

KISSsys Anleitung:

Grafische Elemente in KISSsys

Stichwörter	Darstellung, Kollisionsprüfung automatisch und manuell, Solid Elements, Graphic Elements
Letzte Änderung:	16. Aug. 2005

1 Einleitung

Folgende Anleitung befasst sich mit der 3D Repräsentation eines KISSsys Modells. Die 3D Repräsentation ist im Grunde nur ein kleines Nebenfeature des Programmes, dient aber zur Visualisierung des Modells. Es ist möglich, diese Repräsentation mit gewissen Befehlen zu

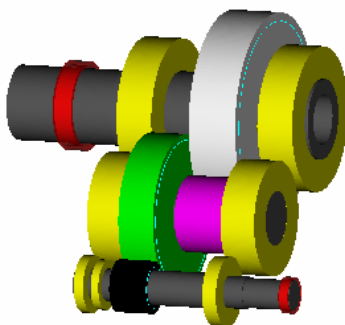


Abbildung 1-1 Anleitungsmoel

manipulieren. Diese Möglichkeiten erlauben es, vorhandene Kollisionen mehrerer Systemkomponenten zu erkennen. Deshalb ist es das Ziel dieser Anleitung, die grafischen Möglichkeiten der 3D Repräsentation anhand eines bestehenden Modells zu erläutern.

Die vorgestellten Tools gehören weniger zum technischen als zum visuellen Aspekt von KISSsys. Dieser kann vor allem an Wichtigkeit gewinnen wenn das Modell einem Kunden vorgestellt wird oder wenn komplexe Getriebe dargestellt werden sollen. Weiter erlaubt die Visualisierung eine ständige Kollisionskontrolle.

2 Lösungsprinzip

Um die Zahnräder besser auseinander halten zu können, ordnet man allen Elementen eine eigene Farbe zu. Anschliessend definiert man die Transparenz eines Rades welches die möglichen Kollisionspunkte verdeckt. Falls man nun immer noch nicht genau erkennen kann ob eine Kollision vorliegt, verfügt KISSsys über eine automatische Kollisionsprüfung. Die Kollisionsprüfung kann auch mit einem Gehäuse erfolgen.

3 Farbgebung

Die Darstellung eines Modells in der 3D Repräsentation geschieht anfangs immer in festgelegten Farben (Blau: Zahnrad, gelb: Lager, rot: Kupplungen, dunkelgrau: Wellen). Die Farbgebung der Elemente kann aber geändert werden. Dies geschieht durch die Anlegung der KISSsys Variable $kSys_3DColor$ für die einzelnen Elemente. Je nach Wert der Variable ändert sich die Farbe des Elementes.

3.1 Beispiel Zahnrad

Das Rad 1 auf der Zwischenwelle soll in der Farbe grün, statt blau erscheinen.

Mit einem Rechtsklick auf *Gear1* unter *Shaft2* und der Wahl *New Variable* wird mit der

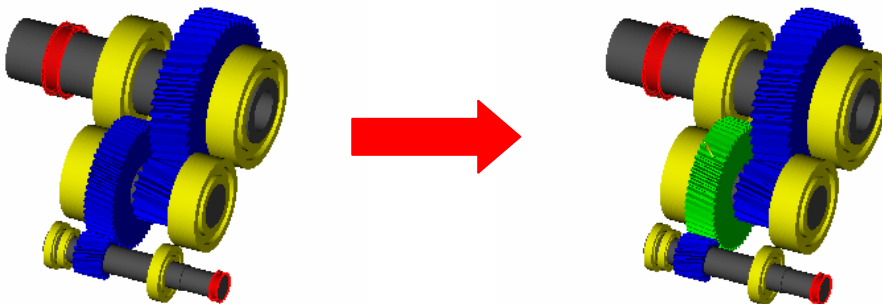


Abbildung 3-1 Farbänderung Rad1 Zwischenwelle

Eingabe von $kSys_3DColor$ als Variablenname, die neue Variable definiert. Der Variable wird der Wert 3 zugeordnet (3 steht für Grün) und anschliessend wird sie mit *Insert* hinterlegt und das Fenster mit *Quit* geschlossen.

