

## パラメトリック修正とスプレッドシート

# コース概要

このコースでは、表紙のようなローマンロックボルトを修復します。はじめに、既存の寸法を変更し穴の位置を変更します。続いて形状を少し修正します。最後にスプレッドシートのさまざまな使用方法を見ていきます。

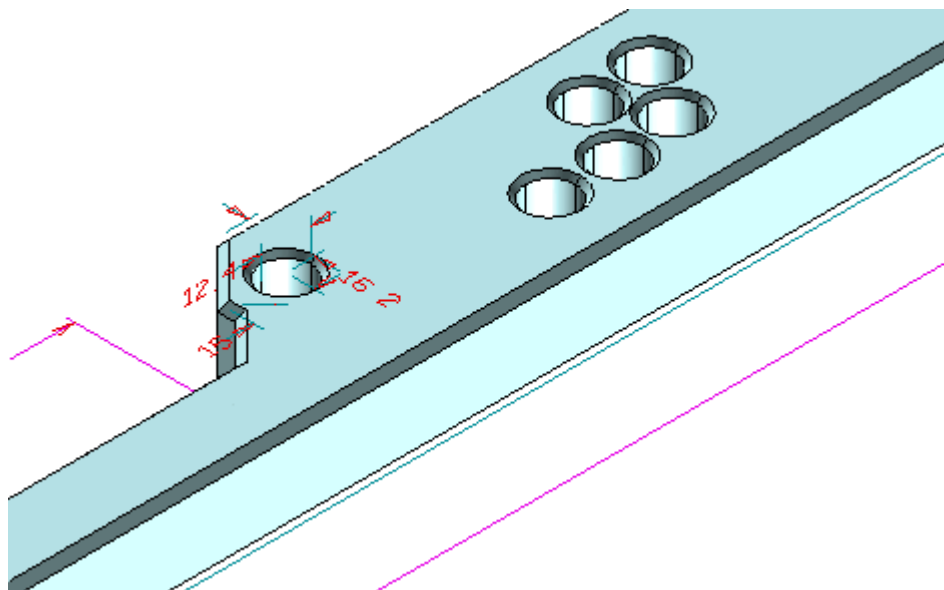
使用するファイル      RomanLock.e3

## 目次

Step 1: 寸法の表示と変更 .....	3
Step 2: パラメトリック修正 .....	6
Step 3: 形状の修正 .....	10
Step 4: スプレッドシートと変数 .....	14
Step 5: スプレッドシートによるフィーチャーの制御 .....	21

## Step 1: 寸法の表示と変更

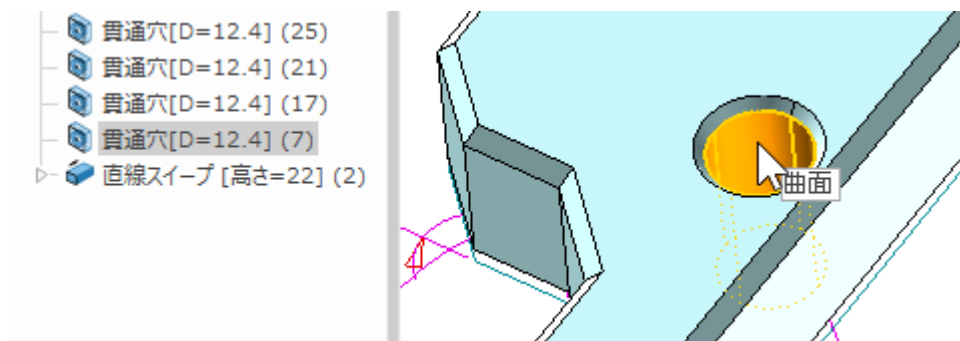
最初のステップでは、駆動寸法を変更して穴の位置を変更します。しかし現在、穴には寸法が表示されてないようです。はじめに穴の駆動寸法を表示します。



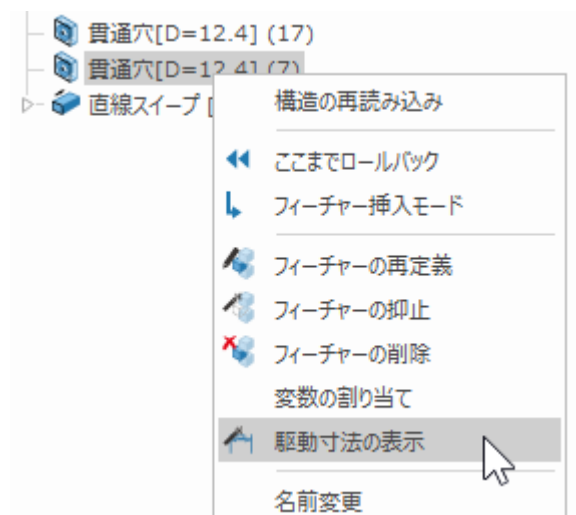
ダウンロードしたファイルから、**RomanLock.e3**を開きます。

穴の辺りを拡大します。

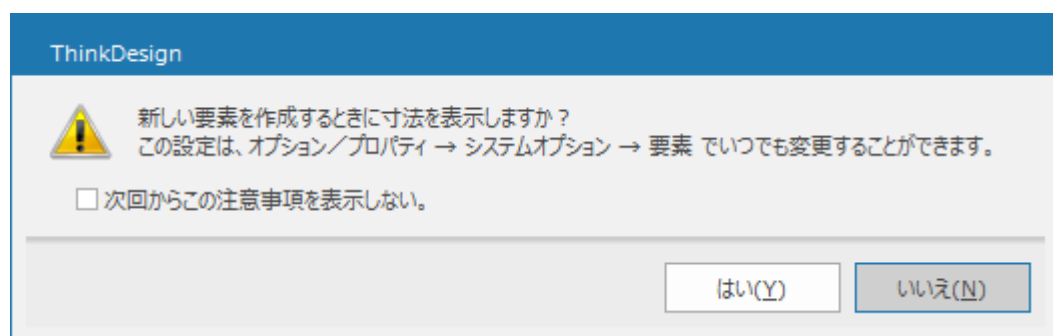
- **ウィンドウ拡大** コマンドで穴の辺りを拡大します。
- 下図に示すように穴の面をクリックします。



モデル構造ツリーで該当するフィーチャーがハイライトするので、右クリックして **駆動寸法の表示** を選択します。



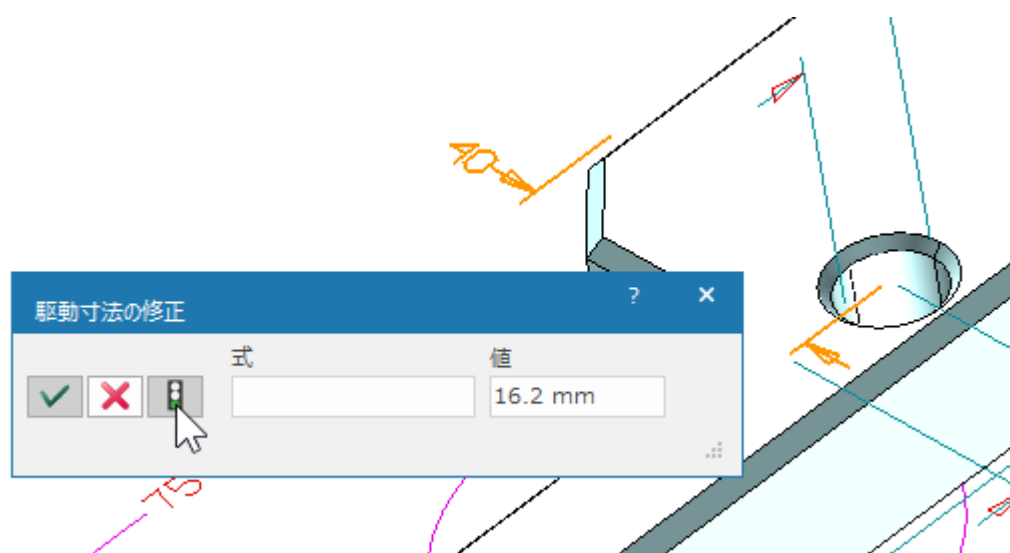
次のようなメッセージが表示されます。



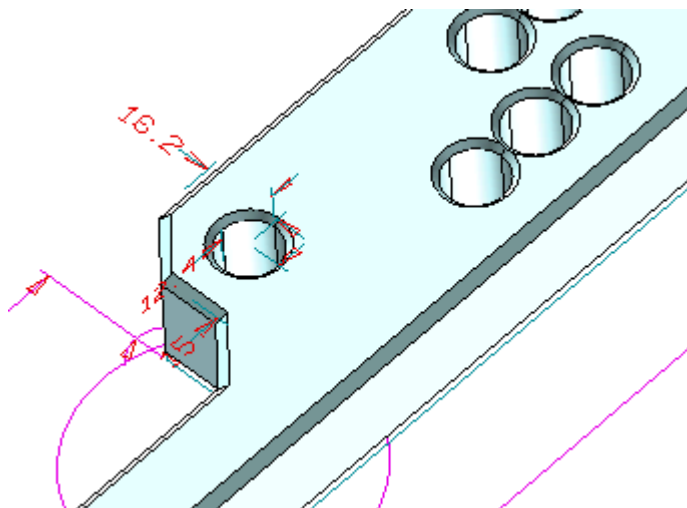
次回から常に寸法を表示する設定にするか？と聞いてきています。  
ここでは **いいえ** を選択します。

