

# Gowin ソフトウェアのメッセージについてのヘルプドキュメント

## 著作権について（2020）

著作権に関する全ての権利は、**Guangdong Gowin Semiconductor Corporation** に留保されています。

何れの団体及び個人も、当社の書面による許可を得ず、本文書の内容の一部もしくは全部を、いかなる視聴覚的、電子的、機械的、複写、録音等の手段によりもしくは形式により、伝搬又は複製をしてはなりません。

## 免責事項

「GOWINSEMI®」、「LittleBee®」、「Arora」、及び GOWINSEMI のロゴは、当社により、中国、米国特許商標庁、及びその他の国において登録されています。商標又はサービスマークとして特定されたその他全ての文字やロゴは、[www.gowinsemi.com.cn](http://www.gowinsemi.com.cn) において記載されているそれぞれの権利者に帰属しています。当社は、**GOWINSEMI Terms and Conditions of Sale**（GOWINSEMI 取引条件）に規定されている内容を除き、（明示的か又は黙示的に拘わらず）いかなる保証もせず、また、知的財産権や材料の使用によりあなたのハードウェア、ソフトウェア、データ、又は財産が被った損害についても責任を負いません。本文書における全ての情報は、予備的情報として取り扱われなければなりません。当社は、事前の通知なく、いつでも本文書の内容を変更することができます。本文書を参照する何れの団体及び個人も、最新の文書やエラッタ（不具合情報）については、当社に問い合わせる必要があります。

## バージョン履歴

日付	バージョン	説明
2020/06/01	1.0J	初版。
2020/08/06	1.01J	<ul style="list-style-type: none"><li>● EX0311 ERROR を EX0210 WARN に変更。</li><li>● EX0206 WARN のコードを更新。</li><li>● EX0310 ERROR のコードを更新。</li></ul>

# 目次

目次 .....	i
<b>Gowin</b> ソフトウェアのメッセージについてのヘルプドキュメント .....	1
概要 .....	1
GowinSynthesis からのユーザーメッセージ .....	1
AG0100 .....	1
AG0101 .....	1
AG0200 .....	2
CK0013 .....	2
CK2027 .....	3
CV0003 .....	5
CV0005 .....	5
CV0008 .....	6
CV0013 .....	6
CV0014 .....	7
DI0002 .....	7
EX0200 .....	8
EX0201 .....	9
EX0203 .....	10
EX0205 .....	11
EX0206 .....	11
EX0210 .....	12
EX0302 .....	14
EX0308 .....	14

EX0309 .....	14
EX0310 .....	15
EX2526 .....	16
EX2598 .....	16
EX2629 .....	18
EX2635 .....	19
EX2656 .....	20
EX2664 .....	20
EX2565 .....	21
EX2666 .....	22
EX2830 .....	23
EX2855 .....	24
EX2932 .....	24
EX2947 .....	25
EX2981 .....	26
EX2987 .....	27
EX2997 .....	28
EX2998 .....	28
EX2999 .....	29
EX3000 .....	29
EX3041 .....	30
EX3044 .....	30
EX3073 .....	31
EX3359 .....	32
EX3413 .....	32
EX3483 .....	33
EX3514 .....	33
EX3534 .....	34
EX3589 .....	35

EX3628 .....	35
EX3638 .....	36
EX3670 .....	37
EX3671 .....	38
EX3680 .....	39
EX3682 .....	39
EX3705 .....	40
EX3706 .....	40
EX3735 .....	41
EX3771 .....	42
EX3779 .....	43
EX3780 .....	44
EX3784 .....	45
EX3786 .....	45
EX3791 .....	46
EX3792 .....	46
EX3794 .....	47
EX3818 .....	48
EX3827 .....	48
EX3829 .....	49
EX3833 .....	50
EX3834 .....	50
EX3858 .....	52
EX3863 .....	52
EX3864 .....	52
EX3872 .....	53
EX3875 .....	53
EX3900 .....	54
EX3902 .....	55

EX3907 .....	55
EX3916 .....	56
EX3927 .....	57
EX3928 .....	57
EX3937 .....	57
EX3945 .....	58
EX3983 .....	59
EX3988 .....	60
IF0003 .....	61
Place & Route からのユーザーメッセージ .....	62
CT1000 .....	62
CT1003 .....	62
CT1005 .....	63
CT1007 .....	63
CT1097 .....	64
CT1098 .....	64
CT1101 .....	64
CT1102 .....	64
CT1108 .....	65
CT1111 .....	65
CT1112 .....	65
CT1113 .....	66
CT1115 .....	66
CT1116 .....	66
CT1117 .....	67
CT1118 .....	67
FS1008 .....	68
FS2001 .....	68
PA1000 .....	68

PA1001.....	69
PA1002.....	71
PA1008.....	71
PA1010.....	72
PA2000.....	73
PA2001.....	75
PA2004.....	76
PA2009.....	79
PA2014.....	80
PA2017.....	82
PA2024.....	82
PA2025.....	82
PA2039.....	83
PA2054.....	84
PA2056.....	86
PA2058.....	87
PA2066.....	89
PR0026.....	90
PR0027.....	91
PR0028.....	91
PR0029.....	91
PR1011.....	91
PR1014.....	93
PR2044.....	94
PR2045.....	94
PR2061.....	95
PR2062.....	95
PR2063.....	95
PR2064.....	95



PR2065 .....	96
PR2066 .....	96
PR2067 .....	96
PR2068 .....	96
PR2069 .....	96
PR2070 .....	97
TA1001 .....	97
TA1004 .....	97
TA1006 .....	98
TA1011 .....	98
TA1012 .....	98
TA1016 .....	99
TA1019 .....	99
TA1027 .....	100
TA1032 .....	100
TA1033 .....	101
TA1048 .....	101
TA1049 .....	101
TA1058 .....	102
TA1059 .....	102
TA1061 .....	102
TA1068 .....	103
TA1076 .....	103
TA1109 .....	104
TA1114 .....	104
TA2002 .....	105

# Gowin ソフトウェアのメッセージについてのヘルプドキュメント

## 概要

このマニュアルは、ユーザーがソフトウェアの使用中表示される warning および error メッセージをすばやく対処できるように作成されています。このマニュアルは、GowinSynthesis からのユーザーメッセージと Place & Route からのユーザーメッセージについて説明します。ソフトウェアのアップデートにより、一部の内容が変更される場合があります。

## GowinSynthesis からのユーザーメッセージ

### AG0100

#### ***WARN (AG0100): Find logical loop signal : <signal>***

デザインにロジックループがある場合、合成ツールで上記のエラーメッセージが表示されます。例えば：

```
module test (in,out);  
  input in;  
  output out;  
  assign out = in & !out;  
endmodule
```

#### 解決法

ロジックループを回避してください。

### AG0101

#### ***WARN (AG0101): The netlist is not one directed acyclic graph including user instantiated primitives***

デザインにロジックループがあり、かつロジックループにインスタンス化されたロジックプリミティブ（LUT、ALU など）がある場合、合成ツ

ルで上記のエラーメッセージが表示されます。

```
module test (a,b,out1,out2);
input a,b;
output out1,out2;
assign out1 = out2 & b;
LUT3 ins1(
    .I0(a),
    .I1(b),
    .I2(out1),
    .F(out2)
);
defparam ins1.INIT=8'hAB;
endmodule
```

#### 解決法

ロジックループを回避してください。

## AG0200

### ***ERROR (AG0200):The netlist is not one directed acyclic graph***

AG0100 と同様です。例えば：

```
module test (in,out);
input in;
output out;
assign out = in & !out;
endmodule
```

#### 解決法

ロジックループを回避してください。

## CK0013

### ***ERROR (CK0013):<signal> is not connected to buf or iodelay.***

<signal>には固定接続があるため、これらのユニットのドライバーまたは宛先にも固定接続があります。例えば、インスタンス **oser4** では、**Q0** が **Q0&Q1** を駆動することは実装できません。

```
module OSER4_ins (Q0_test, D0, D1, D2, D3, TX0, TX1, PCLK, FCLK, RESET);
input D0, D1, D2, D3;
input TX0, TX1;
input PCLK, FCLK, RESET;
output Q0_test;
```

```
    wire Q0;
    wire Q1;
    OSER4 oser4(
        .Q0(Q0),
        .Q1(Q1),
        .D0(D0),
        .D1(D1),
        .D2(D2),
        .D3(D3),
        .TX0(TX0),
        .TX1(TX1),
        .PCLK(PCLK),
        .FCLK(FCLK),
        .RESET(RESET)
    );
    defparam oser4.GSREN = "false";
    defparam oser4.LSREN = "true";
    defparam oser4.HWL = "false";
    defparam oser4.TXCLK_POL = 1'b0;
    assign Q0_test = Q0 & Q1;
endmodule
```

### 解決法

デザインを変更し、Q0 が直接出力ポートを駆動するようにしてください。

## CK2027

### ***ERROR(CK2027): The connection between Instance<inst1>and instance<inst2>is not correct!***

<inst1>と<inst2>には固定接続があるため、これらのユニットのドライバーまたは宛先にも固定接続があります。次の場合、DHCEN がインスタンス化されています。チップデザインによると、DHCEN の CLKOUT は IOLOGIC/CLKDIV/DLL/PLL/DQS などのタイプのクロックポートのみを駆動できるため、エラーメッセージが報告されます。

```
module DHCEN_ins (CLKOUT, CLKIN, CE);
    input CLKIN,CE;
    output CLKOUT;
    DHCEN dhcen(
        .CLKOUT(CLKOUT),
        .CLKIN(CLKIN),
        .CE(CE)
    );
endmodule
```

```
);  
endmodule
```

## 解決法

DHCEN の CLKOUT ポートに駆動されるインスタンスを変更してください。

```
module DHCEN_ins (Q0, CLKIN, CE);  
  input CLKIN,CE;  
  output Q0;  
  wire Q1;  
  wire D0;  
  wire D1;  
  wire D2;  
  wire D3;  
  wire TX0;  
  wire TX1;  
  wire PCLK;  
  wire RESET;  
  wire CLKOUT;  
  DHCEN dhcen(  
    .CLKOUT(CLKOUT),  
    .CLKIN(CLKIN),  
    .CE(CE)  
  );  
  OSER4 oser4(  
    .Q0(Q0),  
    .Q1(Q1),  
    .D0(D0),  
    .D1(D1),  
    .D2(D2),  
    .D3(D3),  
    .TX0(TX0),  
    .TX1(TX1),  
    .PCLK(PCLK),  
    .FCLK(CLKOUT),  
    .RESET(RESET)  
  );  
  defparam oser4.GSREN = "false";  
  defparam oser4.LSREN = "true";  
  defparam oser4.HWL = "false";  
  defparam oser4.TXCLK_POL = 1'b0;  
endmodule
```

## CV0003

### ***WARN(CV0003):Output<port>has undriven bits, assigning undriven bits to 0,simulation mismatch possible***

出力<port>がフローティングのままの場合、GND または高インピーダンスに接続させ、合成結果のシミュレーションは、rtl のシミュレーションと異なる場合があります。合成ツールは上記の警告メッセージを表示し、合成を続行します。例えば、次の場合、出力ポート o2 はフローティングのままです。

```
module test(a,b,o1,o2);
  input a,b;
  output o1,o2; // o2 dangling
  assign o1 = a & b;
endmodule
```

#### 解決法

出力ポートが内部信号に接続されている場合は、GND または VCC を代入してください。

```
module test(a,b,o1,o2);
  input a,b;
  output o1,o2;
  assign o1 = a & b;
  assign o2 = 1'b0; // assign GND to dangling output port
endmodule
```

## CV0005

### ***ERROR(CV0005):Tran switch which all inputs are connected to inout port can not be converted***

双方向スイッチ tran のすべてのピンを inout ポートに接続すると、データが競合し、合成ツールで上記のエラーメッセージが表示され、合成が停止します。例えば：

```
module test(io1,io2,control);
  inout io1,io2;
  input control;
  tran t(io1,io2); //tran D0 and D1 all connect inout port
endmodule
```

#### 解決法

tran の 2 つのピンを異なるタイプのポートに接続してください。

```
module test(o1,control);
output o1;
input control;
tran t(control,o1);//tran D0 connect input port, D1 connect output port
endmodule
```

## CV0008

### ***ERROR(CV0008):Convert tran switch <object> failed***

双方向スイッチ tran <object>のピンの使用が不正な場合、合成ツールで上記のエラーメッセージが表示され、合成が停止します。例えば、次の場合、tran の 1 つのピンはフローティングのままです。

```
module test(o1);
output o1;
wire control;
tran t(o1,control); // tran D1 dangling
endmodule
```

### 解決法

フローティングピンをポートまたは内部信号に接続してください。

```
module test(o1,control);
output o1;
input control;
tran t(control,o1); // connect tran D1 to output port
endmodule
```

## CV0013

### ***ERROR(CV0013):Tri-state signal <signal> connected ERROR, it should only be connected to PAD.***

トライステートの信号<signal>は PAD 以外のタイプの信号には接続する場合、合成ツールで上記のエラーメッセージが表示されます。例えば、PAD に接続された gwBuf のスリーステート信号 io は GND (2' b00) によって駆動されるので、このエラーメッセージが報告されます。

```
module InterFace(
    inout io,
    input in, io_en,
    output lrck
);
IOBUF gwBuf(
```

```
.I(in),  
.OEN(io_en),  
.IO(io),  
.O(lrck)  
);  
assign io = 2'b00; //assign GND to io  
endmodule
```

### 解決法

io のドライバー (2' b00) を削除してください。

```
module InterFace(  
    inout io,  
    input in, io_en,  
    output lrck  
);  
IOBUF gwBuf(  
    .I(in),  
    .OEN(io_en),  
    .IO(io),  
    .O(lrck)  
);  
endmodule
```

## CV0014

### **ERROR(CV0014):Not Support MOS switch<signal>synthesis**

インスタンス化された cmos/rcmos の合成はサポートされていません。  
例えば：

```
module test(in,control1,control2,o);  
    input in,control1,control2;  
    output o;  
    cmos c(o,in,control1,control2);  
endmodule
```

