EXERCISE 2.2 – CREATING A CLOCK ASSEMBLY

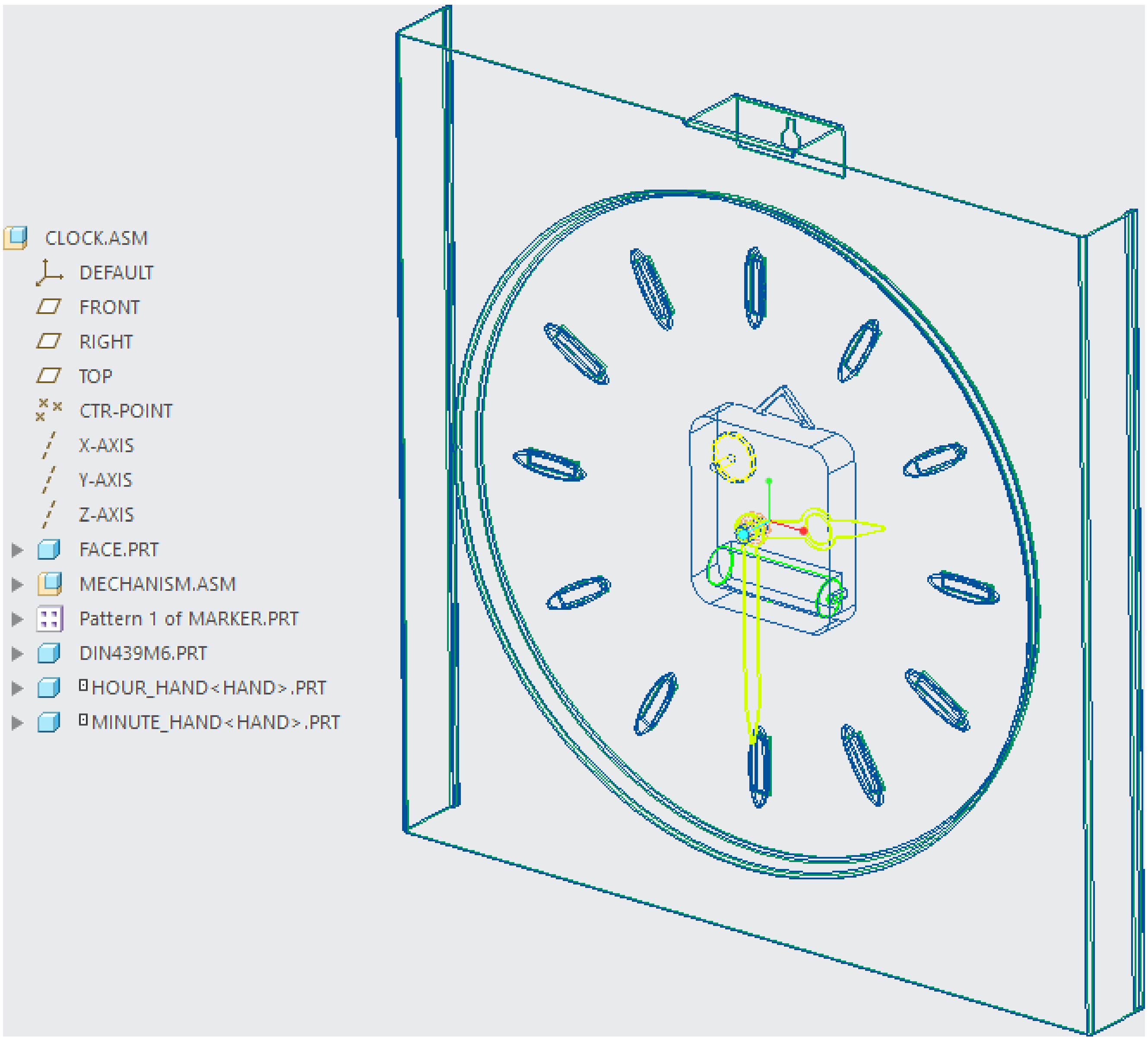


Figure 1: Created assembly and its model tree. *Display Style* as wireframe to see parts behind the clock face.

Learning Targets

In this exercise you will learn

- ✓ to place components to assembly
- ✓ to create sub-assemblies
- ✓ to use mechanism connections
- ✓ to use gears.
- ✓ to export Creo View models.

For this exercise you will need the following clock parts from previous exercises:

- battery.prt (Exercise 1.1)
- face.prt (Exercise 2.1)
- hand.prt (Exercise 1.2)
- mechanism.prt (Exercise 1.3)

You also need the following files from MyCourses:

- din439m6.prt (a nut)
- marker.stp (a neutral file format)
- knob.prt (a knob)

Make sure that you have a copy of each part in the same directory, which you have selected as your working directory (In Aalto classroom setup the default path is Z:\Creo_working_dir\).

Creating Assembly

Let's create a new assembly file by selecting **New** () and selecting **Assembly**. You can name the file for example C*lock* (Figure 2).

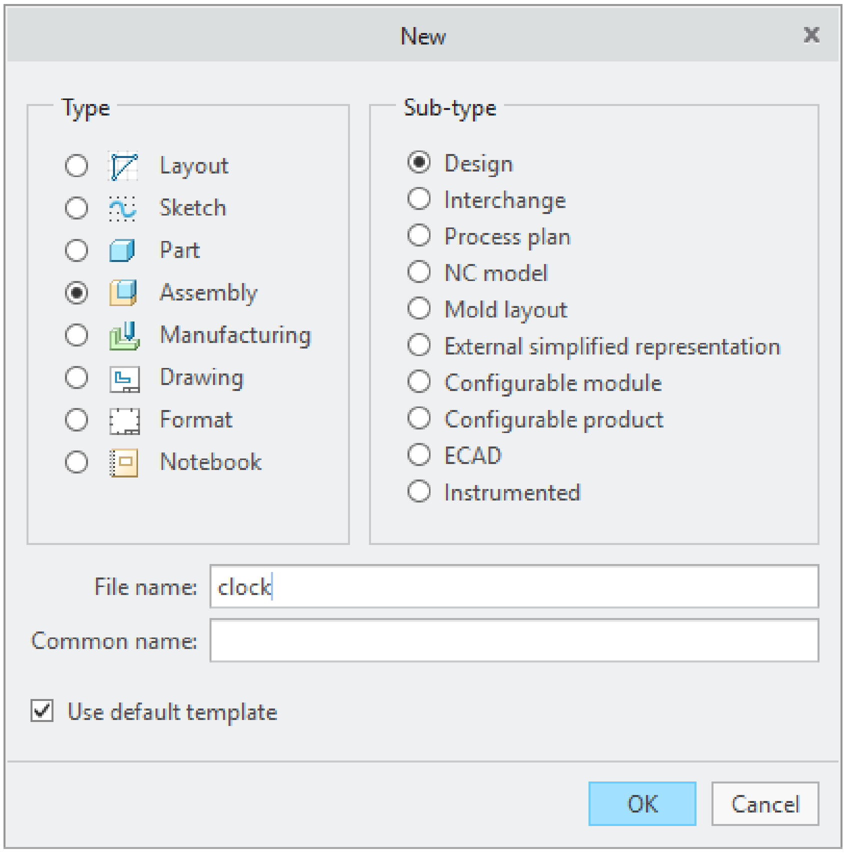


Figure 2: Creating a new assembly.

Assembly files in parametric CAD software define a collection of existing part files, constraints for their positioning, and possibly some assembly-level features. When entering the assembly mode, a list of parts is seen in the left column (instead of the list of the features in the part modeling mode). In Creo, the part list by default only shows the part names, but it can be configured to show also assembly-level features and features inside parts. Luckily, by using Aalto templates, part list shows all features.

Like new part files, new assembly files are not empty by default. They have their own default datum planes and a coordinate system. These are meant to be used as a fixing point for the first component(s).

Adding Components

Face.prt

We can add any part or assembly files in our current working directory (or in internal memory) as components for this new assembly. In the *Component* group, select **Assemble** (). A file browser lets you choose from the files available in your working directory, and by clicking the **Preview**-button you can preview them before selecting. Select **face.prt** and click **Open**.

The selected part will be shown in the 3D-view painted in purple. This means that Creo is waiting for instructions for the placement. The dashboard in the top should look like in Figure 3.

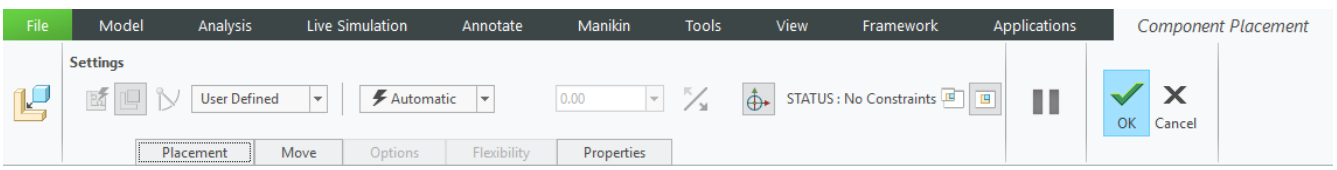


Figure 3: Dashboard. Notice the "STATUS: No Constrains" text.