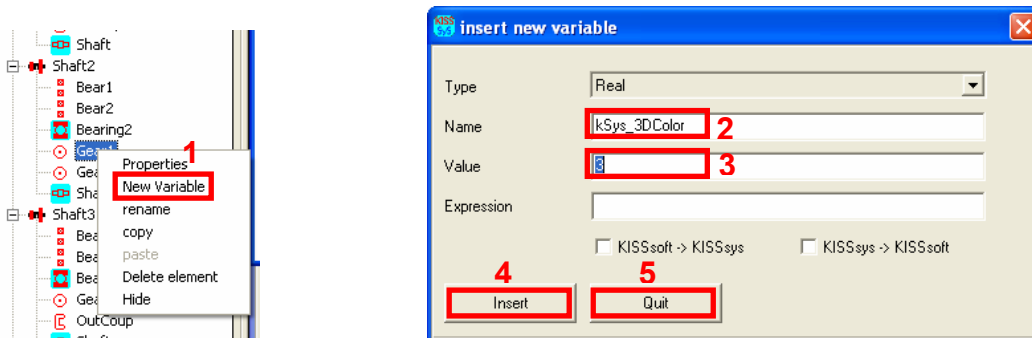


Abbildung 3-2 Variable $kSys_3DColor$ Definieren

Nach einem *Refresh All* wird das Rad grün angezeigt. Die Variable besteht bereits in KISSsys, muss aber durch den oben stehenden Vorgang aktiviert werden.

Die Möglichkeit der Farbänderung besteht für alle Elemente welche in der 3D Repräsentation sichtbar sind. Diese Farbänderung wird bei anderen Elementen gleich durchgeführt wie bei den Zahnradern.

3.2 Farbcodes

KISSsys kann 257 verschiedene Farben generieren welche von 0 – 256 nummeriert sind. Die Farben 0-8 sind aber die geläufigsten und deshalb hier kurz aufgelistet:

Code	Farbe
0	Schwarz
1	Rot
2	Gelb
3	Grün
4	Cyan
5	Blau
6	Pink
7	Weiss
8	Grau

Andere Farbtöne liegen zwischen 9 und 256.

3.3 Integration in ein User Interface

Es ist möglich die Farbwahl im User Interface zu integrieren. Dies kann mit einem Roll-Down Menu geschehen, welches die zu wählenden Farben enthält.

Für jedes Element welches eine Farbänderung durch den normalen User erfahren soll, muss eine Liste mit den zu wählenden Farben erstellt werden. Die Wahl des Listeninhalts wird vom User Interface gesteuert und die Liste dann im Element unter Ausdruck, der Variable *kSys_3DColor* eingelesen.

3.3.1 Erstellen der Liste

Um die Liste zu erstellen muss eine neue Variable in Listenform definiert werden. Mit *Rechtsklick*, zum Beispiel auf das Tabellenicon, des User Interfaces, in der Baumstruktur und

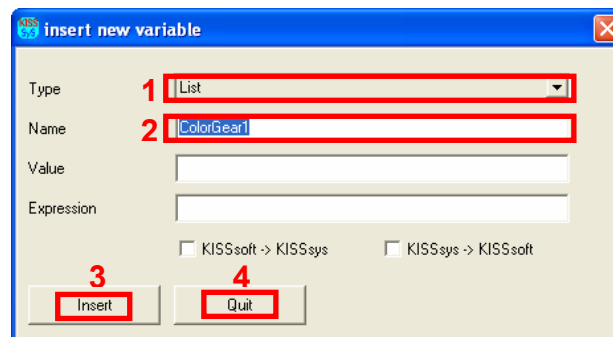


Abbildung 3-3 Listenerstellung

der Anwahl von *neue Variable*, wird der Dialog zur Variabelerstellung geöffnet. Hier muss die Variabelart auf *Liste* gesetzt werden. Der Name der Variable wird eingegeben und das Fenster wird nach Drücken von *Einfügen* mit *Beenden* geschlossen.

