすることにより、出力変数の零レベルを上下に移動することができます。

(15) マーカー表示ボタン (Mkr)

このボタンをクリックするとマーカーモードになりマーカーが表示されます。

このとき、出力変数データリストには、出力変数のマーカ一点のY軸値が表示され、周波数データリストには、マーカ一点の周波数とインデックス番号が表示されます。さらに、CMPウィンドウのタイトルには、X軸値とY軸値が表示されます。このボタンをもう一度クリックすると通常モードに戻ります。通常モードにおいて、コントロールパネルが表示されていれば、CMPウィンドウ内を右クリックしても、このボタンと同じことができます。

(16) マーカーセットボタン (Set)

マーカーモードにおいて、このボタンをクリックするとマーカー範囲モードになり、表示がClr (Clear)となります。もう一度クリックするとクリアされ、表示がSetとなります。マーカー範囲モードで、マーカーを移動すると、セット点から現在点の間のY軸値、X軸値がそれぞれ出力変数データリストとX軸変数データリストに表示されます。

(17) ピークボタン (Pk)

マーカーモードにおいて、このボタンをクリックするとマーカーがCMPウィンドウの表示範囲内において、対象出力変数の最大値または最小値に交互に移動します。

(18) ズームボタン (Zm)

マーカーモードにおいて、セットボタンによりマーカーがセットされた状態でこのボタンをクリックすると、CMPウィンドウの周波数範囲がセット点と現在点の範囲に設定されます。これを解除するには、AdX ボタンを利用します。また、CMPウィンドウ内でドラッグすると矩形枠が表示され、その矩形枠X軸範囲でズームが行われます。Y軸のズームは行われません。このドラッグによるズーム機能は、通常モードでも有効です。

(19) 横スクロールバー

マーカーモードにおいて、この横スクロールバーをスクロールすることにより、マーカーを移動することができます。また、CMP ウィンドウ内でクリックすると、その X 座標値にマーカーが移動します。

(20) 表示線オプション(View:Option:Scope)

View:Option:Scope を利用すると、スコープウィンドウの各表示線と目盛り
の大きさと色を変更することができます。この設定は、WVF, FFT, FRA,
CMP ウィンドウ共通で、WVF ウィンドウの項目を参照してください。

4.7 カスタムウィンドウ

カスタムウィンドウは、カスタム素子用のウィンドウであり、カスタム回路
のパラメータの変更や回路構成の変更を行うウィンドウです。その方法はメイ
ンウィンドウと同じです。ただし、Analysis、Topology モードの変更は、メイ
ンウィンドウで行います。以下のメニューが用意されています。

(1) カスタム回路の読み込み (File:Read)

カスタム回路を読み込みます。このとき、カスタム回路のカスタム端子数と
カスタム素子の端子数が一致していなければなりません。カスタム回路データ
ファイルの拡張子は“CVT”です。

(2) カスタム回路の保存 (File:Save)

カスタム回路データを保存します。

(3) シンボル変更 (Function:Symbol)

シンボルの変更を行います。

(4) クリップボードへのコピー (Function:Copy)

カスタム回路をクリップボードへコピーします。

(5) 表示設定 (Function:Initialize View)

図形全体をメインウィンドウの View:Option:Default で設定されている内容

で表示します。

4. 8 DSPウィンドウ

DSPウィンドウは、DSP専用のウィンドウであり、DSPブロック図の作成変更を行うウィンドウです。その使用方法は、カスタムウィンドウと同じです。以下のメニューが用意されています。

(1) ブロック図の読み込み (Read)

ブロック図を読み込みます。このとき、ブロック図ファイルの拡張子は“CVT”です。

(2) ブロック図の保存 (File:Save)

ブロック図を保存します。

(3) シンボル変更 (File:Symbol)

シンボルの変更を行います。

(4) 全体プログラムの表示(Function:Program)

ブロック図から自動生成された全体プログラムを表示します。

(5) ブロックの実行順序(Function:Block Order)

ブロックの実行順序を表示します。

(6) DSPアクティブ化 (Function:Active)

DSPをアクティブ化、非アクティブ化を行います。

(7) クリップボードへのコピー (Function:Copy)

カスタム回路をクリップボードへコピーします。

(8) 表示設定 (Function:Initialize View)

図形全体をメインウィンドウの View:Option:Default で設定されている内容で表示します。

4. 9 ネットリスト (Window:Information:Net List)

Window:Information:Net List を実行すると、回路データのネットリストがメ

メモ帳に表示されます。この機能はあくまで確認用ですので、ネットリストは書き換えしないでください。

4. 1 0 メモ (Window:Information:Memo)

Window:Information:Memo を実行すると、拡張子 MEM をもったファイルがメモ帳に開かれ、回路の説明が記述できます。

4. 1 1 バージョン表示 (Window:Information:About)

Window:Information:About 実行するとバージョン表示が行われます。

4. 1 2 ウィンドウのクリップボードへのコピー (Window:Copy)

Window:Copy を利用すると以下のウィンドウの内容がクリップボードへコピーされます。

- [1] Main : メインウィンドウ
- [2] State : 状態ウィンドウ
- [3] WVF : W V F ウィンドウ
- [4] FFT : F F T ウィンドウ
- [5] FRA : F R A ウィンドウ
- [6] CMP : C M P ウィンドウ

4. 1 3 プリント出力 (File:Print)

解析結果をプリント出力するには、File:Print をクリックします。すると、プリントダイアログボックスが現れるので、プリント出力したいウィンドウに対するチェックボタンをチェックし、Print ボタンをクリックします。プリントダイアログの Option ボタンをクリックすると、プリントオプションダイアログが現れ、各ウィンドウの文字サイズや線幅などが設定できます。このダイアログの各記号の意味は以下の通りです。

- [1] State : 状態ウィンドウ

C : 文字サイズ

[2] Scope : WV F, F F T, F R A, CMP ウィンドウ

X : ウィンドウの横幅 (単位 : mm)

Y : ウィンドウの縦幅 (単位 : mm)

C : 文字サイズ

これらのパラメータは、使用しているプリンタと用紙に合わせて、適当に変更してください。また、プリントを実行する前に必ず Printer ボタンを実行し、プリンタのセットアップを適切に行ってください。

5. シミュレーション機能

本章では、シミュレーション機能を説明します。

5. 1 過渡応答解析 (Analysis:Transient)

過渡応答を解析するには、Analysis:Transient をクリックします。すると、Cycle 入力ボックスが現れるので、過渡応答において解析するスイッチング周期数を入力します。このとき、アップダウンボタンが利用でき、周期数を100単位で増減させることができます。次に、OK ボタンをクリックして下さい。すると、画面右下にWVFウィンドウが開きます。回路に使用されているスイッチやダイオードなど状態変化素子のオン・オフの状態を決定することを状態解析と呼び、Transient 解析では、指定した周期分の状態解析が行われ、出力変数の軌跡がWVFウィンドウに表示されます。このとき、表示されるのは、出力変数のモードで指定された代表点です。終了すると、小さなメッセージウィンドウが開いて“Congratulations !”と表示されるので OK ボタンをクリックして下さい。

5. 2 波形解析 (Analysis:Waveform)

波形を観測するには、Analysis:Waveform をクリックします。すると、Cycle 入力ボックスが現れます。この入力ボックスでは、波形表示するスイッチング周期数を入力します。入力ボックス右横のアップダウンボタンを使用すると、10単位で周期数を増減できます。このとき、0を指定すると状態解析を行わず、現在決定されている状態の波形表示を1周期分行います。1以上の数を指定すると、状態解析と波形表示を指定された周期分行います。OK ボタンをクリックすると、画面右下にWVFウィンドウが現れ、そこに出力変数の波形が表示されます。

5. 3 ステップ応答解析 (Analysis:Step:Transient, Waveform)

ステップ応答を解析するには、Analysis:Step を使用します。Transient も Waveform も実行方法は同じなので、Transient を例に説明します。Analysis:Step:Transient をクリックすると、ダイアログボックスが表示されますので、以下のパラメータを設定します。

(1) ステップ変数

Symbol リストボックスにスイープ可能な素子シンボルが表示されます。スイープ対象とするパラメータをもつ素子をクリックします。このとき、インダクタ、キャパシタなどの単純な素子に対しては、Symbol を指定するだけでそのパラメータが選択されます。また、その他の素子に対しては、さらに、Parameters リストボックスにスイープ可能なパラメータが表示されるので、スイープしたいパラメータをクリックします。各パラメータについては、パラメータの設定の項目を参照してください。

(2) 解析時間

次に、ステップ応答解析の時間を設定します。Time 1 に次項の Value1 に対する時間、Time 2 に Value2 に対する時間、Total Time に解析全時間を設定します。

(3) パラメータ値

Value1 に Time1 に設定するパラメータ値、Value2 に Time2 に設定するパラメータ値を設定します。

以上の設定を行うと、 $0 < t < \text{Time1}$ でパラメータ値は Value1 になり、 $\text{Time1} < t < \text{Time1} + \text{Time2}$ で Value2 になります。 $\text{Time1} + \text{Time2} < t < \text{Total Time}$ での動作は、Repeat がチェックされていないければ、パラメータ値は Value1 に設定され、チェックされていれば、Value1 と Value2 を交互に Time1 と Time2 の時

間で設定されます。これらの設定をもとに Transient 解析が実行されます。

5. 4 定常解析 (Analysis:Steady)

(1) DC-DCコンバータの定常解析

DC-DCコンバータの定常解析を行うには、Analysis:Steady をクリックします。すると、状態解析が何回かくりかえされた後、定常状態が決定されます。このとき、WVFウィンドウが画面右下に自動的に開き、定常状態における出力変数の波形が1周期分表示されます。このとき、回路の初期状態は定常状態における値に更新されます。ただし、この解析ではニュートン法が使用されており、必ずしも成功するとは限らないので注意が必要です。失敗する場合は、次の(2)の方法を利用してください。

(2) その他のコンバータの定常解析

DC-DCコンバータ以外のコンバータの定常解析を行うには、過渡応答解析を定常状態になるまで続けます。したがって、この場合、過渡応答解析の項目を参照してください。

5. 5 スイープ解析 (Analysis:Sweep:Sweep, Transient, Waveform, Steady)

スイープ解析とは、あるパラメータをある範囲で少しずつ変化させたときの特性変化を解析することです。K492 では、ほとんどのパラメータに対するスイープ解析を行うことができ、解析種類も四つが用意されています。

以下の手順は、上記のスイープ解析に共通です。

Analysis:Sweep のいずれかのメニューをクリックすると、スイープダイアログボックスが現れます。スイープダイアログボックスでは以下のパラメータを設定します。

(1) スイープ変数

Symbol リストボックスにスイープ可能な素子シンボルが表示されます。スイープ対象とするパラメータをもつ素子をクリックします。このとき、インダクタ、キャパシタなどの単純な素子に対しては、Symbol を指定するだけでそのパラメータが選択されます。また、その他の素子に対しては、さらに、Parameters リストボックスにスイープ可能なパラメータが表示されるので、スイープしたいパラメータをクリックします。各パラメータについては、パラメータの設定の項目を参照してください。

(2) スイープ範囲

次に、Begin, End, Division, Cycle 入力ボックスでスイープ範囲の開始値, 終了値, 分割数およびスイッチング周期数をそれぞれ入力します。Cycle 入力ボックスには、解析内容によって以下のように入力します。

[1] Sweep

Cycle に0を入力すれば Steady 解析を行い、正の値を入力すれば、その回数だけ Transient 解析を行います。ただし、Steady 解析が不可能な回路では0は入力できません。Sweep の場合、全ての出力変数の最終値がCMP ウィンドウに表示されます。このとき、CMP ウィンドウの横軸はスイープパラメータになります。

[2] Transient

Cycle には、正の値を入力します。すると、その回数だけ Transient 解析が行われ、WVF ウィンドウで選択されている波形がCMP ウィンドウに登録されます。初期状態は変えずに、この動作を指定したパラメータをスイープしながら繰り返します。

[3] Waveform

Cycle には、正の値を入力します。すると、その回数だけ Waveform 解析が行われ、WVF ウィンドウで選択されている波形がCMP ウィンドウに登録されます。初期状態は変えずに、この動作を指定したパラメータをスイープしながら繰り返します。

[4] Steady

Cycle に 0 を入力すれば Steady 解析を行い、正の値を入力すれば、その回数だけ Transient 解析を行います。ただし、Steady 解析が不可能な回路では 0 は入力できません。そして、WVF ウィンドウで選択されている波形に対して、最終状態におけるスイッチング一周期の波形を CMP ウィンドウに登録します。このとき、初期状態は前回の最終状態に更新されます。この動作を指定したパラメータをスイープしながら繰り返します。

5. 6 入力電流の高調波解析

(1) 整数倍高調波のみの高調波解析

[1] 入力電流と入力電圧を出力変数として設定する。

まず、Parameter:Output を利用して、入力電流と入力電圧を出力変数に設定します。このとき、K492 では、電圧源の電流は直接出力変数に設定できませんので、入力電圧源に小さな抵抗を直列に接続して、この抵抗の電流を入力電流とします。また、入力電流を第 1 番目の出力変数、入力電圧を第 2 番目の出力変数、その他を 3 番目以降の出力変数とする必要があります。出力変数のモードは、入力電流は平均値 (Ave)、入力電圧は任意に設定します。