### 解決法

cmos/rcmos のインスタンスを削除してください。

## DI0002

### **WARN(DI0002):Asynchronous register<asynReg >initial values do**



***not match with the Gowin library, simulation mismatch possible***

非同期セットフリップフロップ<asynReg>の初期値は 1 にのみ設定でき、非同期リセットフリップフロップ<asynReg>の初期値は 0 にのみ設定できます。例えば：

```
module test(clk,d,clear,o);
  input clk,d,clear;
  output o;
  reg o = 1'b1;
  always @(posedge clk or posedge clear) // async register
  if(clear)
    o = 0; // register output 0 if clear, this register will be synthesized
to DFFC
  else
    o = d;
endmodule
```

**解決法**

<asynReg>の初期値を設定しないか、または初期値が<asynReg>の clear/preset 信号のリセット/セット結果と一致するようにしてください。

```
module test(clk,d,clear,o);
  input clk,d,clear;
  output o;
  reg o = 1'b0;
  always @(posedge clk or posedge clear) // async register
  if(clear)
    o = 0; // register output 0 if clear, this register will be synthesized
to DFFC
  else
    o = d;
endmodule
```

**EX0200*****WARN (EX0200): Property <prop>set invalid for <object>***

デザインの<object>に正しくない属性値<prop>が代入されている場合、合成ツールは上記の警告を表示し、デフォルトの初期値を使用します。次のテストケースでは、属性制約 `syn_ramstyle` の値が `null` の場合、上記の警告メッセージが報告されます。

```
module normal1(data_out, data_in, addr, clk, wre,rst);
  output [1:0]data_out;
  input [1:0]data_in;
```

```
input [6:0]addr;
input clk,wre,rst;
reg [1:0] mem [127:0] /* synthesis syn_ramstyle = "" */;
reg [1:0] data_out;
always@(posedge clk or posedge rst)
  if(rst ==1)
    data_out <= 0;
  else
    if(wre == 0)
      data_out <= mem[addr];
  always @(posedge clk)
    if (wre) mem[addr] <= data_in;
endmodule
```

## 解決法

デザインで使用されている属性制約の値が正しいことを確認してください。以下のテストケースでは、`syn_ramstyle` には `registers` が代入されます。

```
module normal1(data_out, data_in, addr, clk, wre,rst);
output [1:0]data_out;
input [1:0]data_in;
input [6:0]addr;
input clk,wre,rst;
reg [1:0] mem [127:0] /* synthesis syn_ramstyle = "registers" */;
reg [1:0] data_out;
always@(posedge clk or posedge rst)
  if(rst ==1)
    data_out <= 0;
  else
    if(wre == 0)
      data_out <= mem[addr];
  always @(posedge clk)
    if (wre) mem[addr] <= data_in;
endmodule
```

## EX0201

### ***WARN (EX0201): Missing INIT parameter on <object> and using default value***

<object>がデザインでインスタンス化されているが、<object>に初期値が設定されていない場合、合成ツールは上記の警告を表示し、デフォルトの初期値に設定します。次のテストケースでは、`LUT3 ins1` がインスタンス化されていますが、`ins1` の初期値はに設定されていません。

```
module test(a,b,c,out);
  input a,b,c;
  output out;
  LUT3 ins1(
    .I0(a),
    .I1(b),
    .I2(c),
    .F(out)
  );
  /*lack defparam of LUT3*/
endmodule
```

### 解決法

上記の警告を解消するには、以下に示すように、インスタンス化された **<object>** に初期値を代入する必要があります。

```
module test(a,b,c,out);
  input a,b,c;
  output out;
  LUT3 ins1(
    .I0(a),
    .I1(b),
    .I2(c),
    .F(out)
  );
  defparam ins1.INIT=8'hEF;
endmodule
```

## EX0203

### ***WARN (EX0203): Top module <modu> has no ports***

デザインのトップモジュール **<modu>** にポートがない場合、合成ツールは上記の警告を表示します。例えば：

```
module test();
  wire a,b,out;
  assign a = 1'b0;
  assign b = 1'b1;
  assign out = a ^ b;
endmodule
```

### 解決法

上記の警告メッセージを解消するには、トップモジュール **<modu>** に少

なくとも 1 つの入力ポートまたは出力ポートを設定する必要があります。

## EX0205

### ***WARN (EX0205): Instance <inst> 's parameter <para> value invalid, replaced by default value <para>***

デザインでインスタンス化された Gowin プリミティブ<inst>の<para>の値が正しくない場合、合成ツールは上記の警告メッセージを表示し、<inst>の<para>をデフォルト値に置き換えます。例えば、以下のデザインの LUT4 の<para>の値が正しくありません。

```
module unittest ( i,out);
input [3:0] i;
output out;
LUT4 lut4_0 (
    .I0(i[0]),
    .I1(i[1]),
    .I2(i[2]),
    .I3(i[3]),
    .F(out)
);
defparam lut4_0.INIT = "wabajd";///< right: 16'h0000 to 16'hFFFF
endmodule
```

### 解決法

上記の警告メッセージを解消するには、『Gowin FPGA プリミティブ ユーザーガイド (SUG283)』を参照して、インスタンス化されたプリミティブの<para>を適切な範囲内に設定する必要があります。

```
module unittest ( i,out);
input [3:0] i;
output out;
LUT4 lut4_0 (
    .I0(i[0]),
    .I1(i[1]),
    .I2(i[2]),
    .I3(i[3]),
    .F(out)
);
defparam lut4_0.INIT = 16'h1234;///< right: 16'h0000 to 16'hFFFF
endmodule
```

## EX0206

***WARN (EX0206): Instance <inst> 's parameter <para> value invalid***

現在指定されているデバイスが、インスタンス化された rPLL などのプリミティブ<inst>の<para>で指定されているデバイスと一致しない場合、合成ツールで上記の警告が表示されます。次の例に示すように、指定されたデバイスがインスタンス化された rPLL で指定されているデバイスと一致しない場合、この警告メッセージが報告されます。

```
module test(i,out);
input [35:0]i;
output [4:0]out;
rPLL rp1l (
    .CLKIN(i[0]),
    .CLKFB(i[1]),
    .FBDSEL(i[7:2]),
    .IDSEL(i[13:8]),
    .ODSEL(i[19:14]),
    .DUTYDA(i[23:20]),
    .PSDA(i[27:24]),
    .FDLY(i[31:28]),
    .RESET(i[32]),
    .RESET_P(i[33]),
    .CLKOUT(out[0]),
    .CLKOUTP(out[1]),
    .CLKOUTD(out[2]),
    .CLKOUTD3(out[3]),
    .LOCK(out[4])
);
defparam rp1l.DEVICE = "GW1N-4B";
endmodule
```

**解決法**

上記の警告を解消するには、指定されたデバイスを、<para>のデバイスと一致させる必要があります。

**EX0210*****WARN (EX0210): Invalid input frequency <freq> to instance <inst>, suitable range is from <num1>MHz to <num2>MHz***

<inst>の<freq>が<num1>から<num2>の範囲内にありません。<inst>の<freq>が設定されていない場合、デフォルトの<freq>値に設定されます。

```
module test(i,out);
```

```
input [35:0]i;
output [4:0]out;
rPLL rp1l (
    .CLKIN(i[0]),
    .CLKFB(i[1]),
    .FBDSEL(i[7:2]),
    .IDSEL(i[13:8]),
    .ODSEL(i[19:14]),
    .DUTYDA(i[23:20]),
    .PSDA(i[27:24]),
    .FDLY(i[31:28]),
    .RESET(i[32]),
    .RESET_P(i[33]),
    .CLKOUT(out[0]),
    .CLKOUTP(out[1]),
    .CLKOUTD(out[2]),
    .CLKOUTD3(out[3]),
    .LOCK(out[4])
);
defparam rp1l.FCLKIN = "100.0";
defparam rp1l.DYN_IDIV_SEL = "false";
defparam rp1l.IDIV_SEL = 0;
defparam rp1l.DYN_FBDIV_SEL = "false";
defparam rp1l.FBDIV_SEL = 0;
defparam rp1l.DYN_ODIV_SEL = "false";
defparam rp1l.ODIV_SEL = 32;
defparam rp1l.PSDA_SEL = "0000";
defparam rp1l.DYN_DA_EN = "false";
defparam rp1l.DUTYDA_SEL = "1000";
defparam rp1l.CLKOUT_FT_DIR = 1'b1;
defparam rp1l.CLKOUTP_FT_DIR = 1'b1;
defparam rp1l.CLKOUT_DLY_STEP = 0;
defparam rp1l.CLKOUTP_DLY_STEP = 0;
defparam rp1l.CLKFB_SEL = "internal";
defparam rp1l.CLKOUT_BYPASS = "false";
defparam rp1l.CLKOUTP_BYPASS = "false";
defparam rp1l.CLKOUTD_BYPASS = "false";
defparam rp1l.DYN_SDIV_SEL = 2;
defparam rp1l.CLKOUTD_SRC = "CLKOUT";
defparam rp1l.CLKOUTD3_SRC = "CLKOUT";
defparam rp1l.DEVICE = "GW1N-4B";
endmodule
```

## 解決法

<freq>値または指定されたデバイスを変更してください。

## EX0302

### ***ERROR (EX0302): No valid top module found***

デザインファイルにモジュールがない場合、合成ツールは上記の警告を表示します。この場合は合成できません。

## EX0308

### ***ERROR(EX0308):GowinSynthesis can not find file ¥"primitive.xml¥". Please reinstall the product***

インストールディレクトリ IDE¥bin の下にあるファイル primitive.xml が移動されると、合成ツールは上記のエラーメッセージを表示します。primitive.xml を元の場所に戻すか、ソフトウェアを再インストールしてください。

## EX0309

### ***ERROR (EX0309): Net <object> has multiple drivers***

<object>に複数の非トライステートドライバーがある場合、合成ツールは上記のエラーメッセージを表示します。エラーメッセージの信号名とファイル情報に従って検索および変更できます。例えば、次に示すように、出力ポート out は、rotate\_1 と rotate\_2 の両方によって駆動されています。

```
module rotate (q1, data, sel1) ;
    output [7:0] q1;
    input [7:0] data;
    input sel1;
endmodule

module top (out, ci, data1, data2);
    output [7:0] out;
    input [7:0] data1, data2;
    input ci;
    rotate rotate_1 (out, data1, ci);
    rotate rotate_2 (out, data2, ci);
endmodule
```

## 解決法

ドライバーを 1 つ削除してください。

```
module rotate (q1, data, sel1) ;
output [7:0] q1;
input [7:0] data;
input sel1;
endmodule

module top (out, ci, data1);
output [7:0] out;
input [7:0] data1;
input ci;
rotate rotate_1 (out, data1, ci);
endmodule
```

## EX0310

### ***ERROR (EX0310): Invalid parameterized value <paraValue>(<para>) specified for instance <inst>***

プリミティブ<inst>の<para>の<paraValue>が正しくありません。例えば、次に示すように、プリミティブ rPLL のパラメータ FCLKIN の値は範囲外です。

```
module test(i,out);
input [35:0]i;
output [4:0]out;
rPLL rp11 (
    .CLKIN(i[0]),
    .CLKFB(i[1]),
    .FBDSEL(i[7:2]),
    .IDSEL(i[13:8]),
    .ODSEL(i[19:14]),
    .DUTYDA(i[23:20]),
    .PSDA(i[27:24]),
    .FDLY(i[31:28]),
    .RESET(i[32]),
    .RESET_P(i[33]),
    .CLKOUT(out[0]),
    .CLKOUTP(out[1]),
    .CLKOUTD(out[2]),
    .CLKOUTD3(out[3]),
    .LOCK(out[4])
);
defparam rp11.DEVICE = "GW1N-4B";
defparam rp11.FCLKIN = "600.0";
```



```
endmodule
```

### 解決法

『Gowin プリミティブ ユーザーガイド (SUG283)』を参照してプリミティブ<inst>の<para>の<paraValue>を適切な値に設定します。

## EX2526

***WARN (EX2526): Entry size <width> at <initvalue>:<initWidth> does not match memory width <memWidth>***

\$readmemh 文が使用され、対応するファイルのデータ幅<width>が<memWidth>と一致しない場合、合成ツールは上記の警告メッセージを表示します。例えば、次の場合、mem 幅は 8 であり、対応する<initvalue>ファイルのデータ幅<initWidth>が 8 ビットでない場合、この警告メッセージが報告されます。

```
module test(in,clk,addr_in,addr_out,out);  
  
input in,clk;  
  
input addr_in;  
  
input addr_out;  
  
output out;  
  
reg mem[7:0];  
  
always @ (posedge clk)  
    mem[addr_in] <= in;  
  
assign out = mem[addr_out];  
  
initial begin  
    $readmemh("initvalue", mem);  
  
end  
  
endmodule
```

### 解決法

対応するファイルのデータ幅<width>が<memWidth>と一致することを確保してください。

## EX2598

***WARN (EX2598): <design> might have mixed concurrent and***

### *procedural assignment*

<design>にブロッキング代入とノンブロッキング代入の両方がある場合、合成ツールは上記の警告メッセージを表示します。例えば：

```
module gene_if(t0,t1,t2,d,clk,t);  
    input t0,t1,t2,clk,t;  
    output d;  
    reg d_reg;  
    localparam S=6;  
    generate  
        if(S<7)  
            assign d_reg=t0|t1|t2;  
        else  
            assign d_reg=t0&t1&t2;  
    endgenerate  
    generate  
        if(S>7)  
            always @(posedge clk)  
                d_reg<=t;  
            else  
                always @(posedge clk)  
                    d_reg<=t0&t1&t2;  
        endgenerate  
    assign d=d_reg;  
endmodule
```

### 解決法

繰り返し代入を削除してください。

```
module gene_if(t0,t1,t2,d,clk,t);  
    input t0,t1,t2,clk,t;  
    output d;
```

```
reg d_reg;

localparam S=6;

generate

    if(S>7)

        always @(posedge clk)

            d_reg<=t;

        else

            always @(posedge clk)

                d_reg<=t0&t1&t2;

    endgenerate

assign d=d_reg;

endmodule
```

## EX2629

### ***WARN (EX2629): Delay control is not supported for synthesis***

**delay** 文は合成不可能な文で、**#10** などの **delay** 文が含まれている場合、合成ツールは上記の警告メッセージを表示し、すべての遅延は無視されます。

```
module top (in0,in1,clk,out);
input in0,in1;
input clk;
output reg out;
always @(posedge clk)
begin
    out <= #10 in0&in1;
end
endmodule
```