Die Liste ist nun erstellt, der Inhalt fehlt aber noch. Mit einem *Rechtsklick* auf das Tabellenicon des User Interfaces und der Wahl von *Eigenschaften* werden die Variablen sichtbar, welche unter dieser Tabelle hinterlegt sind. Ein erneuter *Rechtsklick* auf die Variable *ColorGear1* und die Wahl von *Liste bearbeiten* bringt den *Bearbeitungsdialog für Listen* zum Erscheinen:

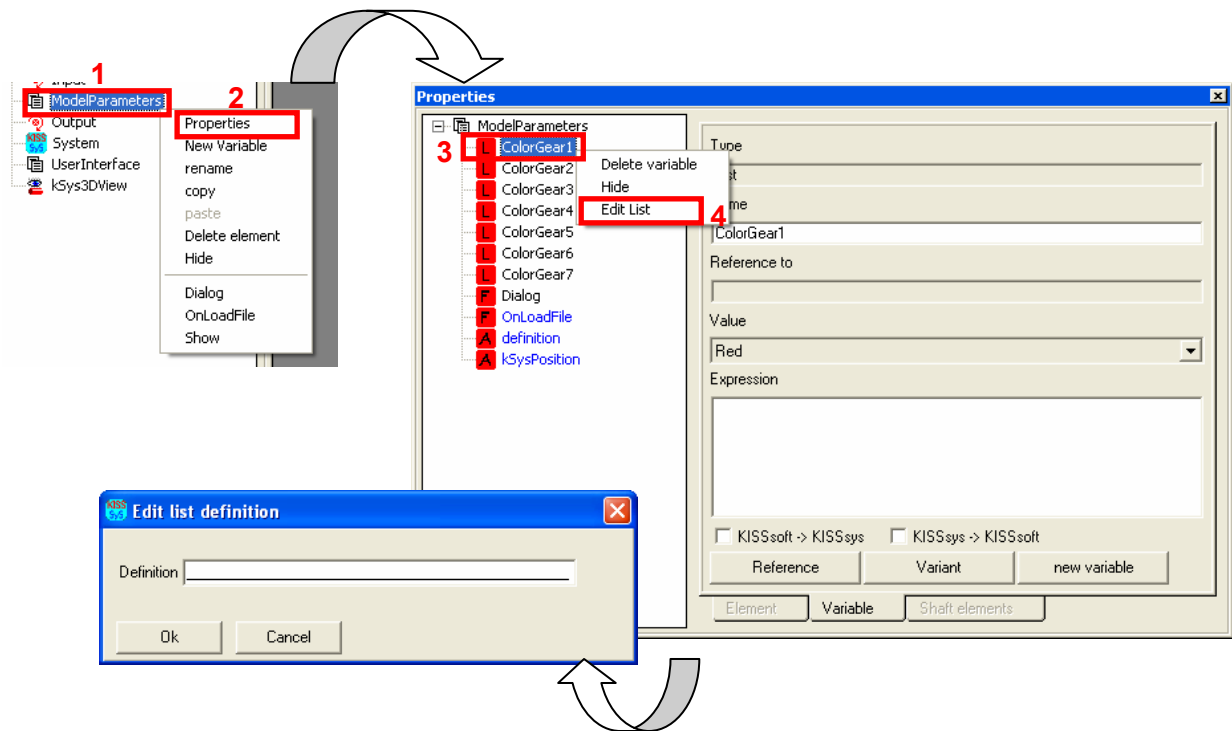


Abbildung 3-4 Aufrufen der Listeneditierung

Im Bearbeitungsdialog wird nun die Liste eingegeben. Die Liste ist mit eckigen Klammern begrenzt und die Positionen werden mit Kommas getrennt. Strings werden zwischen Anführungszeichen gesetzt. Um nun eine Liste der wichtigsten neun Farben zu erstellen muss folgendes in den Bearbeitungsdialog eingegeben werden:

```
[ "Black", "Red", "Yellow", "Green", "Cyan", "Blue", "Pink", "White", "Grey" ]
```