Click **Placement** and you will see the tools for creating placement constraints (Figure 4):

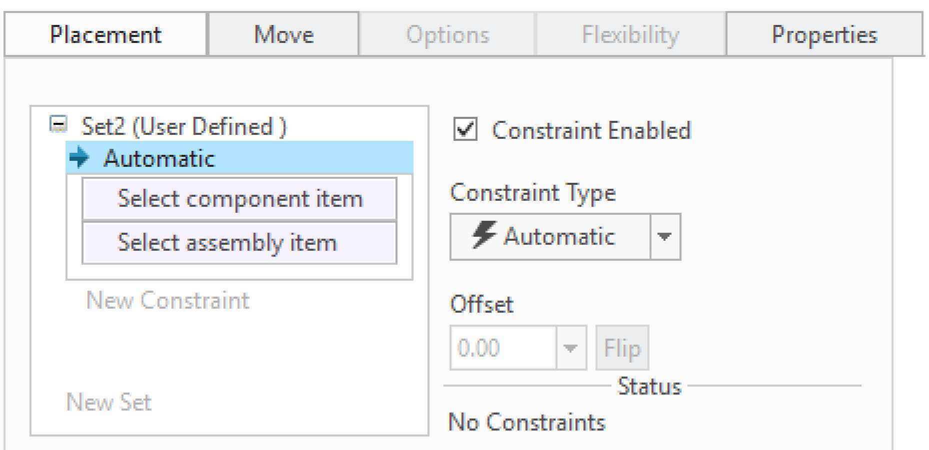


Figure 4: Placement tab opened.

Our aim in placing components in assembly is to lock out all of their degrees of freedom in a purposeful way. When brought into the assembly, all parts have the natural 6 degrees of freedom (3 axis of rotation, 3 axis of translation), and we do not want to leave any of these free if we don’t specifically want to create a mechanism. Unintentionally left degrees of freedom will sooner or later create unintended (often strange) behavior in the model.

Locking out degrees of freedom is done by adding constraints. Available constraint types can be seen in the *Placement* tab, inside the *Constraint Type* selection box or from dashboard list (Figure 5).

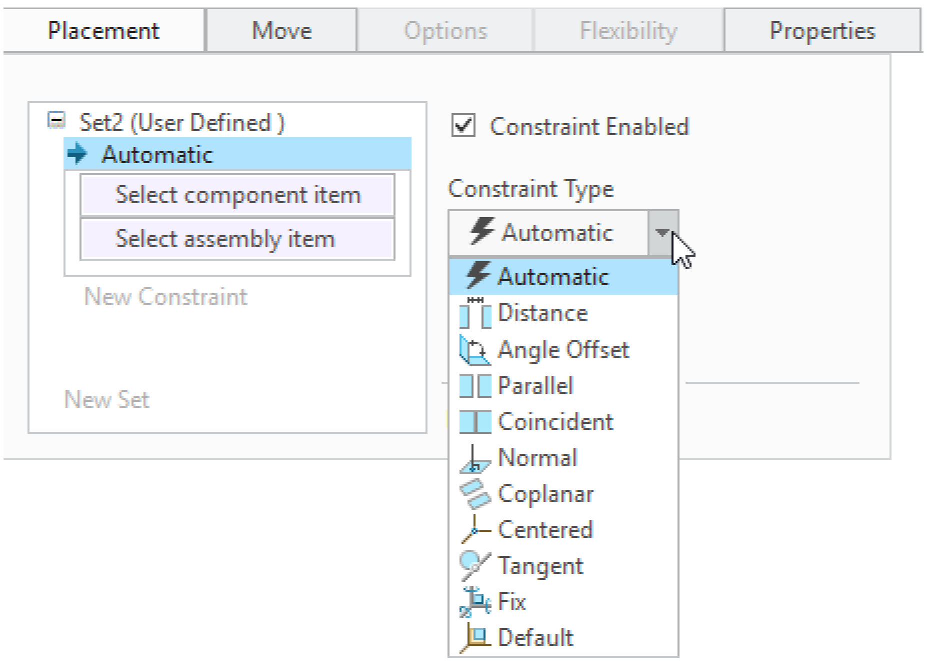
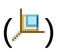


Figure 5: Different kind of Placement Constraints.

All of these constraints (mostly self-explanatory) lock out one or more degrees of freedom between two selected points, surfaces, or coordinate systems.

We want to lock the first component (face.prt) in the middle of the assembly datum planes. The easiest way to do this, is to use the **Default** () constraint. Select it. It will connect two default coordinate systems together by all 6 degrees of freedom, thus fully fixing them according to each other. When all degrees of freedom are fixed, the part will be highlighted in yellow.

It is important to follow the **Status** message. **Fully Constrained** means that all 6 degrees of freedom have been successfully locked. If more constraints would be needed, you should create them by clicking on **New Constraint**. Existing constraints can modify by clicking on them, or deleted by clicking *RMB* on them and selecting *Delete*. None of this is necessary now, so you can accept the placement by clicking on top right or by pressing **MMB**.

The first part appears now to the parts list (CLOCK.ASM) (Figure 6):

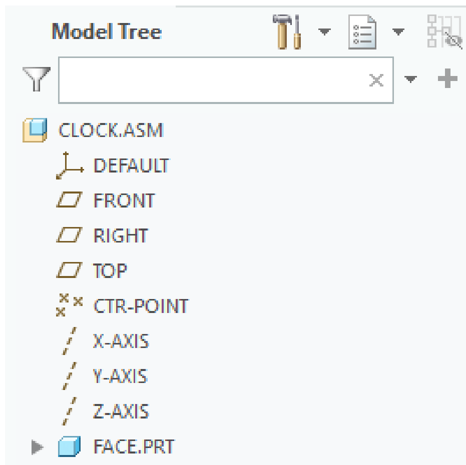


Figure 6: Part list.

If you wish to edit the placement constraints, you click **RMB** on a part and select **Edit Definition**, similar how you can edit features in the part mode.

You have now created a minimal assembly with one part. **Save** this file.

Creating a Sub-assembly

The next component for our clock assembly would be the mechanism of the clock, but let's add it in as a sub-assembly. Large assemblies should always be divided into clever sub-assemblies in order to make constraint definition and component re-usability easier.

Create a new () **assembly** file named *mechanism.asm*, and **Assemble** () the **mechanism.prt** as the first component just as we did before (remember **Default** () constrain). As you can see, you can have a part and an assembly named using the same name.

Battery.prt

Next, assemble ( ) the **BATTERY.PRT** but with different constraints. We will need to align the battery's center axis with the center axis of the battery chamber in the mechanism part. In addition, we need to mate one of the battery's end surfaces to suitable surface on mechanism's battery hole.

Set **Axis Display** ( ) on and select the **Coincident** ( ) constraint and pick two surfaces from the components as seen in Figure 7.

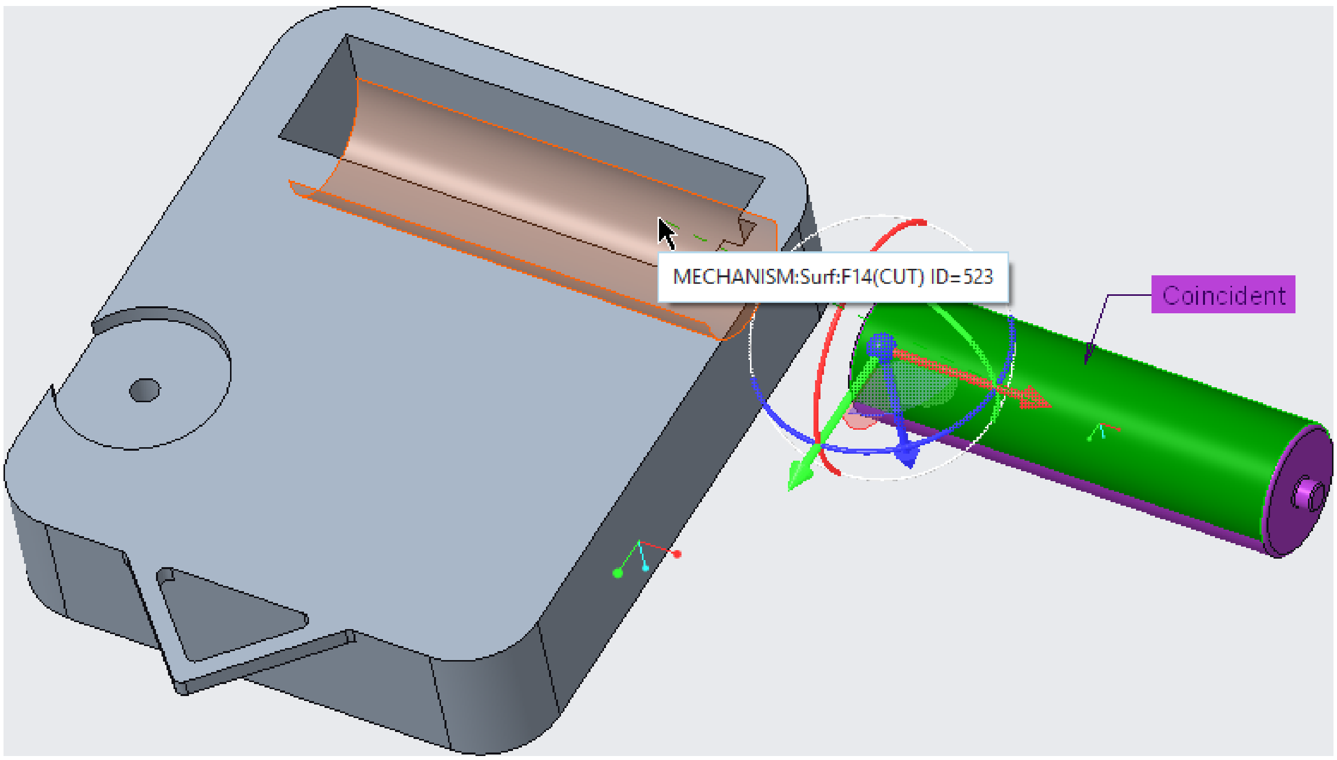


Figure 7: Surface from battery selected, selecting surface from mechanism.