### 解決法

**delay** 文を削除してください。

```
module top (in0,in1,clk,out);
input in0,in1;
input clk;
output reg out;
always @(posedge clk)
```

```
begin
    out <= in0&in1;
end
endmodule
```

## EX2635

### ***WARN (EX2635): Generate block is allowed only inside loop and conditional generate in SystemVerilog mode***

ループでモジュールを生成するために **generate** 文を使用すると、合成ツールで上記のエラーメッセージが表示されます。次の例は、for ループで test モジュールを生成するケースです。これは、**SystemVerilog** でのみサポートされています。

```
module top(in,out);
    input [1:0]in;
    output [1:0]out;
    generate
    begin
        genvar i;
        for (i=0;i<2;i=i+1)
            begin : reg_loop1
                test test1(in[i],out[i]);
            end
        end
    endgenerate
endmodule

module test(in,out);
    input in;
    output out;
    assign out = !in;
endmodule
```

### 解決法

これは、**SystemVerilog** でのみサポートされています。

## EX2656

### ***ERROR (EX2656): SystemVerilog keyword <word> used in incorrect context***

キーワード<word>と同じ名前の変数が定義されている場合、合成ツールで上記のエラーメッセージが表示されます。null は system verilog のキーワードであり、変数名には使用できません。

```
module top (in0,in1,out);  
  input in0,in1;  
  output out;  
  wire null;  
  assign out= in0&in1;  
endmodule
```

#### 解決法

キーワードと同じ名前の変数の使用を回避してください。

```
module top (in0,in1,out);  
  input in0,in1;  
  output out;  
  assign out= in0&in1;  
endmodule
```

## EX2664

### ***WARN (EX2664): Variable <vari> may be used before assigned in always\_comb or always @\* block : might cause synthesis - simulation differences***

always 文のセンシティビティ・リストにある<vari>が変化する場合、合成ツールはこの警告メッセージを表示します。例えば：

```
module top(in,sel,out);  
  input in,sel;  
  output reg out;  
  reg tmp;  
  always@(*)  
  begin  
    if(sel)  
      tmp <= in;
```

```
        else
            tmp <= !tmp;
        end
    assign out= tmp;
endmodule
```

### 解決法

クロック信号を追加し、**tmp** 信号をセンシティブティ・リストの外に移動してください。下図に示すように、

```
module top(in,sel,out,clk);
    input in,sel,clk;
    output reg out;
    reg tmp;
    always@(posedge clk)
        begin
            if(sel)
                tmp <= in;
            else
                tmp <= !tmp;
            end
    assign out= tmp;
endmodule
```

## EX2565

### ***WARN (EX2565): Port <port> is not connected on this instance***

定義された<port>がインスタンスで使用されていない場合、合成ツールは上記の警告メッセージを報告します。例えば：

```
module top (in0,in1,out,out1);
    input in0,in1;
    output out,out1;
    assign out = in0 & !in1;
```

```
test test1(
    .data(in0),
    .out(out1)
);

endmodule

module test (data,out,clk);

input data,clk;

output out;

assign out = !data;

endmodule
```

### 解決法

不要なポートを削除してください。

```
module top (in0,in1,out,out1);

input in0,in1;

output out,out1;

assign out = in0 & !in1;

test test1(
    .data(in0),
    .out(out1)
);

endmodule

module test (data,out);

input data;

output out;

assign out = !data;

endmodule
```

## EX2666

***WARN (EX2666): Unsupported use of clock signal <signal>, clock used as data***

信号<signal>が同時にクロックとデータとして使用されている場合、合成ツールで上記の警告メッセージが表示されます。例えば：

```
module top (clk,out);  
    input clk;  
    output reg out;  
    always@(posedge clk)  
        out <= clk;  
endmodule
```

### 解決法

入力データとクロックを分離してください。

```
module top (in,clk,out);  
    input in,clk;  
    output reg out;  
    always@(posedge clk)  
        out <= in;  
endmodule
```

## EX2830

### ***WARN (EX2830): Data object <object> is already declared***

変数<object>が繰り返し定義された場合、合成ツールは上記のエラーメッセージを報告します。例えば：

```
module top (in,out);  
    input in;  
    output out;  
    wire out;  
    wire out;  
    assign out = !in;  
endmodule
```

### 解決法

繰り返し定義を削除してください。

```
module top (in,out);  
    input in;
```



```
output out;
wire out;
assign out = !in;
endmodule
```

## EX2855

### ***WARN (EX2855): Result of this <oper> operation does not fit in <width> bits***

<oper>演算の結果のビット幅が対応する変数のビット幅を超えた場合、合成ツールで上記の警告メッセージが表示されます。例えば：

```
module top(in,out);
input in;
output [1:0]out;
assign out = 6'd2 ** (16'h77)+in;
endmodule
```

#### 解決法

幅を一致させてください。

```
module top(in,out);
input in;
output [1:0]out;
assign out = 2'b01 ** (2'b10)+in;
endmodule
```

## EX2932

### ***WARN (EX2932): Unknown system task <task> ignored for synthesis***

不明なシステムタスク<task>がある場合、合成ツールは上記の警告メッセージを表示します。例えば：

```
module test(in,out);
input in;
output out;
reg mem;
```

```
assign out = in;

initial begin

    $fsdbDumpMDA(mem);

end

endmodule
```

## 解決法

不明なシステムタスクを削除してください。

```
module test(in,out);

input in;

output out;

reg mem;

assign out = in;

endmodule
```

## EX2947

### ***WARN (EX2947): Input port <port> remains unconnected for this instance***

インスタンス化するとき、ポート<port>が接続されていない場合、合成ツールは上記の警告メッセージを表示します。例えば：

```
module top (top_in0,top_in1,top_out);
input top_in0,top_in1;
output top_out;
sub sub1(
.in0(top_in0),
.in1(),
.out(top_out)
);
endmodule
module sub (in0,in1,out);
input in0,in1;
output out;
assign out = in0 & !in1;
endmodule
```

## 解決法

フローティングポートを接続してください。

```
module top (top_in0,top_in1,top_out);
  input top_in0,top_in1;
  output top_out;
  sub sub1(
    .in0(top_in0),
    .in1(top_in1),
    .out(top_out)
  );
endmodule
module sub (in0,in1,out);
  input in0,in1;
  output out;
  assign out = in0 & !in1;
endmodule
```

## EX2981

### ***WARN (EX2981): Net <objec> is driven by multiple input ports***

入力ポート<port>が繰り返し同じ変数<object>に代入される場合、合成ツールは上記の警告メッセージを報告します。例えば：

```
module test(in,out);
  input [3:0]in;
  output [3:0]out;
  wire [3:0]tmp;
  assign tmp[3:0] = in;
  assign tmp[1:0] = in;
  assign out = tmp;
endmodule
```

## 解決法

繰り返し代入を削除してください。

```
module test(in,out);
  input [3:0]in;
  output [3:0]out;
```

```
wire [3:0]tmp;

assign tmp[3:0] = in;

assign out = tmp;

endmodule
```

## EX2987

### ***WARN (EX2987): Input port <port> is not connected on this instance***

インスタンス化するとき、ポート<port>が接続されていない場合、合成ツールは上記の警告メッセージを表示します。例えば：

```
module top (in0,out1);
  input in0;
  output out1;
  test test1(
    .in1(in0),
    .in(),
    .out(out1)
  );
endmodule
module test (in1,in,out);
  input in1;
  input in;
  output out;
  assign out = !in1;
endmodule
```

### 解決法

不要なポートを削除してください。

```
module top (in0,out1);
  input in0;
  output out1;
  test test1(
    .in1(in0),
    .out(out1)
  );
endmodule
module test (in1,out);
  input in1;
  output out;
```

```
assign out = !in1;
endmodule
```

## EX2997

### ***WARN (EX2997): Net <net> is already driven by input port <port>***

<net>の入力ポート<port>に代入する場合、合成ツールは上記の警告メッセージを報告します。例えば：

```
module test(b,c,d,f);
input b,c,d;
output f;
assign d = c&b;
assign f = b&d;
endmodule
```

#### 解決法

当該代入文を削除してください。

```
module test(b,d,f);
input b,d;
output f;
assign f = b&d;
endmodule
```

## EX2998

### ***WARN (EX2998): Net <object> does not have a driver***

ワイヤまたはレジスタ型変数<object>にドライバーがない場合、合成ツールは上記の警告メッセージを報告します。例えば、ワイヤ **a** にドライバーがありません。

```
module top (in0,in1,out);
input in0,in1;
output out;
wire a;
assign out = in0&in1|a;
endmodule
```

#### 解決法

ワイヤ **a** を削除するか、対応するドライバーを追加してください。

```
module top (in0,in1,out);
input in0,in1;
```

```
output out;  
assign out = in0&in1;  
endmodule
```

## EX2999

### ***ERROR (EX2999): Another driver from here***

複数の入力と同じ出力を駆動する場合、合成ツールは上記のエラーメッセージを表示します。例えば：

```
module test (in0,in1,out);  
input in0,in1;  
output out;  
assign out = in0 & !in1;  
assign out = in0 & in1;  
endmodule
```

#### 解決法

代入文を 1 つ削除してください。

```
module test (in0,in1,out);  
input in0,in1;  
output out;  
assign out = in0 & !in1;  
endmodule
```

## EX3000

### ***ERROR (EX3000): Net <object> is constantly driven from multiple places***

複数の入力と同じ出力<object>を駆動する場合、合成ツールは上記のエラーメッセージを表示します。例えば：

```
module test (in0,in1,out);  
input in0,in1;  
output out;  
assign out = in0 & !in1;  
assign out = in0 & in1;  
endmodule
```

#### 解決法

代入文を 1 つ削除してください。

```
module test (in0,in1,out);
```

```
input in0,in1;
output out;
assign out = in0 & !in1;
endmodule
```

## EX3041

### ***WARN (EX3041): <object> shift count >= width of value***

シフト数が変数のビット幅より大きい場合、合成ツールで上記の警告メッセージが表示されます。例えば、次の場合、**in1** のビット幅は **1** で、左シフト数は **2** ビットです。

```
module top (in0,in1,out);

input in0,in1;

output reg out;

assign out = in0&(in1 << 2);

endmodule
```

### 解決法

無効なシフトを削除してください。

## EX3044

### ***WARN (EX3044): Overwriting previous value of parameter <para>***

同じ<para>に複数回代入を行うと、合成ツールで上記の警告メッセージが表示されます。例えば：

```
module top (in0,in1,out);

input in0,in1;

output reg out;

LUT2 lut2(

.I0(in0),

.I1(in1),

.F(out));

defparam lut2.INIT = 4'h4;

defparam lut2.INIT = 4'h6;

endmodule
```

## 解決法

繰り返し代入を削除してください。

```
module top (in0,in1,out);  
  
input in0,in1;  
  
output reg out;  
  
LUT2 lut2(  
  
    .I0(in0),  
  
    .I1(in1),  
  
    .F(out));  
  
defparam lut2.INIT = 4'h6;  
  
endmodule
```

## EX3073

### ***WARN (EX3073): Port <port> remains unconnected for this instance***

モジュールのポート<port>がモジュールのインスタンスのポートリストに存在しない場合、合成ツールは上記の警告メッセージを表示します。例えば、インスタンス **sub1** に **out1** ポートがありません。

```
module top (top_in,top_out);  
input top_in;  
output top_out;  
sub sub1(  
    .in(top_in),  
    .out0(top_out)  
);  
endmodule  
module sub (in,out0,out1);  
input in;  
output out0;  
output out1;  
assign out0 = !in;  
endmodule
```

## 解決法

モジュールから **out1** ポートを削除するか、インスタンスに **out1** ポートを追加してください。



```
module top (top_in,top_out);
  input top_in;
  output top_out;
  sub sub1(
    .in(top_in),
    .out0(top_out)
  );
endmodule
module sub (in,out0);
  input in;
  output out0;
  assign out0 = !in;
endmodule
```

## EX3359

### ***ERROR (EX3359): Null as source expression is not allowed here***

信号に **null** が代入された場合、合成ツールで上記のエラーメッセージが表示されます。**null** は **system verilog** のキーワードです。

```
module top (in,out);
  input in;
  output out;
  assign out= null;
endmodule
```

#### **Action**

キーワードを使用しないでください。

## EX3413

### ***ERROR (EX3413): Second argument of '\$<object> must be a memory***

**\$<object>**構文が誤って使用されている場合、合成ツールで上記のエラーメッセージが表示されます。例えば：

```
module test(in,clk,addr_in,addr_out,out);

  input in,clk;

  input addr_in;

  input addr_out;

  output out;
```

```
reg [7:0]mem;

always @ (posedge clk)

    mem[addr_in] <= in;

assign out = mem[addr_out];

initial begin

    $readmemh("initvalue", mem);

end

endmodule
```

### 解決法

mem を 2 次元配列として定義してください。

```
module test(in,clk,addr_in,addr_out,out);

input in,clk;

input addr_in;

input addr_out;

output out;

reg [7:0]mem[7:0];

always @ (posedge clk)

    mem[addr_in] <= in;

assign out = mem[addr_out];

initial begin

    $readmemh("initvalue", mem);

end

endmodule
```

## EX3483

### ***ERROR (EX3483): Cannot open Verilog file <file>***

指定されたデザインファイル<file>が存在しないか、開く権限がない場合、合成ツールで上記のエラーメッセージが表示されます。ファイルの有無とその権限を確認してください。

## EX3514

***ERROR (EX3514): Module <modu> in library <lib> is not yet analyzed***

指定した<modu>がライブラリ<lib>に存在しない場合、合成ツールで上記のエラーメッセージが表示されます。指定した<modu>の名前が正しいかどうかを確認してください。

**EX3534*****ERROR (EX3534): Assignment under multiple single edges is not supported for synthesis***

**always** 文のセンシティビティ・リストに立ち上がりエッジトリガと立ち下がりエッジトリガの両方がある場合、合成ツールは上記のエラーメッセージを表示します。例えば：

```
module top (in,out,clk,clear);
input in,clk,clear;
output reg out;
always @(posedge clk or negedge clk)
    if(clear)
        begin
            out <= 1'b0;
        end
    else
        begin
            out <= in;
        end
    end
endmodule
```