[2] 過渡応答解析を行い定常状態を求める。

次に、入力電圧周波数の数周期分の期間だけ Analysis:Transient を実行して定常状態を求めます。

[3] 入力電圧周期の一周期分を WVF ウィンドウに表示する。

定常状態になったら、入力電圧周期の 1 周期以上の波形を Analysis:Transient コマンドを利用して WVF ウィンドウに表示します。このとき、表示範囲は必ずしも厳密に一周期である必要はありません。一周期以上であれば、途中でとぎれていてもかまいません。

[4] コントロールパネルの HA ボタンをクリックする。

次に、WVF ウィンドウのコントロールパネルの HA ボタンをクリックすると、整数倍高調波のみの高調波解析モードに変わります。このボタンはトグルになっており、もう一度クリックすると通常モードにもどります。整数倍高調波のみの高調波解析モードになると、WVF ウィンドウの X 軸範囲は入力電圧の一周期に自動的に設定されます。また、コントロールパネルには、入力電流と入力電圧の実効値 (Irms, Vrms) , 入力電力 (Pin) , 力率 (PF) が表示されます。

[5] コントロールパネルの FFT ボタンをクリックする。

最後に、高調波規制を満足しているかどうかを判定するには、コントロールパネルの FFT ボタンをクリックします。すると、入力電流に対して FFT 解析が行われ、その周波数スペクトラムが FFT ウィンドウに表示されます。規制クラスを設定確認するには、FFT ウィンドウのコントロールパネルの HA Class コンボボックスを利用します。規制クラスに関しては、付録を参照してください。また、全歪率 (Thd) , 補正係数 (HV) が表示されます。補正係数 HV をクリックすると、公称電圧ダイアログボックスが現れます。これを利用すれば、公称電圧を任意の値に変更することができます。

(2) 次数間高調波を含む高調波解析

[1] , [2] は整数倍高調波のみの高調波解析と同じです。

[3] 200ms 分をWVF ウィンドウに表示する。

定常状態になったら、200ms 分波形を Analysis:Transient コマンドを利用してWVF ウィンドウに表示します。このとき、表示範囲は必ずしも厳密に200ms である必要はありません。200ms 以上であれば、途中でとぎれていてもかまいません。

[4] Alt キーを押しながらコントロールパネルの HA ボタンをクリックする。

次に、Alt キーを押しながらWVF ウィンドウのコントロールパネルの HA ボタンをクリックすると、次数間高調波を含む高調波解析モードに変わります。このボタンはトグルになっており、もう一度クリックすると通常モードにもどります。次数間高調波を含む高調波解析モードになると、WVF ウィンドウの X 軸範囲は 200ms に自動的に設定されます。また、コントロールパネルには、入力電流と入力電圧の実効値 (Irms, Vrms) , 入力電力 (Pin) , 力率 (PF) が表示されます。

[5] コントロールパネルの FFT ボタンをクリックする。

最後に、高調波規制を満足しているかどうかを判定するには、コントロールパネルの FFT ボタンをクリックします。すると、入力電流に対して FFT 解析が行われ、その周波数スペクトラムが FFT ウィンドウに表示されます。規制クラスを設定確認するには、FFT ウィンドウのコントロールパネルの HA Class コンボボックスを利用します。また、全歪率 (Thd) , 補正係数 (HV) が表示されます。補正係数 HV をクリックすると、公称電圧ダイアログボックスが現れます。これを利用すれば、公称電圧を任意の値に変更することができます。

さらに、次数間高調波を含む高調波解析では、グループ解析が可能であり、Group コンボボックスより、グループを選択することにより、対応した解析結果が表示されます。各グループに関しては、付録を参照してください。

5. 7 継続解析, 再解析, 解析中止 (Analysis:Continue, Parameter:Update, Analysis:Quit)

Transient 解析と Waveform 解析のとき、以前の解析結果を保持したまま解析を続行する場合には、Analysis:Continue をクリックします。すると、直前に実行した解析の Cycle 入力ボックスが現れるので、継続して解析したいスイッチング周期数を入力します。WVF ウィンドウには、以前の解析結果に続いて新しい結果が表示されます。

Transient 解析と Waveform 解析においては、一度解析が完了しても、状態変

数の初期値などの回路の初期状態は更新されません。これは、初期状態を固定したままで、ほかのパラメータを変更して再解析を行えるようにするためです。回路の初期状態を更新するには、Parameter:Update をクリックします。

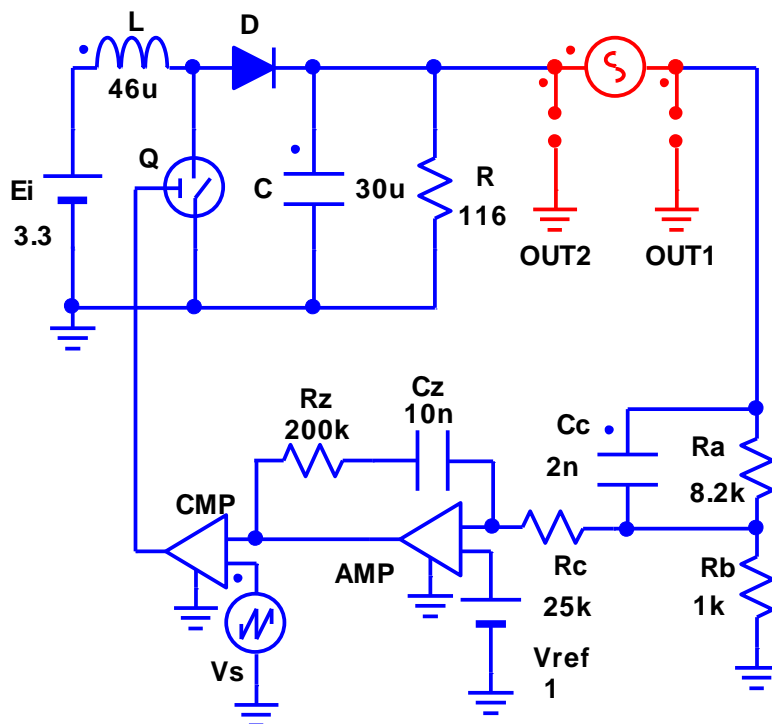
また、解析実行中に途中で解析を中止したい場合には、Analysis:Quit をクリックするか ESC キーを押します。次に、確認メッセージボックスが現れるので、解析を中止する場合には Yes (はい) ボタン、続行する場合には No (いいえ) ボタンをクリックします。このコマンドを実行しても解析結果は有効であり、Analysis:Continue, Parameter:Update が利用できます。

5. 8 周波数特性解析 (Analysis:Frequency:Fast, Slow)

周波数特性解析には二つのコマンドが用意されています。Analysis:Frequency:Fast を使用すると高速かつ容易に周波数特性解析が実行できるので、通常はこれを使用します。Fast でどうしても解析できない場合に Slow を使用します。解析手順は以下のとおりです。

(1) 一巡周波数特性

フィードバック制御された回路の一巡周波数特性を解析するには、下図のように、ループ内に AC Sweep 素子 (Vas) を挿入し、その前後に入力出力波形検出用の Open 素子 (OUT1, OUT2) を接続します。



次に、AC Sweep 素子パラメータの設定を以下のように行います。

[1] 周波数範囲 (Fmin, Fmax)

解析する周波数範囲を設定します。最低周波数を Fmin、最大周波数を Fmax、に設定します。Fast 解析で解析結果が乱れている場合は、Fmin を低く設定すると正しく解析できる場合があります。また、Fmax はスイッチング周波数の約 1 / 4 程度が限度です。

[2] 周波数分割数 (Ndiv) (Slow のみ)

Slow 解析の場合は、周波数の分割数を Ndiv に設定します。Fast 解析の場合は、自動的に設定されます。

[3] 解析時間 (Tan1) (Slow のみ)

次に、Slow を使用する場合、解析時間 Tan1 を設定します。解析時間とは、過渡応答が終わり定常状態になるまでの時間です。周波数特性解析では AC Sweep 素子の周波数や電圧を変化させ、その応答を観測します。このとき、過渡応答が生じますが、これが十分安定しないと正確な測定ができません。通常は、デフォルトの 10ms で良いと思いますが、時定数の大きい回路の場合は、長

く設定する必要があります。もし、解析結果が正確でないようなら 20ms, 30ms・・・等設定してください。

[4] 信号電圧 (Vac)

交流信号源の電圧 Vac を設定します。Vac は、理論的にはなるべく小さいほうが解析は正確になりますが、小さすぎるとリップル等の雑音の影響を受けやすくなります。また、逆に大きすぎる場合には飽和したり波形が歪んだりします。通常は、デフォルトの 1 mV で良いでしょう。1 μ V~1V 程度で設定してください。

[5] PWM-SFM モード (PWM-SFM)

入力波形端子(OUT1)から出力波形端子(OUT2)の回路パスに PWM 素子または SFM 素子が存在すれば PWM-SFM ボタンをチェックし、存在しなければチェックをはずします。上図では PWM 素子が存在しませんのでチェックしません。これがチェックされており、かつ、Parameter:Condition:Compensation がチェックされている場合は、PWM 素子の一周遅れ制御の位相補正が行われます。

次に入力波形と出力波形の設定を行います。入力波形(OUT1)と出力波形(OUT2)を Parameter:Output を使って、出力変数に設定します。出力変数モードは、Ave に設定します。

以上の設定を行ったのち、以下の手順で解析を実行します。

[1] Transient 解析

回路状態を十分定常状態にするため、まず Analysis:Transient 解析を実行し、Parameter:Update を行います。この Transient 解析は、回路パラメータの変更など、回路状態を変化させた場合には、必ず実行します。

[2] Frequency 解析

次に、Frequency を実行します。すると、ダイアログボックスが表示されるので、まず、交流信号源の AC Sweep 素子を選択し、解析対象とする回路範囲を指定するため、入力波形用出力変数(From)と出力波形用出力変数(To)を選択し

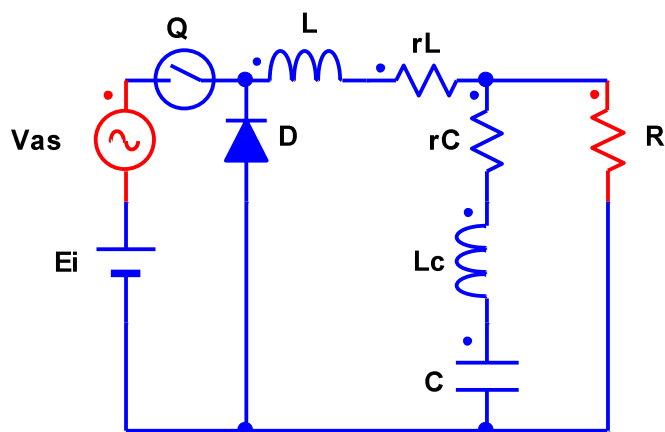
ます。ここでは、FromにOUT1をToにOUT2を選びます。Frequency解析では、Fromの波形に対するToの波形の利得と位相の周波数特性が解析されますので、このように設定すると、一巡周波数特性が得られます。FromとToには任意の出力変数が指定できます。また、Phase Foldをチェックすると、位相特性が -180 度から $+180$ 度の範囲で表示されます。

(2) 部分回路周波数特性

誤差増幅器 Amp の出力 Amp.Vo (出力モード AVE) を出力変数に追加すれば、制御回路部分のみの部分回路の周波数特性も解析できます。このとき、FrequencyのダイアログボックスでFromをOUT1、ToをAmp.Voに設定して実行します。同様に、Fromにコンパレータ Cmp の出力 Cmp.Vo を、ToにOUT2を設定すれば、コンバータのみの周波数特性が得られます。

(3) オープンループ周波数特性

下図のような入出力電圧周波数特性などのオープンループ周波数特性は次のように行います。まず、Vasの電圧 (Vas.V) と Rの電圧を、ModeをAveにして出力変数に設定します。そして、FrequencyのダイアログボックスのFromにVasの電圧を、ToにRの電圧を設定して実行します。



(4) 周波数特性解析の注意点

周波数特性を解析する場合は以下の点に注意してください。

[1] 回路の安定化

周波数特性は、回路が安定に動作しなければ解析できません。設計段階では、各素子のパラメータ設定によっては回路が不安定になる場合がありますが、この場合 Frequency 解析は失敗します。まずは、Transient 解析などを行って、回路が安定に動作しているか確認してください。安定に動作していなければ、素子パラメータの値を変更し回路が安定に動作するようにしてください。

また、PWM 素子や FRM 素子を使用しており、Parameter:Condition の Compensation がチェックされている場合、低周波近似補償が行われます。しかし、Frequency 解析では、この補償が効かなくなるため、回路が安定限界近くで動作している場合、Transient 解析では補償が効いて安定しているのに、Frequency 解析では補償が効かず失敗する場合があります。この場合も、Compensation がチェックされてなくても、回路が安定に動作するように設定しなおして、Frequency 解析を行ってください。

また、AC Sweep の正弦波信号が乱されてしまうので、交流電源や他の理由で波形が振動する場合、周波数特性は解析できません。直流波形にして解析してください。

[2] 解析時間 Tan1 (Slow のみ)