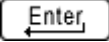
画面に寸法が表示されるので、変更します。

- 40 寸法をダブルクリックします。
- または、右クリックから **寸法編集** を選択します。





- 値を 16.2 に変更し、TAB キーを押します。
- **再構築** ボタンを押すと変更が反映されます。



OK ボタン、あるいは  キーを押しただけでは、値は変更されますが形状にその変更が反映されません。

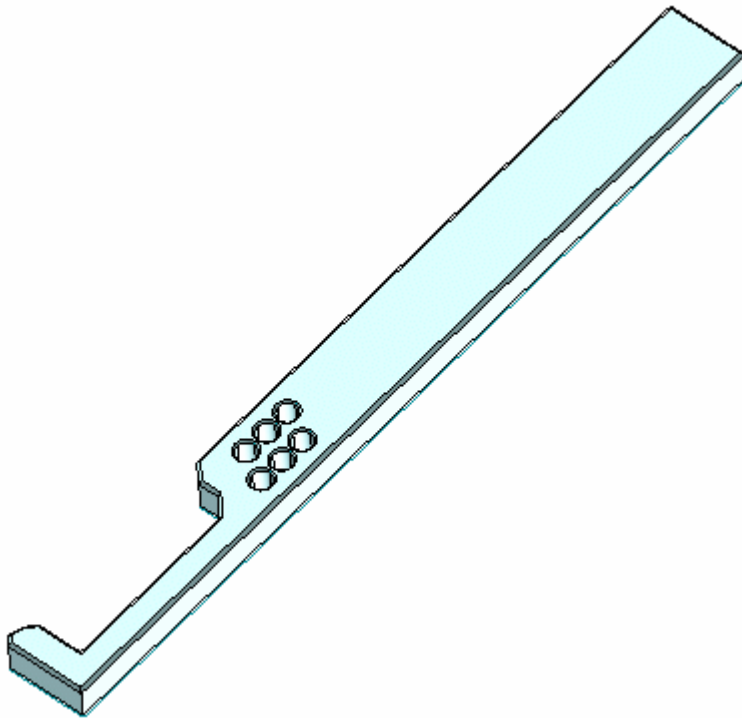
大きなモデルを扱う場合、数値を変更した際に逐一再構築が行われると、再構築だけでかなり多くの時間がかかってしまうことがあります。このような場合に、ThinkDesign では、複数の寸法を値だけ変更しておき、最後に1度だけ再構築を行って変更を形状に反映させることができます。

間違って OK ボタン、あるいは  キーを押した場合は、 **再構築** を選択すると、すぐに変更をモデルに反映させることができます。


それでは Step 2 に進みます。

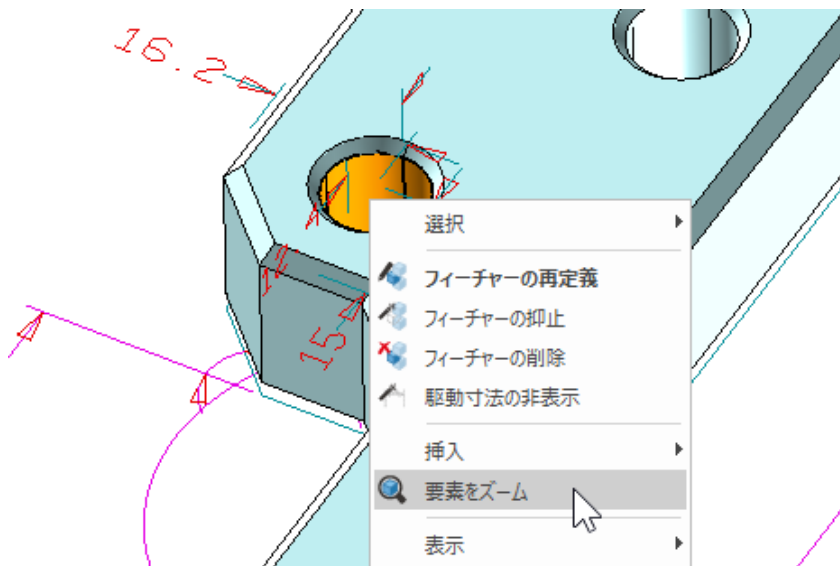
## Step 2: パラメトリック修正

このステップでは、ボルトがキーに合うよう、さらに形状を変更します。



穴の周りを少し拡大します。

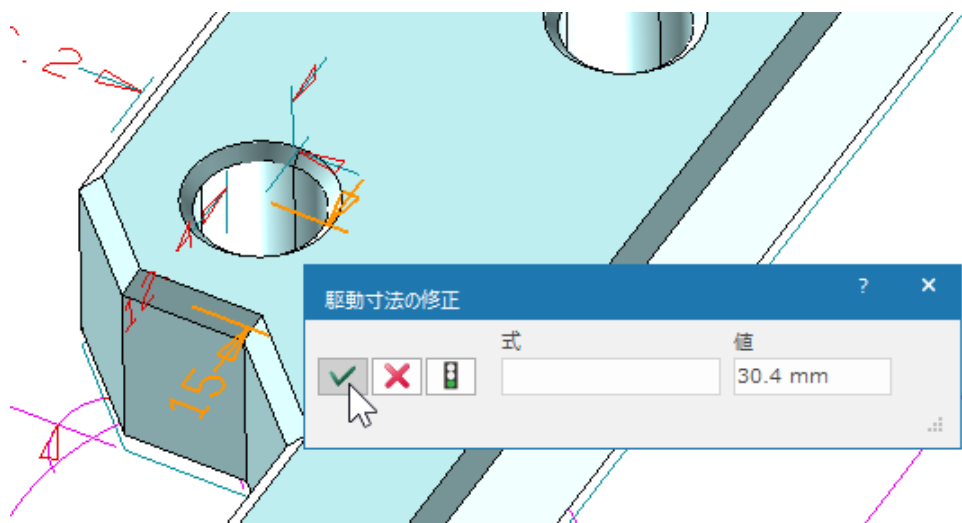
- 穴の面を右クリックして、 **要素をズーム** を選択します。
- 穴の面が画面いっぱいに拡大され、穴の面の中心付近が回転の中心になります。