**解決法**

clk 信号の立ち上がりエッジトリガまたは立ち下がりエッジトリガを削除してください。

```
module top (in,out,clk,clear);
input in,clk,clear;
output reg out;
always @(posedge clk)
    if(clear)
        begin
            out <= 1'b0;
        end
    else
        begin
            out <= in;
        end
    end
endmodule
```

```
        end
    endmodule
```

## EX3589

### ***ERROR (EX3589): Keyword <object> is not allowed here in this mode of Verilog***

キーワード<object>が誤って使用されている場合、合成ツールで上記のエラーメッセージが表示されます。

## EX3628

### ***WARN (EX3628): Redclaration of ansi port <port> is not allowed***

出力ポート<port>が代入文の右辺値として使用される場合、合成ツールは上記の警告メッセージを表示します。例えば：

```
module top(
    input ClkIn,
    input rst,
    output ClkOut
);
    reg ClkOut;
    always@(posedge ClkIn)
    begin
        if(rst) ClkOut = 1'b0;
        else ClkOut = ~ClkOut;
    end
endmodule
```

### 解決法

例えば、レジスタ **tmp** を定義して利用し、最終的にそれを出力ポートに代入することができます。下図に示すように、

```
module top(
    input ClkIn,
```

```
    input rst,
    output ClkOut
);
reg tmp;
always@(posedge ClkIn)
begin
    if(rst) tmp = 1'b0;
    else tmp = ~tmp;
end
assign ClkOut = tmp;
endmodule
```

## EX3638

***WARN (EX3638) : <object> is already implicitly declared on line <lineInfo>***

<lineInfo>で暗黙的に宣言されたワイヤ<object>を使用しているが、使用後に明示的に宣言する場合、合成ツールはこの警告メッセージを表示します。例えば：

```
module top (in0,in1,out);
    input in0,in1;
    output out;
    aa ins(in0,in1,tmp);
    wire tmp;
    assign out = tmp;
endmodule

module aa(in0,in1,out);
    input in0,in1;
    output out;
    assign out=in0|| in1;
endmodule
```

## 解決法

ワイヤ宣言をその前に置くか、明示的な宣言を削除してください。下図に示すように、

```
module top (in0,in1,out);  
  
  input in0,in1;  
  
  output out;  
  
  wire tmp;  
  
  aa ins(in0,in1,tmp);  
  
  assign out = tmp;  
  
endmodule  
  
module aa(in0,in1,out);  
  
  input in0,in1;  
  
  output out;  
  
  assign out=in0|| in1;  
  
endmodule
```

## EX3670

### ***WARN (EX3670):Actual bit length <actlen> differs from formal bit length <forlen>for port <port>***

モジュール<port>をインスタンス化するときに、インスタンスのビット幅<actlen>が定義されたビット幅<forlen>と一致しない場合、合成ツールで上記の警告メッセージが表示されます。例えば、in と top\_in、および out と top\_out の幅が異なっています。

```
module top (top_in,top_out);  
  input top_in;  
  output top_out;  
  test test1(  
    .in(top_in),  
    .out(top_out)  
  );  
endmodule  
module test (in,out);  
  input [2:0]in;  
  output [1:0]out;  
  assign out[0] = in[0];
```

```
assign out[1] = in[1] & !in[2];  
endmodule
```

### 解決法

対応するポートの幅を同じようにしてください。

```
module top (top_in,top_out);  
input [2:0]top_in;  
output [1:0]top_out;  
test test1(  
.in(top_in),  
.out(top_out)  
);  
endmodule  
module test (in,out);  
input [2:0]in;  
output [1:0]out;  
assign out[0] = in[0];  
assign out[1] = in[1] & !in[2];  
endmodule
```

## EX3671

### ***WARN (EX3671): Second declaration of <object> ignored***

変数<object>が繰り返し定義された場合、合成ツールは上記のエラーメッセージを報告します。例えば：

```
module top (in,out);  
input in;  
output out;  
wireout;  
wireout;  
assign out = !in;  
endmodule
```

### 解決法

繰り返し定義を削除してください。

```
module top (in,out);  
input in;  
output out;  
wireout;  
assign out = !in;  
endmodule
```

## EX3680

### ***WARN (EX3680): Concatenation with unsized literal,will interpret as 32 bits***

ビット幅未定義の変数の代入を使用する場合、合成ツールでこの警告メッセージが表示されます。この場合、 32 ビット幅の変数として取り扱われます。

```
module test (in,out);  
    input in;  
    output [15:0]out;  
    assign out = {'b0,in};  
endmodule
```

#### 解決法

例えば、固定したビット幅に設定してください。

```
module test (in,out);  
    input in;  
    output [15:0]out;  
    assign out = {15'b0,in};  
endmodule
```

## EX3682

### ***WARN (EX3682): Variable <vari> might have multiple concurrent drivers***

出力ポート<vari>に複数のドライバーがある可能性がある場合、合成ツールは上記の警告メッセージを報告します。例えば：

```
module top(in,sel,out);  
    input in,sel;  
    output reg out;  
    reg tmp;  
    always@(*)  
        begin
```



```
        if(sel)
            tmp <= in;
        else
            out <= !tmp;
        end
    assign out= tmp;
endmodule
```

### 解決法

複数のドライバーを回避してください。

## EX3705

### ***WARN (EX3705): Macro <object> redefined***

**define** 文で同じパラメータ<object>を複数回定義すると、合成ツールで上記の警告メッセージが表示されます。この場合、後の **define** 文は、前の定義を置き換えます。例えば：

```
`define INIT 1'b0
module test (in,out0,out1);
input in;
output out0,out1;
assign out0 = !in|^INIT;
`define INIT 1'b1
assign out1 = !in|^INIT;
endmodule
```

### 解決法

2 つの異なる **define** 文を使用するか、**if define** 文を使用してください。

```
`define INIT0 1'b0
`define INIT1 1'b1
module test (in,out0,out1);
input in;
output out0,out1;
assign out0 = !in|^INIT0;
assign out1 = !in|^INIT1;
endmodule
```

## EX3706

***WARN (EX3706): Empty port in <modu> declaration***

<modu>の宣言が正しくない場合、合成ツールは上記の警告メッセージを表示します。例えば：

```
module test(in,out,);  
  
input in;  
  
output out;  
  
assign out = !in;  
  
endmodule
```

**解決法**

不要な“,”を削除してください。

```
module test(in,out);  
  
input in;  
  
output out;  
  
assign out = !in;  
  
endmodule
```

**EX3735*****ERROR (EX3735): Port <port> is already connected***

インスタンス化中に同じ<port>に繰り返し値を代入すると、合成ツールは上記のエラーメッセージを表示します。例えば：

```
module test (in,out);  
input in;  
output out;  
assign out = !in;  
endmodule  
module top (in0,in1,out0);  
input in0,in1;  
output out0;  
test test1(  
    .in(in0),  
    .in(in1),  
    .out(out0)  
);  
endmodule
```

## 解決法

代入文を 1 つ削除してください。

```
module test (in,out);
  input in;
  output out;
  assign out = !in;
endmodule
module top (in0,out0);
  input in0;
  output out0;
  test test1(
    .in(in0),
    .out(out0)
  );
endmodule
```

## EX3771

### ***WARN (EX3771): <modu> instantiation should have an instance name***

名前が指定されていない<modu>をインスタンス化する場合、合成ツールは上記の警告メッセージを表示します。次の場合、**test** モジュールのインスタンス名が必要です。合成中にデフォルトの名前が付けられます。

```
module test (in,out);
  input in;
  output [1:0]out;
  assign out = in+1'b1;
endmodule
module top (top_in,top_out);
  input top_in;
  output [1:0]top_out;
  test (
    .in(top_in),
    .out(top_out)
  );
endmodule
```

## 解決法

インスタンスの名前を指定してください。

```
module test (in,out);
  input in;
```

```
output [1:0]out;
assign out = in+1'b1;
endmodule
module top (top_in,top_out);
input top_in;
output [1:0]top_out;
test test1(
.in(top_in),
.out(top_out)
);
endmodule
```

## EX3779

### ***WARN (EX3779):<signal> should be on the sensitivity list***

**always** 文のセンシティビティ・リストに必要な<signal>がない場合、合成ツールはこの警告メッセージを表示します。例えば、次に示すように、in0 と in1 をセンシティビティ・リストに入れる必要があります。そうしないと、合成中に自動的に追加されます。

```
module top (sel,in0,in1,out);
input sel,in0,in1;
output reg out;
always@(sel /*or in0 or in1*/ )
    if(sel == 1'b0)
        begin
            out <= in0;
        end
    else
        begin
            out <= in1;
        end
    end
endmodule
```

### 解決法

対応する信号をセンシティビティ・リストに追加してください。

```
module top (sel,in0,in1,out);
input sel,in0,in1;
output reg out;
always@(sel or in0 or in1)
    if(sel == 1'b0)
        begin
            out <= in0;
```

```
        end
    else
        begin
            out <= in1;
        end
    endmodule
```

## EX3780

### ***WARN (EX3780): Using initial value of <vari> since it is never assigned***

レジスタ変数<vari>初期値のみが代入された場合、合成ツールは上記の警告メッセージを報告します。例えば：

```
module test(in,out);
    input in;
    output reg out;
    reg tmp;
    initial begin
        tmp = 0;
    end
    always@(in or tmp)
        if(in==tmp)
            out <= in;
        else
            out <= !in;
    endmodule
```

### 解決法

役に立たないレジスタ tmp を 1'b0 に置き換えてください。

```
module test(in,out);
    input in;
    output reg out;
    always@(in)
        if(in==1'b0)
```

```
        out <= in;

    else

        out <= !in;

    endmodule
```

## EX3784

### ***WARN (EX3784): Index <width> is out of range <range> for <port>***

データ幅範囲<range>を超えた<port>が使用される場合、合成ツールは上記の警告メッセージを表示します。例えば、次に示すように、ビット幅範囲が 0～1 の out の out[2]への代入は合成中に無視されます。

```
module test (in,out);
input [2:0]in;
output [1:0]out;
assign out[0] = in[0];
assign out[2] = in[1] & !in[2];
assign out[1] = in[1] & in[2];
endmodule
```

### 解決法

out の幅を調整するか、未定義の幅を使用しないでください。

```
module test (in,out);
input [2:0]in;
output [2:0]out;
assign out[0] = in[0];
assign out[2] = in[1] & !in[2];
assign out[1] = in[1] & in[2];
endmodule
```

## EX3786

### ***ERROR (EX3786): Assignment to input <port>***

入力ポート<port>に代入する場合、合成ツールは上記のエラーメッセージを報告します。例えば：

```
module test(b,c,d,f);
input b,c,d;
output f;
assign d = c&b;
assign f = b&d;
```

```
endmodule
```

### 解決法

当該代入文を削除してください。

```
module test(b,d,f);  
input b,d;  
output f;  
assign f = b&d;  
endmodule
```

## EX3791

### ***WARN (EX3791): Expression size <size> truncated to fit in target size <tarSize>***

代入操作前のデータ幅<tarSize>が操作後のデータ幅<size>と異なる場合、合成ツールで上記の警告メッセージが表示されます。例えば、次に示すように、out のデータ幅は 1 で、in0&in1 のデータ幅は 3 であり、この場合、in0&in1 の余分な幅は無効です。

```
module top (in0,in1,clk,out);  
input [2:0]in0,in1;  
input clk;  
output reg out;  
always @( posedge clk)  
begin  
out <= in0&in1;  
end  
endmodule
```

### 解決法

out または in0&in1 の幅を変更してください。例：

```
module top (in0,in1,clk,out);  
input [2:0]in0,in1;  
input clk;  
output reg [2:0] out;  
always @( posedge clk)  
begin  
out <= in0&in1;  
end  
endmodule
```

## EX3792

***WARN (EX3792): Literal value truncated to fit in <num> bits***

範囲外のパラメータ値<num>が定義されている場合、合成ツールはこのエラーメッセージを表示します。次のケースでは、4'h14 は不正な値です。

```
module top (in0,in1,out);
  input in0,in1;
  output out;
  LUT2 lut2(
    .I0(in0),
    .I1(in1),
    .F(out)
  );
  defparam lut2.INIT = 4'h14;
endmodule
```

**解決法**

parameter 値を 4'h0~4'hF に変更してください。

```
module top (in0,in1,out);
  input in0,in1;
  output out;
  LUT2 lut2(
    .I0(in0),
    .I1(in1),
    .F(out)
  );
  defparam lut2.INIT = 4'h4;
endmodule
```

**EX3794*****ERROR (EX3794): Overwriting previous definition of module<modu>***

同じ名前の 2 つの<modu>がある場合、合成ツールはこのエラーメッセージを表示します。例：

```
module test (in0,in1,out);
  input in0,in1;
  output out;
  assign out = in0 & !in1;
endmodule
module test (data,out);
  input data;
  output out;
```



```
assign out = !data;
endmodule
```

### 解決法

いずれかの **module** の名前を変更してください。

```
module test (in0,in1,out);
input in0,in1;
output out;
assign out = in0 & !in1;
endmodule
module test0 (data,out);
input data;
output out;
assign out = !data;
endmodule
```

## EX3818

### **ERROR (EX3818): <inst> expects <num> arguments**

インスタンス化モジュール<inst>のポート数が必要な数を超えている場合、合成ツールは上記のエラーメッセージを報告します。例：

```
module alu_1bit(a,b,din1,din2,sum,cout);
input din1,din2,a,b;
output cout,sum;
ALU sum_cry_0_0 (cout, sum, 0, din2, din1, a, b);
defparam sum_cry_0_0.ALU_MODE=0;
endmodule
```

### 解決法

ポート数を変更してください。

```
module alu_1bit(a,b,din1,din2,sum,cout);
input din1,din2,a,b;
output cout,sum;
ALU sum_cry_0_0 (cout, sum, din2, din1, a, b);
defparam sum_cry_0_0.ALU_MODE=0;
endmodule
```

## EX3827

### **WARN (EX3827): Full\_case directive is effective : might cause synthesis - simulation differences**

**full\_case** 文が使用されている場合、合成ツールで上記のエラーメッセージが表示されます。

```
module top (sel,in0,out);  
input sel,in0;  
output reg out;  
always@(sel or in0)  
begin  
    case(sel)/*synthesis full_case*/  
        1'b0:  
            begin  
                out <= in0;  
            end  
    endcase  
end  
endmodule
```