Nach der Quittierung des Dialoges mit *Ok*, sollte im Feld *Value* eine Liste erscheinen, welche die Farben in der gleichen Reihenfolge wie der oben eingegebene Wert aufführt:

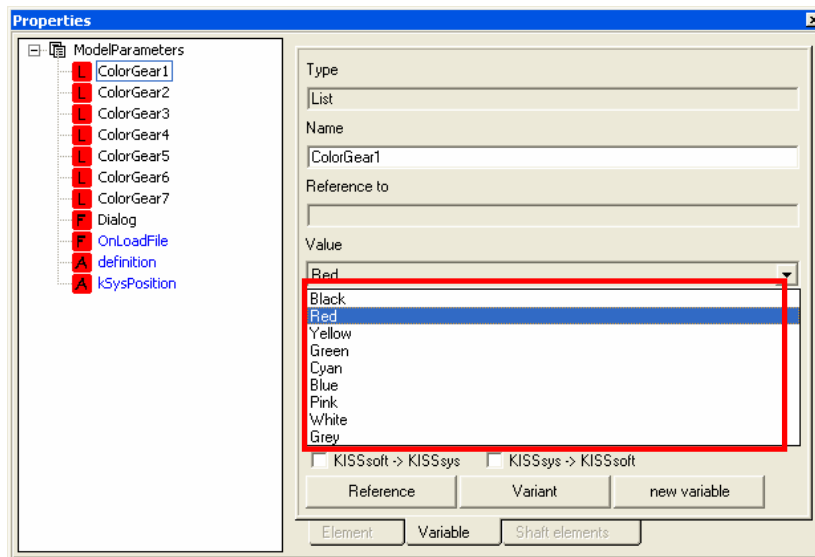


Abbildung 3-5 Liste der möglichen Farbwahlen

3.3.2 Liste Referenzieren

Um die Eingabe im User Interface möglich zu machen, muss in einer Zelle eine Liste eingefügt werden (*Rechtsklick* auf gewünschte Zelle und *Liste einfügen*). Hier wird das Feld zur Variable *ColorGear1* referenziert, indem man auf *Reference* klickt und im Feld *Reference to*, den Pfad der gewünschten Variable zwischen Anführungszeichen eingibt (hier *ModelParameters.ColorGear1*).

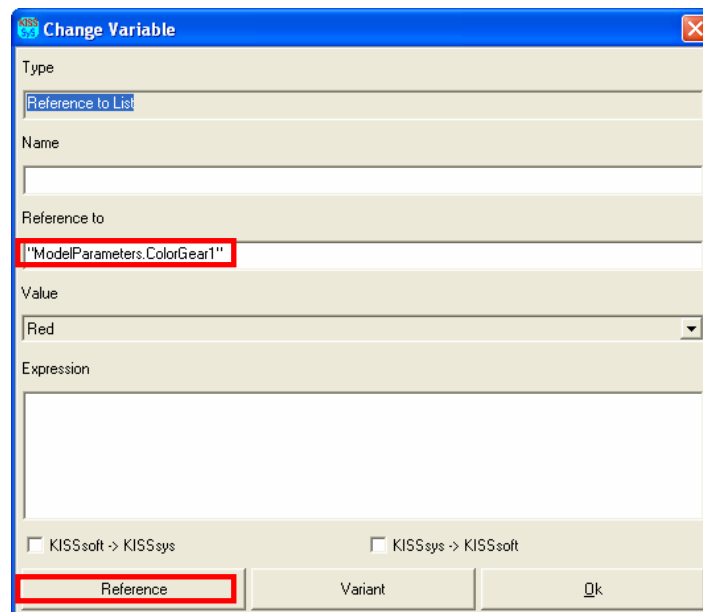


Abbildung 3-6 Referenzieren der Liste

Nach der Quittierung des Fensters mit *Ok*, muss es möglich sein mit einem Doppelklick auf das Fenster, mit der Referenzierung, die ganze Liste anzuzeigen:

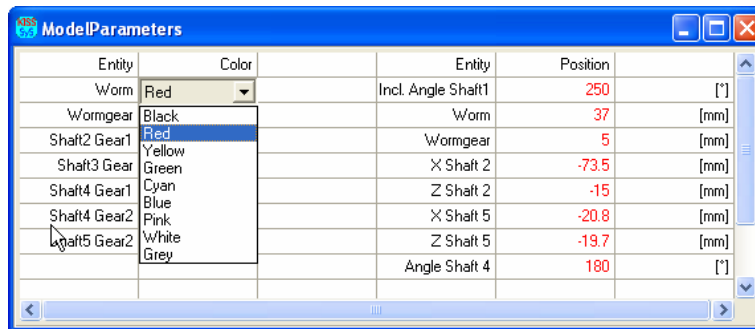


Abbildung 3-7 Listenauswahl im User Interface

3.3.3 Listenwert Einsetzen

Bei der Anwahl des jeweiligen Listenwerts wird nun die Position in der Liste zurückgegeben, und nicht der Wert derselben. Es ist deshalb möglich die Variable einfach in den Ausdruck der Variable $kSys_3DColor$ einzugeben. Rechtsklick auf das Element welches die Farbe zugeordnet erhält, *Eigenschaften* wählen, dann in der Variable $kSys_3DColor$ und unter dem Ausdruck den Pfad der Liste einfügen ($ModelParameters.ColorGear1$).

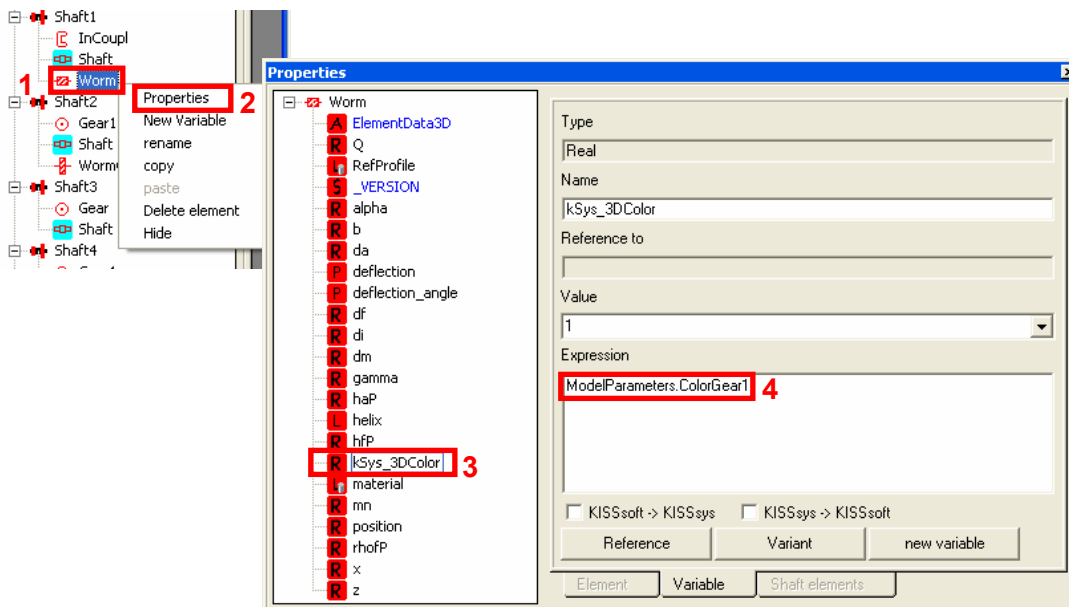


Abbildung 3-8 Variablenwert einsetzen

4 Transparenzfaktor

Mit der Festlegung des Transparenzfaktors $kSys_3DTransparency$ wird die Lichtdurchlässigkeit eines Elementes gesteuert. Dies ist manchmal hilfreich um festzustellen was hinter einem Element passiert.

4.1 Beispiel Zahnrad

Bei vorliegendem Model ist zum Beispiel nicht sichtbar, ob das Rad1 auf der Abtriebswelle die Zwischenwelle berührt.