- **Shift** キー+マウスの右ボタンのドラッグで、少し表示を縮小します。

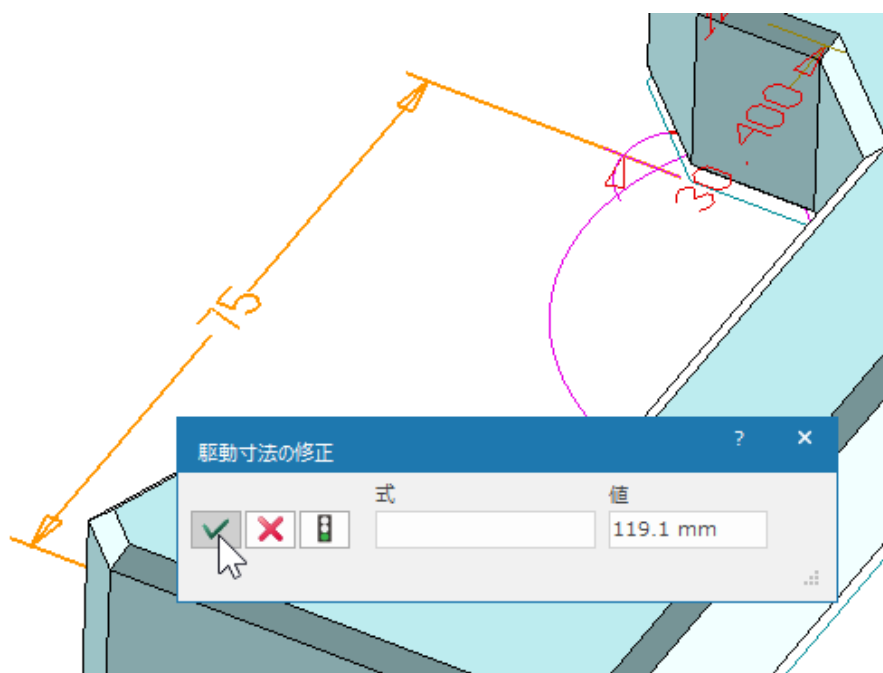
続いて寸法をいくつか変更します。再構築は最後に1度だけ行います。

- 15 寸法をダブルクリックします
- 値を 30.4 に変更します。
- ☒ OK ボタンを押します。

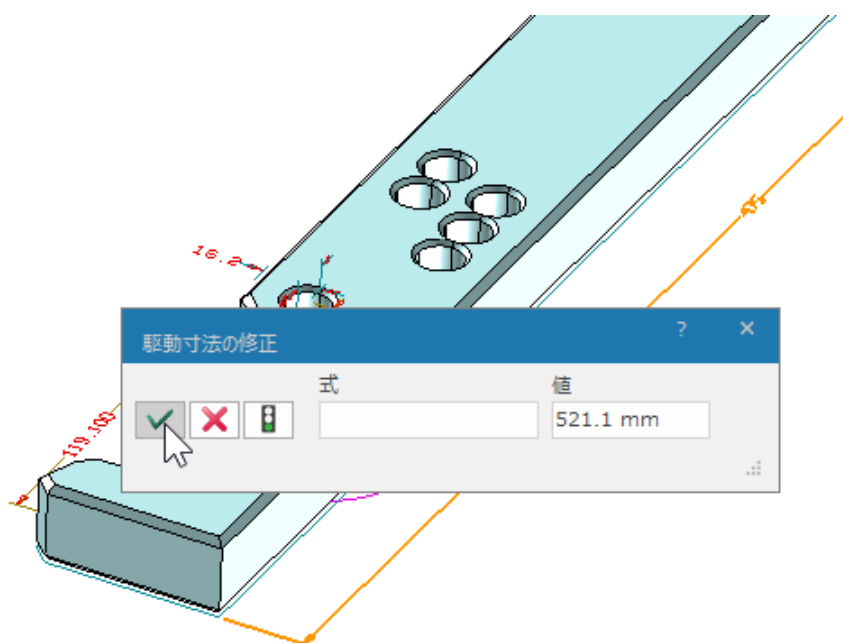


数値は 30.400 のように最後に 0 が付いた形になります。また、ツールバーの再構築アイコンが点灯します。

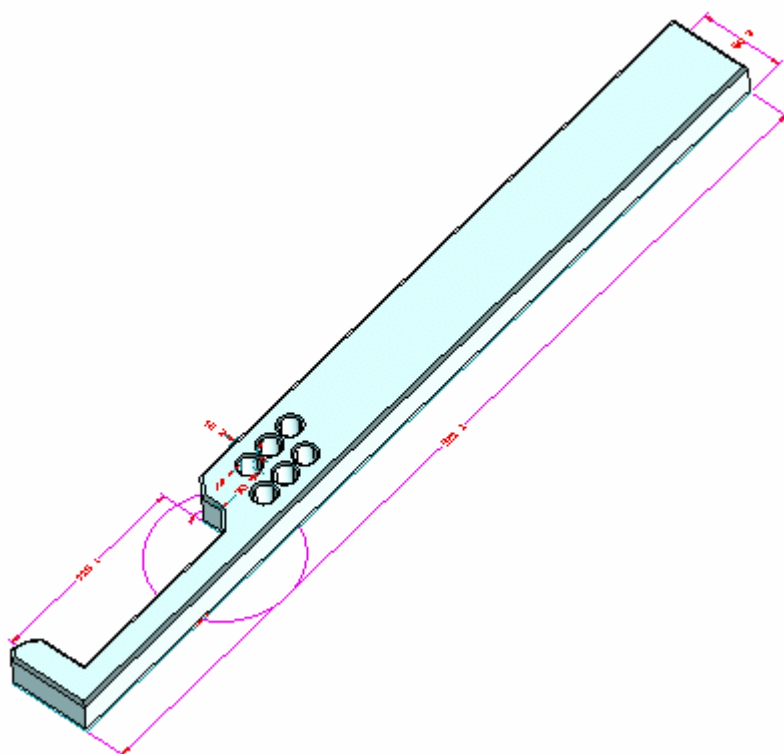
次に、75 寸法を 119.1 に変更します。ここでも、☒ OK ボタンを押します。



- キーボードの **F** キーを押します。すべての要素が画面内に表示されます。
- 475 寸法をダブルクリックして、521.1 に変更します。
- ここでも、☒ OK ボタンを押します。




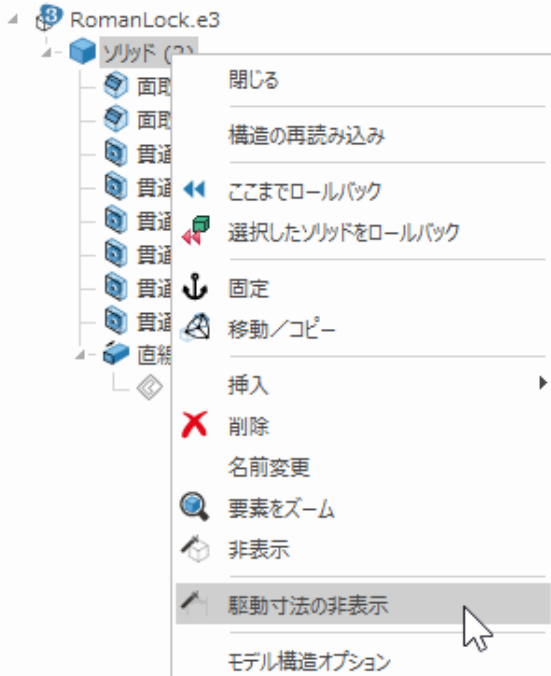
最後にモデルを  **再構築** します。



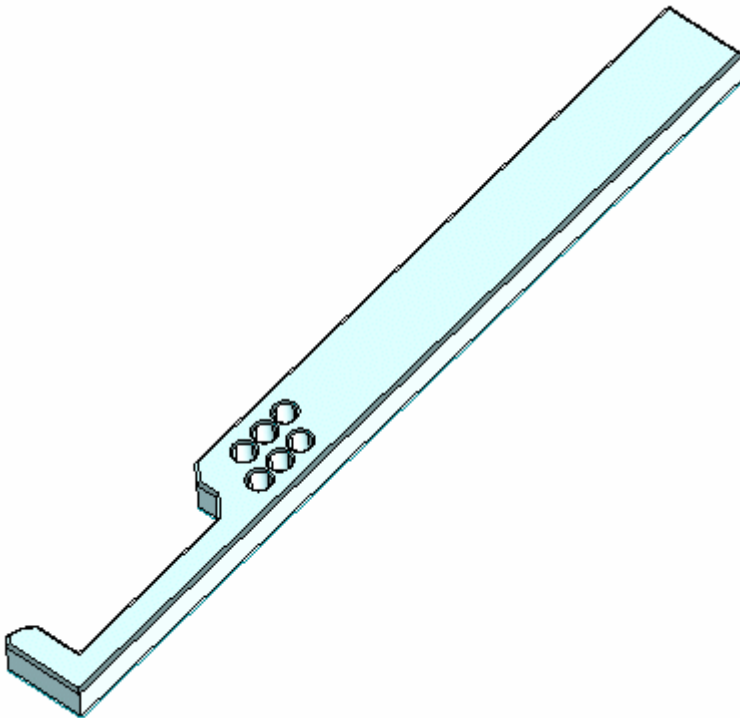


今度はモデルの寸法を非表示にします。モデル全体の寸法を非表示にするには、モデル構造ツリーで、ソリッドを選択して駆動寸法の非表示を選択します。ソリッドを選択した場合はソリッドごと、フィーチャーを選択した場合は、フィーチャーごとに表示／非表示を制御することができます。

- ソリッドを右クリックします。
- いったん  **駆動寸法の表示** を選択します。すべての非表示になっている寸法が表示されます。
- 同じ操作をもう一度行います。コンテキストメニューには今度は **駆動寸法の非表示** と表示されており、これですべての駆動寸法が非表示になります。