#### 解決法

他の **case** 条件を追加してください。

## EX3829

### **ERROR (EX3829): Port <port> is not defined**

ポート<port>がポートリストにない場合、合成ツールは上記のエラーメッセージを報告します。例：

```
module test (in,out);  
input in;  
output out;  
output out1;  
assign out = !in;  
endmodule
```

#### 解決法

out1 の宣言を削除してください。

```
module test (in,out);  
input in;
```

```
output out;
assign out = !in;
endmodule
```

## EX3833

### **ERROR (EX3833): If-condition does not match any sensitivity list edge**

**always** 文に複数の信号が含まれ、内部の **if** 条件信号がセンシティブティ・リストにない場合、合成ツールはこのエラーメッセージを表示します。  
例：

```
module top (in,out,clk1,clk2,clear);
input in,clk1,clk2,clear;
output reg out;
always @(posedge clk1 or posedge clk2)
begin
    if(clear)
        out <= 1'b0;
    else
        out <= in;
end
endmodule
```

#### 解決法

clk2 を削除して、clear をセンシティブティ・リストに追加してください。

```
module top (in,out,clk1,clear);
input in,clk1,clear;
output reg out;
always @(posedge clk1 or posedge clear)
begin
    if(clear)
        out <= 1'b0;
    else
        out <= in;
end
endmodule
```

## EX3834

### **WARN (EX3834): Case condition never applies**

**case** 文の状況が発生する可能性がない場合、合成ツールはこの警告メッ

セージを表示します。例：

```
module top (sel,in0,in1,out);
input sel,in0,in1;
output reg out;
always@(sel or in0 or in1 )
begin
    case(sel)
        1'b0:
            begin
                out <= in0;
            end
        1'b1:
            begin
                out <= in1;
            end
        3'b101:
            begin
                out <= 1'b0;
            end
    endcase
end
endmodule
```

### 解決法

不要な **case** 条件を削除してください。

```
module top (sel,in0,in1,out);
input sel,in0,in1;
output reg out;
always@(sel or in0 or in1 )
begin
    case(sel)
        1'b0:
            begin
                out <= in0;
            end
        1'b1:
            begin
                out <= in1;
            end
    endcase
end
endmodule
```

## EX3858

### ***WARN (EX3858): System task <task> ignored for synthesis***

<task>文はサポートされていません。合成中に<task>は無視され、この警告メッセージが表示されます。

## EX3863

### ***ERROR (EX3863): Syntax ERROR near <object>***

<object>の周りに構文エラーがある場合、合成ツールは上記のエラーメッセージを表示します。例えば、次の場合の最初の行では、最後にセミコロンを追加する必要があります。

```
module test (in,out)
input in;
output out;
assign out = !in;
endmodule
```

#### 解決法

構文エラーを修正してください。

```
module test (in,out);
input in;
output out;
assign out = !in;
endmodule
```

## EX3864

### ***WARN (EX3864): <port> was previously declared with a different range***

<port>が異なる幅のワイヤまたはレジスタ型変数として宣言された場合、合成ツールは上記の警告メッセージを報告します。例えば、

```
module top(in,out);
input in;
output [1:0]out;
wire [2:0] out = 3'b0+in;
endmodule
```

## 解決法

幅を一致させてください。

```
module top(in,out);  
  
input in;  
  
output [2:0]out;  
  
wire [2:0] out = 3'b0+in;  
  
endmodule
```

## EX3872

### ***ERROR (EX3872): <port> is not declared***

ポート<port>が宣言されていない場合、合成ツールは上記のエラーメッセージを報告します。

```
module test (in,out,out1);  
input in;  
output out;  
assign out = !in;  
endmodule
```

## 解決法

定義されていないポートを削除するか、このポートの定義を追加してください。

```
module test (in,out);  
input in;  
output out;  
assign out = !in;  
endmodule
```

## EX3875

### ***ERROR (EX3875) : No definition for port <port>***

ポート<port>が定義されていない場合、合成ツールは上記のエラーメッセージを報告します。例：

```
module test(in,out);  
input in;  
assign out = !in;  
endmodule
```

```
module top (top_in,top_out);  
  input top_in;  
  output top_out;  
  test test1(  
    .in(top_in),  
    .out(top_out)  
  );  
endmodule
```

### 解決法

out の入出力を定義してください。

```
module test(in,out);  
  input in;  
  output out;  
  assign out = !in;  
endmodule  
module top (top_in,top_out);  
  input top_in;  
  output top_out;  
  test test1(  
    .in(top_in),  
    .out(top_out)  
  );  
endmodule
```

## EX3900

### ***ERROR (EX3900): Procedural assignment to a non-register <nreg> is not permitted***

非レジスタ型<nreg>に対するノンブロッキング代入がある場合、合成ツールは上記のエラーメッセージを報告します。例えば、次の場合、out はレジスタ型でなければなりません。如下案例中 out 声明应为寄存器类型。

```
module top (in,out,clk);  
  input in,clk;  
  output out;  
  wire out;  
  always @( posedge clk)  
    begin  
      out <= in;  
    end  
endmodule
```

## 解決法

out をレジスタ型に変更してください。

```
module top (in,out,clk);
  input in,clk;
  output out;
  reg out;
  always @( posedge clk)
    begin
      out <= in;
    end
endmodule
```

## EX3902

### ***ERROR (EX3902): Port <port> is already defined***

ポート<port>が既に定義された場合、合成ツールは上記のエラーメッセージを報告します。例えば、次に示すように、出力ポート **out** が繰り返し定義されています。

```
module top (in,out);
  input in;
  output out;
  output out;
  assign out = !in;
endmodule
```

## 解決法

繰り返し定義を削除してください。

```
module top (in,out);
  input in;
  output out;
  assign out = !in;
endmodule
```

## EX3907

### ***ERROR (EX3907): Parameter <para>is not defined in this module***

現在のモジュールで定義されていない<para>を設定した場合、合成ツールで上記のエラーメッセージが表示されます。エラーメッセージのパラメータ名とファイル情報に従って検索および変更できます。例えば、次に示すように、INIT\_0 は module DFF のパラメータではありません。



```
module test(a,clk,out);
input a,clk;
output out;
DFF ins1(
    .D(a),
    .CLK(clk),
    .Q(out)
);
defparam ins1.INIT=1'b0;
defparam ins1.INIT_0=8'hEF;
endmodule
```

### 解決法

このパラメータを削除してください。

```
module test(a,clk,out);
input a,clk;
output out;
DFF ins1(
    .D(a),
    .CLK(clk),
    .Q(out)
);
defparam ins1.INIT=1'b0;
endmodule
```

## EX3916

***WARN (EX3916): No support for synthesis of mixed edge and level triggers. Assume level triggers only.***

**always** 文にエッジトリガー信号とレベルトリガー信号の両方がある場合、合成ツールは、上記の警告メッセージを報告します。このとき、エッジトリガー信号は無視されます。

```
module top (in,out,clk,clear);
input in,clk,clear;
output reg out;
always @(posedge clk or clear)
    if(clear)
        begin
            out <= 1'b0;
        end
    else
        begin
```

```
        out <= in;
    end
endmodule
```

### 解決法

**clear** 信号をセンシティビティ・リストから削除してください。

```
module top (in,out,clk,clear);
input in,clk,clear;
output reg out;
always @(posedge clk)
    if(clear)
    begin
        out <= 1'b0;
    end
    else
    begin
        out <= in;
    end
end
endmodule
```

## EX3927

### **ERROR (EX3927): Module<modu> remains a black box, due to ERRORS in its contents**

このメッセージは、他のエラーメッセージと同時に表示されます。  
<modu>に他のエラーがある場合、合成ツールは上記のエラーメッセージを報告して合成を終了します。

## EX3928

### **ERROR (EX3928): Module <modu> ignored due to previous ERRORS**

このメッセージは、他のエラーメッセージと同時に表示されます。  
<modu>に他のエラーがある場合、合成ツールは上記のエラーメッセージを報告して合成を終了します。

## EX3937

### **ERROR (EX3937): Instantiating unknown module <modu>**

定義されていない、他社のモジュール<modu>をインスタンス化しようとする場合、合成ツールでこのエラーメッセージが表示されます。例えば、

次に示すように、**test** は定義されていません。

```
module top (in,out);
  input in;
  output out;
  test test1(
    .in0(in),
    .out0(out)
  );
endmodule
```

### 解決法

上記のエラーを解消するには、**<modu>**を定義する必要があります。モジュールの内部実装は空でもかまいません。内部実装が空の場合、ブラックボックスになります。下図に示すように、

```
module top (in,out);
  input in;
  output out;
  test test1(
    .in0(in),
    .out0(out)
  );
endmodule
module test(in0,out0);
  input in0;
  output out0;
  assign out0 = !in0;
endmodule
```

## EX3945

### **ERROR (EX3945): Incorrect use of predefined macro <include>.** **Expected <filePath>**

**<filePath>**の指定のために**<include>**構文が誤って使用されている場合、合成ツールで上記のエラーメッセージが表示されます。例えば：

```
`include param.v;

module top(in,sel,out);

  input in,sel;

  output reg [size:0]out;

  assign out = in+sel;
```

```
endmodule

//param.v ファイルの内容

/*

parameter size = 2;

*/
```

### 解決法

`include` 文のファイルパスを二重引用符で囲んでください。

```
`include "param.v";

module top(in,sel,out);

input in,sel;

output reg [size:0]out;

assign out = in+sel;

endmodule

//param.v ファイルの内容

/*

parameter size = 2;

*/
```

## EX3983

### ***WARN (EX3983): Case condition never applies due to comparison with x or z***

`case` 文に **X** 値と **Z** が含まれている場合、合成ツールは上記の警告メッセージを表示します。例えば、次のような場合、**X** と **Z** は合成中に無視されます。

```
module top (sel,in0,in1,out);
input sel,in0,in1;
output reg out;
always@(sel or in0 or in1 )
begin
    case(sel)
        1'b0:
            begin
                out <= in0;
```

```
        end
        1'b1:
        begin
            out <= in1;
        end
        1'bX:
        begin
            out <= 1'b0;
        end
        1'bZ:
        begin
            out <= 1'b1;
        end
    endcase
end
endmodule
```

### 解決法

X と Z を削除してください。

```
module top (sel,in0,in1,out);
input sel,in0,in1;
output reg out;
always@(sel or in0 or in1 )
begin
    case(sel)
        1'b0:
        begin
            out <= in0;
        end
        1'b1:
        begin
            out <= in1;
        end
    endcase
end
endmodule
```

## EX3988

### **WARN(EX3988): Cannot open file <file>**

必要な構成ファイル<file>が存在しないか、開く権限がない場合、合成ツールで上記のエラーメッセージが表示されます。例えば、\$readmemh 文

の **initvalue** ファイルが存在しないか、開く権限がない場合、上記のエラーメッセージが表示されます。

```
module test(in,clk,addr_in,addr_out,out);  
  
input in,clk;  
  
input addr_in;  
  
input addr_out;  
  
output out;  
  
reg [7:0]mem [7:0];  
  
always @ (posedge clk)  
    mem[addr_in] <= in;  
  
assign out = mem[addr_out];  
  
initial begin  
  
    $readmemh("initvalue", mem);  
  
end  
  
endmodule
```

### 解決法

initvalue ファイルの有無と開く権限を確認してください。

## IF0003

### **ERROR (IF0003): Cannot infer <signal> due to multiple write clocks**

RAM Inference は、最大で 2 つのクロック書き込みのみをサポートします。

```
module normal5(data_out0,data_out1, data_in0,  
data_in1,data_in2,addr,addr0, addr1,addr2,clk0,clk1,clk2,ce, wre,rst);  
input [2:0]data_in0;  
input [2:0]data_in1;  
input [2:0]data_in2;  
input [3:0]addr,addr0, addr1,addr2;  
input clk0,clk1,clk2,wre,ce,rst;  
reg [2:0] mem [7:0] ;  
output reg [2:0] data_out0;  
output reg [2:0] data_out1;  
always@(posedge clk0)
```

```
        if(ce==1 & wre == 0)
            data_out0 <= mem[addr0];
    always@(posedge clk1)
        if(ce==1 & wre == 0)
            data_out1 <= mem[addr1];
    always @(posedge clk0)
        if (ce & wre) mem[addr0] <= data_in0;
    always @(posedge clk1)
        if (ce & wre) mem[addr1] <= data_in1;
    always @(posedge clk1)
        if (ce & wre) mem[addr2] <= data_in2;
endmodule
```

### 解決法

rtl 設計を変更してください。

## Place & Route からのユーザーメッセージ

### CT1000

***WARN (CT1000):<file>:<line> / This constraint of <name> is defined again, so this will overwrite the previous***

重複した制約があり、後者の制約のみが保持されます。

```
INS_LOC uut R3C4;

INS_LOC uut R4C5;
```

### 解決法

重複した制約を削除してください。

```
INS_LOC uut R4C5;
```

### CT1003

***WARN (CT1003) :<file>:<line> / Group(<name>) location is already defined, so this will overwrite the previous***

重複したグループ制約があり、後者のグループ制約のみが保持されます。

```
GROUP grp = { "ins1" "ins2" }

GRP_LOC grp R3C[3:5];

GRP_LOC grp R[4:5]C8;
```

## 解決法

重複した制約を削除してください。

```
GROUP grp = { "ins1" "ins2" }  
  
GRP_LOC grp R[4:5]C8;
```

## CT1005

***WARN (CT1005):Conflicting multiple constraints specified for location of Instance <name>(type: <type>); Or constrained location for the Instance is not available; Or constrained location type is not matched with the instance***

複数の制約の間に競合があるか、制約の位置が正しくないか、または制約の位置がプリミティブと一致していません。

## 解決法

警告メッセージに従って制約ファイルを変更してください。

## CT1007

***WARN (CT1007):There is no intersection between multiple group constraints specified for instance <name>***

制約オブジェクトは、共通の制約位置がない複数の制約グループに存在するため、制約オブジェクトには正しい制約位置がありません。

```
GROUP grp1={ "ins1" "ins2" "ins3" };  
  
GRP_LOC grp1 R2C[5:6];  
  
GROUP grp2 = { "ins1" "ins4" };  
  
GRP_LOC grp2 R4C[5:6];
```