Now, the next constraints should lock the movement about this axis (notice the red arrow). Select **New Constraint** and **Coincident** ( ). Select surfaces shown in Figure 8.

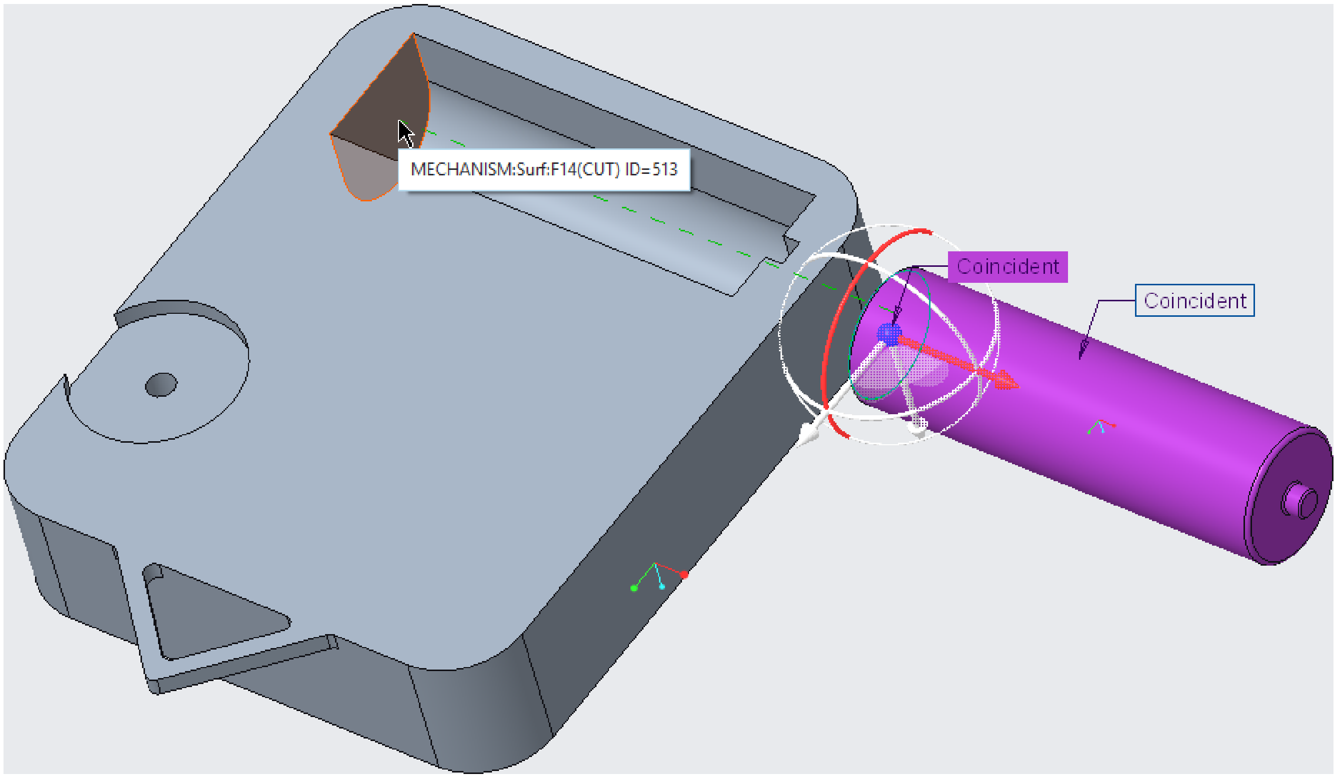


Figure 8: First surface selected, selecting second surface.

Take a look at the *STATUS* field (in the **Placement** tab), it has a new checkbox called *Allow Assumptions*. Also, it says that our placement is *Fully Constrained*, which we know is not yet true (the battery should still be able to rotate about itself!). What happens here, is that when there is only one degree of freedom left, Creo tries to <u>*guess*</u> the last constraint for us. These guesses are called assumptions, and they can be disabled by unchecking the *Allow Assumptions* checkbox. But there is seldom need for this, as the assumptions are usually just what we want (as is the case this time). Accept the placement (✓).

Knob.prt

You have now successfully added two basic parts to this assembly. Now, on your own, assemble () one more part: KNOB.PRT. Feel free to create the constraints as you wish, just make sure that the assembly stays is **Fully Constrained**. After this, the mechanism sub-assembly is ready (Figure 9), **Save** it and **close** its window (⊠ in the top-left bar).

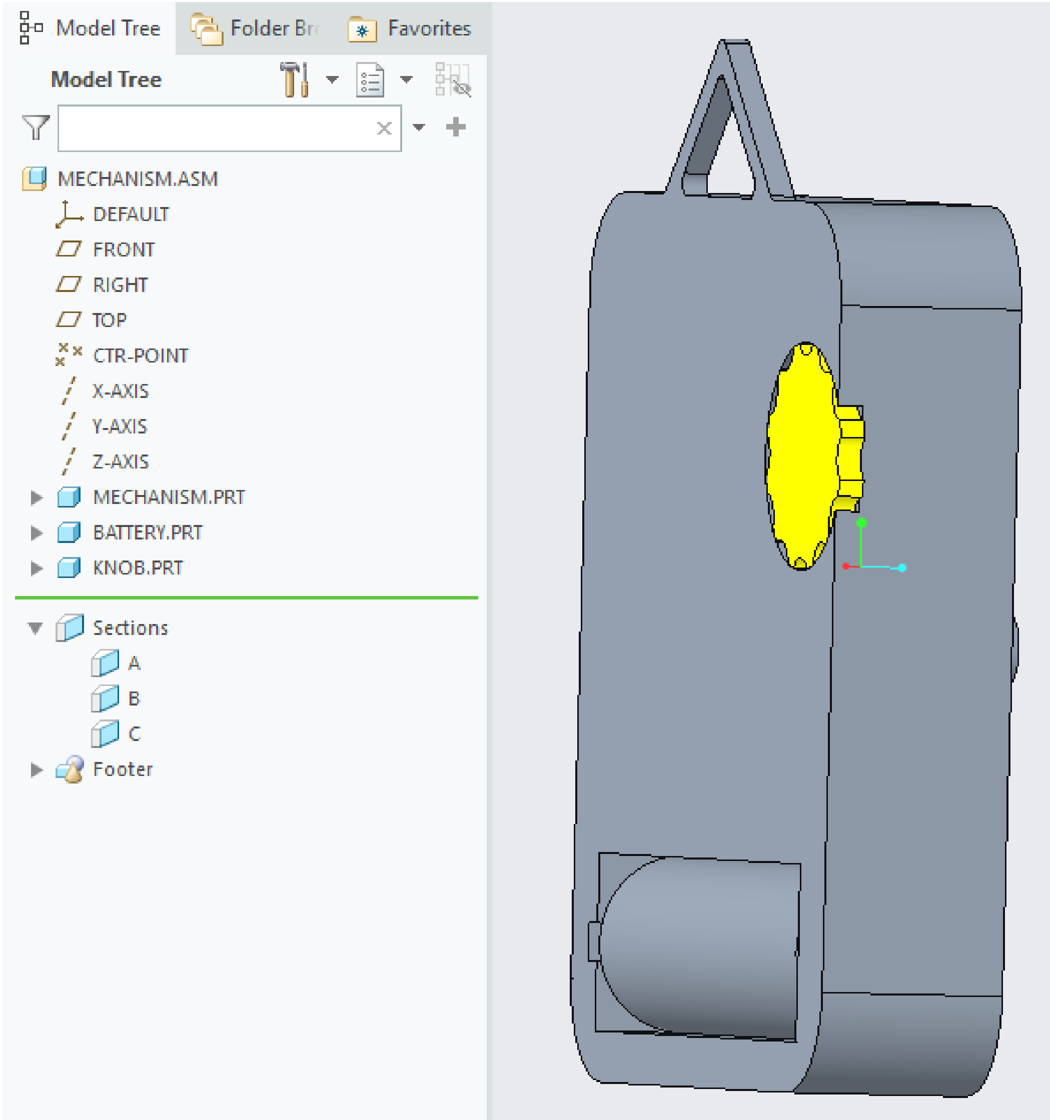


Figure 9: Ready subassembly.

Colors

Note: You can use **Appearance Gallery** (, **View** tab, in *Appearance* group) to add some colors to your parts! Open part, select **Appearance Gallery**, select color, then part name from the model tree (Battery.prt) and **MMB** to accept (Figure 10). Remember to save your part after this.