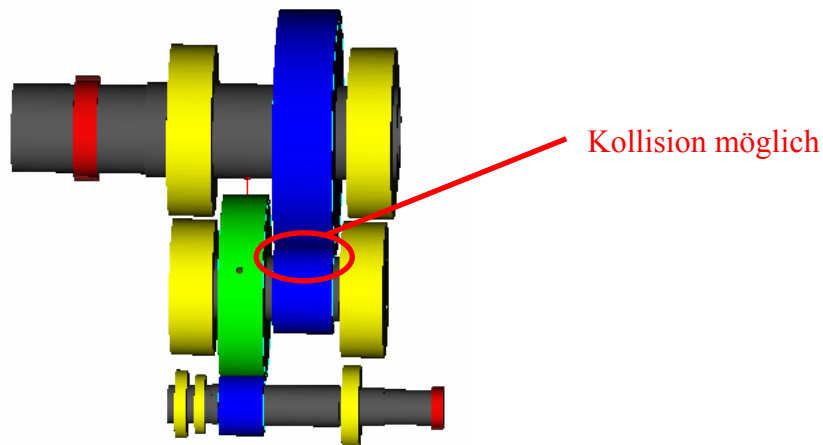


Abbildung 4-1 Kollisionsgefahr bei Vorliegendem Modell

Hier kann es von Vorteil sein, die Lichtdurchlässigkeit von Rad2 auf der Zwischenwelle zu steuern, bzw. es unsichtbar zu machen. Hier muss auf dem gewünschten Element eine Variable *kSys_3DTransparency* eingefügt werden. Der Vorgang für die Erstellung einer neuen Variablen wurde in Kapitel 3.1 bereits erklärt. Der Name der Variablen sowie deren Wert ändern nun aber:

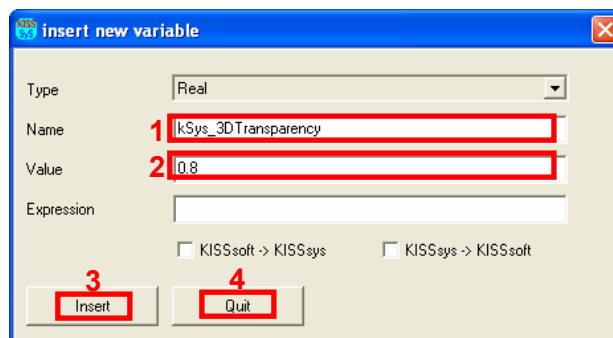


Abbildung 4-2 Definition Transparenzvariable

Der Wert der Variable geht von 0 bis 1, wobei 0 lichtundurchlässig und 1 durchsichtig ist. Nach dem Drücken des *Refresh All* Knopfes kann man sich vergewissern dass im kritischen Bereich keine Kollision vorliegt:

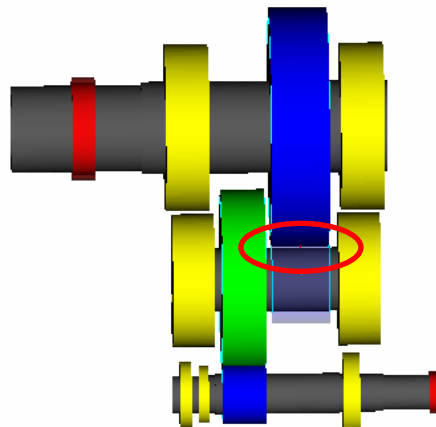


Abbildung 4-3 Modell mit Beinahe Transparentem 3. Zahnrad

Die Transparenzen können ebenfalls für alle Elemente angewendet werden welche in 3D repräsentiert werden können.

5 Grafische Darstellung

Es gibt zwei unterschiedliche Darstellungen der Elemente in der 3D Repräsentation.

5.1 Graphic Elements

Graphic Elements stellt die Elemente nur grafisch und rudimentär dar. Die Darstellung reicht aber meist aus um Kollisionen zu erkennen und erfordert keine lange Rechenzeit beim Auffrischen der Modelle. Stirnräder werden als Zylinder mit Aussendurchmesser (entspricht Kopfkreisdurchmesser) dargestellt.