AC Sweep 素子の解析時間 Tan1 (デフォルト 10ms) は、およその値を設定すれば自動的に最適値が計算されます。このとき、最適値の範囲は Tan1 の 100 分の 1 から 10 倍までとなっていますので、この範囲以外で最適値が存在する場合や最適値が不適切な場合は Tan1 を変更する必要があります。このような場合、特性がなめらかではなくジグザグ状になることが多く、そのときは、Tan1 を 20ms, 30ms・・・という具合に増加してください。特に、時定数が非常に大きい回路では注意してください。

[3] 入力波形、出力波形の出力変数モード

基本的には、入力波形、出力波形には任意の出力変数が指定できます。出力

変数モードは任意ですが、周波数特性としては、平均値の特性が一般的と考えられるので通常 Ave を設定します。

5. 9 スクリプト解析 (Analysis:Script, Window:Script:New, Edit)

K. 492 では、スクリプト言語を利用することによりシミュレーション全体を自動化できます。このスクリプト言語はC言語に似た言語で、これを用いてテキストでプログラムを書き、拡張子を scr としてファイルに保存します。これを行うには、新規作成の場合は Window:Script:New を、既存のスクリプトファイルを編集するには Window:Script:Edit を利用します。スクリプトファイルを実行するには、K. 492 を起動したのち、Analysis:Script をクリックし、スクリプトファイルを指定するか、コマンドラインでスクリプトファイルをパラメータとして K. 492 を起動します。コマンドライン起動については Windows のマニュアルを参照してください。また、スクリプト言語の文法については、付録の「スクリプト言語の文法」を参照してください。以下に、2つのスクリプトファイルの例を示します。

(1) ステップ解析と F F T 解析

```
ReadCircuit("c:\¥scat¥cvt¥buck. cvt"); (回路を読み込む)
ClearOutput(); (出力変数をクリアする)
SetOutput("R", "V", "AVE"); (負荷Rの電圧平均値を出力変数に設定する)
OpenWVF(false); (WVFウィンドウを画面に表示せず生成する)

SetParam("R", "Value", 10); (負荷Rの値を10Ωに設定する)
Steady(); (定常解析を行う)
Transient(500); (トランジエント解析を500周期行う)
SetParam("R", "Value", 1); (負荷Rの値を1Ωに設定する)
SetAC(0); (0番目の出力変数の表示をACに設定する)
ShowWVF(true); (WVFウィンドウを表示する)
```

Continue(); (継続解析にする)

Transient(1000); (トランジエント解析を1000周期行う)

ExecuteFFT(0, true); (0番目の出力変数に対してFFTを実行し、
FFTウィンドウを表示する)

(2) スイープ解析

ReadCircuit("c:\¥scat¥cvt¥buck.cvt"); (回路を読み込む)

ClearOutput(); (出力変数をクリアする)

SetOutput("R", "V", "AVE"); (負荷Rの電圧平均値を出力変数に設定する)

SetAC(0); (0番目の出力変数の表示をACに設定する)

OpenWVF(true); (WVFウィンドウを画面に表示して生成する)

Rbgn = 100; (負荷の開始値)

Rend = 1; (負荷の終了値)

n = 10; (繰り返し数)

i = 1; (初期設定)

while (i <= n) { (繰り返しループ)

 Ro = Rbgn + i * (Rend - Rbgn) / n; (スイープ値の計算)

 SetParam("R", "Value", Ro); (負荷Rをスイープ値に設定する)

 Steady(); (定常解析を行う)

 Waveform(0); (現在の波形を表示する)

 ExecuteCMP(0, true); (0番目の出力変数をCMPウィンドウに登録し表
示する)

 i = i + 1; (繰り返しの更新)

}

6. 制御方法

本章では、制御回路を付加した場合のシミュレーション方法を説明します。

6. 1 制御素子

K492 には非常に豊富な制御に関係した素子があります。まず、最も一般的な PWM回路の構成素子には次のような素子があります。

(1) のこぎり波発生器(Saw Tooth)

のこぎり波（三角波）を発生させる素子で、次のコンパレータとともに PWM回路の基本的構成素子です。

(2) コンパレータ素子(Comparator)

コンパレータは、PWM回路の基本構成素子です。また、ラッチなど、他のコンパレータ関連素子は、パルス波形の変化を検出する素子です。

(3) オペアンプ素子(OpAmp)

オペアンプ関連素子は、主に誤差増幅や位相補償回路の素子として使用されます。

K492 では、さらに、シミュレーションを容易にする以下のような素子が用意されています。

(4) パルス発生器 (Pulse)

パルス発生器は、スイッチを駆動するための矩形波電圧を出力する素子です。

(5) PWMスイッチ (PWM Switch)

PWMスイッチとは、パルス発生器とスイッチを組み合わせた素子です。通

常、スイッチはパルス発生器により駆動されますが、この二つのデバイスが完全に同期している場合、すなわち、パルス発生器がスイッチに直接接続されている場合はPWMスイッチを使用したほうが便利です。

(6) パルス変調器 (PWM)

パルス変調器は入力電圧に基づいて、主回路のパルス発生器またはPWMスイッチの状態変化時刻を変調し、PWM制御を行います。次節の低周波近時補正が行われます。

(7) 周波数変調器 (SFM)

周波数変調器は入力電圧に基づいて、主回路のスイッチング周波数またはスイッチング周期を変調し、スイッチング周波数制御またはスイッチング周期制御を行います。次節の低周波近時補正が行われます。

(8) 素子パラメータ変調器 (DPM)

素子パラメータ変調器は、入力電圧に基づいて素子パラメータ値を変調します

(9) 同期制御器 (SYC)

同期制御器は、複数の状態変化時刻を基準時刻に同期させて制御する場合に使用します。

また、補助的な制御用素子として以下の素子が用意されています。

(10) 演算素子(Operator, Logic)

演算関連素子には、演算素子(Operator)と論理素子(Logic)があります。これらの素子は、入力に対して様々な演算を行う素子であり、複雑な制御を行うために使用します。

(11) プログラマブル素子

プログラマブル素子は、(1)～(9)の素子では実現が困難な制御をソフトウェアで実現する強力な素子です。

(12) DSPとDSPブロック素子

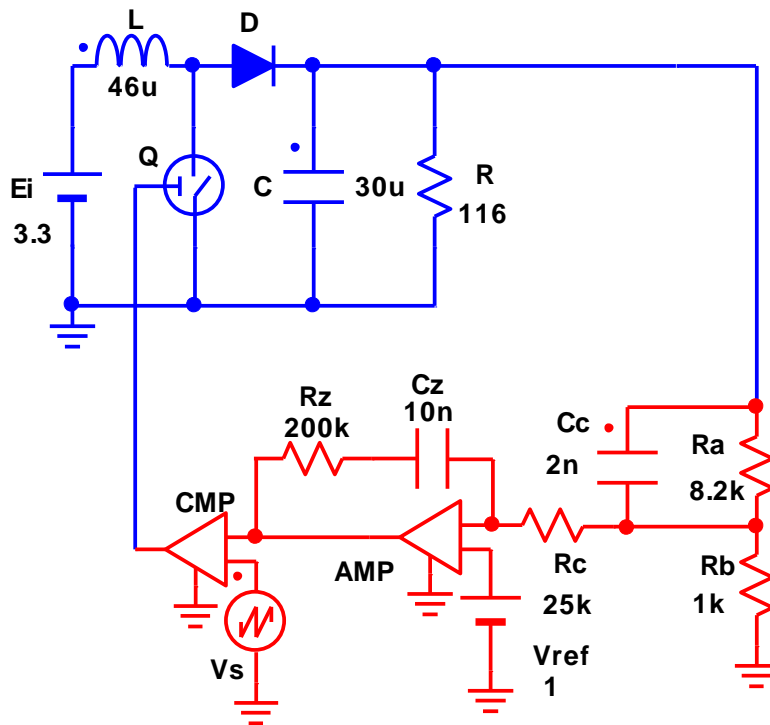
DSPはプログラマブル素子の一つですが、プログラムをブロック図で記述します。これにより、今後、主流となるデジタル制御のシミュレーションが劇的に容易になります。K492の新しい機能です。

6.2 低周波近似

PWM、SFM、DPM、SYC、PRCの素子は、各スイッチングの終わりに動作します。特に、PWM、SFM、DPMでは、入力電圧としてスイッチングの最終値がサンプル評価されます。さらに、PWM、SFMでは、Parameter:ConditionのCompensationがチェックされていると、低周波近似補正と呼ばれる補正が行われます。この補正は、主にノコギリ波発生器とコンパレータとで構成された実際のPWM制御回路をPWM素子で模擬するために行います。また、いくつかの演算素子は平均サンプルホールド動作をします。したがって、これらの素子を使用する場合、入力電圧のリップルはなるべく小さく、またその変動周波数はスイッチング周波数より十分小さいほうが望ましいでしょう(十分の一以下推奨)。

6.3 一般的なPWM制御

下図の赤色部分は、PWM制御の一例です。このような構成は、良く知られているので、ここでは簡単に説明します。



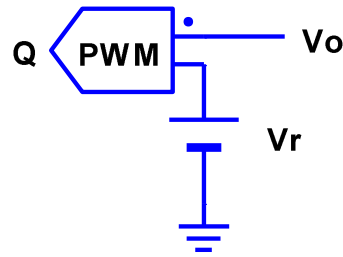
まず、コンバータの出力電圧が、 R_a 、 R_b の分圧回路で分圧された後、誤差増幅器 AMP に入力されます。次に、基準電圧 V_{ref} と比較され、その誤差電圧が増幅された後、コンパレータ CMP に入力されます。そして、この信号とのこぎり波発生器 V_s の出力がコンパレータで比較されることにより、コンパレータの出力には矩形波が発生し、スイッチ Q をドライブします。キャパシタは位相補償用であり、 C_c は微分補償、 C_z は積分補償を行います。これら、誤差増幅器まわりの回路構成は、個々の電源システムで多少変わります。

6. 4 PWM素子を用いたPWM制御

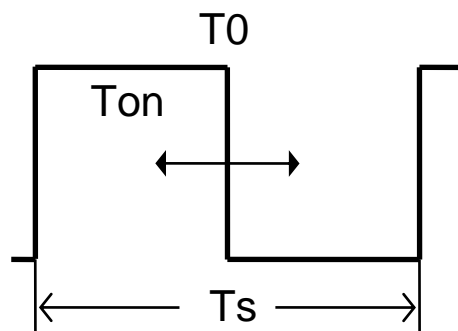
次に、パルス変調器を利用したPWM制御について説明します。PWM 素子など、K492 に用意されている変調器を利用すると、複雑な制御も容易にシミュレーションできます。

(1) PWM時比率制御

下図のようなパルス変調器 (PWM) を主回路に接続します。



この図において、 V_o は主回路の出力電圧、 V_r は基準電圧を表し、さらに、パルス変調器の入力信号電圧を $V_i (= V_o - V_r)$ とし、パルス変調器が変調する素子（PWMスイッチかパルス発生器）を Q とします。また、 Q のタイムチャートが下図のように設定されているとし、図のようにオン時間をPWM制御するとします。



このとき、パルス変調器は Q の時比率 $D (= T_{on} / T_s)$ を以下の式にしたがって変調します。

$$D = \text{Gain} * V_i + D_{ref}$$

$$D < D_{min} \rightarrow D = D_{min}$$

$$D > D_{max} \rightarrow D = D_{max}$$

D : 変調時比率

V_i : 入力信号電圧

Gain : 変換ゲイン

D_{ref} : 基準時比率

D_{min} : 最小時比率

D_{max} : 最大時比率

すなわち、時比率 D は最小時比率 D_{min} と最大時比率 D_{max} の間では、入力信号電圧 V_i によって線形的に変化し、それ以外では D_{min} と D_{max} で制限されます。

ここで、実際にパラメータを設定してみましょう。たとえば定格出力電圧を 5V、そのときの基準時比率を 0.3 とし、ゲインを -0.5、最小時比率を 0.1、最大時比率を 0.9 とします。まず、回路図において基準電圧 V_r を右クリックして、パラメータダイアログボックスを表示し、Value を定格出力電圧の 5 に設定します。次に、パルス変調器 PWM のパラメータダイアログボックスを表示し、Gain を -0.5、Dref を 0.3、 D_{min} を 0.1、 D_{max} を 0.9 と設定します。そして、時比率制御なので Mode の Ratio をチェックし、次に Modulated で Q:T0 と入力します。シンボル ":" でくぎられた 2 つパラメータの意味は、最初のパラメータは変調対象となる素子シンボルであり、この例の場合、"Q" となります。2 番目のパラメータは、状態変化時刻番号です。状態変化時刻番号とは、Q の状態変化時刻に割りふられる番号であり、0 からはじまり順次 1, 2・・・と割り振られます。この例の Q の場合、変調すべき状態変化時刻は、時比率 0.3 のオンからオフに変化する時刻であり、この時刻の状態変化時刻番号は 0 となります。入力するときは、頭に "T" をつけて "T0" のように指定します。オプションの Active ボタンをチェックすればパルス変調器が機能し、チェックしなければ機能しません。最後に OK ボタンをクリックすればパルス変調器の設定は終了です。後のシミュレーション方法は、制御しない場合と同じです。