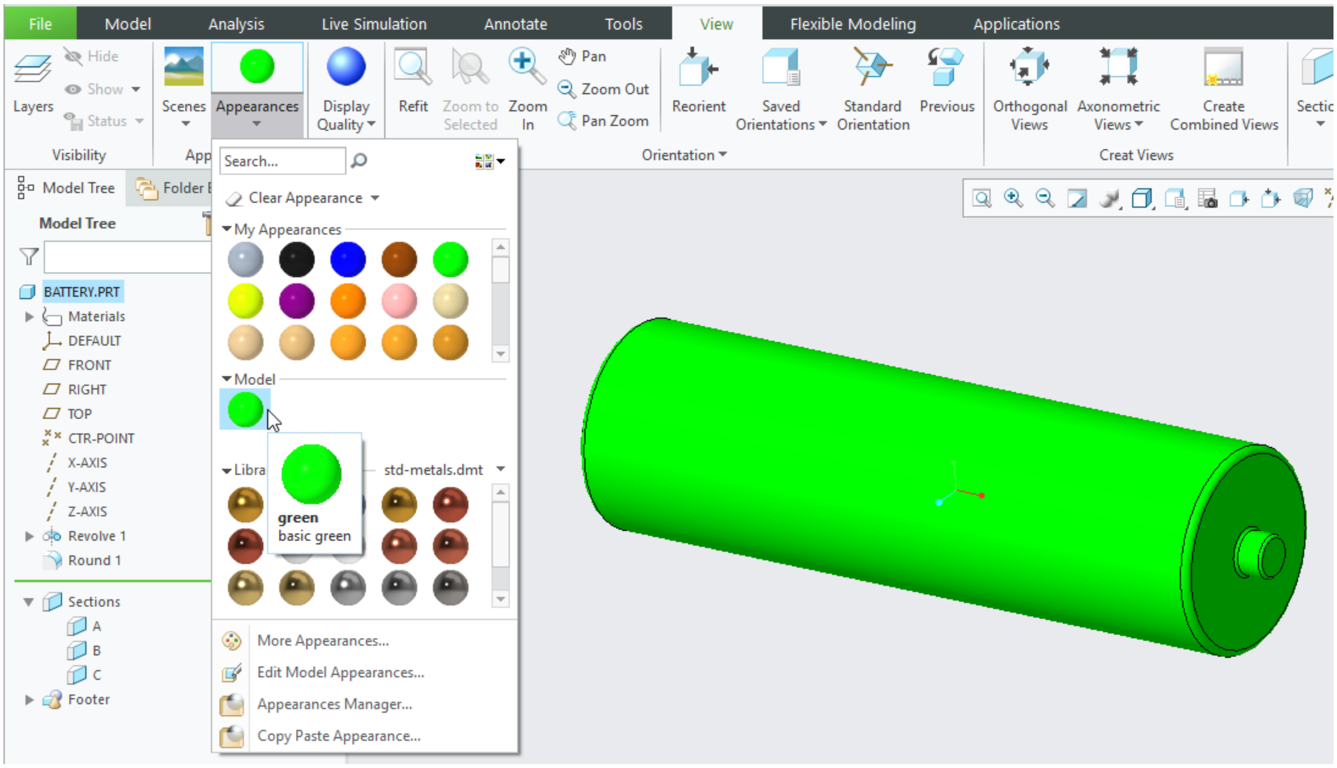


Figure 10: Selecting color.

Adding Sub-assembly to the Main Assembly

Open the previously saved CLOCK.ASM and assemble ( ) the newly created MECHANISM.ASM just like adding a new part. To create the necessary constraints, first **coincident** the mechanism's hollow axle into the center hole of the clock face ( ). Then **coincident** the mechanism front surface to the face's back surface ( ). Once again, it's safe to *Allow Assumptions*. Your model should look like in Figure 11.

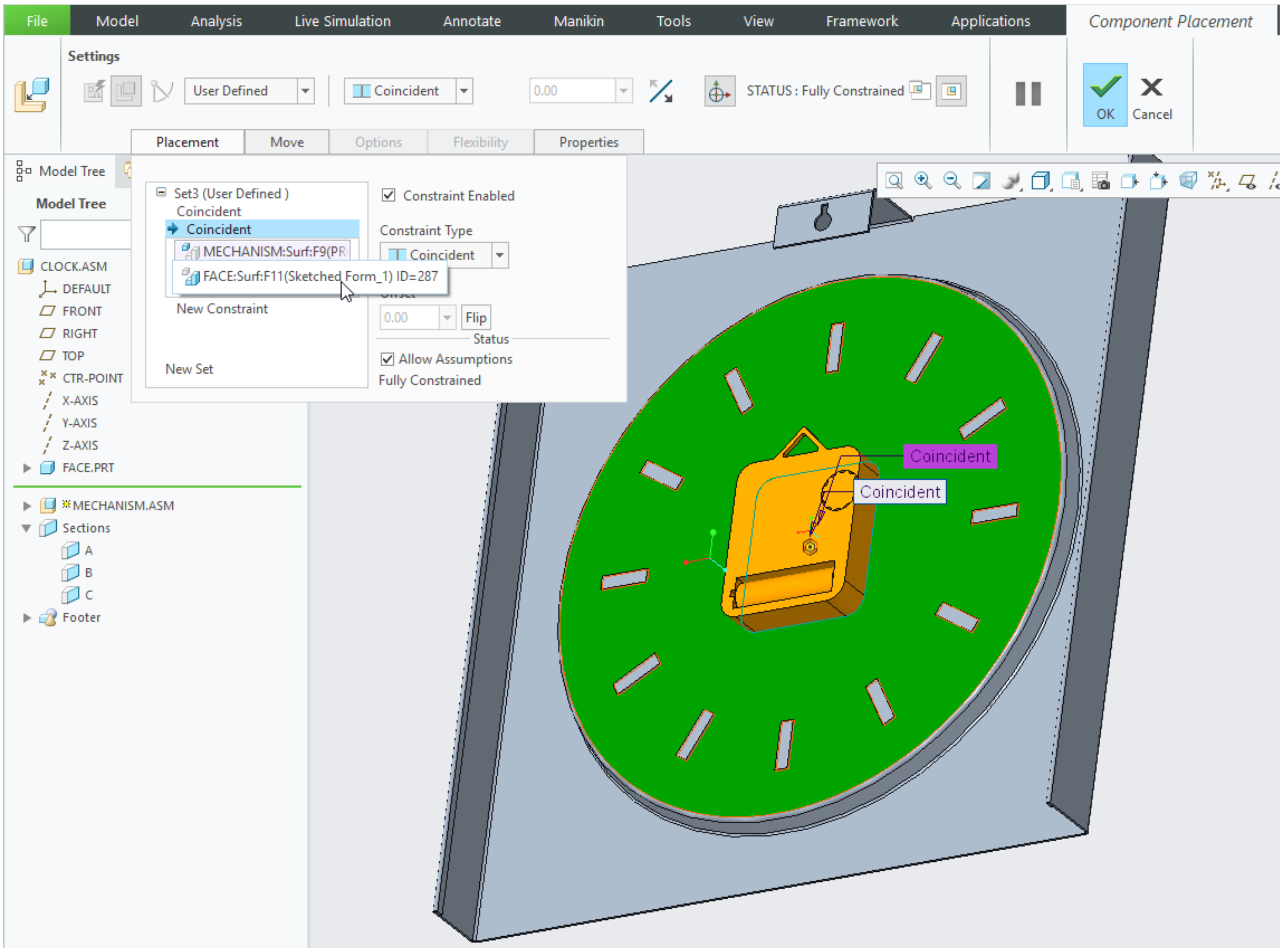


Figure 11: Added sub-assembly and its constraints.

Keep in mind, that when adding an assembly as a component to another assembly, you can create constraints between any parts inside the assemblies. It is easy to get them scattered unwisely, which results in too much dependencies between too low level parts. In the future, you want to be able to remove or replace low level parts (like batteries, knobs, bolts) without re-creating the whole upper level assembly structure. For example: you do not want to attach a bolt to the clock face, and then attach the mechanism to the bolt – if you later decide to remove the bolt, you will have to remove the mechanism too! So think before creating constraints, and try not to create traps for yourself.

Adding a File from another CAD Program

Using Neutral File Format

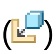
In the beginning of this exercise, you copied a file called marker.stp. This file was made with Solid Edge (another CAD software by Siemens). By default, model created with other CAD programs are not compatible. To export geometry (only geometry, not features!) a neutral file format (commonly used are STEP, IGES and Parasolid) can be utilized.

To export part to another CAD system:

1) Open part in native CAD program (in this case Solid Edge).
2) Select Save As -> Save AS Translated.
3) Save part in a neutral file format (in this case STEP).

What just happened? We exported native Solid Edge file to neutral file format (STEP, other formats are for ex. IGES and Parasolid). This exports only geometry, not features! In Creo, we can't see how the part is made, we can only see its geometry.

Marker.stp

Assemble () marker.stp to our clock assembly. Remember to change *Type* to STEP to see previously created file (Figure 12). **Import**.

Note!
Creo 6.0 can also open some native files from selected CAD programs (Catia, NX, Solidworks, Inventor, Solid Edge) directly, but still can only see its geometry, not how it is made. Advantage is, that if you do changes to the native file, it can be directly updated in Creo. With STEP, you need to replace existing file in the assembly.