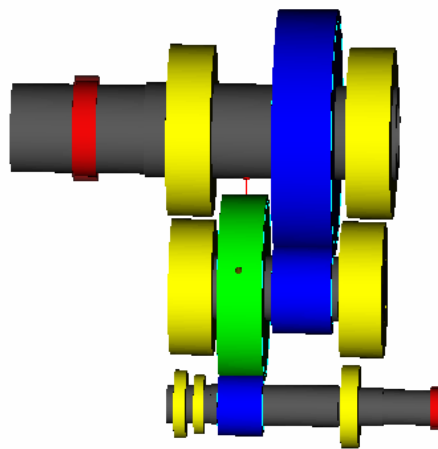


Abbildung 5-2 Modell aus Grafischen Elementen

5.2 Solid Elements

Solid Elements stellt die Elemente in einer detaillierten Art dar. Diese Darstellung kommt der Realität näher als die Darstellung mit grafischen Elementen, die Zahnform z. B. wird vereinfacht dargestellt. Diese Einstellung braucht Rechenzeit beim Auffrischen und sollte nur gewählt werden, wenn das Model mehr repräsentativ oder zur Kollisionskontrolle eingesetzt wird. Diese Darstellung kann auch Hilfreich sein wenn es um die Erkennung des Schrägungswinkels bei Stirnrädern geht. Die Verzahnung von Kegelrädern wird nicht dargestellt.

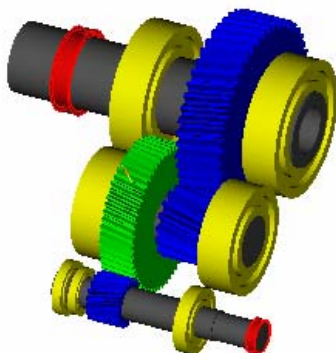


Abbildung 5-1 Modell aus Soliden Elementen

5.3 Einstellung

Um die Art der Grafischen Ansicht in der 3D Repräsentation aus zu wählen muss wie folgt vorgegangen werden:

Rechtsklick auf *System*, *Properties* und Anwahl der Variable *kSys3DElements*, hier kann im Rolldown Menu unter *Value* zwischen *Graphic Elements* und *Solid Elements* gewählt werden.

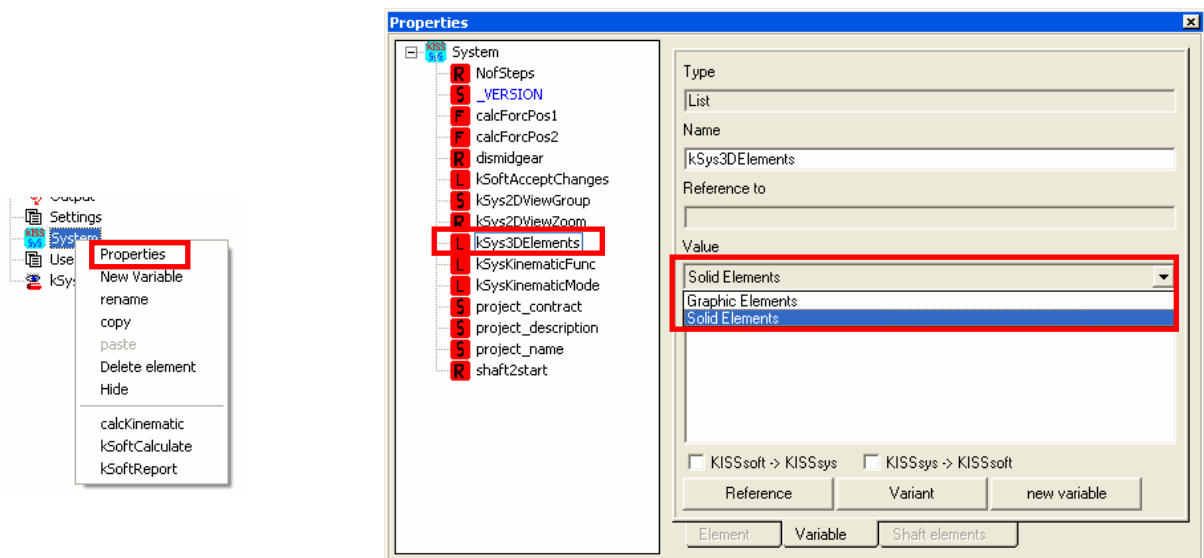


Abbildung 5-3 Einstellen der Darstellungsart

Nach einem *Refresh All* wird die Änderung sichtbar.