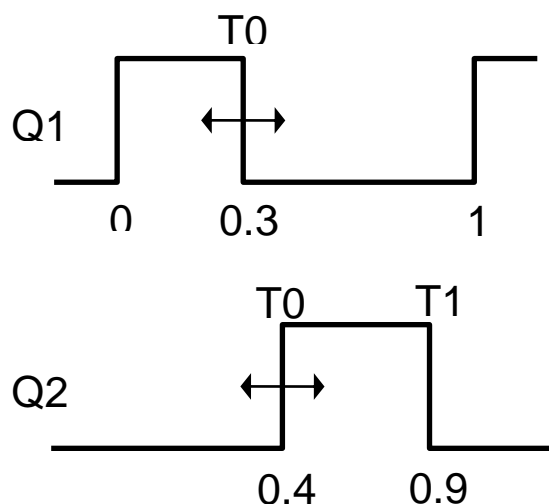
パルス変調器のパラメータのなかで、あらかじめ分からないのが基準時比率 Dref の値です。これは、次のようにして決定すればよいでしょう。まず、パルス変調器の Active ボタンのチェックをはずし、パルス変調器が機能しないようにします。そこで、Q のパラメータダイアログボックスを利用して、変調する状態変化時刻の時比率を変えながら、定常解析を繰り返して実行します。出力電圧がある程度目標出力電圧に近くなったら、そのときの Q の状態変化時刻の時比率をパルス変調器の基準時比率 Dref とします。そして、最後にパルス変調器の Active ボタンをチェックします。このようにすれば、パルス変調器は出力電圧を目標出力電圧に近づくように Q を変調します。

(2) PWM時刻制御

パルス変調器は、通常、時比率を変調します。ところが、スイッチング周波数制御やトリガを併用した制御によりスイッチング周期が変動する場合は、これによっても時比率が変化することになりPWM制御に影響を与えます。スイッチング周期の変化の影響をうけずに状態変化時刻そのものをPWM制御するには、PWM時刻制御を行います。PWM時刻制御を行うにはパルス変調器のパラメータダイアログボックスで Mode の Time をチェックします。この場合、Dref, Dmin, Dmax はすべて時比率ではなく時刻となります。また、Gain も時刻に対応した値にする必要があります。たとえば、スイッチング周波数 100kHz (スイッチング周期 10e-6s) で時比率 0.5 の時刻を基準としてPWM時刻制御する場合、Dref の値は 5e-6 となります。

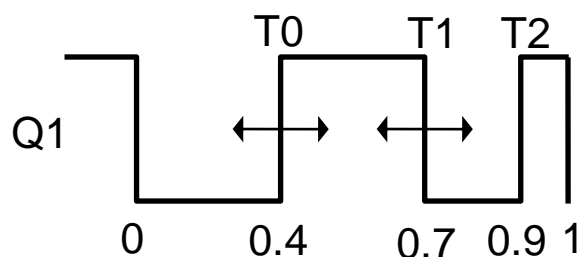
(3) 同期PWM制御 (デッドタイム制御)

次に、複数のスイッチを同期して制御する場合や一つのスイッチの複数の状態変化時刻を同期して制御する場合など、複数の状態変化時刻を同期してPWM制御する場合について説明します。このような制御を行うにはパルス変調器と同期制御器を使用します。まず、2つのスイッチを制御する場合について説明します。2つのPWMスイッチ Q1, Q2 のタイムチャートが下図のようになっているとします。



ただし、この図において波形上部の数字は状態変化時刻番号を表しています。これらを同期PWM制御するには、まず、パルス変調器の Dref を 0.3, Modulated を Q1:T0 とします。次に、同期制御器の Base Type を Transition, Mode を Ratio, Transition Base を Q1:T0, Synchronized を Q2:T0:0.1 とします。この例の場合、Q1:T0 の状態変化時刻を基準にして、Q2:T0 の状態変化時刻を同期させているので、同期基準のタイプを表す Base Type は Transition, 同期基準時刻を指定する Transition Base は Q1:T0 となります。もし、同期基準がスイッチング周期の始めまたは終わりであれば、Base Type を Begin または End に設定し、Transition Base には何も設定しません（設定していても無視されます）。さらに、時比率を対象にしているので Mode は Ratio です。同期させる状態変化時刻は Q2:T0 で同期基準時刻 Q1:T0 との時比率差は 0.1 なので、Synchronized は Q2:T0:0.1 となります。このようにして、Synchronized には任意の数の同期時刻が設定できます。Q1:T0 と Q2:T0 の差を時間差で同期制御する場合には、同期制御器の Mode を Time にし Synchronized の 3 番目のパラメータの数値を Q2:0:1e-6 のように時間で指定します。

次に、一つのスイッチの2つの状態変化時刻を同期制御する場合について説明します。PWMスイッチ Q1 のタイムチャートが下図のようになっています。



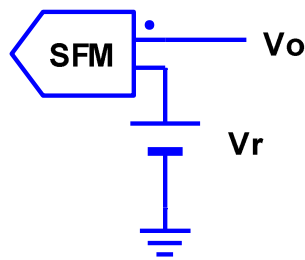
この場合は、パルス変調器の Dref を 0.4, Modulated を Q1:T0 とします。次に、同期制御器の Base Type を Transition, Mode を Ratio, Transition Base を Q1:T0, Synchronized を Q1:T1:0.3 とします。

6. 5 スイッチング周波数制御

次に、周波数変調器を利用した制御方法について説明します。

(1) スイッチング周波数制御

下図のよう周波数変調器 (SFM) を主回路に付加します。



この図において、 V_o は主回路の出力電圧、 V_r は基準電圧であり、さらに、周波数変調器の入力信号電圧を $V_i (= V_o - V_r)$ とします。このとき、周波数変調器は主回路のスイッチング周波数 F_s を以下の式にしたがって変調します。

$$F_s = \text{Gain} * V_i + F_{\text{ref}}$$

$$F_s < F_{\text{min}} \rightarrow F_s = F_{\text{min}}$$

$$F_s > F_{\text{max}} \rightarrow F_s = F_{\text{max}}$$

F_s : 変調周波数

V_i : 入力信号電圧

Gain : 変換ゲイン

F_{ref} : 基準周波数

F_{min} : 最小周波数

F_{max} : 最大周波数

すなわち、スイッチング周波数 F_s は最小周波数 F_{min} と最大周波数 F_{max} の間では、入力信号電圧 V_i によって線形的に変化し、それ以外では F_{min} と F_{max} で制限されます。

ここで、実際にパラメータを設定してみましょう。たとえば定格出力電圧を 5V、そのときの基準周波数を 100kHz とし、変換ゲインを $1e+5$ 、最小周波数を

20kHz, 最大周波数を 200kHz とします。まず、回路図において基準電圧 Vr を右クリックして、パラメータダイアログボックスを表示し、Value を定格出力電圧の 5 に設定します。次に、パルス変調器 SFM のパラメータダイアログボックスを表示し、Gain を 100k, Fref を 100k, Fmin を 20k, Fmax を 200k と設定します。スイッチング周波数変調の場合、Modulated は Frequency をチェックします。また、オプションの Active ボタンをチェックすれば周波数変調器が機能し、チェックしなければ機能しません。最後に OK ボタンをクリックすれば周波数変調器の設定は終了です。後のシミュレーション方法は、制御しない場合と同じです。

周波数変調器のパラメータのなかで、あらかじめ分からないのが基準周波数 Fref の値です。これは、次のようにして決定すればよいでしょう。まず、周波数変調器の Active ボタンのチェックをはずし、周波数変調器が機能しないようにします。そこで、主回路のスイッチング周波数を変えながら定常解析を繰り返しかえし実行します。出力電圧がある程度目標出力電圧に近くなったら、そのときのスイッチング周波数を周波数変調器の基準周波数 Fref とします。そして、最後に周波数変調器の Active ボタンをチェックします。このようにすれば、周波数変調器は出力電圧を目標出力電圧に近づくようにスイッチング周波数を変調します。

周波数変調器を使用した場合、パルス発生器や PWM スイッチのモードが 'Ratio' であれば、それらの状態変化時刻は時比率が一定になるように変調されます。一方、モードが 'Time' であれば変調されず固定されたままです。

(2) スイッチング周期制御

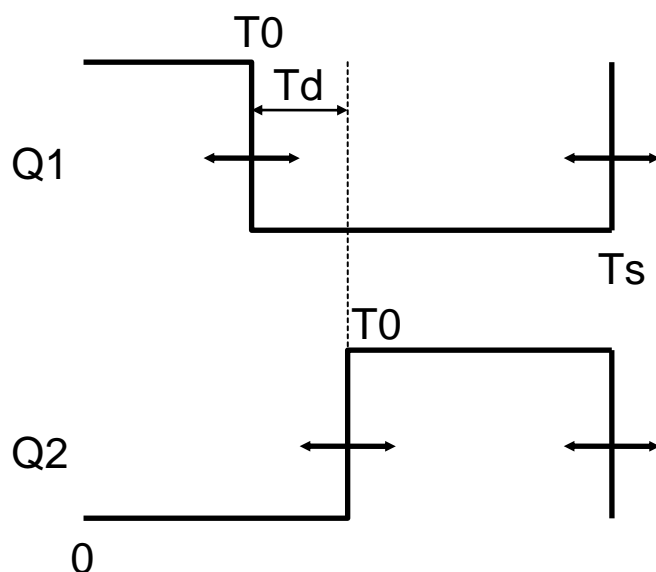
スイッチング周期制御を行うには、周波数変調器のパラメータダイアログボックスにおいて、Modulated の Period をチェックします。後は、スイッチング周波数をスイッチング周期と置き換えて (1) と同様に行います。

(3) 時比率一定制御

通常のスイッチング周波数制御またはスイッチング周期制御を行うと、スイッチング周期が変化します。しかし、パルス発生器やPWMスイッチの状態変化時刻は固定されていますので、これらの状態変化時刻の時比率は変化することになります。このとき、状態変化時刻をスイッチング周期の変化とともに変化させ、その時比率を一定に保つ制御が時比率一定制御です。これを行うには、パルス発生器やPWMスイッチのモードを'Ratio'にして、スイッチング周波数制御を行います。

(4) 時比率一定制御と同期制御

時比率一定制御と同期制御を同時に行う場合は、周波数変調器と同期制御器を利用します。例をあげて説明します。2つのPWMスイッチ（またはパルス発生器）Q1, Q2のタイムチャートが下図のようになっています。

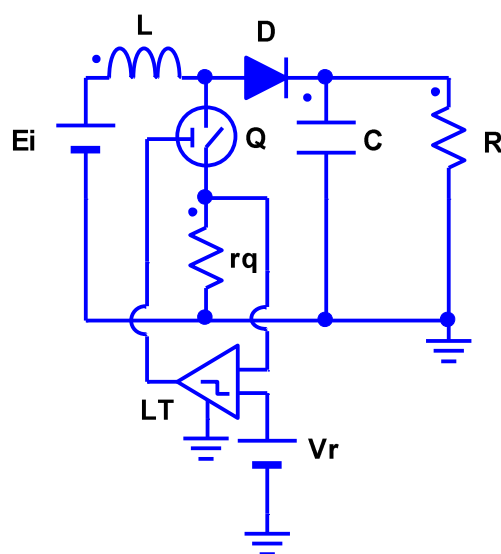


ここで、状態変化時刻 Q1:T0 を時比率一定制御し、さらに、この時刻に同期して状態変化時刻 Q2:T0 を制御する場合（時刻の差 Td 一定）を考えます。周波数変調器と同期制御器を用意します。まず、(3) で説明したように周波数変調器を利用して Q1:T0 を時比率一定制御します。次に、同期制御器の Base Type を Transition, Mode を Time, Transition Base を Q1:T0, Synchronized を Q2:T0:Td とします。このように設定すると、まず、周波数変調器によりスイッ

チング周波数が変化し、Q1:T0 が時比率一定となるように変化します。つぎに、同期制御器の同期時刻制御により Q1:T0 に同期して Q2:T0 が変化します。

6. 6 ピーク電流制御

(1) ピーク電流制御



スイッチのピーク電流を制限して制御する方法をピーク電流制御といいます。ラッチ素子を使用すれば容易にピーク電流制御が実現できます。下図は、ピーク電流制御を施した昇圧形コンバータを示しています。スイッチに直列に接続されている抵抗 r_q は抵抗値が十分小さい電流検出用抵抗であり LT がラッチです。また、直流電圧 V_r はしきい電圧を表しています。次に、このラッチの動作を説明します。ラッチには、 V_{on} 、 V_{off} のパラメータ値を設定する必要があり、この例では、それぞれ-5、5 と設定されています。まず、スイッチング周期の始めではラッチはオフ状態にあり、出力電圧は $V_{off}=5V$ となっています。したがって、スイッチ Q のしきい電圧が 1V のためスイッチはオン状態となり、入力電圧、スイッチを介して抵抗 r_q に電流が流れその電圧は上昇していきます。ラッチの入力電圧が正、すなわち、抵抗 r_q の電圧がラッチのしきい電圧 V_r を超えるとラッチはオン状態となります。このとき、ラッチの出力電圧は $V_{on}=-5V$

となり、スイッチはオフ状態となります。ラッチは一度オフ状態からオン状態に変化すると、オン状態をそのスイッチング周期が終了するまで保持します。スイッチング周期が更新されるとラッチは自動的にオフ状態に復帰します。このようにしてピーク電流制御が実現できます。