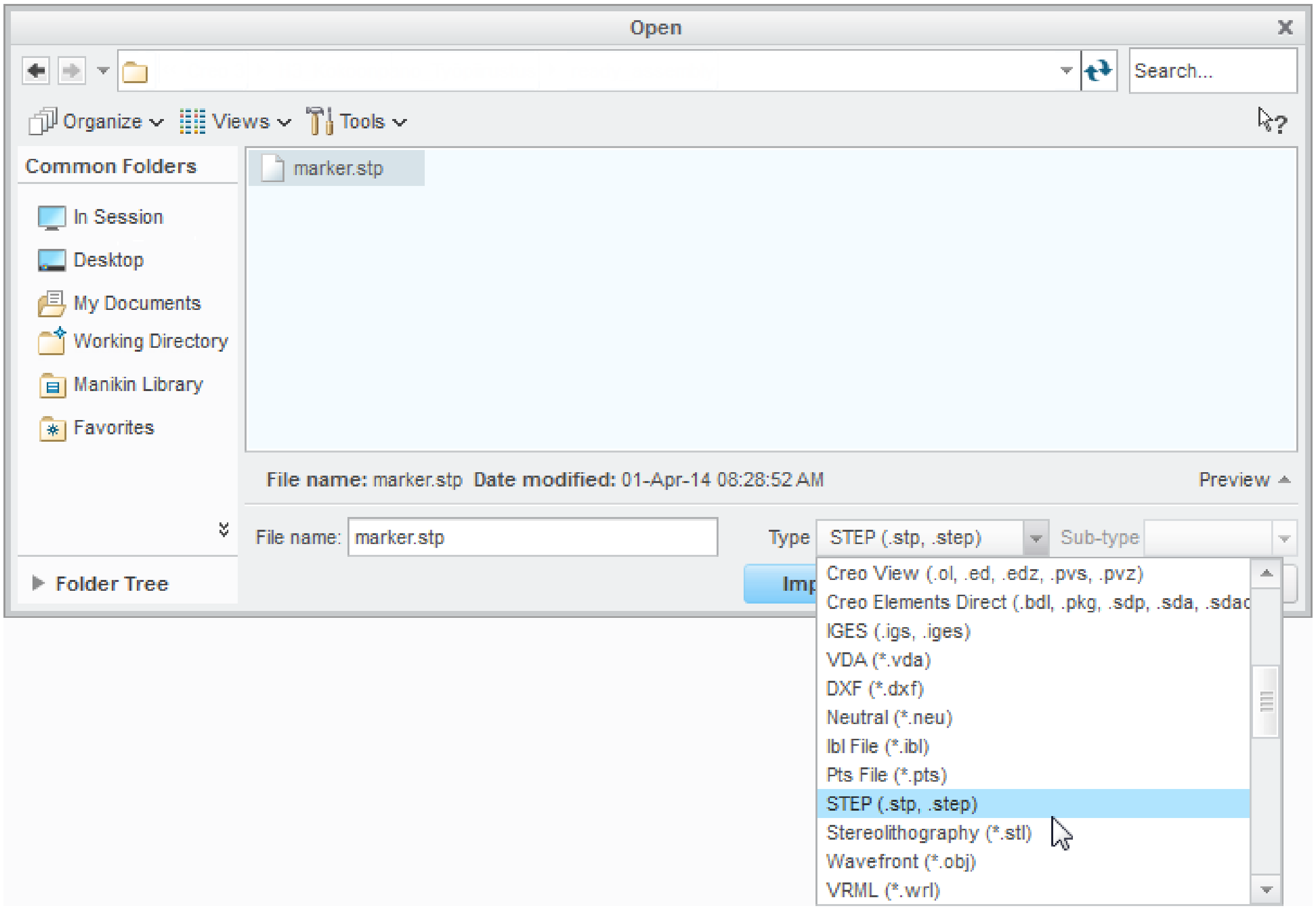


Figure 12: Changing Type to STEP.

Import New Model window opens. Default settings are fine, **OK**. Click ( ) to see assembled part in small preview window. Using **Coincident** constraints ( ), assemble marker as seen in Figure 13 to location seen in Figure 14.

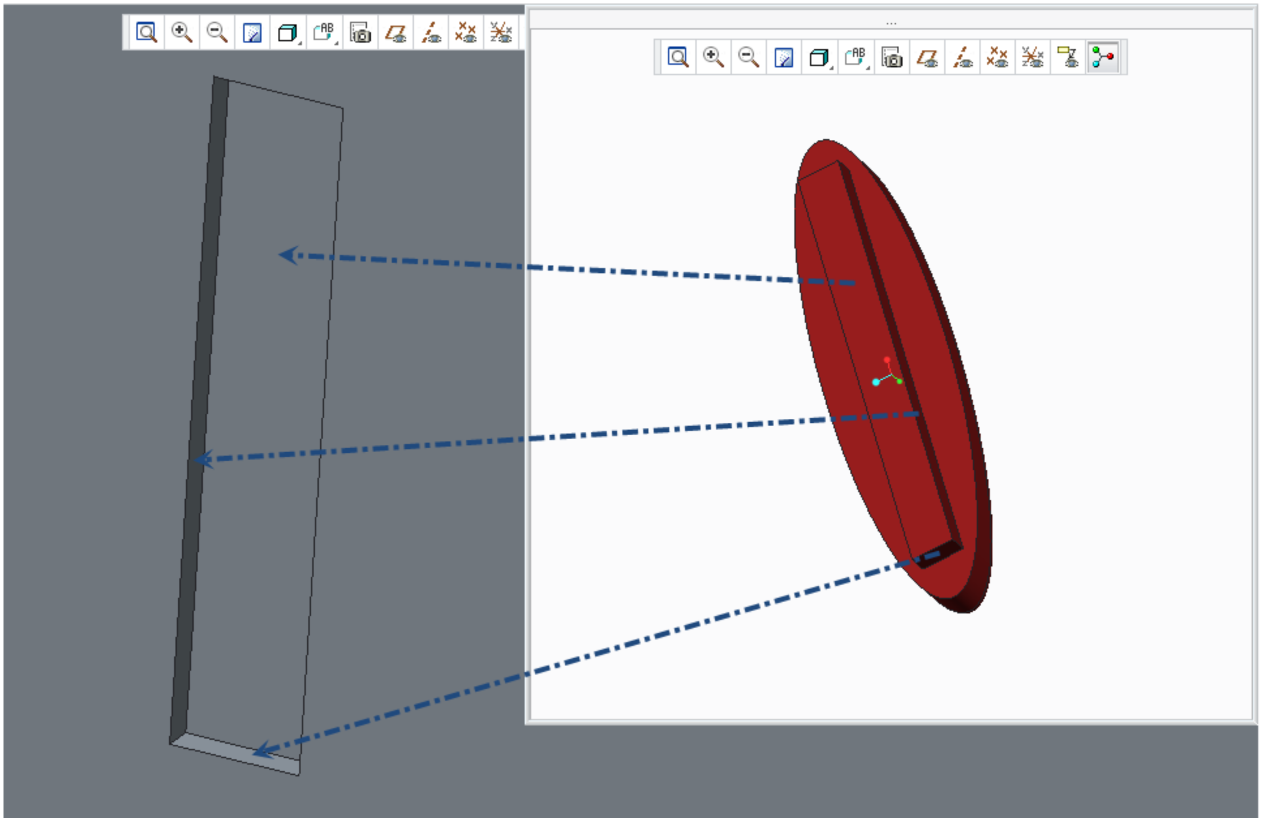


Figure 13: Surfaces to be Coincident with each other.

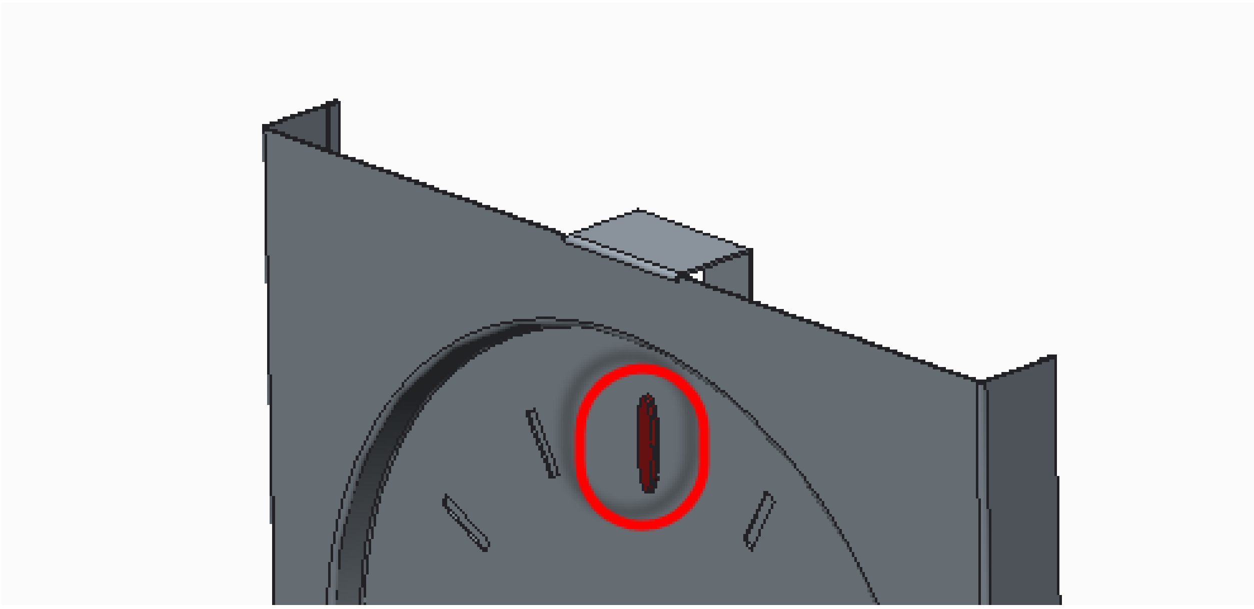


Figure 14: Location for marker.

Patterning Components

Select MARKER.PRT from model tree, hold **RMB** and select (**Pattern**)

Creo should recognize that the part is constrained only to one patterned feature, which makes the *Reference* type pattern the default option in the patterning dashboard (Figure 15).

Figure 15: Reference as a pattern type.

At this point, the pattern preview should show the positions of the planned instances by drawing uellow circles with black dots (Figure 16). You could turn instances on/off by clicking on these circles. Note: if you didn't see the yellow circles, you have not constrained marker-part to the feature that has patterned (the first instance of the pattern). No panic, just accept the feature and choose Edit Definition for pattern, then you see the circles.

