

U_Sheet1
Sheet1.SchDoc



M4



NXP_LOGO

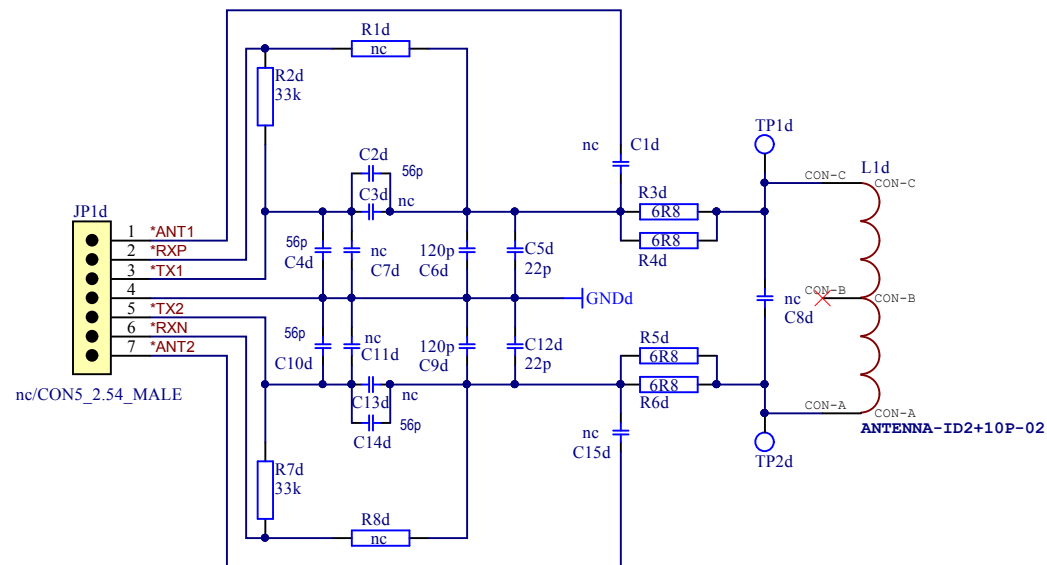
Antenna Kit Rev.B

Antenna
NXP Semiconductors Hamburg



LID 2272

Version 1
Revision B
Designer R.R.
Sheet A4 1 of 3
Date 04.08.2017
Time 12:04:29



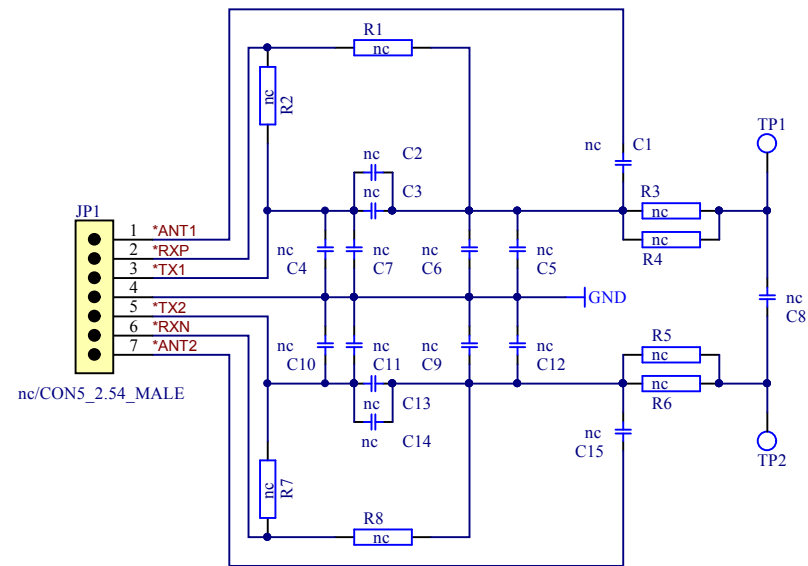
Antenna Kit Rev.B

Antenna
NXP Semiconductors Hamburg



LID 2272

Version 1
Revision B
Designer R.R.
Sheet A4 2 of 3
Date 04.08.2017
Time 12:04:29



Network 1-8

Antenna Kit Rev.B

Antenna
NXP Semiconductors Hamburg



LID 2272

Version 1
Revision B
Designer R.R.
Sheet A4 3 of 3
Date 04.08.2017
Time 12:04:29

```

R:\antennenkit\Protel_DXP_Daten\Protel_DXP_Daten\LayerRpt_philips.pas
[.....]
/ Summary Generate a PCB Layers report based on Layer Stack of PCB document
/ Copyright (c) 2003 by Altium Limited
[.....]
[.....]
Function ConvertDielectricTypeToString (DT : TDielectricType): String;
Begin
    Result := 'Unknown Type';

    Case DT Of
        eNoDielectric : Result := 'No Dielectric';
        eCore          : Result := 'Core';
        ePrePreg       : Result := 'PrePreg';
        eSurfaceMaterial : Result := 'Surface Mat.';
    End;
End;
[.....]
Function GetLayerInfo(BoardHandle : IPCB_Board; Var LayerID : TLayer): String;
Var
    LayerObj : IPCB_LayerObject;
    GerberExt : String;
    MaterialAndThickness : String;
    DielectricConst : String;
    Layertype : String;
    const MilToMM = 0.00000254;
Begin
    LayerObj := BoardHandle.LayerStack.LayerObject[LayerID];

    If ((LayerObj.LayerID > 1) and (LayerObj.LayerID < 32) Then
        GerberExt := '.G' + FloatToStr(LayerObj.LayerID-1);
    If ((LayerObj.LayerID > 38) and (LayerObj.LayerID < 55) Then
        GerberExt := '.GP' + FloatToStr(LayerObj.LayerID-38);
    If (LayerObj.LayerID = 37) Then
        Begin
            GerberExt := '.GTS';
            MaterialAndThickness := LayerObj.LayerStack.DielectricTop.DielectricMaterial + ', '
            + Format('%5.3f', (LayerObj.LayerStack.DielectricTop.DielectricHeight * MilToMM * 1.0));
            DielectricConst := ', Er = ' + FloatToStr(LayerObj.LayerStack.DielectricTop.DielectricConstant);
            Layertype := 'Solder Layer: ';
        End;
    If (LayerObj.LayerID = 38) Then
        Begin
            GerberExt := '.GBS';
            MaterialAndThickness := LayerObj.LayerStack.DielectricBottom.DielectricMaterial + ', '
            + Format('%5.3f', (LayerObj.LayerStack.DielectricBottom.DielectricHeight * MilToMM * 1.0));
            DielectricConst := ', Er = ' + FloatToStr(LayerObj.LayerStack.DielectricBottom.DielectricConstant);
            Layertype := 'Solder Layer: ';
        End;
    If (LayerObj.LayerID = 1) Then
        GerberExt := '.GTL';
    If (LayerObj.LayerID = 32) Then

```

```