(2) PWMピーク電流制御

PWM制御を行いながら、ピーク電流制御を同時に行うには、ラッチの出力とパルス発生器を直列に接続したものをスイッチのゲート・ソース間に加えます。そして、ラッチのオフ電圧を0 V、オン電圧をパルス発生器のオン電圧を打ち消す電圧に設定します。たとえば、パルス発生器のオン電圧が5 Vとし、ラッチのオン電圧を-6 Vとすれば、ラッチがオフの期間では、通常のPWM制御が行われますが、一旦、ラッチの検出電流がしきい値に達して、ラッチがオンすれば、スイッチには-1 Vが加わりスイッチがオフします。

6. 7 素子パラメータ制御

ある信号電圧により、素子パラメータの値を変調する場合には、素子パラメータ変調器を利用します。素子パラメータ変調器はModulatedとParametersで指定された素子パラメータ値Xを以下の式にしたがって変調します。

$$X = \text{Gain} * V_i + X_{\text{ref}}$$

$$X < X_{\text{min}} \rightarrow X = X_{\text{min}}$$

$$X > X_{\text{max}} \rightarrow X = X_{\text{max}}$$

X : 変調パラメータ値

V_i : 入力信号電圧

Gain : 変換ゲイン

X_{ref} : 基準パラメータ値

X_{min} : 最小パラメータ値

X_{max} : 最大パラメータ値

すなわち、変調素子パラメータ値Xは最小パラメータ値 X_{min} と最大パラメータ

値 X_{max} の間では、入力信号電圧 V_i によって線形的に変化し、それ以外では X_{min} と X_{max} で制限されます。

6. 8 プログラム制御

以上の制御方法では実現困難な制御を行う場合、プログラムでその制御を実現するプログラマブル素子を使用します。プログラマブル素子では、C 言語に似た言語により柔軟な制御が行えます。プログラマブル素子には、以下のような素子があります。

(1) プログラマブル演算器 (Operator:Program)

二つの入力を持ち、それらをプログラムにより演算し出力する演算器です。

二つの入力はプログラム内では、 $PrgVi1$, $PrgVi2$ で参照でき、出力は $PrgVo$ で参照します。一周期の終わりに変調器が実行される前に実行されます。その他のパラメータとして

V_{min} : 出力最小値

V_{max} : 出力最大値

を設定します。

例 :

/* 加算器 */

$PrgVo = PrgVi1 + PrgVi2;$

(2) プログラマブル電圧源 (Source:Prg Voltage)

プログラムにより定義された電圧を発生する電圧源です。時間は t で参照でき、出力電圧は $PrgV$ で参照します。この素子は一周期の初めに実行されます。

例 :

/* 正弦波を出力 */

$V_a = 100 * \text{sqrt}(2);$

$f = 60;$

```
th = PI / 3;  
PrgV = Va * sin(PI2 * f * t + th);
```

(3) プログラマブル制御器 (Modulator:PRC)

プログラムにより制御を行う制御器です。時間は t で参照できます。一周期の終わりに実行されます。

例

```
/* 時間 t = 3 m s で負荷 R を 1 0 Ω から 2 0 Ω に、 t = 5 m s で再び 1  
0 Ω に戻す */  
if (t < 3m) R0 = 10;  
else if (t < 5m) R0 = 20;  
else R0 = 10;  
setparam("R", "Value", R0);
```

プログラマブル制御器では、プログラムを Lib ボタンで読み込み、Save ボタンで保存することが出来ます。このときのファイルの拡張子は BLK です。

これらの三つのプログラマブル素子は、基本的に同じ素子であり、形状と実行のタイミングが若干異なるだけです。これらの素子で使用できるプログラム言語の文法は付録を参照してください。

6. 9 DSP制御

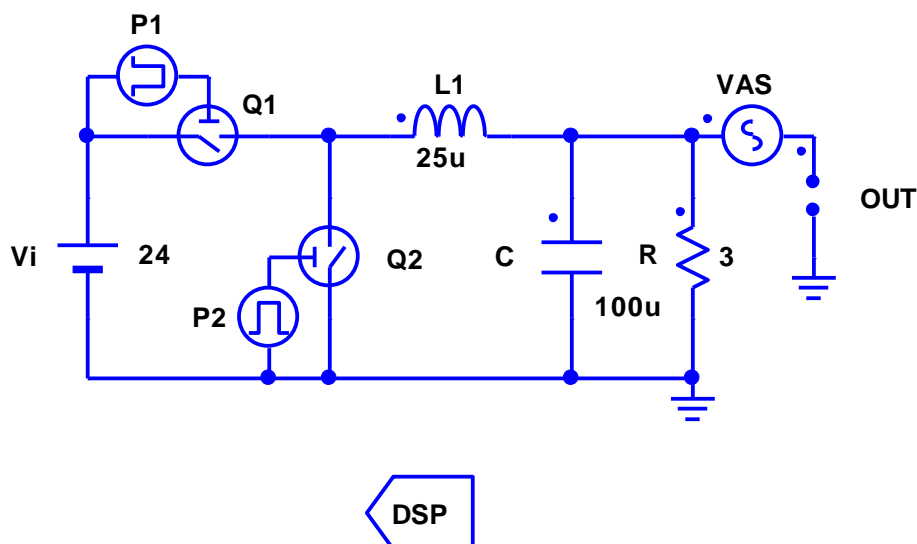
K492 では、今後主流になる DSP によるデジタル制御を解析するために DSP 素子と DSP ブロックが用意されており、以下のような特長があります。

- [1] DSP プログラムはブロック図で作成する。
- [2] 各ブロックは、単純な機能を持っており、内部にその機能を現す小さなプログラムが記述されている。

- [3] 標準的なブロックプログラムはライブラリから読み込んで使用する。
- [4] 全体のプログラムは、ブロック図から自動生成される。
- [5] 作成された DSP プログラムは、K492 の持つ解析機能によって、容易に検証できる。

(1) メイン回路の作成

DSPで制御するメイン回路を準備します。ここでは、下図のような同期整流による降圧形コンバータを例として使用します。そこで、Topology モードに移行して、作図できるようにします。



次に、素子メニューの DSP:DSP をクリックして DSP 素子を作成し、これを右クリックすると DSP ウィンドウが表示されます。DSP ウィンドウについては、別項を参照してください。

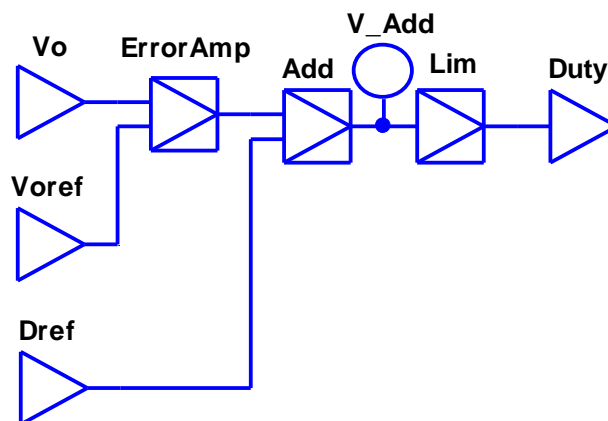
(2) ブロック図の作成

次に、DSP 素子で実行するプログラムを DSP ウィンドウ内にブロック図で構成します。この段階では、プログラムの詳細については、全く考える必要はありません。実行したい内容全体をおおまかに考えてブロック図にします。

ここでは、単純なPWM制御を作ってみます。PWM制御の大まかな流れは、以下のようになります。()内は、下図の各ブロックの素子メニュー名とシンボルを表しています。

- [1] 出力電圧の検出(Input, Vo)
- [2] 基準出力電圧の設定(Input, Voref)
- [3] 基準時比率の設定(Input, Dref)
- [4] 誤差信号生成(Block2, ErrorAmp)
- [5] 誤差信号と基準時比率の加算(Block2, Add)
- [6] 時比率制限(Block1, Lim)
- [7] 時比率設定(Output, Duty)
- [8] 内部信号検出(Detector, V_Add)

これを、そのままブロック図に表すと下図のようになります。



このブロック図は、ほとんど回路と同様な感覚で作成できます。ブロックにその機能を端的に表すシンボルをつけ、信号の流れを考えて、回路を接続するときと同じように各ブロックを接続します。このブロック図だけで、プログラムのおおまかな動作が簡単に理解できると思います。

(3) ブロックプログラムの読み込み

次に、各ブロックのプログラムの作成を行います。通常は、ブロックプログ

ラムは、ほとんど作成する必要はなく、ライブラリから読み込むだけです。

[1] ブロック Vo

ブロックを右クリックすると、パラメータダイアログボックスが表示されます。プログラム記述部分には、空をあらわす ?EMPTY? が表示されています。そこで、LIB ボタンをクリックして、あらかじめ容易されているライブラリから適当なプログラム（拡張子は BLK）を読み込みます。ライブラリについては、付録を参照してください。ブロック Vo の機能は、出力電圧の検出なので、"SENSE_V.BLK"を読み込みます。すると、次のようなプログラムが表示されます。

```
/* PROGRAM */  
$OUT = output("?SYM", "V", "AVE");
```

最初の行は、コメント文で、プログラムセクションであることを示しています。次の行が電圧を検出する K492 特有の記述です。これらの文法については、付録を参照してください。ここで、"?SYM"のように?で始まる部分は、ユーザーが必ず書き直して設定しなければなりません。ここでは、次のように電圧検出するメイン回路の素子名"OUT"を設定します。

```
$OUT = output("OUT", "V", "AVE");
```

ここで、変数\$OUT は、ブロックの出力を表す特殊変数です。これについては、後述します。以上でこのブロックのプログラム作成は終了ですので、OK ボタンをクリックします。

[2] ブロック Voref

このブロックでは、基準電圧を設定します。ライブラリから"INPUT.BLK"を読み込むと、次のようなプログラムが表示されます。

```
/* PROGRAM */  
$OUT = ?VAL;
```

ここで、?VAL を基準電圧値に変更します。ここでは、12V とします。

```
$OUT = 12;
```

[3] ブロック Dref

これも、ブロック Vref と同様にして、基準時比率を 0.5 に設定します。

```
$OUT = 0.5;
```

[4] ブロック ErrorAmp

このブロックは、誤差増幅器です。“DIF_AMP.BLK”を読み込むと次のようなプログラムが表示されます。

```
/* PROGRAM */  
$OUT = ?GAIN * ($IN1 - $IN2);
```

ここで、?Gainを書き換えて-0.5とします。\$IN1と\$IN2はブロックの入力を表す特殊変数で、ブロック上側入力が\$IN1、下側入力が\$IN2となります。複数入力ブロックの場合、同様にして\$IN1～\$IN4まであります。一入力ブロックの場合は、単に\$INとなります。

```
$OUT = -0.5 * ($IN1 - $IN2);
```

[5] ブロック Add

このブロックは、加算器です。“ADD_2.BLK”を読み込むと次のようなプログラムが表示されるので、そのまま使用します。

```
/* PROGRAM */  
$OUT = $IN1 + $IN2;
```

[6] ブロック Lim

このブロックは、時比率用のリミッタです。“LIMITER.BLK”を読み込みます。

```
/* PARAMETER */  
define @MAX (?VAL)  
define @MIN (?VAL)  
/* PROGRAM */
```

・
・

このプログラムは、PARAMETER セクションと PROGRAM セクションの二つのセクションに分かれています。このような場合は、PARAMETER セクションに define 文を使用した定数設定プログラムが記述されているので、?VAL を時比率上限値 0.9 と下限値 0.1 に設定します。

```
define @MAX (0.9)
```

```
define @MIN (0.1)
```

[7] ブロック Duty

このブロックは、計算した時比率をメイン回路のパルス発生器 P1、P2 に設定します。“SET_PARAM_2.BLK”を読み込みます。

```
/* PROGRAM */
```

```
setparam("?SYM1", "?PAR1", $IN);
```

```
setparam("?SYM2", "?PAR2", $IN);
```

ここで、“?SYM1”、“?PAR1”を“P1”、“T0”に、“?SYM2”、“?PAR2”を“P2”、“T0”に設定します。ただし、P1 と P2 の Initial State をそれぞれ ON と OFF にしておきます。

```
setparam("P1", "T0", $IN);
```

```
setparam("P2", "T0", $IN);
```

[8] ブロック V_Add

このブロックは、プログラム内部信号を検出します。ブロックの出力端子に接続して適当なシンボルを設定します。すると、接続した検出ポイントがブロックの大域シンボル名（ここでは“DSP.V_Add.V”）で、メインメニューの Parameter:Output で出力変数として表示されますので、出力変数に設定すれば波形観測が可能になります。これで、ブロック図の完成です。

(4) ブロック図の保存と読み込み

次に、作成したブロック図を Menu:Save を使用して保存します。また、カスタム回路と同様に、メインウィンドウの Save コマンド実行時に、固有のファイル名でブロック図も保存されます。このとき、ファイル名の最初に“DSP”が付加されます。また、既存のブロックファイルを Menu:Read で読み込むことができます。この読み込みは解析、図形の両モードで可能で、解析モードで読み込んだ場合、そのブロック図で直ちに解析可能です。

(5) 出力変数の設定

次に、解析モードに以降し、必要な出力変数を設定します。設定する出力変数は、ブロック V_o で検出するコンバータ出力のオープン素子 OUT の電圧と、ブロック V_Add で検出する内部信号 "DSP.V_Add.V" です。特に、OUT の電圧のように、プログラムの output 関数で使用される出力は、出力変数として定義していなければプログラムエラーとなります。