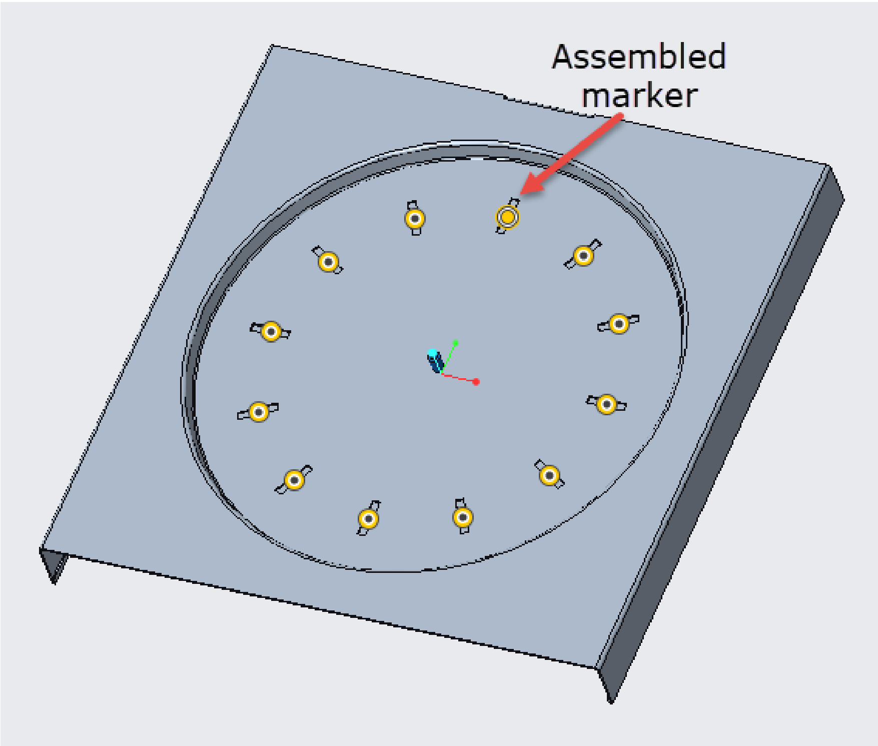


Figure 16: Pattern preview. Notice the location of the assembled marker.

Adding Components

DIN439M6.prt

Place DIN439M6.prt (nut to the assembly) as seen in Figure 17.

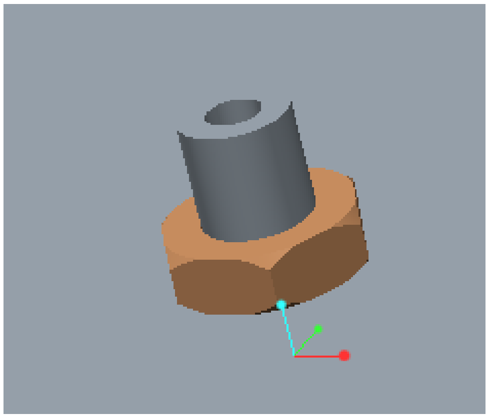


Figure 17: Nut in the assembly.

Mechanism Connections

Hour_hand

Next **assemble** () hand.prt to the assembly. Select HOUR_HAND as an instance. This time we want that this part can move, i.e. it has one degree of freedom (rotation over its connection axis). From the dashboard, select **Pin** () as a connection type (Figure 18).

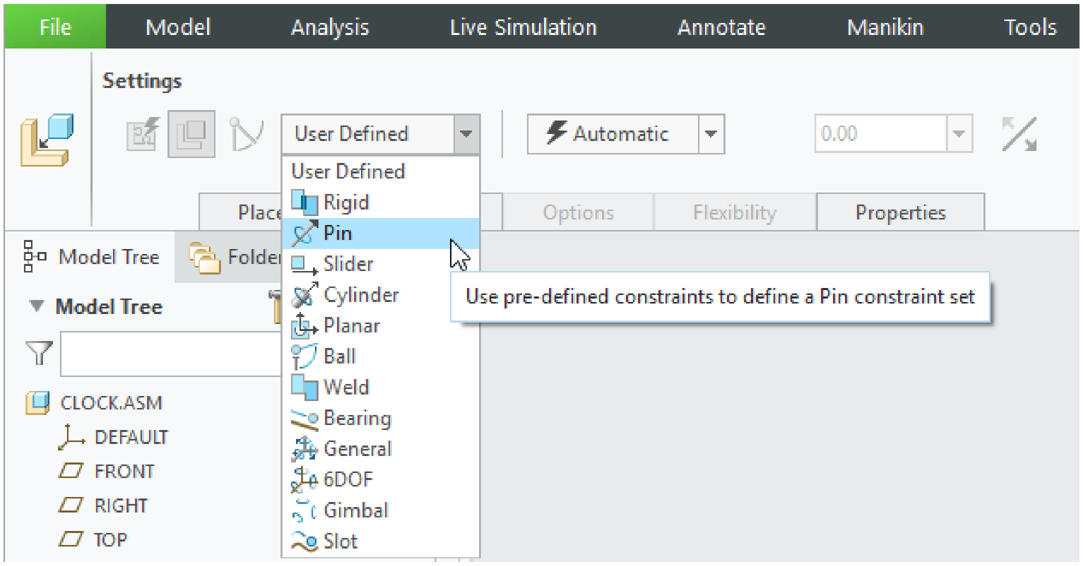


Figure 18: Selecting Pin as a connection type.

Pin connection lets component to have one degree of freedom. Open **Placement** tab to see what kind of constraints this pin connection needs. First we need to define *Axis alignment* (i.e. rotation axis), as seen in Figure 19.

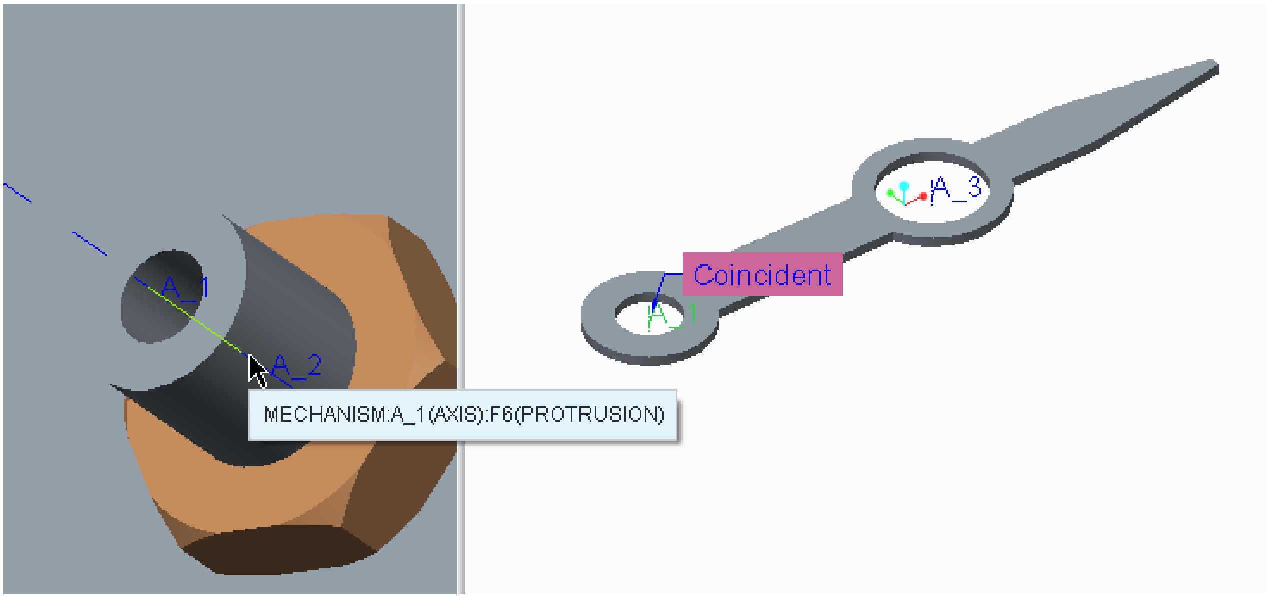


Figure 19: Pin Connection - Axis alignment.

Next *Translation* need to be defined (Figure 20).

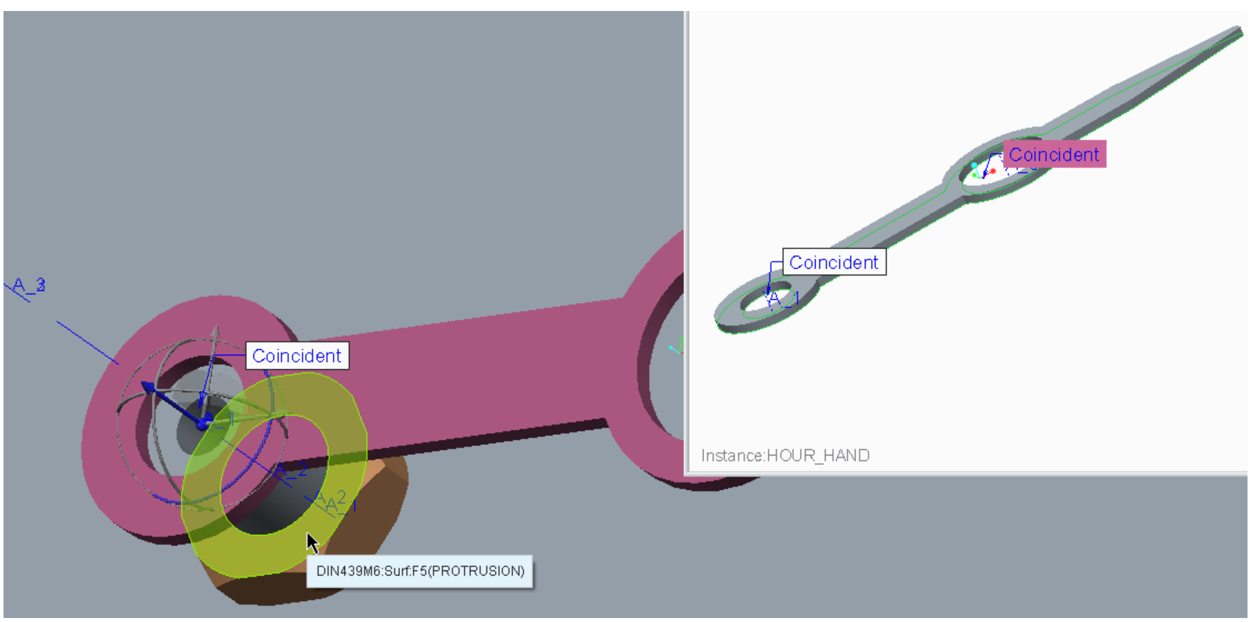


Figure 20: Pin Connection - Translation.

