model {

#Priors

for(i in 1:16){

bet[i] ~ dt(0,pow(1.566267,-2),7.63179)

}

for(i in 1:7){

alpha[i] ~ dt(0,pow(1.566267,-2),7.63179)

}

for(i in 1:5){

phi[i] ~ dt(0,pow(1.566267,-2),7.63179)

}

#Occurence model

for(i in 1:n){

z[i] ~ dbern(psi[i])

psi[i] <- exp(logit.psi[i])/(1+exp(logit.psi[i]))

logit.psi[i] <- bet[1] + bet[2]\*year[i] + bet[3]\*depth[i,1] + bet[4]\*depth2[i,1] +

bet[5]\*lat[i,1] + bet[6]\*lat2[i,1] + bet[7]\*temp[i,1] + bet[8]\*temp2[i,1] +

bet[9]\*livebot.l[i] + bet[10]\*livebot.m[i] + w[10]\*bet[11]\*livebot.h[i] +

bet[12]\*hardsub.l[i] + bet[13]\*hardsub.m[i] + bet[14]\*hardsub.h[i] +

bet[15]\*relief.m[i] + bet[16]\*relief.h[i]

#Detection model (Chevron trap)

y[i,1] ~ dbern(z[i]\*p[i,1])

p[i,1] <- exp(logit.p[i,1])/(1+exp(logit.p[i,1]))

logit.p[i,1] <- alpha[1] + alpha[2]\*temp[i,1] + alpha[3]\*temp2[i,1] + alpha[4]\*soak[i,1] +

alpha[5]\*cdir.p[i,2] + alpha[6]\*cdir.a[i,1] + alpha[7]\*cspeed[i,1]

#Detection model (Cammera trap)

y[i,2] ~ dbern(z[i]\*p[i,2])

p[i,2] <- exp(logit.p[i,2])/(1+exp(logit.p[i,2]))

logit.p[i,2] <- phi[1] + phi[2]\*turb[i,2] + phi[3]\*cdir.p[i,2] + phi[4]\*cdir.a[i,2] +

phi[5]\*cspeed[i,2]

} #end loop i

}