(6) 全体プログラムの確認

ブロック図から全体プログラムが自動生成されます。作成された全体プログラムを確認するには、解析モードで DSP ウィンドウを表示して、Menu:Program をクリックします。また、プログラムエラー表示の行番号は、この全体プログラムの行番号を表しています。Menu:Block Order をクリックすると、ブロックの実行順序が表示されます。

(7) ブロックプログラムの作成法

ブロックのプログラムは、(3) では、ライブラリから読み込んで作成しましたが、ライブラリに存在しない機能は、ユーザー自身が作成する必要があります。その作成方法は以下のとおりです。基本的な言語文法は PRC などのプログラマブル素子の言語と同じです。

[1] セクション

ブロックプログラムは、次の3つのセクションから構成されます。このセクションによる分割は必須ではありませんが、プログラムを理解しやすくするために、三つのセクションから適当なものを選択して、必ず実行してください。

①説明セクション(/* EXPLANATION */)

コメント文でプログラムの内容を説明するセクションです。自明な場合は省略してもかまいません。

②パラメータ設定セクション(/* PARAMETER */)

プログラムで使用される定数を設定するセクションです。通常、define 文を利用して行います。単純な代入文でも可能ですが、定数設定を明確にするために、define 文を利用します。特に、プログラムの使用状況に応じて変更される定数を記述します。

③プログラムセクション(/* PROGRAM */)

機能を表すプログラムを記述するセクションです。

[2] 入出力変数名

入出力を表す変数名は、次のように定義されていますので、他の意味で使用できません。

\$IN : 一入力ブロックの入力

\$INn (nは1～4) : 複数入力ブロックの入力

\$OUT : 出力

これらの変数名は、全体プログラム生成時に、出力の場合は、そのブロックの大域シンボルに置き換えられ、入力の場合は、接続しているブロックの大域シンボルに置き換えられます。ただし、ピリオドは全てアンダーバーに置き換えられます。以下同様です。たとえば、ブロック1の出力がブロック2の入力に接続している場合、DSP素子のシンボルが"DSP"、ブロック1、ブロック2のシンボルが"Block1"、"Block2"の場合、ブロック2のプログラムの\$IN と\$OUTは次のように変換されます。

\$IN -> DSP_Block1

\$OUT -> DSP_Block2

[3] 定数名、変数名

プログラムで使用する定数や変数は、プログラム言語で認識される通常の定数名や変数名も使用できますが、名前の衝突を避けるために、先頭に次の特殊記号のどれかをつけて使用します。

①\$を付加

記号\$は、全体プログラムが生成される時、ブロックの大域シンボルと置き換えられます。たとえば、DSP素子のシンボルが"DSP"で、ブロックのシンボルが"Block"の場合、そのブロックのプログラムの変数名\$Var は次のように変換されます。通常、変数はこの方法で使われます。

\$Var -> DSP_Block_Var

②@を付加

ブロックの大域シンボルを全て大文字にして付加します。通常、define で定数を定義する時に使われます。

@Var -> DSP_BLOCK_Var

③#を付加

ブロックの大域シンボルを全て小文字にして付加します。

#Var -> dsp_block_Var

[4] プログラム記述

ブロックプログラムは、そのブロックの機能だけ、すなわち、入力\$IN に対して出力\$OUT がどうなるのかだけを記述します。他のブロックのことは一切考える必要はありません。したがって、プログラム記述が局所化され非常に容易になります。

[5] ライブラリ化

作成したプログラムは、汎用性があればライブラリ化しておきます。このとき、ユーザーに設定させる部分には特殊記号?を使用して汎用化しておきます。既存のライブラリを参考にしてください。ライブラリ化するには、ブロックのパラメータダイアログボックスの Save ボタンを使います。また、テキストエディタを使用して作成してもかまいません。拡張子は"BLK"です。

[6] C (C++) 言語への移植

K492 で使われているプログラミング言語は、オリジナル言語ですが、文法

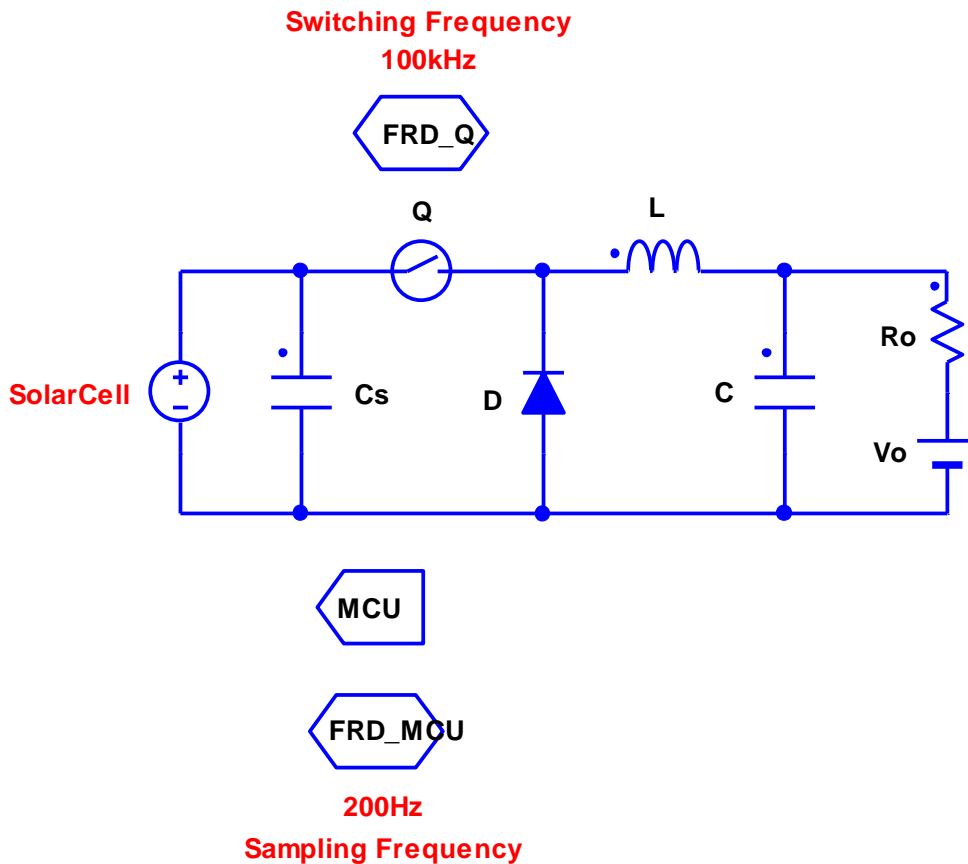
仕様はCやC++の仕様とほぼ同じですので、作成したDSPプログラムは、簡単にこれらの言語に移植できます。ただし、次のような違いがあります。

- ①データ型や変数宣言がない
- ②関数を使用できない
- ③define 文には先頭の#がない
- ④空文を表す ; のみの文は使用できない

移植するには、回路データファイルの CVT ファイルの DSP 部分に全体プログラムが記述されているので、これをコピーして使用します。

6. 10 太陽電池最大電力点追跡 (MPPT) 制御

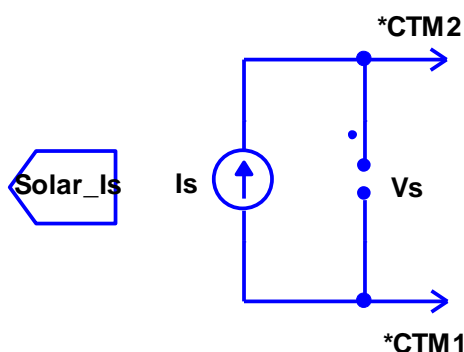
本節では、太陽電池のMPPT制御について説明します。例として下図のような降圧形コンバータを使用した回路で説明します。この回路は、ライブラリのサンプル回路 (SCATLIB¥SOLAR¥SAMPLE¥SolarCell_Buck.cvt) として用意されていますので参考にしてください。



(1) 太陽電池のモデル化

まず、カスタム回路を使用して太陽電池をモデル化します。まず、カスタム化する二端子素子を素子メニューから選択します。ここでは、PrgVoltage 素子を選び Customize コマンドでカスタム化します。カスタム化については別項を参照してください。次に、この素子を右クリックしてカスタムウィンドウを開いてシンボルを SolarCell に設定します。次に、ライブラリから下図のような太陽電池モデルのカスタム回路を読み込みます。太陽電池モデルはダイオードモデル (SCATLIB¥SOLAR¥MODEL¥solarcell-diode-i.cvt) と数学モデル (SCATLIB¥SOLAR¥MODEL¥solarcell-math-i.cvt) の2種類が用意されており、ここでは、特性パラメータ設定が容易な数学モデルを使用します。このカスタム回路から分かるように、太陽電池は電流源 I_s で表現されており、P R C素子 Solar_Is の中に太陽電池の特性を表すプログラムが書かれています。プログラムでは、太陽電池の電圧 V_s を検出して数学モデルを用いて太陽電池 I_s を計算

し電流源に設定しています。



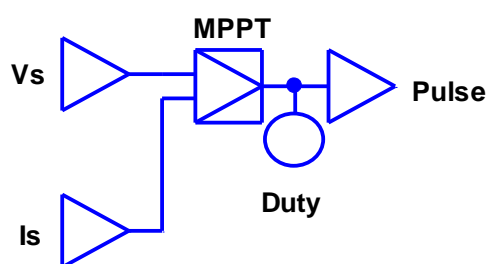
次に、Solar_Is を右クリックすると、P R Cのパラメータダイアログボックスが現れ特性プログラムが表示されます。そこで、太陽電池の最大電圧 (V_{oc})、最大電力点電圧 (V_{mp})、最大電流 (I_{sc})、最大電力点電流 (I_{mp})を設定し、ダイアログボックスを閉じます。これで、太陽電池のカスタム回路によるモデル化は完成です。Solar_Is は後ほど設定するメイン周波数で動作します。最後に、Solar_Is が検出する太陽電池の電圧 SolarCell.Vs をモード FIN で出力変数に設定します。

(2) コンバータ回路の作成

次に、M P P Tを行うコンバータ回路を作成します。ここでは、降圧形コンバータを使用しており、負荷には蓄電池 V_o を想定しています。 R_o は蓄電池の内部抵抗です。また、太陽電池の電圧を安定にするため平滑キャパシタ C_s を太陽電池に並列に接続します。コンバータを作成した後、F R D素子を準備し、コンバータのスイッチング素子をこのF R Dに登録して、F R Dにスイッチング周波数を設定します。ここでは、F R Dのシンボルを FRD_Q とし、スイッチ Q をFRD_Qに登録します。周波数は100kHzとします。次に、メイン周波数をスイッチング周波数の少なくとも10倍の以上に設定します。ここでは1MHzとします。

(3) MPPT制御部の作成

最後に、MPPT制御部を、DSPを用いて作成します。まず、DSP素子を準備してシンボルをMCUとします。次に、ライブラリから下図のようなMPPTブロックファイル(SCATLIB¥SOLAR¥MODEL¥mppt-gradient-power.cvt)を読み込みます。このファイルのMPPTアルゴリズムは、標準的な山登り法です。



次に、MPPTに必要な太陽電池の電圧 V_s と電流 I_s を検出する設定を行います。入力ブロック V_s を右クリックしてプログラムを表示し、Output 関数で太陽電池の電圧のシンボルを設定してください。このとき、モードは必ず INI にします。ここでは、太陽電池のシンボルが SolarCell なので、SolarCell.Vs とします。同様にして、入力ブロック I_s で、太陽電池の電流を設定してください。次に、出力パルスを設定します。出力ブロック $Pulse$ を右クリックして setparam 関数でコンバータのスイッチ Q のパルスを設定してください。その後、これらの検出する太陽電池の電圧 SolarCell.Vs と電流 SolarCell.Is をモード INI で出力変数に設定します。また、太陽電池の電力波形を得るために出力変数の演算機能を用いて、SolarCell.Vs と SolarCell.Is の積を出力変数として設定します。さらに、出力パルスを確認する場合は、MCU.Duty.V を出力変数に設定します。モードは任意です。最後に、FRD素子を用意して、MCU を登録し、このFRDでサンプリング周波数を設定します。ここでは 200Hz とします。これで、全ての回路が完成です。Tranjient 解析を 100000 回ほど行い、MPPTが行われるか試してください。

(4) 太陽電池特性やMPPTアルゴリズムの変更

K492 では、太陽電池の特性やMPPTアルゴリズムはプログラムを変更することにより、ユーザーが自由に変更定義できます。たとえば、太陽電池モデルを数学モデルからダイオードモデルに変更する場合は、SolarCell を右クリックし、さらに Solar_Is を右クリックしダイアログボックスを表示させます。次に、Lib ボタンでライブラリの SCATLIB¥SOLAR¥BLK¥SOLAR_CELL_DIODE_I.BLK を読み込んで、必要な特性パラメータを設定すれば完了です。特性パラメータについては付録を参照してください。また、SolarCell のカスタムウィンドウでカスタム回路レベルでも変更できます。SolarCell のカスタムウィンドウで、ライブラリの SCATLIB¥SOLAR¥MODEL¥solarcell-diode-i.cvt を読み込んで、Solar_Is の特性パラメータを設定します。同様に、MPPTアルゴリズムも変更できます。ライブラリには山登り法しか登録されていないので、ユーザー独自のアルゴリズムをライブラリ登録して変更してみてください。