Click on *Rotation Axis* (yellow circle next to it says that this is optional), select TOP from HAND.PRT and RIGHT from MECHANISM.PRT as references (Figure 21). To select plane behind other plane, click RMB to flip between possible references (or hold **RMB** and **Pick From List**, Figure 22).

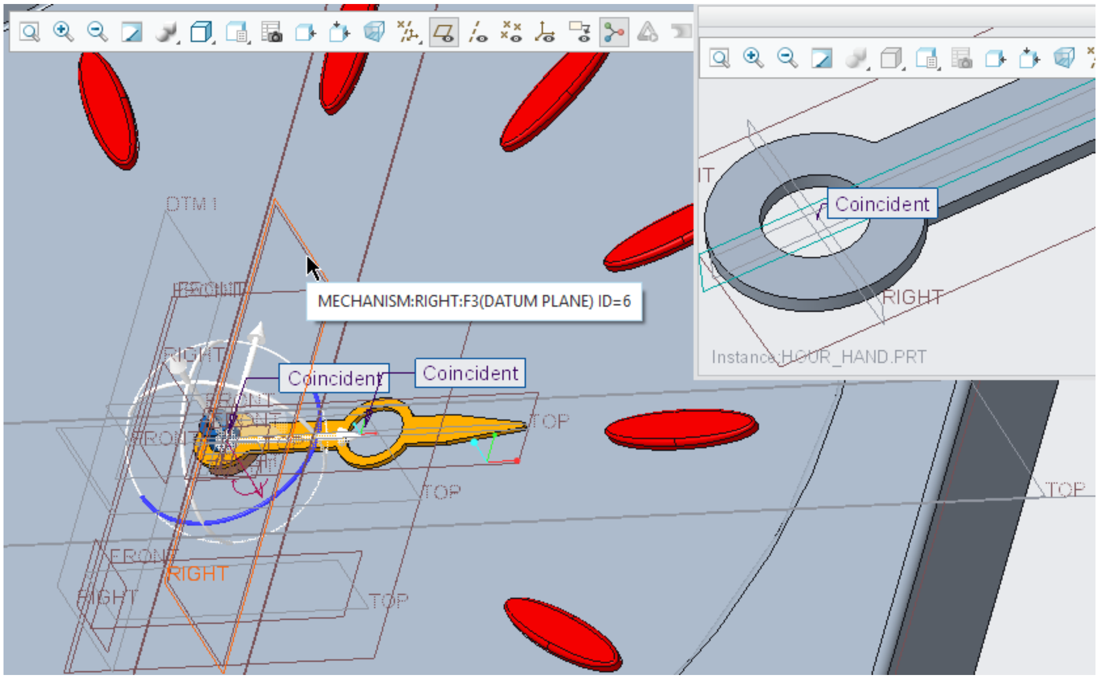


Figure 21: Pin Connection - Rotation Axis.

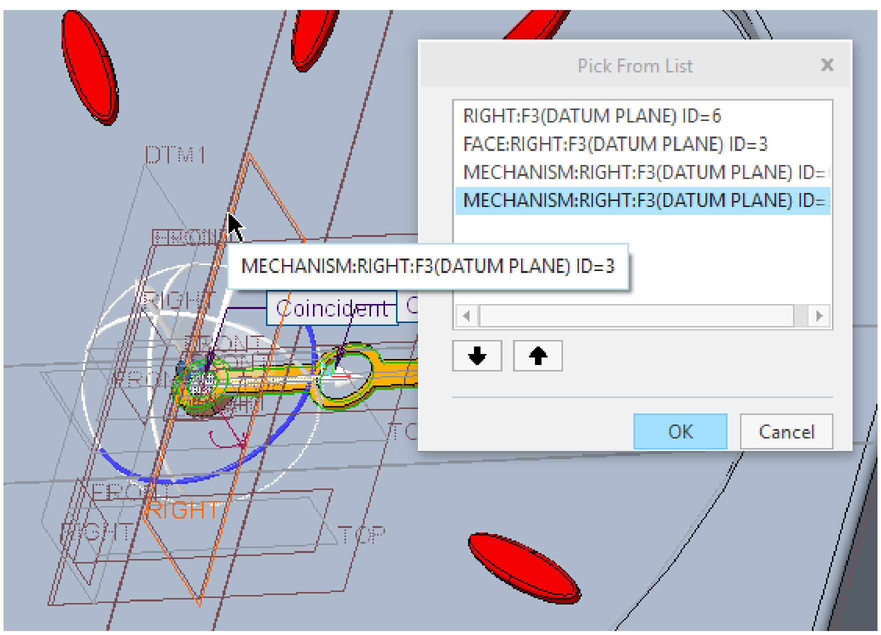


Figure 22: Pick From List menu, RIGHT from mechanism selected.

To *Current Position* field, give **0 or 180** as a value (depending witch direction is up).

Our Pin connection is ready (notice the status text), accept ( ) it.

MINUTE_HAND

Assemble minute_hand using the same kind of connection as for hour_hand (Figure 23), and ensure, that the minute_hand is connected to the nut upper surface, not to hour_hand (we want these two parts not to be depended on each other), so you can use **distance** ( ) constraint with **1** mm from the nut surface.

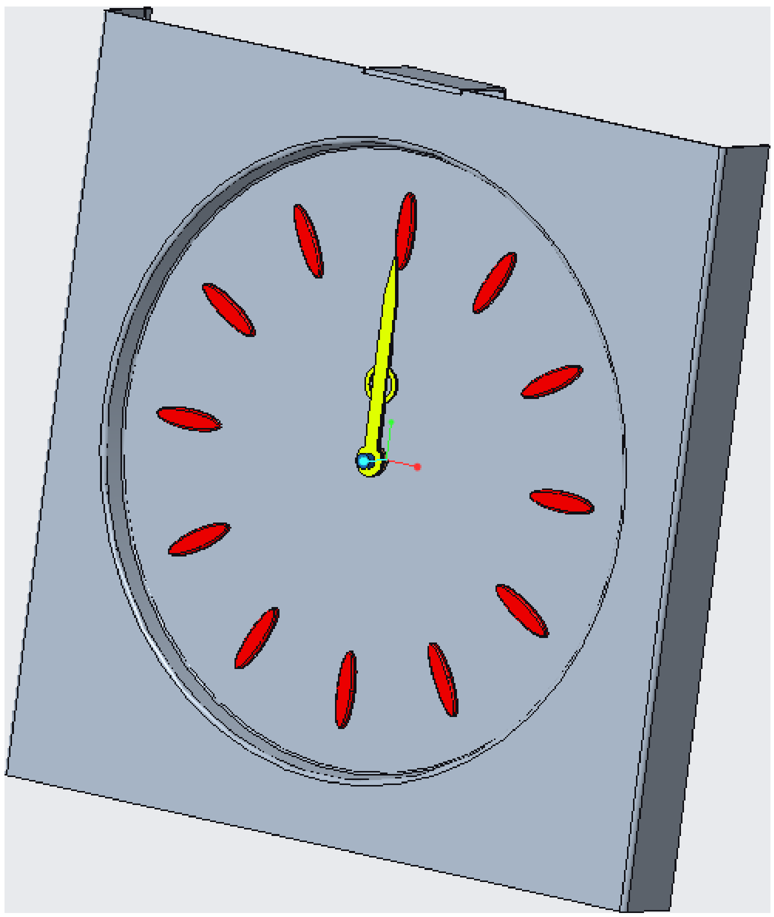


Figure 23: Both hands added.

You can use **Drag Components** (, in *Components* group) tool to move hands, try it out now! After testing, place components as seen in Figure 23 (**Edit Definition**, **Rotation Axis**, give value).

Adding a Gear

There is a gear in the real clock, which allows one motor to move both hands. Select **Application** tab and **Mechanism** (Figure 24). This will open a new mode.

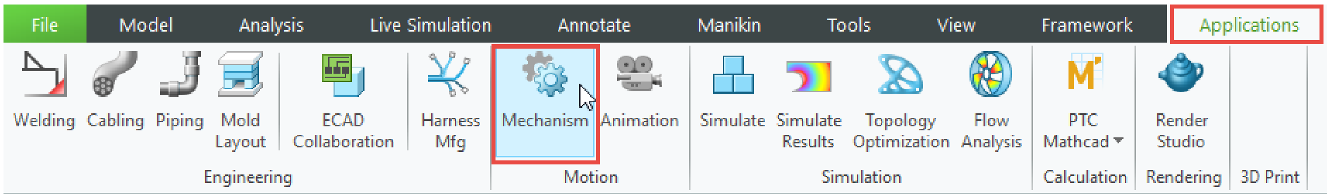


Figure 24: Selecting Mechanism from Applications tab.