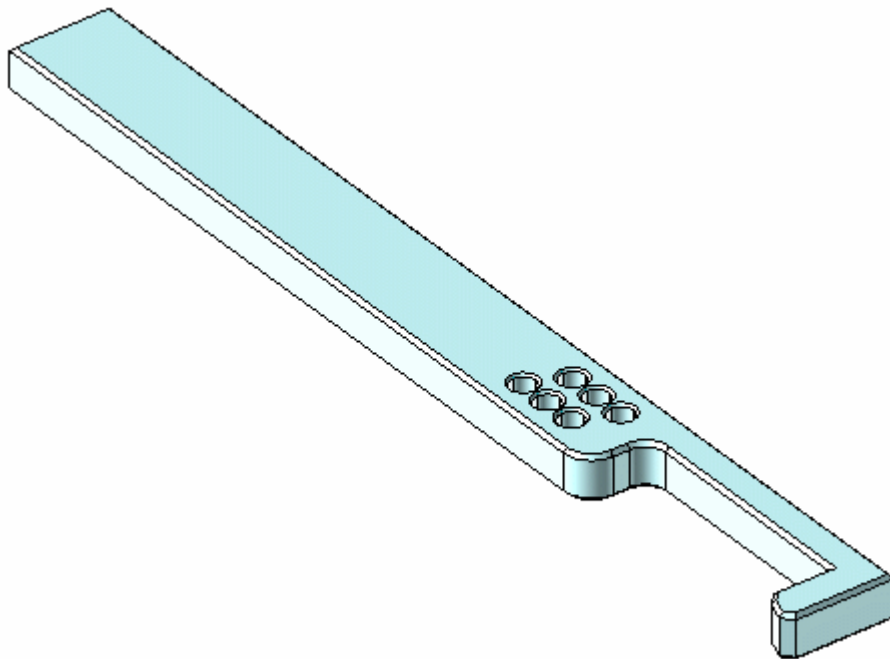
すべての寸法が非表示になりました。



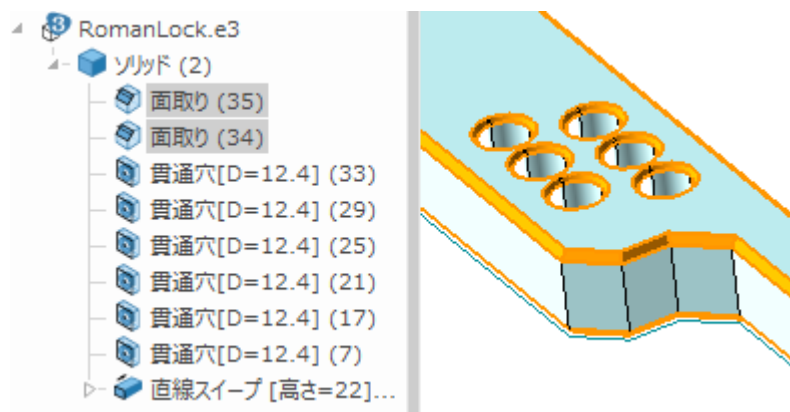
次のステップでは、形状を部分的に修正します。

### Step 3: 形状の修正

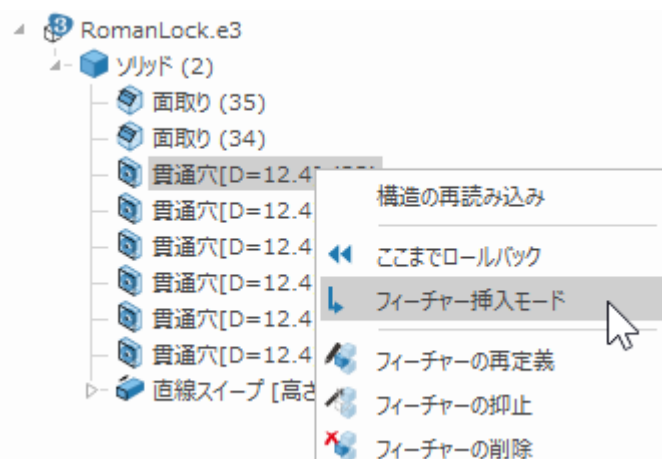
形状の角部が面取りされていますが、ここをフィレットに変更します。



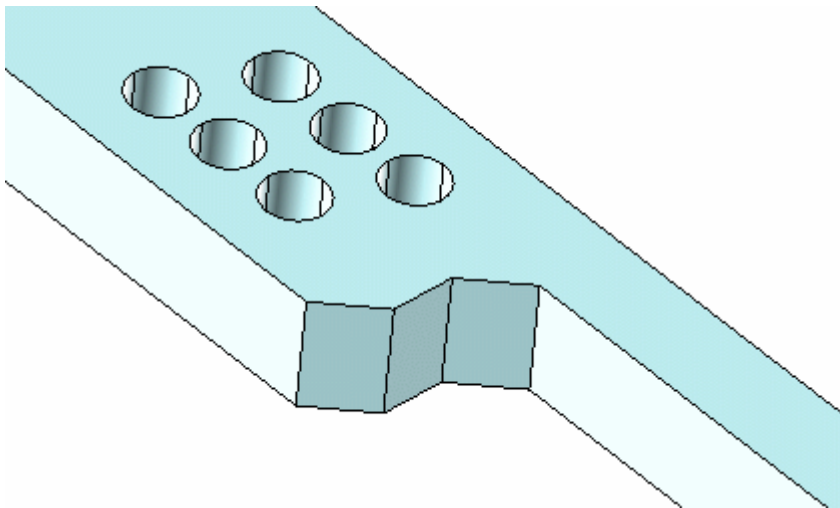
モデルがどのように作成されているかわからないので、最後に追加されている面取りの前に戻って、直接モデルを変更します。



- モデル構造ツリーで面取りの下の穴フィーチャーを右クリックして、**フィーチャー挿入モード**を選択します。

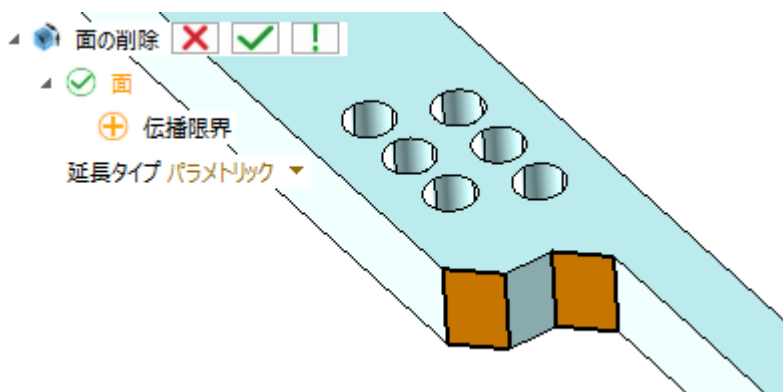


形状が面取りを追加する前に戻りました。

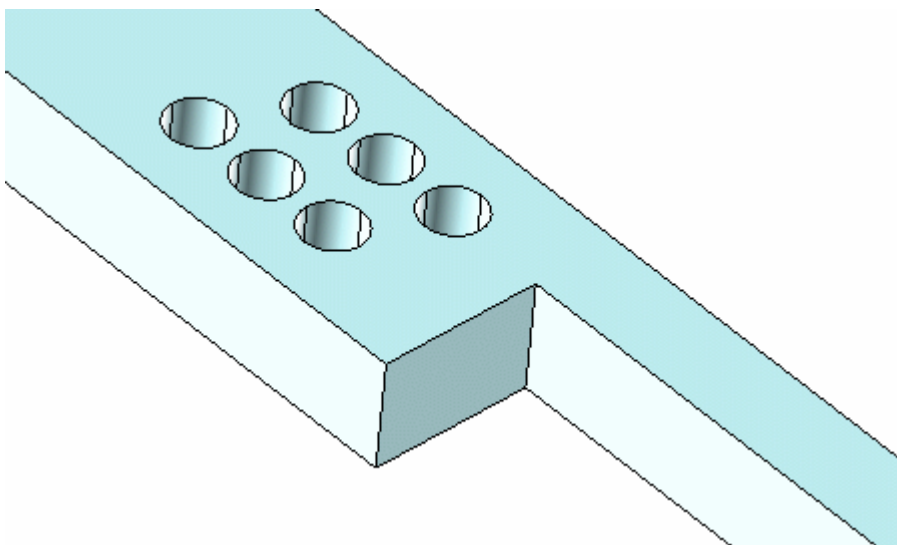



形状の角部の2面を削除します。


- 修正 インタラクティブモデリング 面の削除 を選択します。
- 下図に示した2面を選択します。

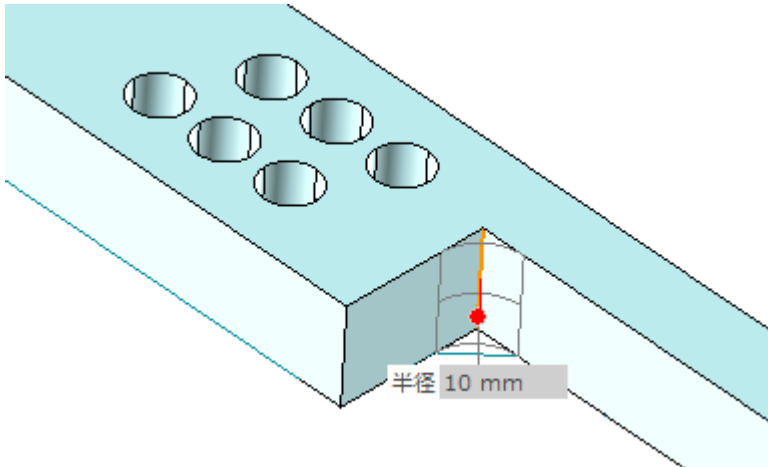




- OK を選択してコマンドを終了します。

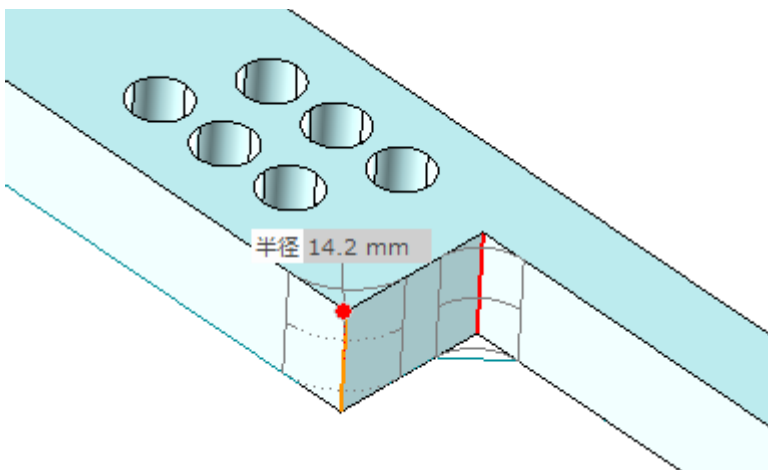



 エッジフィレット コマンドであらためて角部にフィレットを追加します。

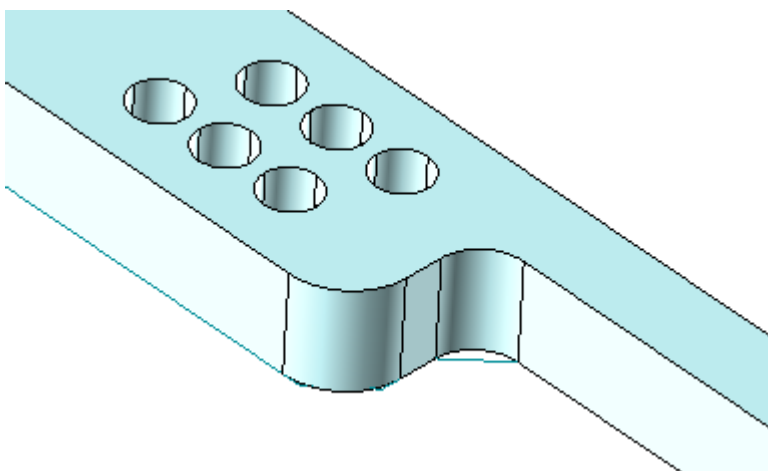
-  エッジフィレット コマンドを選択します。
- 右のエッジを選択し、半径 10 mm と指定します。







