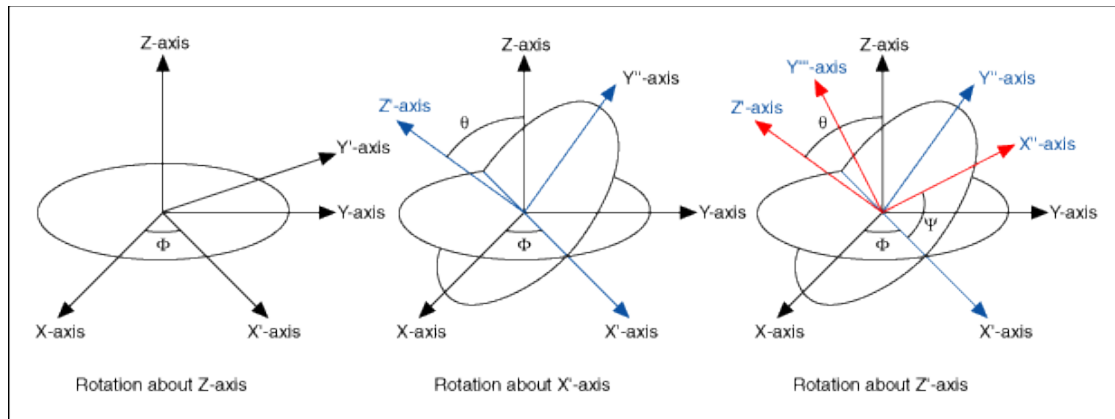


Euler vs. Kvaternió

Rutin készítése az orientáció kvaternió értékeinek euler szögre történő átszámítására és vissza.



A robot programozási nyelvekben akár egy pontról (rapidban robtarget), egy szerszámról (rapidban tool), vagy koordináta rendszerről (rapidban workobject) van szó, és egyéb más helyeken is (pl.: robot terhelés, loaddata) az orientációs információk megadásához jellemzően az Euler szögek hármas egységét használják. Ettől eltérően az ABB rapid programnyelve a kvaterniókat használja ezen orientációs információ megadására. Ennek vannak bizonyos előnyei, de hátrányos lehet, ha manuálisan akarjuk a pont vagy koordináta rendszer orientációját beállítani. Itt jelenthet segítséget az a program, mely a két rendszer között elvégzi a szükséges konverziókat.

Az ABB rapid nyelvén megadott orientációs információ egy ORIENT típusú ROT (rotation) nevű változóban kerül megadásra, mely változó egy RECORD, ami 4 db elemi NUM (numeric) típusú változóból áll. Ezen változók nevei sorrendben q1, q2, q3, q4.:

```
CONST robtargget myRobtarget := [[100,200,300],[1,0,0,0],[0,0,0,0],[9E9,9E9,9E9,9E9,9E9,9E9]];
```

```
Position data
RECORD robtargget
  pos trans;
  orient rot;
  confdata robconf;
  extjoint extax;
ENDRECORD
```

```
CONST robtargget myRobtarget := [[100,200,300],[1,0,0,0],[0,0,0,0],[9E9,9E9,9E9,9E9,9E9,9E9]];
```

```
robtargget.orient.q1
```

A robtarget változóban található orientációs adatokat (ésminden más is) módosítani csak akkor lehet, ha a robtarget módosítható, vagyis PERS (persistent) típusúként van definiálva.

```
PERS robtarget myRobtarget:=|
```

Amint ez biztosított, az adatokat a következő módon érhetjük el:

```
q1:=myrobtarget.rot.q1;  
q2:=myrobtarget.rot.q2;  
q3:=myrobtarget.rot.q3;  
q4:=myrobtarget.rot.q4;
```

Erre szükség lesz, ha a kiszámított értékeket szeretnénk a robtargetben alkalmazni.

Kvaterniókkal megadott orientációs adatokból visszanyerni az Euler szögeket a következő módon lehet:

```
PROC euler(robtarget position)  
  
  VAR num q1;  
  VAR num q2;  
  VAR num q3;  
  VAR num q4;  
  
  var num ff;  
  var num fs;  
  var num sf;  
  var num tf;  
  var num ts;  
  
  q1:=position.rot.q1;  
  q2:=position.rot.q2;  
  q3:=position.rot.q3;  
  q4:=position.rot.q4;  
  
  ! roll (x-axis rotation)  
  ff:=2*(q1*q2+q3*q4);  
  fs:=1-2*(q2*q2+q3*q3);  
  a:=atan2(ff,fs);  
  
  ! pitch (y-axis rotation)  
  sf:=2*(q1*q3-q4*q2);  
  b:=asin(sf);  
  
  ! yaw (z-axis rotation)  
  tf:=2*(q1*q4+q2*q3);  
  ts:=1-2*(q3*q3+q4*q4);  
  c:=atan2(tf,ts);  
  
  kiir:=numtostr(a,5)+" , "+numtostr(b,5)+" , "+numtostr(c,5);  
  TPWrite kiir;  
  quater:=numtostr(q1,5)+" , "+numtostr(q2,5)+" , "+numtostr(q3,5)+" , "+numtostr(q4,5);  
  TPWrite quater;  
  
ENDPROC
```

A rutin bemenő paramétere egy rotarget, mely tartalmazza azt az orientációs változót melyet Euler szögek formájában szeretnénk visszakapni. Első lépésben a bemeneti pontból kiolvassuk az orientációs adatokat, melyeket változóban tárolunk. Az átmeneti változók felhasználásával kiszámoljuk a megfelelő Euler szögeket melyek a kimeneten az **a**, **b**, **c** numerikus változóban érhetőek el.

A rutin a kezelőpanelra kiírja a bemenetként kapott kvaterniókat és az eredmény Euler szögeket is.

Euler szögekkel megadott orientációs adatokat kvaterniókká alakítani, a következő módon lehet:

```
PROC quatern(num r,num p,num y)

    ! roll (X), pitch (Y), yaw (Z)

    ! cos and sin Yaw
    VAR num cy;
    VAR num sy;

    ! cos and sin Pitch
    VAR num cp;
    VAR num sp;

    ! cos and sin Roll
    VAR num cr;
    VAR num sr;

    cy:=cos(y*0.5);
    sy:=sin(y*0.5);
    cp:=cos(p*0.5);
    sp:=sin(p*0.5);
    cr:=cos(r*0.5);
    sr:=sin(r*0.5);

    !QW = rapid Q1

    !MATRIX
    qw:=(cy*cp*cr)+(sy*sp*sr);
    qx:=(cy*cp*sr)-(sy*sp*cr);
    qy:=(sy*cp*sr)+(cy*sp*cr);
    qz:=(sy*cp*cr)-(cy*sp*sr);

    quater:=numtostr(qw,5)+" , "+numtostr(qx,5)+" , "+numtostr(qy,5)+" , "+numtostr(qz,5);
    TPWrite quater;

ENDPROC
```

A rutinnak három numerikus paramétere van, melyen keresztül adhatok meg az egyes tengelyek szerinti orientációs adatok. A rutin kimenetként megjelenő négy (qw, qx, qy, qz) változó érték hordozza a bemeneti adatoknak megfelelő kvaterniókat.

A rutin a kezelőpanelra kiírja az eredmény kvaterniókat.