6. 1 1 制御順序

パルス変調器、周波数変調器、同期制御器、トリガなどを組み合わせて様々な制御を行うことができます。これらの素子が複数存在する場合の制御順序は以下のとおりです。

- [1] まず、主回路に設定されたスイッチング周波数またはトリガによって、スイッチング動作の一周期が終了します。
- [2] 次に、演算器素子が機能し演算器素子の出力が更新されます。
- [3] 次に、DSP 素子が機能します。
- [4] 次に、PRC 素子が機能します。
- [5] 次に、[4]までに決定された状態において、周波数変調器の入力電圧が計算され、それに基づいてスイッチング周波数が更新されます。
- [6] 次に、[5]までに決定された状態において、パルス変調器の入力電圧が計算され、それによって変調される状態変化時刻が更新されます。このとき、時比率制御を行っている場合の時比率の計算には、[5]までに決定されたスイッチン

グ周期が使用されます。

[7] 次に、[6]までに決定された状態において、素子パラメータ変調器の入力電圧が計算され、それによって変調される素子パラメータ値が更新されます。

[8] 次に、[7]までに決定された状態において、同期制御器によって制御される状態変化時刻が更新されます。

[9] 同じ素子が複数ある場合は、回路図に登録された順番で機能します。

7. アルゴリズムの概要

本章では、アルゴリズムに関する事項について説明します。

7. 1 トポロジー条件

K492 では、インダクタ、キャパシタ、電圧源、電流源の接続のしかたによっては、解析できない場合があります。このような場合、エラーメッセージが表示されます。この不正な接続を見分けるには、まず、電圧源は短絡、電流源は開放とします。この状態でインダクタの並列接続やキャパシタの直列接続が不正となります。この問題を回避する最も簡単な方法は、インダクタには小さな抵抗を直列に、キャパシタと電圧源には大きな抵抗を並列に付加します。

インダクタの直列接続やキャパシタの並列接続は不正な接続ではありませんが、いくつかの状態変数は従属になり状態変数ではなくなります。状態変数ではなくなったインダクタの電流やキャパシタの電圧の初期値は設定できません。

また、電圧、電流が無限大になりうるような回路も解析できません。例えば、直流電流源でキャパシタを直接充電するような回路は、キャパシタの電圧が無限に増加していくので解析できません。この場合も適当に損失をいれることにより避けることができます。

7. 2 解析アルゴリズム

本節では、K492 で採用されているアルゴリズムについて、その概略を説明します。

(1) 区分的線形性

従来のシミュレータは定常解析に多数の逐次計算を必要とし非常に時間がかかります。これらのシミュレータは対象回路の最も短い時定数に対して十分な解析精度を得るようにきざみ時間を調整します。スイッチング電源の場合、時定数の非常に短いスイッチング動作が行われますが、基本特性はフィルタの長

い時定数により決定されます。このような解析を、従来のアルゴリズムで忠実に
行えば1個の定常点を求めるにも莫大なくり返し計算が必要になります。これ
が、従来のシミュレータが定常解析に長時間かかる理由です。一方、K492は
スイッチモデルを簡略化し、スイッチング時間を考慮しません。このとき、区
分的線形性という性質が利用でき、きざみ時間は状態解析のためある程度必要
になるものの、それほど細かくきざむ必要はないので大幅な時間の節約になり
ます。それでは、区分的線形性とはどういう性質をいうのでしょうか。スイッ
チやダイオードを抵抗で置き換えれば、スイッチング電源は、単なる線形回路
と見なせます。ただし、これらの抵抗値はスイッチのオン・オフ状態に応じて
変化します。このように、スイッチング動作を線形回路の切り替わり動作とみ
なせる性質を区分的線形性と呼びます。この場合、様々な線形解析手法が区
分的に利用でき解析が容易になります。しかし、この前提条件として次に述べる
状態解析が必要です。

(2) 状態解析アルゴリズム

状態解析とは、一周期のスイッチ素子のオン・オフ状態を決定することを言
い、スイッチング電源解析の中核です。これが完全に行えれば、定常解析と過
渡応答解析は比較的簡単に実現できます。パルス発生器やPWMスイッチの状態
変化時刻はあらかじめ与えられるので問題ありませんが、スイッチやダイオ
ードの状態変化は回路の内部状態に依存します。K492のこれらの状態決定には
他のシミュレータにはない様々な独自の効率的なアルゴリズムが用いられてい
るので、高速かつ正確です。K492はスイッチやダイオードのバイアス電圧を監
視し、その状態変化を検出します。このとき、ある程度時間をきざむ必要があ
りますが、K492は独自のきざみ時間の最適化アルゴリズムにより、これを実行
時に動的に変化させ、解析時間を最小限に押さえるようプログラムされていま
す。また、矛盾した決定をした場合の誤り訂正機能も施されています。それで
も状態決定に失敗した場合、状態ウィンドウに以下のようにその旨を知らせま
す。

[1] S : 状態決定失敗

[2] T : 状態変化時刻決定失敗

(3) トポロジー解析

トポロジー解析とは、与えられた回路データから解析に必要な行列やベクトルを導出することをいいます。K492 のトポロジー解析は行列、ベクトルの従属、独立性の理論に基づいて考案された独自のアルゴリズムで、素子数が多くなっても正確に計算できるように工夫されています。

(4) 過渡応答解析アルゴリズム

過渡応答解析アルゴリズムは比較的容易です。スイッチング周期の更新に伴って、状態変数の初期値などを更新しながら、(1) で述べた状態解析アルゴリズムを繰り返すだけです。このとき、交流電源や波形発生器の出力および変調器で変調される信号などは一周期内で一定とみなされます。したがって、交流電源、波形発生器、変調信号などの周波数はスイッチング周波数に比べて十分低いことが必要です。ただし、変調信号に対しては K492 は誤差を補正する機能を持っています。したがって目安として、交流電源や波形発生器の周波数はスイッチング周波数の百分の一以下、変調信号の周波数は十分の一以下であれば問題はないでしょう。この変調信号の周波数は一般に変調器の変換ゲインが大きくなるにつれて高くなるので、解析においては、変換ゲインはなるべく小さくしておいてください。

(5) 定常解析アルゴリズム

定常特性は(1)でも述べたように、シミュレータにとって解析が最も困難なものの一つです。K492 は区分的線形性を単に利用するだけでなく、DC-DCコンバータに対してはニュートン法を用いて定常解析を行います。ただし、この定常解析は必ずしも成功するとは限らないので注意が必要です。

7. 3 解析誤差

K492 の解析誤差は、0.1%程度です。たとえば、電圧値の振幅が数Vの場合、数mVの誤差が含まれていると考えてよいでしょう。スイッチング電源の場合、部品や測定の誤差を考えるとこれぐらいの解析精度で十分です。

8. エラーメッセージ

K492 では、不正な計算や動作を行った場合、エラーメッセージまたは警告メッセージを表示します。本章では、これらのメッセージの意味とその対処方法を説明します。

8. 1 解析エラーメッセージ

解析エラーメッセージには以下のようなメッセージがあります。

[1] "Inverse matrix may be invalid."

逆行列が不正です。

[2] "Reduced matrix may be invalid."

縮約行列が不正です。

[3] "Topology may be invalid."

回路のトポロジーが不正です。

これらのメッセージは、ほとんどの場合、回路がトポロジー条件を満たしていないのが原因です。インダクタ、キャパシタ、電圧源、電流源の接続関係を変更する必要があります。これらの素子に直列または並列に適切な抵抗を接続すればほとんどの場合解決します。トポロジー条件については「7. 1 トポロジー条件」を参照してください。

8. 2 入出力エラーメッセージ

ファイルの読み込みやプリント関係のエラーメッセージには以下のようなメッセージがあります。

[1] "Failed to open a notepad."

メモ帳がありません。

[2] "Failed to print."

プリンターが接続されていないか接続不良です。

[3] "Failed to read data files."

回路データファイルの読み込みに失敗したか、データの整合性があっていません。

これらのエラーに対しては、データを設定または作成しなおしてください。

8. 3 警告メッセージ

警告メッセージには以下のメッセージがあります。

[1] "An invalid device is detected."

不正な素子があります。パラメータを正しく設定してください。主な誤りとしては、シンボルの欠如や重複、値の範囲エラー、変調器の変調信号の対象エラーなどがあります。

[2] "Invalid connection is detected."

不正な接続があります。接続を正しく行ってください。

[3] "No outputs are set."

出力変数が設定されていません。Parameter メニューの Output を実行して出力変数を設定してください。

[4] "No printers are installed."

プリンタが接続されていません。Windows の設定方法にしたがってプリンタを設定してください。

[5] "No state analysis is done."

状態解析が一度も行われていない状態で波形解析を 0 サイクルで行うことはできません。正のサイクル数で行ってください。

[6] "Invalid data."

入力データが不正です。データを訂正してください。

[7] "Cannot delete an operation operand."

演算出力変数の演算対象出力変数を削除しようとしています。まず、演算出力変数を削除してください。

付録 A 高調波規制の概略

スイッチング電源に関する高調波規制の概略を説明します。高調波規制は、ライン入力電流に対する規制であり、以下の項目があります。

*高調波規制でのクラス判別及びグループの詳細は最新の高調波規制関係法令・省令・政令・ガイドラインなどを参照ください。

1. クラス

クラスは、機器に応じてクラスA～D、その他と分けられ、スイッチング電源の対象となるおもなクラスはクラスAとクラスDです。

2. 周波数スペクトラム規制値

(1) クラスA

奇数高調波次数	最大許容高調波電流 [A]
3	$2.30 * h$
5	$1.14 * h$
7	$0.77 * h$
9	$0.40 * h$
11	$0.33 * h$
13	$0.21 * h$
$15 \leq n \leq 39$	$(0.15 * 15 / n) * h$
偶数高調波次数	最大許容高調波電流 [A]
2	$1.08 * h$
4	$0.43 * h$
6	$0.30 * h$
$8 \leq n \leq 40$	$(0.23 * 8 / n) * h$

(2) クラスB

クラスAの最大許容高調波電流の1.5倍

(3) クラスC

奇数高調波次数	基本波入力電流の百分率 [%]
3	30 * 回路力率
5	10
7	7
9	5
$11 \leq n \leq 39$	3
偶数高調波次数	基本波入力電流の百分率 [%]
2	2

(4) クラスD

高調波次数	電力比例限度値 [mA]
3	$3.4 * P_{in} * h$
5	$1.9 * P_{in} * h$
7	$1.0 * P_{in} * h$
9	$0.5 * P_{in} * h$
11	$0.35 * P_{in} * h$
$13 \leq n \leq 39$	$(3.85 / n) * P_{in} * h$

h : 230 / 公称入力電圧 [V]

P_{in} : 入力電力 [W]

3. グループ

(1) Group 1 : 高調波グループ

$$G_{g,n}^2 = \frac{C_{k-5}^2}{2} + \sum_{i=-4}^4 C_{k+i}^2 + \frac{C_{k+5}^2}{2} \quad \{50\text{Hz system}\}$$

$$G_{g,n}^2 = \frac{C_{k-6}^2}{2} + \sum_{i=-5}^5 C_{k+i}^2 + \frac{C_{k+6}^2}{2} \quad \{60\text{Hz system}\}$$

(2) Group 2 : 高調波サブグループ

$$G_{s,g,n}^2 = \sum_{i=-1}^1 C_{k+i}^2$$

(3) Group 3 : 次数間高調波中心サブグループ

$$G_{i \in g,n}^2 = \sum_{i=2}^8 C_{k+i}^2 \quad \{50\text{Hz system}\}$$

$$G_{i \in g,n}^2 = \sum_{i=2}^{10} C_{k+i}^2 \quad \{60\text{Hz system}\}$$

(4) Group 4 : 次数間高調波グループ

$$G_{i \in g,n}^2 = \sum_{i=1}^9 C_{k+i}^2 \quad \{50\text{Hz system}\}$$

$$G_{i \in g,n}^2 = \sum_{i=1}^{11} C_{k+i}^2 \quad \{60\text{Hz system}\}$$

付録 B プログラマブル素子言語文法

プログラマブル素子で利用できるプログラム言語の文法は以下のとおりです。

(1) 予約語

if, else, while, init, define, return, quit

(2) 予約定数

PI : π

PI2 : 2π

true : ブール値の真

false : ブール値の偽

ON : スイッチング素子のオン状態

OFF : スイッチング素子のオフ状態

(3) 数学関数

sin(x) : 正弦

cos(x) : 余弦

tan(x) : 正接

asin(x) : 逆正弦

acos(x) : 逆余弦

atan(x) : 逆正接

atan2(y, x) : y/x の逆正接

sqrt(x) : ルート

log(x) : 自然対数

log10(x) : 常用対数

exp(x) : 指数関数

pow(x, y) : x の y 乗を計算する冪関数

abs(x) : 絶対値
min(x, y) : 最小値
max(x, y) : 最大値
mod(x, y) : x を y で割った余りを出す剰余関数
int(x) : 整数化する

(4) 機能関数

output("sym", "type", "mode") :

シンボル sym、タイプ type、モード mode で指定された出力変数の現在値を返す関数。シンボルは"sym"のように二重引用符で挟む。以下同様とする。

output("sym", "type", "mode", time) :

シンボル sym、タイプ Type、モード mode で指定された出力変数の時間 time における値を返す関数

param("sym", "par") :