-  グループ2 の  新規グループ で左のエッジを選択します。
- 半径は、半径 14.2 mm と指定します。

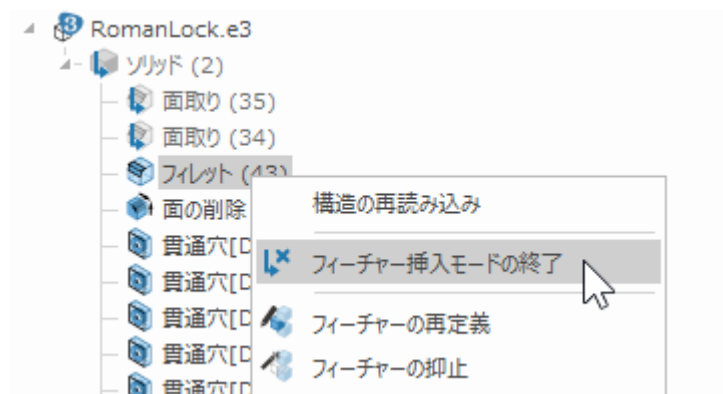


-  OK を選択してコマンドを終了します。

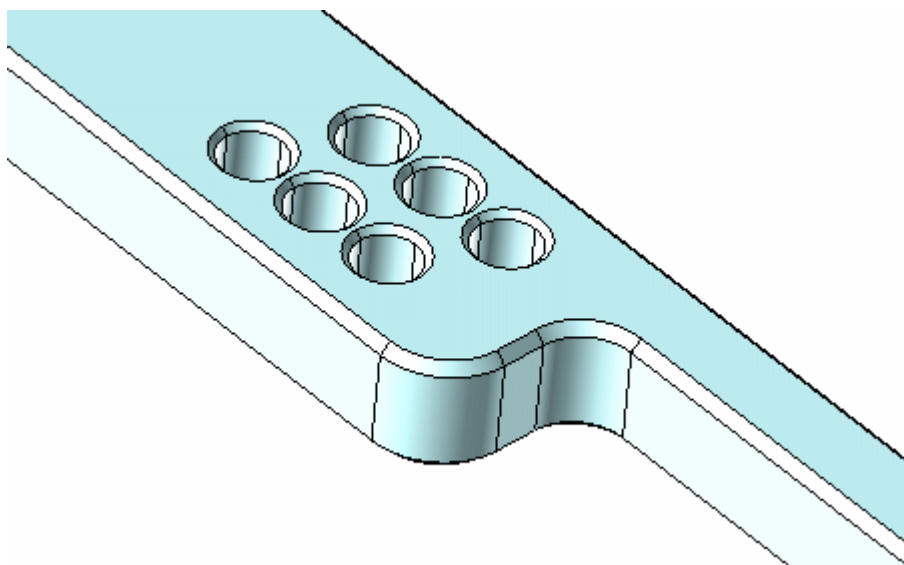


モデル構造ツリーには、今追加した  **面の削除** と  **エッジフィレット** が面取りの下に追加されています。遡った履歴を元に戻し、面取りを復活させます。

- モデル構造ツリーで、 **エッジフィレット** を右クリックします。
-  **フィーチャー挿入モードの終了** を選択します。これでフィーチャー挿入モードを抜けます。



- モデルを  **再構築** します。



遡った履歴が元に戻りました。面取りだった部分をフィレットに変更しました。




次のステップでは、スプレッドシートと変数の使い方を見ていきます。

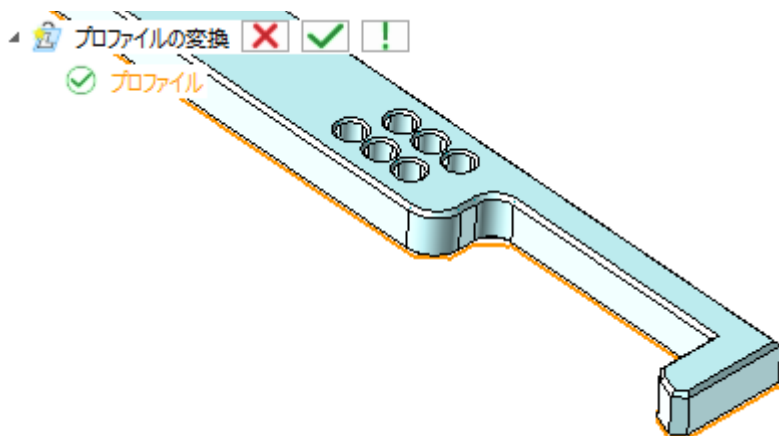
## Step 4: スプレッドシートと変数

次に、いくつかの寸法に変数を割り当てます。変数を割り当てられた寸法は、スプレッドシートからコントロールすることができますようになります。

このモデルは少し古いバージョンで作成されているようです。ThinkDesign はバージョン 2006.2 からスケッチャーを一新し、新しい環境でプロファイルを作成するようになっています。このモデルのプロファイルの寸法に変数を設定するには、古いスケッチャーで作成されているプロファイルを新しいプロファイルに変換しておく必要があります。

プロファイルの変換は  **プロファイルの変換** コマンドを使用します。

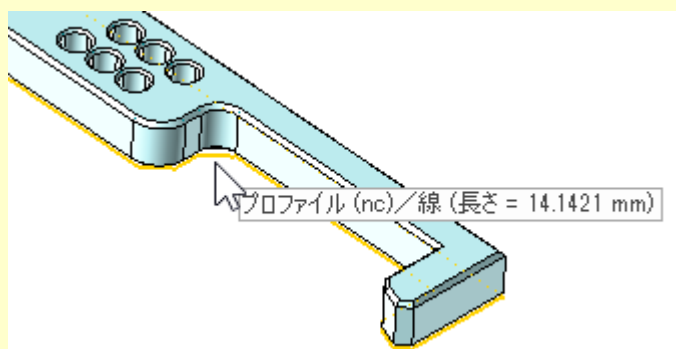
- **修正**  **プロファイル**  **変換** コマンドを選択します。
- プロファイルを選択します。
-  **OK** ボタンを押してコマンドを終了します。



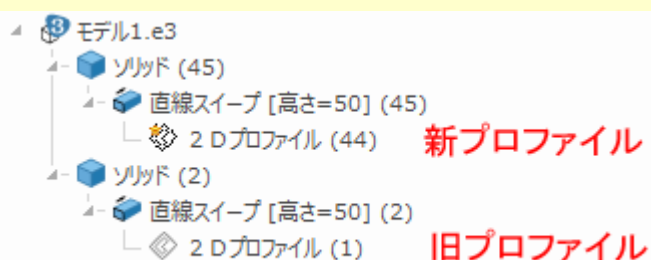
- モデルを  **再構築** します。

### 新旧のプロファイル:

古いスケッチャーで作成されているプロファイルは、ツールチップに プロファイル(nc) と表示されます。

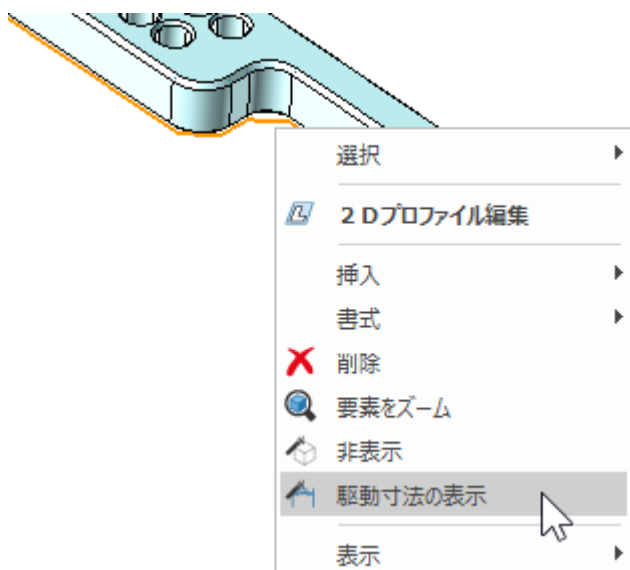


またモデル構造ツリーでは、異なったアイコンで表示されます。




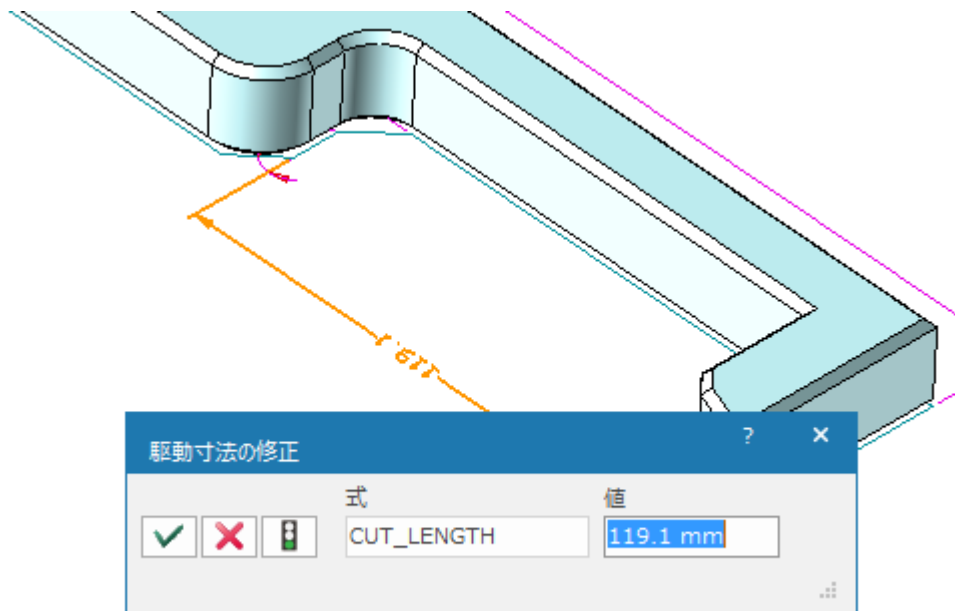
ステップ2で、モデルの寸法を非表示にしました。今度はプロファイルの寸法のみを表示します。