C:\Rf\antennenkit\Protel_DXP_Daten\Protel_DXP_Daten\LayerRpt_philips.pas
GerberExt := '.GBL';
If ((LayerObj.LayerID < 37) or ((LayerObj.LayerID > 38) Then
Begin
MaterialAndThickness := 'Copper' + ', ' + Format('%5.3f',[LayerObj.CopperThickness * MilToMM* 1.0]);
LayerType := 'Copper Layer: ';
End;

Result := LayerType
      + Layer2String(LayerID)
      + ', ' +
      LayerObj.Name
      + ', ' +
      GerberExt
      + ', ' +
      MaterialAndThickness
      + 'mm' +
      DielectricConst
      + #13#10 + #13#10;

//Check if the current layer contains dielectric information
If LayerObj.Dielectric.DielectricType <> eNoDielectric Then
Begin
Result := Result + 'Core: '
      + ConvertDielectricTypeToString(LayerObj.Dielectric.DielectricType) + ', ' +
      LayerObj.Dielectric.DielectricMaterial
      + ', ' +
      Format('%5.3f',[LayerObj.Dielectric.DielectricHeight * MilToMM* 1.0]) + 'mm' + ', ' +
      'Er = ' + FloatToStr(LayerObj.Dielectric.DielectricConstant)
      + #13#10 + #13#10;
End;

LayerObj := BoardHandle.LayerStack.NextLayer(LayerObj);

If LayerObj <> Nil Then
LayerID := LayerObj.LayerID
Else
LayerID := eNoLayer;
End;
{.....}
{.....}
Var
BoardHandle : IPCB_Board;
Str : String;
Filename : String;
OutFile : TextFile;
Layer : TLayer;
ReportDocument : IServerDocument;
TopSolder : Float;
BotSolder : Float;
Begin

// This main routine obtains details for each existing PCB layer that is defined by the layer stack.
BoardHandle := PCBServer.GetCurrentPCBBoard;
If BoardHandle = Nil Then
Begin
ShowError('The current document is not PCB document');
Exit;
End;

Str := 'PCB Layers with dielectric information report ' + #13#10 + (BoardHandle.FileName) + #13#10 + #13#10 + #13#10;

```

```

C:\Raf\antennenkit\Protel_DXP_Daten\Protel_DXP_Daten\LayerRpt_philips.pas
Str := Str + 'Solder Layer: [Layer], [Layer Name], [Extension], [Material], [Thickness], [Constant]' + #13#10 + #13#10
+ 'Copper Layer: [Layer], [Layer Name], [Extension], [Material], [Thickness]' + #13#10 + #13#10
+ 'Core      : [Type], [Material], [Height], [Constant] ' + #13#10 + #13#10 + #13#10 + #13#10;

TopSolder := 37;
Layer := TopSolder;
Str := Str + GetLayerInfo(BoardHandle, Layer);

Layer := MinLayer;
Repeat
  Str := Str + GetLayerInfo(BoardHandle, Layer);
Until Layer = eNoLayer;

BotSolder := 38;
Layer := BotSolder;
Str := Str + GetLayerInfo(BoardHandle, Layer);

// Write dielectric data for the layer stack of the current PCB out to a text file.
FileName := ChangeFileExt(BoardHandle.FileName, '') + ' LayerRpt.txt';
Try
  AssignFile(OutFile, FileName);
  Rewrite(OutFile);
  Write(OutFile, Str);
Finally
  CloseFile(OutFile);
End;

// Display the dielectric data in a dialog.
ShowMessage(Str);
End.
{.....}
{.....}

```

Ritzkante Bottom
Board Outline