## 解決法

制約オブジェクトを複数の制約グループに入れることを回避してください。

```
GROUP grp1={ "ins2" "ins3" };  
  
GRP_LOC grp1 R2C[5:6];  
  
GROUP grp2 = { "ins1" "ins4";  
  
GRP_LOC grp2 R4C[5:6];
```



## CT1097

***WARN (CT1097) :<file>:<line> / Please define group <name> first before define the constraint at line <number>***

制約される制約グループはまだ定義されていません。

```
GRP_LOC grp1 R2C[5:6];
```

### 解決法

制約を行う前に、まず制約グループを定義してください。

```
GROUP grp1={ "ins2" "ins3" };  
GRP_LOC grp1 R2C[5:6];
```

## CT1098

***WARN (CT1098) :<file>:<line> / Group name <name> is already defined***

制約グループの重複定義があります。例えば：

```
GROUP grp1={ "ins2" "ins3" };  
REL_GROUP grp1={ "ins4" "ins5" };  
GRP_LOC grp1 R2C[5:6];
```

### 解決法

重複した定義を削除してください。

```
GROUP grp1={ "ins4" "ins5" };  
GRP_LOC grp1 R2C[5:6];
```

## CT1101

***WARN (CT1101) :<file>:<line> / Location column <number> is out of chip range(<maxColumn>)***

制約位置情報の列がチップの範囲外です。

### 解決法

列がチップの範囲を超えないように、制約位置情報を変更してください。

## CT1102

***WARN (CT1102): <file>:<line> / Location row <number> is out of the chip range(<maxRow>)***

制約位置情報の行がチップの範囲外です。

#### 解決法

行がチップの範囲を超えないように、制約位置情報を変更してください。

## CT1108

***WARN (CT1108) :<file>:<line> / Illegal port attribute value specified <attribute> = <value> on <instName>***

属性値が正しくありません。例えば：

```
IO_PORT bufIns DRIVE=20;
```

#### 解決法

属性値を変更してください。

```
IO_PORT bufIns DRIVE=8;
```

## CT1111

***WARN (CT1111): Instance <name>(<type>) constrained to unsuitable location***

制約オブジェクトは、不合理な制約位置に制約されています。例えば：

```
INS_LOC dll_inst_2 PLL_R;
```

#### 解決法

制約オブジェクトを正しい制約位置に制約してください。

```
INS_LOC dll_inst_2 DLL_BR;
```

## CT1112

***WARN (CT1112) :<file>:<line> / Invalid range location <location>, please constrained in the same side***

レンジ制約を実行する場合、レンジの開始位置と終了位置は同じ側にある必要があります。

```
INS_LOC bufIns IOR4:IOL9;
```

## 解決法

レンジの開始位置と終了位置が同じ側にあるように変更してください。

```
INS_LOC bufIns IOR4:IOR9;
```

## CT1113

***WARN (CT1113) :<file>:<line> | Cannot find pad location <pin> in current package***

現在のパッケージにはこの制約位置がありません。

## 解決法

制約位置を変更してください。

## CT1115

***WARN (CT1115): Attribute <name> can only be set when the port is located to bank <index>. Please set the corresponding location constraint of port <portName>***

属性制約を行う前に、位置の制約が必要です。

```
IO_PORT i0 IO_TYPE=RSDS25E DIFF_RESISTOR=ON;
```

## 解決法

属性制約の前に、位置の制約を実行してください。

```
IO_LOC i0 IOT4;
```

```
IO_PORT i0 IO_TYPE=RSDS25E DIFF_RESISTOR=ON;
```

## CT1116

***WARN (CT1116): Attribute <name> can only be set when the port is located to bank <index>. Please set the corresponding location constraint of port <portName> or <portName>***

差動ポートの属性制約を行う前に、位置の制約が必要です。

```
IO_PORT I IO_TYPE=RSDS25E DIFF_RESISTOR=ON;
```

## 解決法

属性制約の前に、ポート **IB** またはその差動ポート **I** に位置の制約を実行してください。

```
IO_LOC IB IOT4;
```

```
IO_PORT I IO_TYPE=RSDS25E DIFF_RESISTOR=ON;
```

或 :

```
IO_LOC I IOT4;
```

```
IO_PORT I IO_TYPE=RSDS25E DIFF_RESISTOR=ON;
```

## CT1117

***WARN (CT1117): Attribute <name> can only be set when the port is located to bank <index>, but the constraint location of port <portName> include other bank***

属性制約の値と制約位置が一致していません。

```
IO_LOC i0 IOB4;
```

```
IO_PORT i0 IO_TYPE=RSDS25E DIFF_RESISTOR=ON;
```

### 解決法

属性制約または制約位置を変更してください。

```
IO_LOC i0 IOT4;
```

```
IO_PORT i0 IO_TYPE=RSDS25E DIFF_RESISTOR=ON;
```

## CT1118

***WARN (CT1118): Attribute <name> can only be set when the port is located to bank <index>, but the constraint location of port <portName> or <portName> include other bank***

属性制約の値と制約位置が一致していません。

```
IO_LOC I IOB4;
```

```
IO_PORT I IO_TYPE=RSDS25E DIFF_RESISTOR=ON;
```

### 解決法

ポート I またはその差動ポート IB の属性制約または位置制約を変更してください。

```
IO_LOC I IOT4;
```

```
IO_PORT I IO_TYPE=RSDS25E DIFF_RESISTOR=ON;
```

或 :

```
IO_LOC IB IOT4;
```

```
IO_PORT I IO_TYPE=RSDS25E DIFF_RESISTOR=ON;
```

## FS1008

***WARN (FS1008):Device <device type> is not supported AES encryption, please uncheck in bitstream configurations***

AES 暗号化はサポートされていません。

### 解決法

構成オプションを変更して暗号化をキャンセルしてください。

## FS2001

***ERROR (FS2001) :Cannot read corrupted fse file***

fse ファイルの読み出しに失敗しました。

### 解決法

現在のソフトウェアと一致する **fse** ファイルを使用してください。また、fse ファイルを削除または変更しないでください。

## PA1000

***WARN (PA1000):Dangling net <netName> in module <moduleName> has no source instance***

モジュール内のワイヤには信号ソースがありません。

```
module test (i0,i1,i2,i3,out);  
  
  input i0;  
  
  input i1;  
  
  input i2;  
  
  input i3;  
  
  output out;  
  
  wire i4;  
  
  LUT4 uut (  
    .I0(i0),  
    .I1(i1),  
    .I2(i2),
```

```
.I3(i4),  
    .F(out)  
);  
  
endmodule
```

### 解決法

ワイヤには信号ソースがあるかどうか確認してください。フローティングのままにする必要がある場合は、この警告メッセージを無視してください。

```
module test (i0,i1,i2,i3,out);  
    input i0;  
    input i1;  
    input i2;  
    input i3;  
    output out;  
    LUT4 uut (  
        .I0(i0),  
        .I1(i1),  
        .I2(i2),  
        .I3(i3),  
        .F(out)  
    );  
  
endmodule
```

## PA1001

***WARN (PA1001): Dangling net  
<netName>(source:<instanceName>) in module <moduleName> has  
no destination***

指定されたモジュールのワイヤが対象プリミティブに接続されていません。

```
module test (i0,i1,i2,i3,out);  
    input i0;
```

```
input i1;

input i2;

input i3;

output out;

wire out_c;

LUT4 uut (

    .I0(i0),

    .I1(i1),

    .I2(i2),

    .I3(i3),

    .F(out_c)

);

endmodule
```

### 解決法

ワイヤが正しく接続されたことを確認してください。フローティングのままにする必要がある場合は、この警告メッセージを無視してください。

```
module test (i0,i1,i2,i3,out);

input i0;

input i1;

input i2;

input i3;

output out;

wire out_c;

LUT4 uut (

    .I0(i0),

    .I1(i1),

    .I2(i2),

    .I3(i3),

    .F(out_c)

);

endmodule
```

```
);  
  
OBUF buf_ins (  
    .I(out_c),  
    .O(out)  
);  
  
endmodule
```

## PA1002

***WARN (PA1002): <file>:<line> / Invalid parameterized value <value>( <parameter>) specified for instance <instanceName>***

指定された場所のプリミティブには、間違ったパラメータ値が構成されています。

### 解決法

プリミティブのパラメータ値を正しく構成してください。

## PA1008

***WARN (PA1008): <file>:<line> / Object <name> is already defined***

重複定義があります。例えば：

```
module test (i0,i1,i2,i3,out);  
  
    input i0;  
  
    input i1;  
  
    input i2;  
  
    input i3;  
    input i3;  
  
    output out;  
  
    LUT4 uut (  
        .I0(i0),  
        .I1(i1),  
        .I2(i2),  
        .I3(i3),
```



```
.F(out)

);

endmodule
```

### 解決法

重複定義を削除してください。

```
module test (i0,i1,i2,i3,out);

input i0;

input i1;

input i2;

input i3;

output out;

LUT4 uut (

    .I0(i0),

    .I1(i1),

    .I2(i2),

    .I3(i3),

    .F(out)

);

endmodule
```

## PA1010

***WARN (PA1010): <file>:<line> / Dangling pin(<name>) is not connect with net***

ピンが正しく接続されていません。例えば：

```
module test (i0,i1,i2,i3,out);

input i0;

input i1;

input i2;

input i3;

output out;
```

```
LUT4 uut (  
    .I0(i0),  
    .I1(i1),  
    .I2(i2),  
    .I3(i3),  
    .F()  
);  
  
endmodule
```

### 解決法

ピンを正しく接続してください。フローティングのままにする必要がある場合は、この警告メッセージを無視してください。

```
module test (i0,i1,i2,i3,out);  
    input i0;  
    input i1;  
    input i2;  
    input i3;  
    output out;  
  
    LUT4 uut (  
        .I0(i0),  
        .I1(i1),  
        .I2(i2),  
        .I3(i3),  
        .F(out)  
    );  
  
endmodule
```

## PA2000

***ERROR (PA2000): <file>:<line> / Syntax error near token <name>***

構文エラーがあります。例えば：

```
module test (i0,i1,i2,i3,out);
```

```
input i0;

input i1;

input i2;

input i3;

ouput out;

LUT4 uut (

    .I0(i0),

    .I1(i1),

    .I2(i2),

    .I3(i3),

    .F(out)

);

endmodule
```

## 解決法

構文エラーを直してください。

```
module test (i0,i1,i2,i3,out);

input i0;

input i1;

input i2;

input i3;

output out;

LUT4 uut (

    .I0(i0),

    .I1(i1),

    .I2(i2),

    .I3(i3),

    .F(out)

);

endmodule
```

## PA2001

### ***ERROR (PA2001): <file>:<line> / Module <moduleName> is already defined***

モジュールの重複定義があります。例えば：

```
module test (i0,i1,i2,i3,out);  
    input i0;  
    input i1;  
    input i2;  
    input i3;  
    output out;  
    LUT4 uut (  
        .I0(i0),  
        .I1(i1),  
        .I2(i2),  
        .I3(i3),  
        .F(out)  
    );  
endmodule  
  
module test (I0,I1,OUT);  
    input I0;  
    input I1;  
    output OUT;  
    LUT2 uut (  
        .I0(I0),  
        .I1(I1),  
        .F(OUT)  
    );  
endmodule
```

## 解決法

重複したモジュールの名前を変更してください。

```
module test (i0,i1,i2,i3,out);  
  
    input i0;  
  
    input i1;  
  
    input i2;  
  
    input i3;  
  
    output out;  
  
    LUT4 uut (  
        .I0(i0),  
        .I1(i1),  
        .I2(i2),  
        .I3(i3),  
        .F(out)  
    );  
  
endmodule  
  
module testLut2 (I0,I1,OUT);  
  
    input I0;  
  
    input I1;  
  
    output OUT;  
  
    LUT2 uut (  
        .I0(I0),  
        .I1(I1),  
        .F(OUT)  
    );  
  
endmodule
```

## PA2004

***ERROR (PA2004) :<file> / In module <name>: Net <netName> driven by multiple source instances***

モジュールのワイヤに複数の信号ソースがあります。例えば：

```
module test (i0,i1,i2,i3,out, out1);  
  
    input i0;  
  
    input i1;  
  
    input i2;  
  
    input i3;  
  
    output out;  
  
    output out1;  
  
    wire out_c;  
  
    LUT4 uut (  
        .I0(i0),  
        .I1(i1),  
        .I2(i2),  
        .I3(i3),  
        .F(out_c)  
    );  
  
    LUT2 uut2 (  
        .I0(i0),  
        .I1(i1),  
        .F(out_c)  
    );  
  
    OBUF bufIns (  
        .I(out_c),  
        .O(out)  
    );  
  
endmodule
```

## 解決法

指定されたモジュールの接続関係を変更してください。

```
module test (i0,i1,i2,i3,out,out1);
```

```
input i0;
```

```
input i1;
```

```
input i2;
```

```
input i3;
```

```
output out;
```

```
output out1;
```

```
wire out1_c;
```

```
wire out_c;
```

```
LUT4 uut (
```

```
    .I0(i0),
```

```
    .I1(i1),
```

```
    .I2(i2),
```

```
    .I3(i3),
```

```
    .F(out_c)
```

```
);
```

```
LUT2 uut2 (
```

```
    .I0(i0),
```

```
    .I1(i1),
```

```
    .F(out1_c)
```

```
);
```

```
OBUF bufIns (
```

```
    .I(out_c),
```

```
    .O(out)
```

```
);
```

```
OBUF bufIns1 (
```

```
    .I(out1_c),
```

```
.0(out1)

);

endmodule
```

## PA2009

***ERROR (PA2009): The port <name> connected to <instName>(instType) defined error direction which should be <portType> according to connection***

ポートのタイプが正しくありません。例えば：

```
module test (i0,i1,i2,i3,out);

input i0;

input i1;

input i2;

input i3;

input out;

LUT4 uut (

    .I0(i0),

    .I1(i1),

    .I2(i2),

    .I3(i3),

    .F(out)

);

endmodule
```