- プロファイルを右クリックして、 **駆動寸法の表示** を選択します。

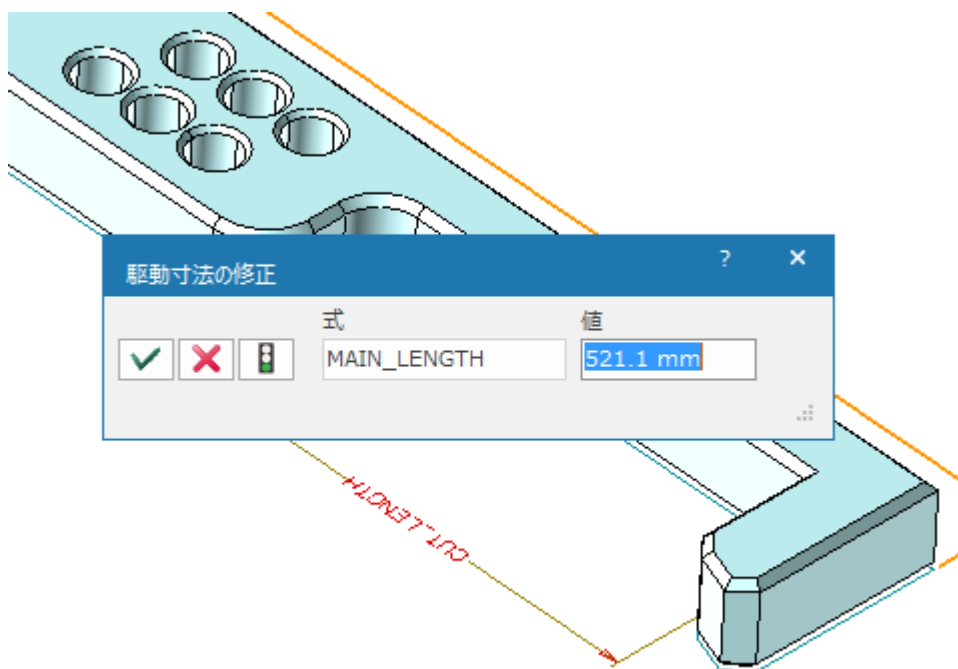



表示された寸法に変数を設定します。

- はじめに 119.1 寸法をダブルクリックします。
- 式 欄に **CUT\_LENGTH** と入力します。変数は、大文字と小文字を区別するので、間違えないようにすべて大文字で入力してください。
-  **再構築** ボタンを押します。




- 次に、521.1 寸法をダブルクリックします。
- 式 欄に **MAIN\_LENGTH** と入力します。
-  **再構築** ボタンを押します。



 **スプレッドシート** を開きます。2つの変数が設定されています。





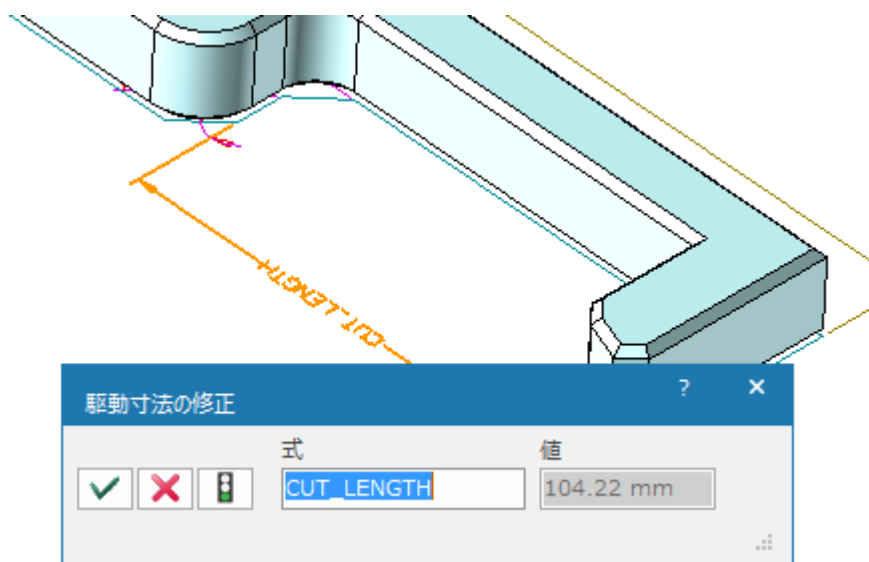
- 変数 **CUT\_LENGTH** の式欄を編集します。
- MAIN\_LENGTH/5** と入力し、 **再構築** ボタンを押します。



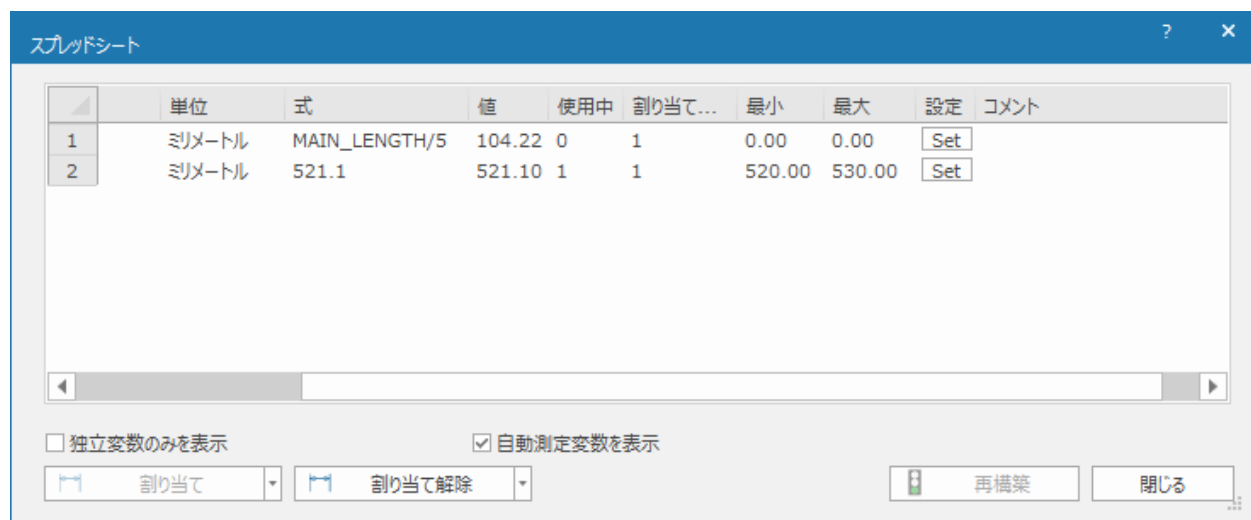
グラフィック領域で寸法 **CUT\_LENGTH** をダブルクリックします。

表示されるダイアログの値欄がグレーアウトしていることがわかります。

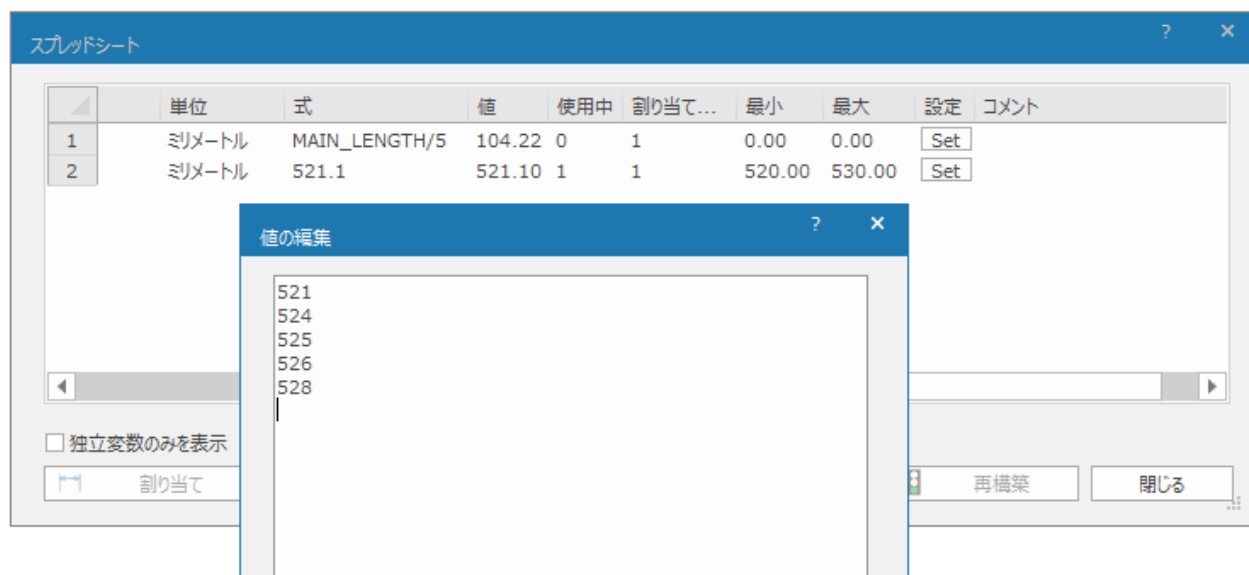
このことは、この寸法が他の変数にリンクしており、単独で変更することができないことを示しています。

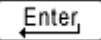


スプレッドシートでは、値の上限と下限を指定することもできます。



また、Set ボタンを押すと、いくつかの値を入力することができます。



各値は、 キーを押して区切ります。

OK ボタンを押すと、変数 **MAIN\_LENGTH** の値が 521 に変わります。  
値のリストを設定すると、強制的にリストの中で最も近い値に変更されます。