6 Kollisionsprüfer

KISSsys verfügt über einen automatischen Kollisionsprüfer. Dieser arbeitet nur in der Einstellung *Solid Elements* und ist rechenintensiv. In folgendem Modell ist es beispielsweise nicht mehr möglich, visuell zu bestimmen ob das Rad1 auf der Zwischenwelle mit dem Lager kollidiert:

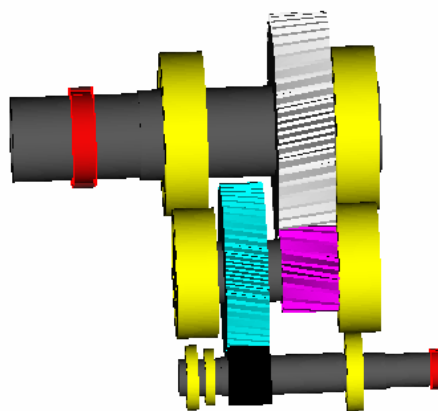


Abbildung 6-1 Modell mit Möglicher Kollision

Hier ist der Einsatz des Kollisionsprüfers sinnvoll.

6.1 Benützung

Ein anschliessender *Rechtsklick* auf *kSys3DView* und die Wahl von *CheckCollision* führt zur Aktivierung des folgenden Fensters:

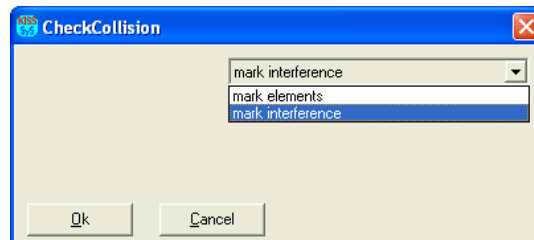


Abbildung 6-2 Kollisionsprüfer

Im Rolldownmenu kann gewählt werden ob das Programm die kollidierenden Elemente (*mark elements*) oder den Bereich der Kollision (*mark interference*) anzeigt. Nach dem Klicken von *Ok*, wird gerechnet und die allfälligen Kollisionen angezeigt. Falls keine Kollision besteht, passiert nach einer ganz kurzen Rechenphase nichts weiter. Falls Kollision vorliegt, werden die kollidierenden Elemente oder Oberflächen in grüner Farbe und transparent angezeigt:

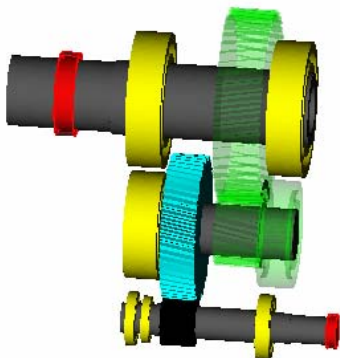


Abbildung 6-3 Mark Elements

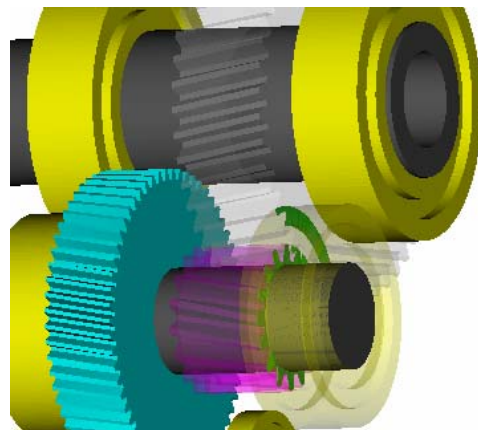


Abbildung 6-4 Mark Interference

Mit einem *Refresh All* wird das ursprüngliche Model wieder sichtbar gemacht.