シンボル sym を持つ素子の par で指定されたパラメータ値を返す関数。par には、各素子のパラメータダイアログボックスに表示されているパラメータシンボルを指定する。

setparam("sym", "par", value) :

シンボル sym を持つ素子の par で指定されたパラメータ値を value に設定する関数。par には、各素子のパラメータダイアログボックスに表示されているパラメータシンボルを指定する。

(5) 演算子

四則演算子 +, -, *, /

余り演算子 %

インクリメント、デクリメント演算子 ++, --

シフト演算子 <<, >>

不等号 <, >, <=, >=

比較演算子 ==, !=

論理演算子 !(NOT), &&(AND), ||(OR)

(6) 変数

全ての変数は、宣言なしで使用でき型はない。必ず0で初期化される。

(7) 文

文末には終端記号の';'をつける。

例 c = a + b;

(8) if 文

```
if (condition) {  
    文1 ;  
    文2 ;  
}  
else {  
    文3 ;  
    文4 ;  
}
```

condition が真のとき文1, 文2を、偽のとき文3, 文4を実行する。

else 文は省略可能。また、単文のときは { } は省略可能。

(9) while 文

```
while (condition) {  
    文1 ;  
    文2 ;  
}
```

condition が真のときループする。また、単文のときは { } は省略可能。

(10) init 文

```
init {  
    文1 ;  
    文2 ;  
}
```

プログラムの最初に一回だけ実行される。単文のときは { } は省略可能。

(11) define 文

```
define 定数名 (数または式)
```

定数を定義する。定数を変更しようとするときエラーになる。定数部分の括弧は必ず付ける。文末の終端記号は不要である。

(12) return 文

```
return;
```

プログラム実行を終了し解析に戻る。

(13) quit 文

```
quit;
```

プログラム実行を終了し解析も終了する。

(14) コメント

コメント文は /* と */ で、文との間にスペースを入れて挟む。

付録 C スクリプト言語文法

スクリプト言語の文法は以下のとおりです。大文字小文字は区別されます。

(1) 予約語

if, else, while, define, return

(2) 予約定数

PI : π

PI2 : 2π

true : ブール値の真

false : ブール値の偽

ON : スイッチング素子のオン状態

OFF : スイッチング素子のオフ状態

(3) 数学関数

sin(x) : 正弦

cos(x) : 余弦

tan(x) : 正接

asin(x) : 逆正弦

acos(x) : 逆余弦

atan(x) : 逆正接

atan2(y, x) : y/x の逆正接

sqrt(x) : ルート

log(x) : 自然対数

log10(x) : 常用対数

exp(x) : 指数関数

pow(x, y) : x の y 乗を計算する冪関数

abs(x) : 絶対値
min(x, y) : 最小値
max(x, y) : 最大値
mod(x, y) : x を y で割った余りを出す剰余関数
int(x) : 整数化する

(4) 機能関数

Clear() : 初期状態をクリアする。
ClearContinue() : 継続モードを解除する。
ClearOutput() : 出力変数を全て消去する。
CloseCMP() : CMP ウィンドウを閉じる。
CloseFFT() : FFT ウィンドウを閉じる。
CloseFRA() : FRA ウィンドウを閉じる。
CloseMain() : メインウィンドウを閉じる。
CloseState() : 状態ウィンドウを閉じる。
CloseWVF() : WVF ウィンドウを閉じる。
Continue() : 継続解析を行う。
DeleteOutput(id) : id 番目の出力変数を削除する。
ExecuteCMP(id, show) :
CMP ウィンドウを生成し、id 番目の出力変数を CMP ウィンドウに登録する。show が true なら CMP ウィンドウを表示し、false なら表示しない。
ExecuteFFT(id, show) :
FFT ウィンドウを生成し、id 番目の出力変数の FFT 解析を行う。show が true なら FFT ウィンドウを表示し、false なら表示しない。
ExecuteFRA("sym", id1, id2, show) :
FRA ウィンドウを生成し、シンボル sym をもつ ACSweep 素子によって、id1 番目の出力変数に対する id2 番目の出力変数の周波数特性解析を行

う。show が true なら F R A ウィンドウを表示し、false なら表示しない。
シンボルは、"sym" のように二重引用符で挟む。以下同様とする。

OpenState(show) :

状態ウィンドウを生成する。show が true なら F R A ウィンドウを表示し、false なら表示しない。

OpenWVF(show) :

W V F ウィンドウを生成する。show が true なら F R A ウィンドウを表示し、false なら表示しない。

Output(id) : id 番目の出力変数の現在値を返す。

OutputID("sym", "type", "mode") :

シンボル sym、タイプ type、モード mode で指定された出力変数の ID 番号を返す。

Param("sym", "par") :

シンボル sym を持つ素子の par で指定されたパラメータ値を返す。par には、各素子のパラメータダイアログボックスに表示されているパラメータシンボルを指定する。

ReadCircuit("file") : ファイル名 file の C V T 回路データを読み込む。

ReadCircuit("filehead", n) :

ファイル名 filehead+シリアル番号 n (例 circuit5 など) の C V T 回路データを読み込む。

SaveCircuit("file") :

現在の回路を、ファイル名 file の C V T ファイル保存する。

SaveCircuit("filehead", n) :

現在の回路を、ファイル名 filehead+シリアル番号 n (例 circuit5 など) の C V T 回路データに保存する。

SaveCMP("file") :

C M P ウィンドウの波形データを、ファイル名 file の C M P ファイルに保存する。

SaveCMP("filehead", n) :

CMP ウィンドウの波形データを、ファイル名 filehead+シリアル番号 n (例 circuit5 など) の CMP ファイルに保存する。

SaveFFT("file") :

FFT ウィンドウの波形データを、ファイル名 file の FFT ファイルに保存する。

SaveFFT("filehead", n) :

FFT ウィンドウの波形データを、ファイル名 filehead+シリアル番号 n (例 circuit5 など) の FFT ファイルに保存する。

SaveFRA("file") :

FRA ウィンドウの波形データを、ファイル名 file の FRA ファイルに保存する。

SaveFRA("filehead", n) :

FRA ウィンドウの波形データを、ファイル名 filehead+シリアル番号 n (例 circuit5 など) の FRA ファイルに保存する。

SaveWVF("file") :

WVF ウィンドウの波形データを、ファイル名 file の WVF ファイルに保存する。

SaveWVF("filehead", n) :

WVF ウィンドウの波形データを、ファイル名 filehead+シリアル番号 n (例 circuit5 など) の WVF ファイルに保存する。

SetAC() : 全ての出力変数を AC モードにする。

SetAC(id) : id 番目の出力変数を AC モードにする。

SetDC() : 全ての出力変数を DC モードにする。

SetDC(id) : id 番目の出力変数を DC モードにする。

SetOutput("sym", "type", "mode") :

シンボル sym、タイプ type、モード mode で指定された出力変数を設定する。

SetParam("sym", "par", value) :

シンボル sym を持つ素子の par で指定されたパラメータ値を value に設定する。par には、各素子のパラメータダイアログボックスに表示されているパラメータシンボルを指定する。

ShowCMP(show) :

show が true なら、CMP ウィンドウを表示し、false なら非表示にする。

ShowFFT(show) :

show が true なら、FFT ウィンドウを表示し、false なら非表示にする。

ShowFRA(show) :

show が true なら、FRA ウィンドウを表示し、false なら非表示にする。

ShowMain(show) :

show が true なら、メインウィンドウを表示し、false なら非表示にする。

ShowState(show) :

show が true なら、状態ウィンドウを表示し、false なら非表示にする。

ShowWVF(show) :

show が true なら、WVF ウィンドウを表示し、false なら非表示にする。

Steady() : Steady 解析を行う。

Transient(cycle) : Transient 解析を cycle 周期行う。

Update() : 初期状態を更新する。

Waveform(cycle) : Waveform 解析を cycle 周期行う。

(5) 演算子

四則演算子 +, -, *, /

余り演算子 %

インクリメント、デクリメント演算子 ++, --

シフト演算子 <<, >>

不等号 <, >, <=, >=

比較演算子 ==, !=

論理演算子 !(NOT), &&(AND), ||(OR)

(6) 変数

全ての変数は、宣言なしで使用でき型はない。必ず0で初期化される。

(7) 文

文末には終端記号の';'をつける。

例 c = a + b;

(8) if 文

```
if (condition) {  
    文1 ;  
    文2 ;  
}  
else {  
    文3 ;  
    文4 ;  
}
```

condition が真のとき文1, 文2を、偽のとき文3, 文4を実行する。

else 文は省略可能。また、単文のときは { } は省略可能。

(9) while 文

```
while (condition) {
```

```
    文1 ;  
    文2 ;  
}
```

condition が真のときループする。また、単文のときは { } は省略可能。

(10) define 文

```
define 定数名 (数または式)
```

定数を定義する。定数を変更しようとするときエラーになる。定数部分の括弧は必ず付ける。文末の終端記号は不要である。

(11) return 文

スクリプトプログラム実行を終了する。

(12) コメント

コメント文は /* と */ で、文との間にスペースを入れて挟む。

付録 D SCATLIB ライブラリ

標準的に添付されているライブラリについて解説します。

1. ファイル構成

ライブラリのディレクトリ構成は次のようになっています。

SCATLIB - DSP --- BLK	D S P機能プログラム
- SOLAR - BLK	太陽電池D S P機能プログラム
- MODEL	太陽電池カスタム回路
- SAMPLE	太陽電池サンプル回路

2. D S P機能プログラム

D S P制御でよく使用される機能プログラムライブラリです。これらのプログラムでは、演算は浮動小数で行われており、あくまで参考的なものと考えて使用してください。

(1) ADD_2.BLK

Block2用の加算器です。入力を加算します。設定パラメータはありません。

(2) ADD_3.BLK

Block3用の加算器です。入力を加算します。設定パラメータはありません。

(3) ADD_4.BLK

Block4用の加算器です。入力を加算します。設定パラメータはありません。

(4) DIF_AMP.BLK

Block2用の差動増幅器です。入力1から入力2を引いた差をゲイン倍します。設定パラメータはゲイン(GAIN)です。

(5) GET_PARAM.BLK

Input用です。メイン回路のパラメータの取得を行います。設定パラメータは、素子シンボル(SYM)とパラメータシンボル(PAR)です。

(6) INPUT.BLK

Input 用です。外部定数設定を行います。設定パラメータは、外部定数値 (VAL) です。

(7) LIMIT.BLK

Block1 用のリミッタです。入力値を制限して出力します。設定パラメータは上限値 (MAX) と下限値 (MIN) です。

(8) SET_PARAM.BLK

Output 用です。メイン回路のパラメータを設定します。設定パラメータは、素子シンボル (SYM) とパラメータシンボル (PAR) です。

(9) SET_PARAM_2.BLK

Output 用です。メイン回路のパラメータを2つ設定します。設定パラメータは、素子シンボル (SYM1, SYM2) とパラメータシンボル (PAR1, PAR2) です。

(10) PID_AMP.BLK

Block2 用の P I D 制御用の誤差増幅器です。一次の微分補償と積分補償が施されています。設定パラメータは、直流ゲイン (GAIN)、積分極周波数 (FPI)、積分零周波数 (FZI)、微分零周波数 (FZD)、微分極周波数 (FPD)、サンプリング周波数 (FS) です。

(11) SENSE_I.BLK

INPUT 用です。電流検出を行います。検出電流は、出力変数に設定している必要があります。設定パラメータは、素子シンボル (SYM) です。

(12) SENSE_V.BLK

INPUT 用です。電圧検出を行います。検出電圧は、出力変数に設定している必要があります。設定パラメータは、素子シンボル (SYM) です。

(13) ZERO_POLE.BLK

Block1 用です。一次の零と極を持つ伝達関数として働きます。設定パラメータは、零周波数 (FZ)、極周波数 (FP)、サンプリング周波数 (FS) です。

3. 太陽電池DSP機能プログラム

太陽電池特性とMPPT用の機能プログラムライブラリです。

(1) SOLAR_CELL_DIODE_I.BLK

太陽電池特性のダイオードモデルです。設定パラメータは、太陽電池の最大電圧(Voc)、最大電流(Isc)、セル直列接続数(n)、絶対温度(T)です。

(2) SOLAR_CELL_MATH_I.BLK

太陽電池特性の数学モデルです。設定パラメータは、太陽電池の最大電圧(Voc)、最大電力点電圧(Vmp)、最大電流(Isc)、最大電力点電流(Imp)です。

(3) MPPT_GRADIENT_POWER.BLK

山登り法によるMPPTアルゴリズムです。設定パラメータは、最小時比率(Dmin)、最大時比率(Dmax)、初期時比率(Dini)、刻み時比率(dDm)dです。

4. 太陽電池カスタム回路

太陽電池に関連したカスタム回路ライブラリです。

(1) solarcell-diode-i.cvt

太陽電池特性のダイオードモデル回路です。

(2) solarcell-math-i.cvt

太陽電池特性のダイオードモデル回路です。

(3) mppt-gradient-power.cvt

山登り法によるMPPTアルゴリズムを使用したDSPブロック回路です。

5. 太陽電池サンプル回路

太陽電池のMPPTサンプル回路例です。

(1) SolarCell_Boost.cvt

昇圧形コンバータを用いたMPPT回路です。

(2) SolarCell_Buck.cvt

降圧形コンバータを用いたMPPT回路です。