例えば **MAIN\_LENGTH** に 523 を入力してみます。  
この時は、最も近い値 524 に強制的に変更されます。

モデルを  **再構築** します。

続いて、空白領域のコンテキストメニューから **変数の作成** を選択します。新しい変数 **HOLES** を作成します。




**単位** 欄をクリックします。ここでは、**スカラー** を指定します。

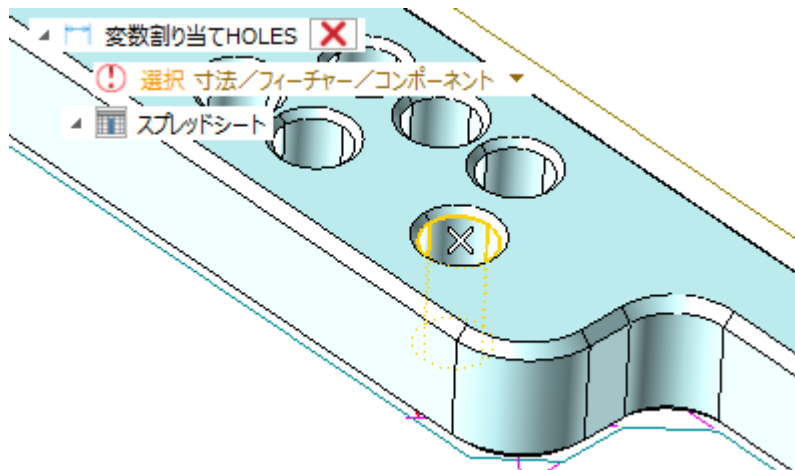



左端の 3 をクリックして、変数全体を選択します。続いて、左下の **割り当て** ボタンをクリックします。

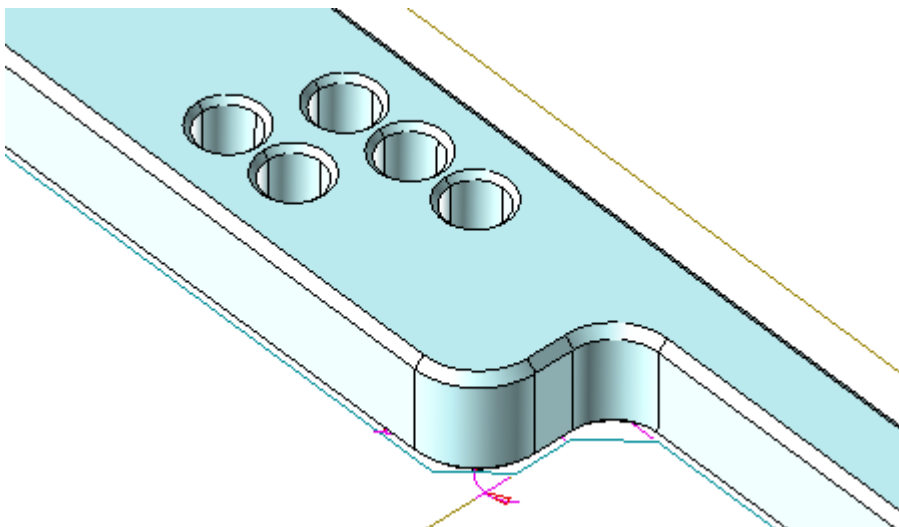


変数は寸法の外、フィーチャーにも割り当てることができます。

- 穴フィーチャーを選択します。
- コマンドを  キャンセル します。



モデルを  **再構築** すると穴フィーチャーが無くなります。(抑止されます。)




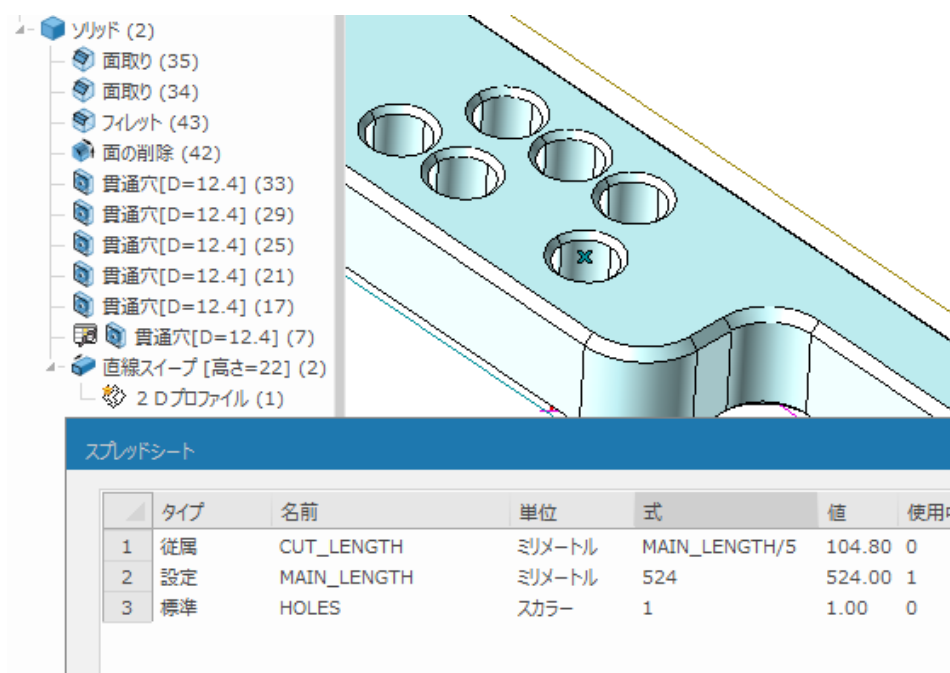
フィーチャーに割り当てた変数が 0 の時、そのフィーチャーは抑止されます。

次のステップで、さらに変数によるフィーチャーの制御を見ていきます。

## Step 5: スプレッドシートによるフィーチャーの制御

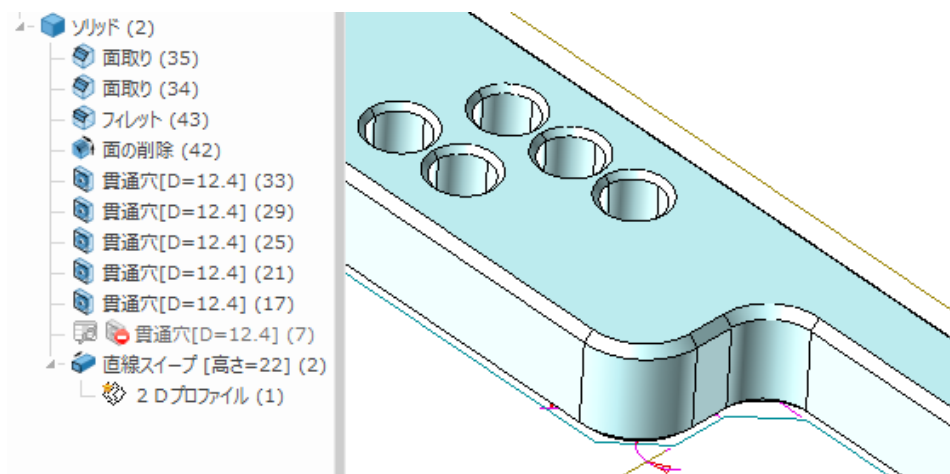
再び  **スプレッドシート** を開きます。

さきほど 0 に設定した変数 **HOLES** を 1 に変更して、 **再構築** を行います。フィーチャーの抑止が解除され、穴がまた開きます。

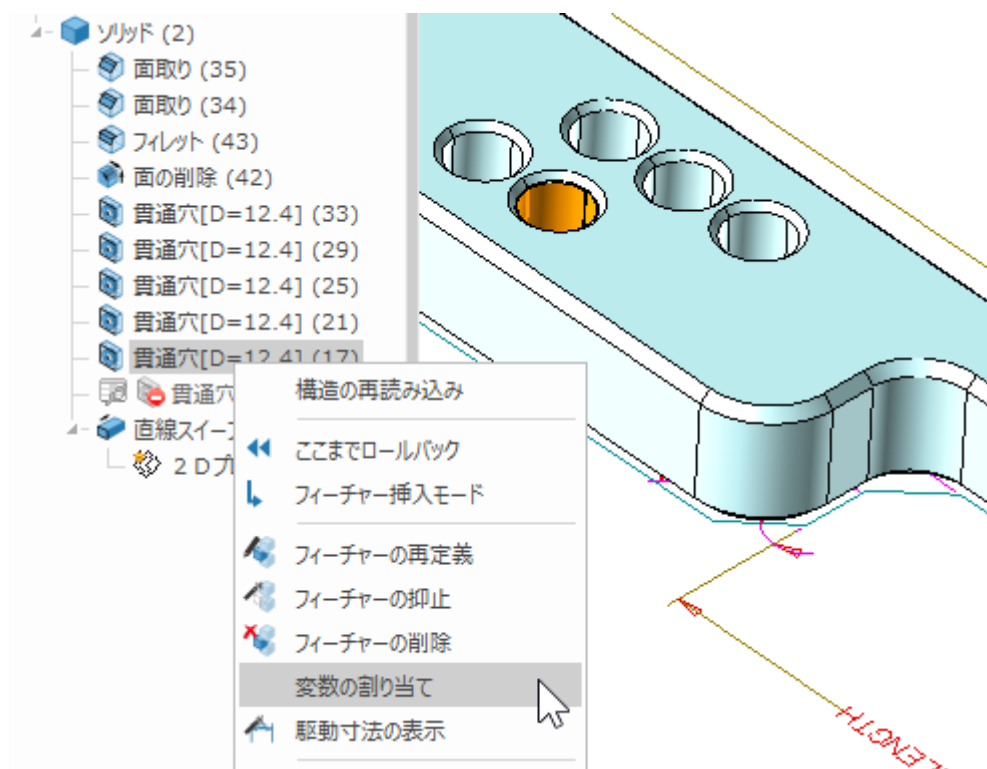


モデル構造ツリーに注目すると、フィーチャーのアイコンの横にスプレッドシートのアイコンが表示されていることがわかります。これは、このフィーチャーは変数に関連付けられていることを示しています。