### 解決法

ポートのタイプを変更してください。

```
module test (i0,i1,i2,i3,out);

input i0;

input i1;

input i2;

input i3;
```



```
output out;
```

```
LUT4 uut (  
    .I0(i0),  
    .I1(i1),  
    .I2(i2),  
    .I3(i3),  
    .F(out)  
);  
  
endmodule
```

## PA2014

***ERROR (PA2014): Pin(<name>) of <instName>(<instType>) does not connect to port***

指定されたインスタンスのピンは、モジュールのポートに接続されていません。例えば：

```
module test (i0,i1,i2,i3,out);  
  
    input i0;  
    input i1;  
    input i2;  
    input i3;  
  
    output out;  
  
    wire i0_c;  
    wire VCC;  
    wire io;  
  
    LUT4 uut (  
        .I0(i0_c),  
        .I1(i1),  
        .I2(i2),  
        .I3(i3),  
        .F(out)
```

```
);  
  
IOBUF bufIns (  
    .I(i0),  
    .O(i0_c),  
    .IO(io),  
    .OEN(VCC)  
);  
  
VCC vcc (  
    .V(VCC)  
);  
  
endmodule
```

### 解決法

インスタンスのピンとモジュールのポートの接続を確認してください。

```
module test (i0,i1,i2,i3,out,io);  
  
    input i0;  
  
    input i1;  
  
    input i2;  
  
    input i3;  
  
    output out;  
  
    inout io;  
  
    wire i0_c;  
  
    wire VCC;  
  
    wire io;  
  
    LUT4 uut (  
        .I0(i0_c),  
        .I1(i1),  
        .I2(i2),  
        .I3(i3),  
        .F(out)
```

```
);  
  
IOBUF bufIns (  
    .I(i0),  
    .O(i0_c),  
    .IO(io),  
    .OEN(VCC)  
);  
  
VCC vcc (  
    .V(VCC)  
);  
  
endmodule
```

## PA2017

***ERROR (PA2017): The number(<value>) of <instType> in the design exceeds the resource limit(<maxValue>)of current device***

プリミティブ数が制限を超えています。

### 解決法

プリミティブ数を減らすか、リソースの多いチップを使用してください。

## PA2024

***ERROR (PA2024): The number(<value>) of ports exceeds the resource limit <maxValue> regular I/Os(include <value> dedicated I/Os) and <value> shared I/Os of current device***

トップモジュールのポート数が制限を超えています。

### 解決法

パッケージを変更するか、リソースの多いチップを使用してください。

## PA2025

***ERROR (PA2025): No <instType> resource in current device***

サポートされないリソースがあります。

## 解決法

デザインを変更するか、当該リソースをサポートする他のチップを使用してください。

## PA2039

### ***ERROR (PA2039): Net <name>is used in module <moduleName> but not declared in wire list***

宣言されていないワイヤがあります。例えば：

```
module test (i0,i1,i2,i3,out);  
  
    input i0;  
  
    input i1;  
  
    input i2;  
  
    input i3;  
  
    output out;  
  
    LUT4 uut (  
        .I0(i0),  
        .I1(i1),  
        .I2(i2),  
        .I3(i3),  
        .F(out_c)  
    );  
  
    OBUF obufIns (  
        .O(out),  
        .I(out_c)  
    );  
  
endmodule
```

## 解決法

当該ワイヤを宣言してください。

```
module test (i0,i1,i2,i3,out);  
  
    input i0;
```

```
input i1;

input i2;

input i3;

output out;

wire out_c;

LUT4 uut (

    .I0(i0),

    .I1(i1),

    .I2(i2),

    .I3(i3),

    .F(out_c)

);

OBUF obufIns (

    .O(out),

    .I(out_c)

);

endmodule
```

## PA2054

***ERROR (PA2054): <file>:<line> / <name> is already declared***

インスタンスの名前が重複しています。

```
module test (i0,i1,i2,i3,out);

input i0;

input i1;

input i2;

input i3;

output out;

wire out_c;

LUT4 uut (
```

```
.I0(i0),  
.I1(i1),  
.I2(i2),  
.I3(i3),  
.F(out_c)  
);  
  
OBUF uut (  
    .O(out),  
    .I(out_c)  
);  
  
endmodule
```

### 解決法

重複した名前を変更してください。

```
module test (i0,i1,i2,i3,out);  
    input i0;  
    input i1;  
    input i2;  
    input i3;  
    output out;  
    wire out_c;  
    LUT4 uut (  
        .I0(i0),  
        .I1(i1),  
        .I2(i2),  
        .I3(i3),  
        .F(out_c)  
    );  
    OBUF obufIns (  
        .O(out),
```

```
.I(out_c)

);

endmodule
```

## PA2056

***ERROR (PA2056): <file>:<line> / Error pin name(<name>) found in instance <instName>***

インスタンスのピン名が正しくありません。例えば：

```
module test (i0,i1,i2,i3,out);

input i0;

input i1;

input i2;

input i3;

output out;

wire out_c;

LUT4 uut (

    .I0(i0),

    .I1(i1),

    .I2(i2),

    .I4(i3),

    .F(out_c)

);

OBUF bufIns (

    .O(out),

    .I(out_c)

);

endmodule
```

### 解決法

ピン名を修正してください。

```
module test (i0,i1,i2,i3,out);
```

```
input i0;

input i1;

input i2;

input i3;

output out;

wire out_c;

LUT4 uut (

    .I0(i0),

    .I1(i1),

    .I2(i2),

    .I3(i3),

    .F(out_c)

);

OBUF bufIns (

    .O(out),

    .I(out_c)

);

endmodule
```

## PA2058

***ERROR (PA2058): <file>:<line> / Error pin number within instance <name>(<type>) of module<name>***

インスタンスのピン数が正しくありません。例えば：

```
module test (i0,i1,i2,i3,out);

input i0;

input i1;

input i2;

input i3;

output out;

wire out_c;
```



```
LUT4 uut (  
    .I0({i0,i1}),  
    .I1(i1),  
    .I2(i2),  
    .I3(i3),  
    .F(out_c)  
);  
  
OBUF bufIns (  
    .O(out),  
    .I(out_c)  
);  
  
endmodule
```

### 解決法

ピン数を変更してください。

```
module test (i0,i1,i2,i3,out);  
  
    input i0;  
    input i1;  
    input i2;  
    input i3;  
    output out;  
    wire out_c;  
  
    LUT4 uut (  
        .I0(i0),  
        .I1(i1),  
        .I2(i2),  
        .I3(i3),  
        .F(out_c)  
    );  
  
    OBUF bufIns (  
        .O(out),  
        .I(out_c)  
    );  
  
endmodule
```

```
.O(out),  
.I(out_c)  
);  
endmodule
```

## PA2066

***ERROR(PA2066): <file>:<line> / Invalid parameter name <name> setting to object <instName>***

パラメータがサポートされていません。例えば：

```
module test (i0,i1,i2,i3,out);  
    input i0;  
    input i1;  
    input i2;  
    input i3;  
    output out;  
    wire out_c;  
    LUT4 uut (  
        .I0(i0),  
        .I1(i1),  
        .I2(i2),  
        .I3(i3),  
        .F(out_c)  
    );  
    defparam uut.INIT_1=16'h0000;  
    OBUF bufIns (  
        .O(out),  
        .I(out_c)  
    );  
endmodule
```

## 解決法

パラメータを変更してください。

```
module test (i0,i1,i2,i3,out);  
  
    input i0;  
  
    input i1;  
  
    input i2;  
  
    input i3;  
  
    output out;  
  
    wire out_c;  
  
    LUT4 uut (  
        .I0(i0),  
        .I1(i1),  
        .I2(i2),  
        .I3(i3),  
        .F(out_c)  
    );  
  
    defparam uut.INIT=16'h0000;  
  
    OBUF bufIns (  
        .O(out),  
        .I(out_c)  
    );  
  
endmodule
```

## PR0026

***ERROR (PR0026): CLKOUTN pin of <name> is not connected to any other iologic***

DHCENC の CLKOUTN pin は IOLOGIC を駆動していません。

## 解決法

DHCENC の CLKOUTN pin が IOLOGIC を適切に駆動するようにデザインを変更してください。

## PR0027

**ERROR (PR0027): Instance <name> connected to CLKIN pin of instance <name> is unsupported**

このインスタンスを PLL または DLL の CLKIN pin に接続することはサポートされていません。

### 解決法

このインスタンスを PLL または DLL の CLKIN pin に接続しないようにデザインを変更してください。

## PR0028

**ERROR (PR0028): Instance <name> connected to CLKFB pin of instance <name> is unsupported**

このインスタンスを PLL の CLKFB pin に接続することはサポートされていません。

### 解決法

このインスタンスを PLL の CLKFB pin に接続しないようにデザインを変更してください。

## PR0029

**ERROR (PR0029): Instance <name>(INS\_DHCENC) cannot drive two CLKDIVs**

DHCENC は、同時に 2 つの CLKDIV を駆動することはできません。

### 解決法

DHCENC が 2 つの CLKDIV を同時に駆動しないようにデザインを変更してください。

## PR1011

**ERROR (PR1011): Failed to capture gao signal:<name>, because there's no wire to route for the signal**

GAO は信号をキャプチャできませんでした。例えば、次に示すように、ALU チェーンの信号” c0\_c” は配線できないため、キャプチャは失敗します。

```
module test (i0, i1, i2, i3, o0, o1, o2);
```

```
input i0, i1, i2, i3;

output o0, o1, o2;

wire i0_c, i1_c, i2_c, i3_c, c0_c, c1_c, s0_c, s1_c, GND;

GND GND_C(.G(GND));

IBUF ibuf_i0(.I(i0), .O(i0_c));

IBUF ibuf_i1(.I(i1), .O(i1_c));

IBUF ibuf_i2(.I(i2), .O(i2_c));

IBUF ibuf_i3(.I(i3), .O(i3_c));

ALU
alu_0(.I0(i0_c), .I1(i1_c), .I3(GND), .CIN(GND), .COUT(c0_c), .SUM(s0_c));

defparam alu_0.ALU_MODE = 0;

ALU
alu_1(.I0(i2_c), .I1(i3_c), .I3(GND), .CIN(c0_c), .COUT(c1_c), .SUM(s1_c));

defparam alu_1.ALU_MODE = 0;

OBUF obuf_sum0(.I(s0_c), .O(o0));

OBUF obuf_sum1(.I(s1_c), .O(o1));

OBUF obuf_cout(.I(c1_c), .O(o2));

endmodule
```

## 解決法

キャプチャできない場合は、論理的に前のレベルまたは次のレベルの信号をキャプチャして解析してください。例えば：

```
module test (i0, i1, i2, i3, o0, o1, o2);

input i0, i1, i2, i3;

output o0, o1, o2;

wire i0_c, i1_c, i2_c, i3_c, c0_c, c1_c, s0_c, s1_c, GND;

GND GND_C(.G(GND));

IBUF ibuf_i0(.I(i0), .O(i0_c));

IBUF ibuf_i1(.I(i1), .O(i1_c));

IBUF ibuf_i2(.I(i2), .O(i2_c));

IBUF ibuf_i3(.I(i3), .O(i3_c));
```

```
    ALU
alu_0(.I0(i0_c),.I1(i1_c),.I3(GND),.CIN(GND),.COUT(c0_c),.SUM(s0_c));

    defparam alu_0.ALU_MODE = 0;

    ALU
alu_1(.I0(i2_c),.I1(i3_c),.I3(GND),.CIN(c0_c),.COUT(c1_c),.SUM(s1_c));

    defparam alu_1.ALU_MODE = 0;

    OBUF obuf_sum0(.I(s0_c), .0(o0));

    OBUF obuf_sum1(.I(s1_c), .0(o1));

    OBUF obuf_cout(.I(c1_c), .0(o2));

endmodule
```

## PR1014

***WARN(PR1014): Generic routing resource will be used to clock signal<name> by the specified constraint. And then it may lead to the excessive delay or skew***

Gowin Router は、クロック信号の配線に論理配線があることを検出します。これにより、クロック遅延またはスキューが発生する可能性があります。例えば、次のテストケース（デバイスは **GW1N-4**）では、クロック信号” **clk\_c**” のソースが非クロックポートに制約されているため、クロック信号がいくつかの論理配線リソースを通過することになります。

```
top.vh

Module test_clk()

input i0, i1

output o0

IBUF ibuf_data(.I(i0), .0(d_c));

IBUF ibuf_clk(.I(i1), .0(clk_c));

DFF dff_c(.D(d_c), .CLK(clk_c), .Q(q_c));

OBUF obuf_c(.I(q_c), .0(o0));

endmodule

top.cst
```

```
IO_LOC "ibuf_data" IOB18A;
```

### 解決法

クロック信号のソースがクロック信号ソースであるかどうか、またはクロック信号が接続されているポートの物理制約位置がクロックポートであるかどうかを確認してください。例えば、次のテストケース（デバイスはGW1N-4）では、IOB20A がクロックポートであるため、クロック信号“clk\_c”をこのポートに制限します。

```
top.vh
Module test_clk()
  input i0, i1
  output o0

  IBUF ibuf_data(.I(i0), .O(d_c));
  IBUF ibuf_clk(.I(i1), .O(clk_c));
  DFF dff_c(.D(d_c), .CLK(clk_c), .Q(q_c));
  OBUF obuf_c(.I(q_c), .O(o0));

endmodule

top.cst

IO_LOC "ibuf_data" IOB20A;
```

## PR2044

***WARN (PR2044): FCLK port of <name> conflicts with FCLK port of <name> and <FCLK or HCLKIN> port of <name>***

最初のインスタンスの FCLK は、2 番目のインスタンスの FCLK および 3 番目のインスタンスの FCLK または HCLKIN とコリニアである必要があります。