Select **Gears** ( ) from *Connections* group. This opens *Gear Pair Definition* window. As *Gear1* select hour_hand's connection (lowest one) and as *Gear2* select minute_hand's connection (topmost). Select **Properties** tab. Select **User Defined** as *Gear Ratio*. For values in D1 and D2, put values from normal clock (if minute_hand rotates 360, how much rotates hour_hand?).

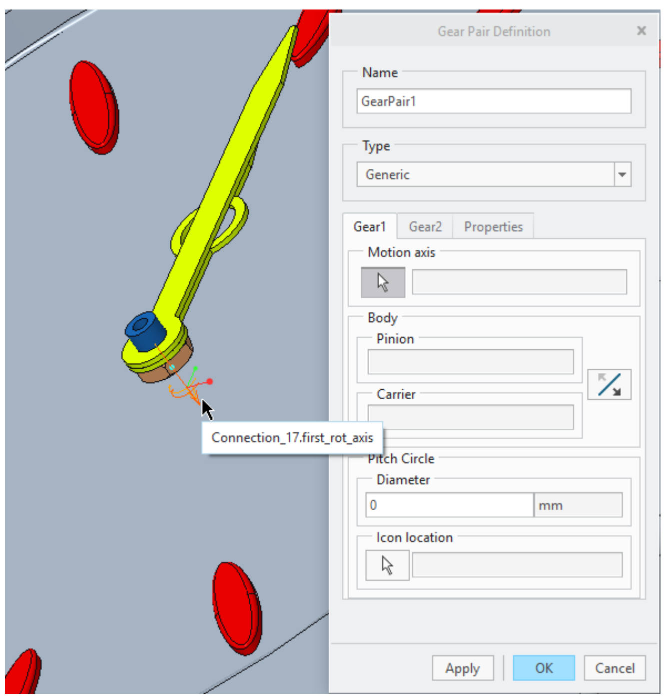


Figure 25: Selecting lower connector (hour hand) as Gear 1.

When ready, accept feature (OK). You can use **Drag Components** ( ) to test your gear. If changes are needed (like changing gearing direction, **Gear2** and), GearPair connection can be found in Mechanism Tree (Figure 26).

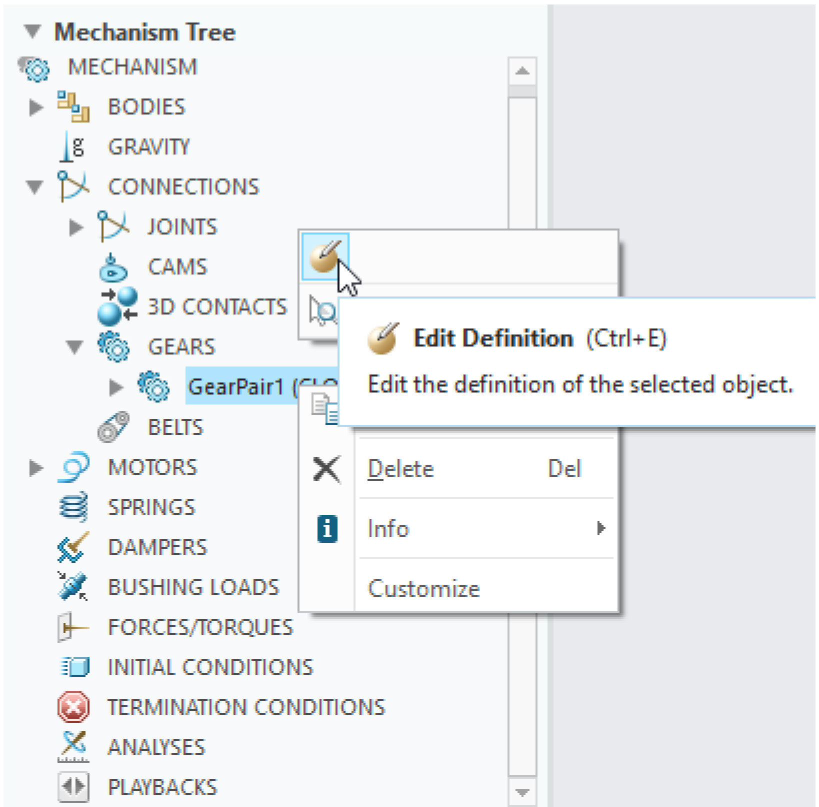


Figure 26: Mechanism Tree and selecting Edit Definition for GearPair.

When gear works like in a real clock, close the Mechanism mode ( ). You are now in the normal assembly mode. Test *Drag Components* ( ) here.

Viewer Model

Several CAD programs have viewers, which allow to see CAD models without owning specific program and its costly licenses. PTC has a freely downloadable *Creo View* for this purpose[1]. To export your assembly to Creo View compatible format, select **Save → Save As → Save a Copy**. From *Type* list, select **Creo View (*.pvz)** and **OK**.

[1] https://www.ptc.com/en/products/plm/plm-products/creo-view

To check that this model works, go to your working directory (default save location) and double-click *.pvz file (Figure 27) to open it with Creo View (select MCAD as type if needed).

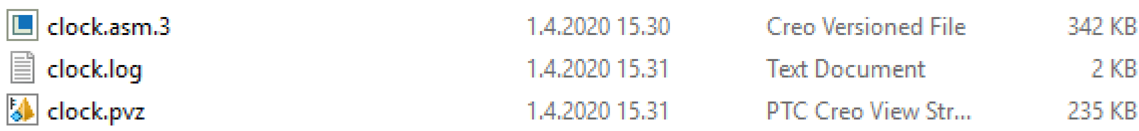


Figure 27: Created pvz file and its log file.

By default, all files are hidden. Check CLOCK.ASM from Model tree to see the whole assembly (Figure 28).

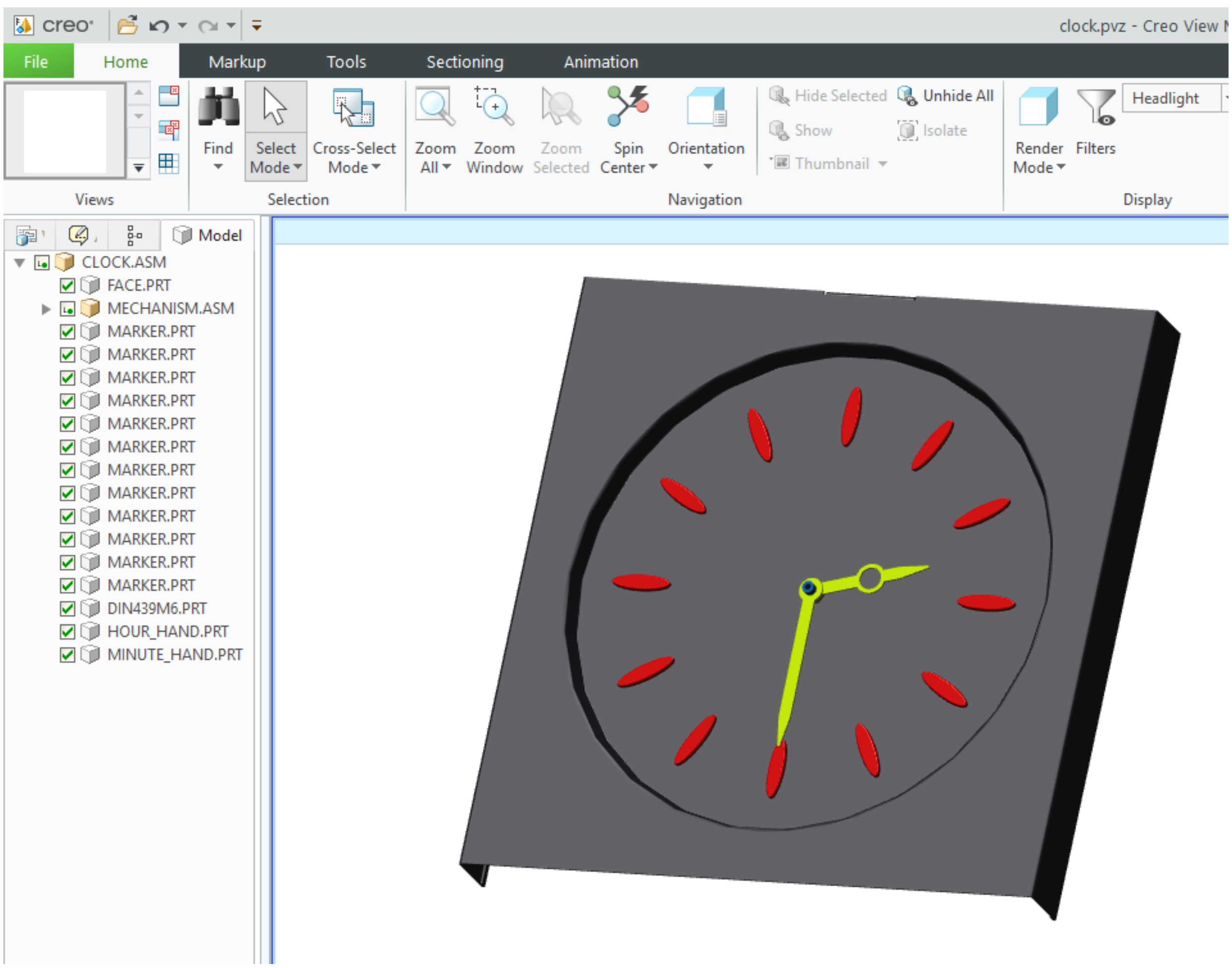


Figure 28: Creo View model of the clock. You may have different colors.

This concludes this exercise; remember to save your work! Return the *.pvz file into MyCourses.