また、値を 0 に戻して再構築します。穴は再び抑止されます。



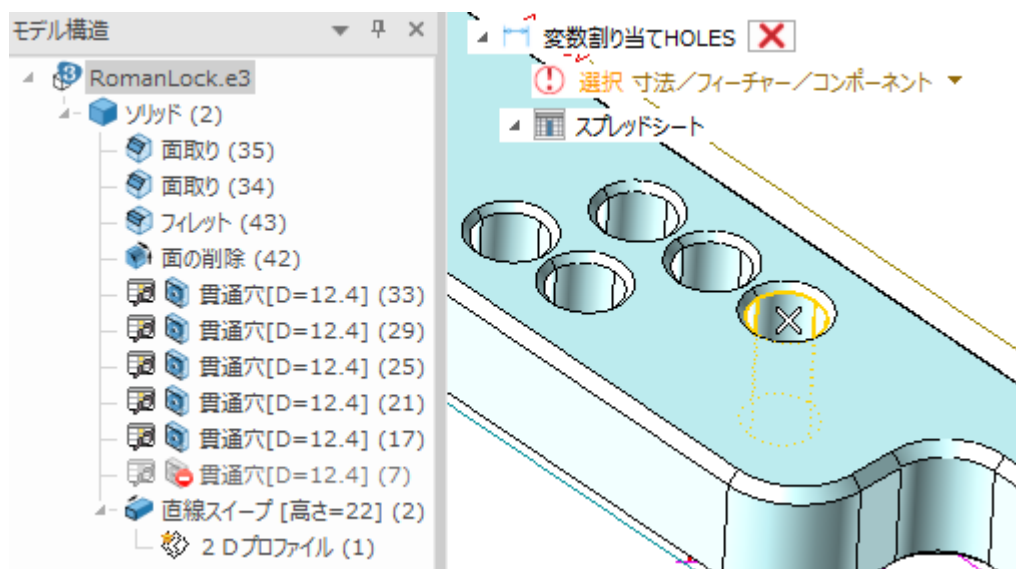
続いて、モデル構造ツリーから穴フィーチャーを右クリックして、**変数の割り当て**を選択します。



続いてスプレッドシート上で、左端の数字をクリックして、変数 **HOLES** を選択します。

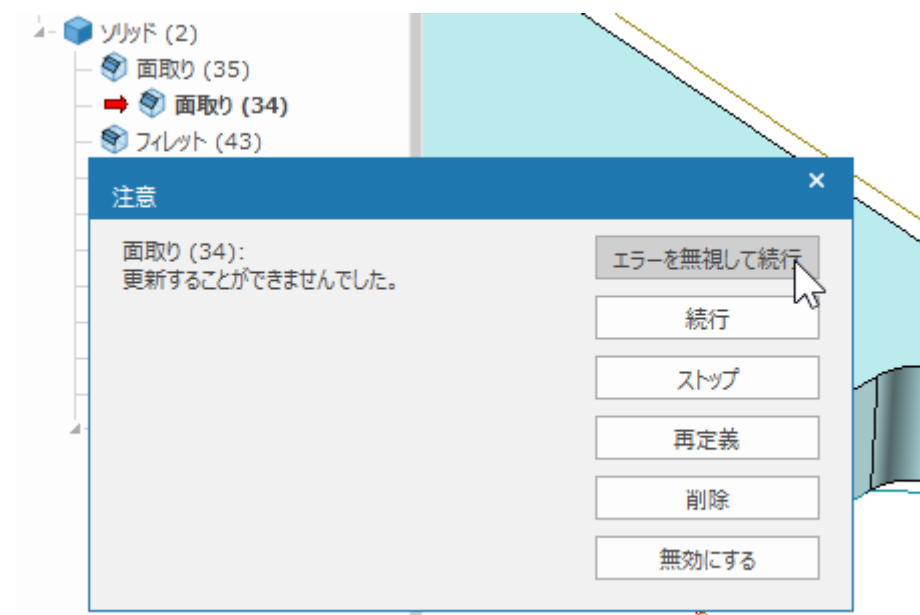
スプレッドシート左下の **割り当て** ボタンを押すとフィーチャーに変数を割り当てることができます。

**変数割り当て** コマンドが起動するので、続けて他の穴を選択します。すべての穴に変数 **HOLES** を割り当てます。



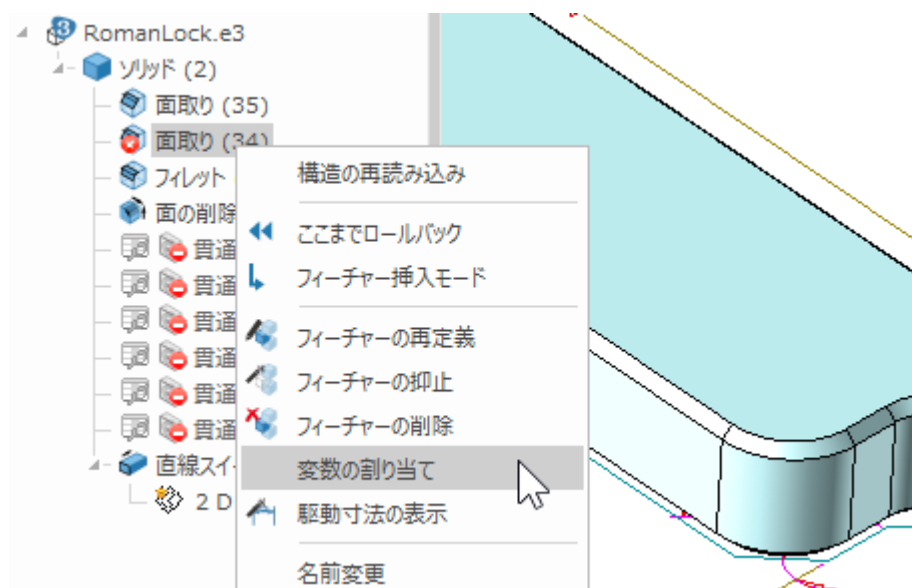
 **再構築** します。

するとエラーメッセージが表示されます。



穴の縁に追加した面取りが、穴が無くなったためにエラーになっています。ここではとりあえず、**エラーを無視して続行** を選択します。

この面取りフィーチャーも穴と同時に抑止するのがよいようです。モデル構造ツリーの面取りアイコンを右クリックして **変数の割り当て** を選択し、このフィーチャーにも変数 **HOLES** を割り当てます。



これで穴と面取りの両方を変数 **HOLES** で制御できるようになりました。

もう少し手を加えます。変数 **HOLES** の式欄に次のように記入します。

- `iff(MAIN_LENGTH.gt.525,1,0)`

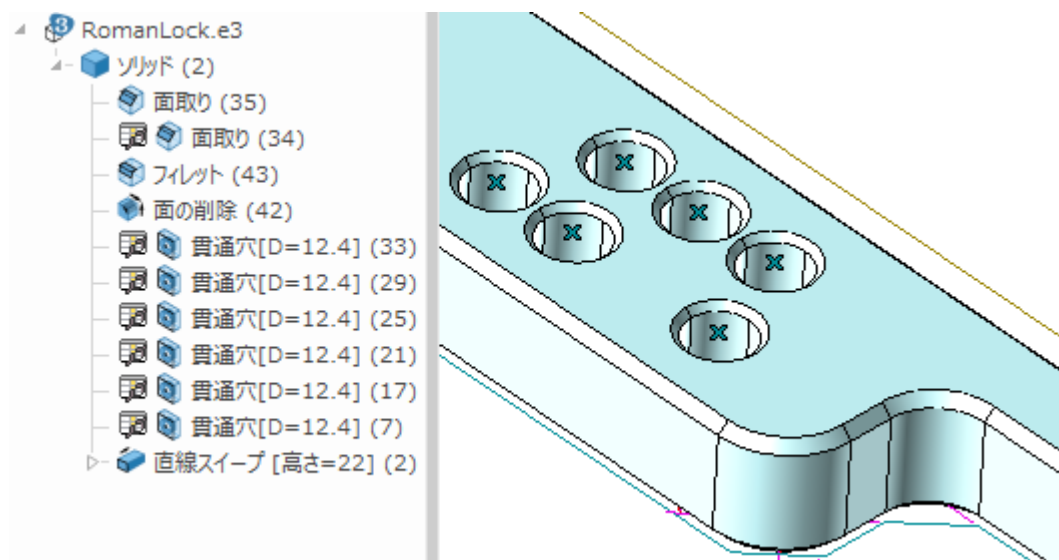
スプレッドシート						
	タイプ	名前	単位	式	値	使用中
1	従属	CUT_LENGTH	ミリメートル	MAIN_LENGTH/5	104.80	0
2	設定	MAIN_LENGTH	ミリメートル	524	524.00	2
3	従属	HOLES	スカラー	iff(MAIN_LENGTH.gt.525,1,0)	0.00	0

この記述は、変数 **MAIN\_LENGTH** の値が 525 より大きい場合 (greater than)、変数 **HOLES** は 1 になり、そうでない場合は 0 になる、ということを意味しています。

.gt. の他に、以下の演算子が利用できます。

- .ge. / .lt. / .le. / .eq. / .ne.

これで変数 **MAIN\_LENGTH** の値によって、穴フィーチャーのあるなしを制御できるようになりました。いくつか値を変えて効果を確認してください。



うまく動作していますか？ これでこのコースは終了です。