### 解決法

コリニアになるようにデザインを変更してください。

## PR2045

***WARN (PR2045): FCLK port of <name> conflicts with FCLK port of <name>***

最初のインスタンスの FCLK は 2 番目のインスタンスの FCLK とコリニアである必要があります。

#### 解決法

コリニアになるようにデザインを変更してください。

### PR2061

#### ***ERROR (PR2061): There is no position to place <name>***

このインスタンスを配置する位置がありません。

#### 解決法

このインスタンスの配置に影響を与える他のインスタンスの配置位置を変更して、配置に使用可能なリソースを確保します。

### PR2062

#### ***ERROR (PR2062): Objects driven by CLKOUT pin of <name> must be placed on same side***

同じ DHCENC の CLKOUT により駆動される IOLOGIC は同じ側にある必要があります。

#### 解決法

他の使用可能な位置に変更してください。

### PR2063

#### ***ERROR (PR2063): Objects driven by CLKOUTN pin of <name> must be placed on same side with buffer <name>***

同じ DHCENC の CLKOUTN により駆動される IOLOGIC はその BUFFER と同じ側にある必要があります。

#### 解決法

他の使用可能な位置に変更してください。

### PR2064

#### ***ERROR (PR2064): Buffer<name> driving DHCENC must be placed to GCLK***

DHCENC を駆動する BUFFER を GCLK に配置する必要があります。



### 解決法

他の使用可能な GCLK 位置に変更してください。

## PR2065

***ERROR (PR2065): Buffer<name> driving DLLDLY must be placed to GCLK***

DLLDLY を駆動する BUFFER を GCLK に配置する必要があります。

### 解決法

他の使用可能な GCLK 位置に変更してください。

## PR2066

***ERROR (PR2066): Iologics need more than two hclk on <chip side>***

IOLOGICs には 2 つ以上の hclk が必要です。

### 解決法

hclk リソースが十分で利用可能な位置に変更してください。

## PR2067

***ERROR (PR2067): Instance <name> must have constraint***

GW1N-9C と GW1NR-9C では、このインスタンスには制約が必要です。

### 解決法

このインスタンスに制約を追加してください。

## PR2068

***ERROR (PR2068): Instance <name> must have unique constraint***

GW1N-9C と GW1NR-9C では、このインスタンスには一意の制約が必要です。

### 解決法

このインスタンスに一意の制約を追加してください。

## PR2069

***ERROR (PR2069): The constrained location of <name> cannot be***

## **IO BLOCK**

インスタンスの制約位置を IOB にすることはできません。

### **解決法**

IOB 以外の位置に制約してください。

## **PR2070**

### ***ERROR (PR2070): Instance <name> connected to IODELAYC cannot be placed to bottom side***

IODELAYC に接続された BUFFER を bottom 側に制約することはできません。

### **解決法**

他の使用可能な位置に変更してください。

## **TA1001**

### ***WARN(TA1001): Either option "-name" or option "<source objects>" should be specified***

create\_clock のとき、対象位置もクロックの名前も指定されていません。

```
create_clock -period 10 -waveform {0 5}
```

### **解決法**

sdc 制約を変更し、クロック名と対象位置を指定してください。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]
```

## **TA1004**

### ***WARN(TA1004): Clock waveform should not contain two edges with time larger than one clock period***

-waveform で指定されたクロックエッジが -period で指定された周期より大きいです。

```
create_clock -name clk1 -period 10 -waveform {0 15} [get_ports {clk}]
```

### **解決法**

クロックエッジが 1 クロックサイクル内になるように、-waveform または -period の値を変更してください。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]
```

## TA1006

### ***WARN(TA1006): The waveform list is not monotonically increasing***

-waveform で指定されたクロックエッジはインクリメンタルではありません。

```
create_clock -name clk1 -period 10 -waveform {5 0} [get_ports {clk}]
```

#### 解決法

指定したクロックエッジがインクリメンタルであるように-waveform を変更してください。

```
create_clock -name clk1 -period 10 -waveform {5 10} [get_ports {clk}]
```

## TA1011

### ***WARN(TA1011): Option "-rise" and option "-fall" are mutually exclusive***

set\_input\_delay/set\_output\_delay のとき、-rise と -fall が同時に使用されています。-rise と -fall は相互に排他的です。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]
set_input_delay -clock clk1 1 -rise -fall -max [get_ports {in01}]
```

#### 解決法

-rise または -fall を削除してください。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]
set_input_delay -clock clk1 1 -rise -max [get_ports {in01}]
set_input_delay -clock clk1 1 -fall -max [get_ports {in01}]
```

## TA1012

### ***WARN(TA1012): Option "-max" and option "-min" are mutually exclusive***

set\_input\_delay/set\_output\_delay のとき、-max と -min が同時に使用されています。-max と -min は相互に排他的です。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]
set_input_delay -clock clk1 1 -rise -max -min [get_ports {in01}]
```

## 解決法

-max または -min を削除してください。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
set_input_delay -clock clk1 1 -rise -max [get_ports {in01}]  
set_input_delay -clock clk1 1 -rise -min [get_ports {in01}]
```

## TA1016

***WARN(TA1016):Options "-edges -edge\_shift" and options "-divide\_by -multiply\_by -duty\_cycle -phase -offset" are mutually exclusive: specify either of the two ways***

sdc 制約 create\_generated\_clock では、派生クロックの周波数と位相を決定する方法が 2 つあります。1 つは -edges および -edge\_shift を使用する方法、もう 1 つは -divide\_by / -multiply\_by / -duty\_cycle / -phase / -offset を使用する方法です。ただし、これら 2 つの方法を混合してはなりません。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
create_generated_clock -name genClk -source [get_ports {clk}]  
-master_clock clk1 -edges {1 3 5} -edge_shift {1 1 1} -divide_by 2 [get_pins  
{reg0_0_Z/Q}]
```

## 解決法

これら 2 つの方法を混合しないでください。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
create_generated_clock -name genClk -source [get_ports {clk}]  
-master_clock clk1 -divide_by 2 [get_pins {reg0_0_Z/Q}]  
  
create_generated_clock -name genClk2 -source [get_ports {clk}]  
-master_clock clk1 -edges {1 3 5} -edge_shift {1 1 1} [get_pins  
{reg0_0_Z/Q}] -add
```

## TA1019

***WARN(TA1019): Option "-edges" must be in non-decreasing order***

-edges で指定されたパラメータはインクリメンタルではありません。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
create_generated_clock -name genClk -source [get_ports {clk}]
```

```
-master_clock clk1 -edges {3 1 5} -edge_shift {1 1 1} [get_pins {reg0_0_Z/Q}]
```

### 解決法

指定したクパラメータがインクリメンタルであるように `create_generated_clock` を変更してください。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
  
create_generated_clock -name genClk -source [get_ports {clk}]  
-master_clock clk1 -edges {1 3 5} -edge_shift {1 1 1} [get_pins {reg0_0_Z/Q}]
```

## TA1027

### ***WARN(TA1027): Missing required clock latency delay***

`set_clock_latency` 制約では、遅延値は指定されていません。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
  
set_clock_latency -source -late -fall [get_ports {clk}] -clock  
[get_clocks {clk1}]
```

### 解決法

遅延値を指定してください。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
  
set_clock_latency -source -late -fall [get_ports {clk}] -clock  
[get_clocks {clk1}] 1
```

## TA1032

### ***WARN(TA1032): Option "-from" must be used with "get\_clocks"***

`set_clock_uncertainty` 制約では、`-from` の後に `get_clocks` を使用する必要があります。他の方法が使用されている場合、この情報が報告されます。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
  
set_clock_uncertainty 1 -setup -from [get_ports {clk}] -to [get_clocks  
{clk1}]
```

### 解決法

`-from` の後に `get_clocks` を使用してください。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
  
set_clock_uncertainty 1 -setup -from [get_clocks {clk1}] -to [get_clocks  
{clk1}]
```

## TA1033

### ***WARN(TA1033): Option "-to" must be used with "get\_clocks"***

`set_clock_uncertainty` 制約では、`-to` の後に `get_clocks` を使用する必要があります。他の方法が使用されている場合、この情報が報告されます。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
  
set_clock_uncertainty 1 -setup -from [get_clocks {clk1}] -to [get_ports {clk}]
```

#### 解決法

`-to` の後に `get_clocks` を使用してください。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
  
set_clock_uncertainty 1 -setup -from [get_clocks {clk1}] -to [get_clocks {clk1}]
```

## TA1048

### ***WARN(TA1048): Existing clock <name> is overwritten***

名前重複のため、既存のクロックは上書きされます。`sdc` コマンドで作成されたクロックの名前は、既存のクロックと同じためです。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
  
create_clock -name clk1 -period 20 -waveform {0 5} [get_ports {clk}] -  
add
```

#### 解決法

クロックの名前を変更してください。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
  
create_clock -name clk2 -period 20 -waveform {0 5} [get_ports {clk}] -  
add
```

## TA1049

### ***WARN(TA1049): Object<name>alreadyhas one clock applied on it, if you want one more, please use -add option***

同じ位置に複数のクロックを追加するには、2 番目のクロックから制約に `-add` を追加する必要があります。`-add` を使用してください。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
create_clock -name clk2 -period 20 -waveform {0 5} [get_ports {clk}]
```

### 解決法

2 番目のクロックから制約に **-add** を追加してください。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
create_clock -name clk2 -period 20 -waveform {0 5} [get_ports {clk}] -  
add
```

## TA1058

### ***WARN(TA1058): Input ports list has output ports <name>***

sdc 制約 **set\_input\_delay** では、入力ポートリストに出力ポートがあります。**set\_input\_delay** には、入力ポートしか追加できません。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
set_input_delay -clock clk1 1 -min -fall [get_ports {out}]
```

### 解決法

入力ポートリストから出力ポートを削除してください。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
set_input_delay -clock clk1 1 -min -fall [get_ports {in}]
```

## TA1059

### ***WARN(TA1059): Output ports list has input ports <name>***

sdc 制約 **set\_output\_delay** では、出力ポートリストに入力ポートがあります。**set\_output\_delay** は、出力ポートにしか追加できません。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
set_output_delay -clock clk1 1 -min -fall [get_ports {in}]
```

### 解決法

出力ポートリストから入力ポートを削除してください。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
set_output_delay -clock clk1 1 -min -fall [get_ports {out}]
```

## TA1061

***WARN(TA1061): Cannot find objects matching with <name>***

sdc 制約で指定されたオブジェクトが見つかりません。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {port_no_exist}]
```

**解決法**

sdc 制約で指定されるオブジェクトの名前を変更してください。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]
```

**TA1068*****WARN(TA1068):Previous IO timing constraints are overwritten***

I/O 制約が上書きされます。set\_input\_delay / set\_output\_delay は、-max/-min と -rise/-fall の 2 つのパラメーターセットを使用して制約範囲を指定できます。-max/-min が指定されていない場合、max と min の両方がデフォルトで分析されます (-rise / -fall も同様)。次の制約範囲が以前の制約範囲と重複する場合にこの警告メッセージが報告されます。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]
set_input_delay -clock clk1 1 -max -fall [get_ports {in01}]
set_input_delay -clock clk1 2 [get_ports {in01}]
```

**解決法**

-max/-min および -rise/-fall を明示的に指定することをお勧めします。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]
set_input_delay -clock clk1 1 -max -fall [get_ports {in01}]
set_input_delay -clock clk1 2 -min -rise [get_ports {in01}]
```

**TA1076*****WARN(TA1076): Generated clock Source list has source object <object> which specified by option "-source", this generated clock will be ignored***

sdc 制約 create\_generated\_clock では、対象位置はメインクロックのソース位置です。create\_generated\_clock の対象位置をソース位置と同じにすると、この警告メッセージが報告されます。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]
```



```
create_generated_clock -name genClk -source [get_ports {clk}]  
-master_clock clk1 -divide_by 2 [get_ports {clk}]
```

### 解決法

対象位置をメインクロックのソース位置以外の位置に設定してください。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
  
create_generated_clock -name genClk -source [get_ports {clk}]  
-master_clock clk1 -divide_by 2 [get_pins {reg0_0_Z/Q}]
```

## TA1109

### ***WARN (TA1109): Invalid speed grade is specified"***

**sdc** では不正なスピードグレードが設定されています。GW1N シリーズのスピードグレードは **5,6** (コマーシャル)、**4,5** (インダストリアル)、および **4** (オートモーティブ) です。GW2A シリーズのスピードグレードは **7,8** (コマーシャル)、**6,7** (インダストリアル)、および **6** (オートモーティブ) です。

```
set_operating_conditions -grade c -model slow -speed 1
```

### 解決法

スピードグレードが要件を満たしていることを確認してください。

```
set_operating_conditions -grade c -model slow -speed 5
```

## TA1114

### ***WARN(TA1114): Invalid access is specified***

**sdc** では、アクセス方法が正しくありません。このメッセージには **3** つの理由が考えられます。

1. **set\_clock\_groups** 制約では、**-group** の後のアクセス方法は **get\_clocks** または **all\_clocks** ではありません。
2. パス制約 (**set\_false\_path** / **set\_max\_delay** / **set\_min\_delay** / **set\_multicycle\_path**) **-rise\_from** / **-fall\_from** / **-rise\_to** / **-rise\_fall** の後のアクセス方法は **get\_clocks** または **all\_clocks** ではありません。
3. **report\_timing** / **report\_exceptions** の後、**-from\_clock** と **-from [get\_clocks {}]** または **-to\_clock** と **-to [get\_clocks {}]** の両方があります。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]  
  
create_clock -name clk2 -period 10 -waveform {0 5} [get_ports {clk}] -add
```

```
set_clock_groups -exclusive -group [get_ports {clk}]

set_false_path -rise_from [get_ports {in1}] -fall_to [get_ports {out00}]

report_timing -setup -from_clock [get_clocks {clk1}] -to_clock [get_clocks {clk1}] -from [get_clocks {clk1}] -to [get_clocks {clk2}]
```

### 解決法

上記の 3 つの状況を避け、正しいアクセス方法を採用してください。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]
create_clock -name clk2 -period 10 -waveform {0 5} [get_ports {clk}] -add
set_clock_groups -exclusive -group [get_clocks {clk1 clk2}]

set_false_path -rise_from [get_clocks {clk1}] -fall_to [get_clocks {clk2}]

report_timing -setup -from_clock [get_clocks {clk1}] -to_clock [get_clocks {clk1}]
```

## TA2002

### ***ERROR(TA2002): Cannot get clock with name <name>***

使用するクロックが使用前に正しく作成されていない場合、このエラーが報告されます。

```
create_generated_clock -name genClk -source [get_ports {clk}]
-divide_by 2
```

### 解決法

クロックが使用前に正しく作成されていることを確認してください。

```
create_clock -name clk1 -period 10 -waveform {0 5} [get_ports {clk}]

create_generated_clock -name genClk -source [get_ports {clk}]
-master_clock clk1 -divide_by 2
```

