```
restart

with(LinearAlgebra):
with(VectorCalculus):
with(CodeGeneration):
```

Coordinate system origins

A - Origin at inital pivot point at the end of Link 1
B - Origin at Hip
C - Origin at impact point at the end of Link 2

Distance and velocities for Link 1 about Hip

$$r1_B := Vector([-a\cdot\sin(-q1), -a\cdot\cos(-q1), 0]):$$
$$r1_A := Vector([b\cdot\sin(-q1), b\cdot\cos(-q1), 0]):$$
$$rH_C := Vector([-l\cdot\sin(q2), l\cdot\cos(q2), 0]):$$
$$v1_- := CrossProduct(Vector([0, 0, Dq1_-]), r1_A):$$
$$v1_+ := CrossProduct(Vector([0, 0, Dq1_+]), r1_B) + CrossProduct(Vector([0, 0, Dq2_+]), rH_C):$$

```
BasisFormat(false):
```

#Distance and velocities for complete biped about Impact

$$r2_C := Vector([-b\cdot\sin(q2), b\cdot\cos(q2), 0]):$$
$$r1_C := rH_C + r1_B:$$
$$rH_A := Vector([l\cdot\sin(-q1), l\cdot\cos(-q1), 0]):$$
$$r2_B := Vector([a\cdot\sin(q2), -a\cdot\cos(q2), 0]):$$

$$vh_- := CrossProduct(Vector([0, 0, Dq1_-]), rH_A):$$
$$v2_- := CrossProduct(Vector([0, 0, Dq2_-]), r2_B) + vh_-:$$

$$v2_+ := CrossProduct(Vector([0, 0, Dq2_+]), r2_C):$$
$$vh_+ := CrossProduct(Vector([0, 0, Dq2_+]), rH_C):$$

#Angular momentum

$$AMt := CrossProduct(r1_B, m1\cdot(v1_+)) = CrossProduct(r1_B, m1\cdot(v1_-)):$$
$$AMr := CrossProduct(r1_C, m1\cdot(v1_+)) + CrossProduct(r2_C, m2\cdot(v2_+)) + CrossProduct(rH_C, mH\cdot(vh_+)) = CrossProduct(r1_C, m1\cdot(v1_-)) + CrossProduct(r2_C, m2\cdot(v2_-)) + CrossProduct(rH_C, mH\cdot(vh_-)):$$

#Simplify expressions

$$AMtsimp := lhs(AMt(3)) - rhs(AMt(3)):$$

```
AMtsimp := simplify(AMtsimp,'trig') :
AMtsimp := collect( algsubs(sin(q1)·sin(q2) + cos(q1)·cos(q2) = cos(q2 − q1), AMtsimp),
    [Dq1_+, Dq2_+, Dq1_-, Dq2_-]);

AMrsimp := lhs(AMr(3)) − rhs(AMr(3)) :
AMrsimp := simplify(AMrsimp,'trig') :
AMrsimp := collect( algsubs(sin(q1)·sin(q2) + cos(q1)·cos(q2) = cos(q2 − q1), AMrsimp),
    [Dq1_+, Dq2_+, Dq1_-, Dq2_-]);
```

#Formulating map

```
A0 := Matrix(2, 2) :
B0 := Matrix(2, 2) :

for i from 1 to 2 do
  A0(1, i) := subs( Dq1_+ = i mod 2, Dq2_+ = i − 1 mod 2, Dq1_- = 0, Dq2_- = 0, AMrsimp) :
  A0(2, i) := subs( Dq1_+ = i mod 2, Dq2_+ = i − 1 mod 2, Dq1_- = 0, Dq2_- = 0, AMtsimp) :
end do:


for i from 1 to 2 do
  B0(1, i) := -subs( Dq1_- = i mod 2, Dq2_- = i − 1 mod 2, Dq1_+ = 0, Dq2_+ = 0, AMrsimp) :
  B0(2, i) := -subs( Dq1_- = i mod 2, Dq2_- = i − 1 mod 2, Dq1_+ = 0, Dq2_+ = 0, AMtsimp) :
end do:
```

#Inserting relabeling of legs

```
Perm := Matrix([[0, 1], [1, 0]]);
Qp := Q_+ = MatrixMatrixMultiply(A0, Perm);
Qn := Q_- = B0;
```

#Simplify

```
par6 := m1 l^2 + l^2 mH + b^2 m2 :
par7 := a m1 l :
par8 := a^2 m1 :

Qp0 := algsubs( par6 = p6, Qp) :
Qp0 := algsubs( par7 = p7, Qp0) :
Qp0 := algsubs( par8 = p8, Qp0) :

Qp0;

par9 := a m1 b :
par10 := b m2 l + l^2 mH + m1 b l :
par11 := b m2 a :
```

```
Qn0 := algsubs(par9=p9, Qn) :
Qn0 := algsubs(par10=p10, Qn0) :
Qn0 := algsubs(par11=p11, Qn0) :

Qn0;
```

#Creating Matlab code

```
Matlab(rhs(Qp0), resultname='Qp');
Matlab(rhs(Qn0), resultname='Qn');

Matlab(par6, resultname='p6');
Matlab(par7, resultname='p7');
Matlab(par8, resultname='p8');
Matlab(par9, resultname='p9');
Matlab(par10, resultname='p10');
Matlab(par11, resultname='p11');


save Qp0, Qn0, "impact_map.m";
```