//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//

//The following is the code for DarkSabre's Mission Possible device at the 2005 National Tournament.

//

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

int RCX=2; // Change this according to which RCX is being programmed

//sensor triggering variables

int ball1=999; //480; //bernoulli ball present \*\*triggers when higher\*\*

int precip1=999; //408; //precipitate present \*\*triggers when higher\*\*

int magFO1=440; //electromag released FO are blocked \*\*triggers when lower\*\*

int heattouch=500; //bimetallic has pressed the sensor \*\*triggers when lower\*\*

int chemLED1=0; //chemical battery LEDs are on \*\*triggers when lower\*\*

int sound1=200; //piezo detection \*\*triggers when lower\*\*

int lacetouch=500; //lace tying trigger \*\*triggers when lower\*\*

//motor timing variables

int motorswitcher=300;

int springsmotor=300;

int chembatterymotor=0;

int ballthrowermotor=1000;

int pneumaticmotor=100;

int precipmotor=0;

int laceknotmotor=2600;

int laceknotmotor2=600;

//special ops variables

int bernoulli=0;

int transformer=0;

int aa=0;

int transformer2=0; //Rapid checker Averager \*\*triggers when lower\*\*

int thejudgefriendlywaitcommand=300;

// true/false transfer markers

int i=0;

int j=0;

int k=0;

int r=0;

task main()

{

 SetPower(OUT\_A && OUT\_B && OUT\_C, OUT\_FULL);

 SetDirection(OUT\_A && OUT\_B && OUT\_C, OUT\_FWD); //cords need to go up!

 SetSensorType(SENSOR\_1, SENSOR\_TYPE\_NONE); //tells the RCX that the sensors are generic

 SetSensorType(SENSOR\_2, SENSOR\_TYPE\_NONE);

 SetSensorType(SENSOR\_3, SENSOR\_TYPE\_NONE);

 SetSensorMode(SENSOR\_1, SENSOR\_MODE\_RAW); //reads sensors in raw data form

 SetSensorMode(SENSOR\_2, SENSOR\_MODE\_RAW);

 SetSensorMode(SENSOR\_3, SENSOR\_MODE\_RAW);

 start Identify;

 if (RCX==1)start RCX1;

 if (RCX==2)start RCX2;

 if (RCX==3)start RCX3;

 if (RCX==4)start RCX4;

}

task RCX1()

{

 while(true)

 {

 if(SENSOR\_1<=magFO1 && i==0) //electromagnet drops ball and FO is blocked

 {

 repeat(50)

 {

 aa=(SENSOR\_2+aa); //sums the ambient readings and stores them as "aa"

 Wait(1);

 }

 precip1=((aa/50)+5); //finds the average of the ambient conditions

 ClearTimer(1); //timer for resetting the precip dumper

 On(OUT\_A); //precip dumper

 until(SENSOR\_2>=precip1)

 precipmotor=(Timer(1)\*10); //sets the reset time in the correct units

 Off(OUT\_A);

 i=1;

 }

 if(SENSOR\_2>=precip1 && j==0) //precipitate is present

 {

 repeat(50)

 {

 aa=(SENSOR\_3+aa); //sums ambient readings and stores them as "aa"

 Wait(1);

 }

 ball1=((aa/50)+20); //finds the average of the ambient conditions

 PlayTone(800,100);

 Wait(thejudgefriendlywaitcommand);

 OnFor(OUT\_B,100); //pneumatic switch

 j=1;

 Float(OUT\_B); //Float puts motors into "neutral"

 }

 if(SENSOR\_3>=ball1 && k==0) //bernoulli ball is present

 {

 while(bernoulli==0)

 {

 ClearTimer(0);

 until(Timer(0)==2||SENSOR\_3<=ball1) //the ball must stay for 2 seconds

 if(SENSOR\_3>=ball1) bernoulli=1;

 }

 PlayTone(800,100);

 Wait(thejudgefriendlywaitcommand);

 OnFor(OUT\_C, springsmotor); //springs motor (release and store)

 k=1;

 Float(OUT\_C);

 }

 if(i==1 && j==1 && k==1 && r==0) //resetting code

 {

 Toggle(OUT\_A); //switches motor direction

 OnFor(OUT\_A,precipmotor); //precip

 r=1;

 }

 }

}

task RCX2()

{

 while(true)

 {

 if(SENSOR\_1<=sound1 && i==0) //piezoelectric microphone

 {

 repeat(50)

 {

 aa=(SENSOR\_2+aa); //averager of ambient conditions

 Wait(1);

 }

 transformer2=(aa/50);

 PlayTone(800,100);

 Wait(thejudgefriendlywaitcommand);

 OnFor(OUT\_A, motorswitcher); //completes transformer circuit

 Float(OUT\_A);

 i=1;

 }

 ClearTimer(1); //for transformer hits counting

 while(i==1 && j==0) //transformer LEDs

 {

 if(SENSOR\_2<=transformer2) transformer=transformer+1;

 if(transformer>=5 && Timer(1)<=3) //triggers if it gets enough hits in the given time

 {

 aa=0;

 repeat(50)

 {

 aa=(SENSOR\_3+aa); //averager

 Wait(1);

 }

 chemLED1=((aa/50)-1);

 ClearTimer(0); //timer for chemical battery resetting

 On(OUT\_B); //completes chemical battery

 until(SENSOR\_3<=chemLED1)//lowers until immersed in vinegar

 Wait(100);

 until(SENSOR\_3<=chemLED1) //makes sure it is in vinegar

 Off(OUT\_B);

 chembatterymotor=(Timer(0)\*10); //sets reset time

 Float(OUT\_B);

 j=1;

 }

 if(transformer<=5 && Timer(1)>=3) //if it doesn't get enough hits in the given time

 {

 ClearTimer(1); //start over

 transformer=0;

 }

 Wait(1);

 }

 if(SENSOR\_3<=chemLED1 && k==0) //chemical battery LED

 {

 OnFor(OUT\_C, ballthrowermotor); //launches ball

 Float(OUT\_C);

 k=1;

 }

 if(i==1 && j==1 && k==1 && r==0) //resetting code

 {

 Toggle(OUT\_B);

 OnFor(OUT\_B, chembatterymotor); //resets chemical battery

 r=1;

 }

 }

}

task RCX3()

{

 while(true)

 {

 if(SENSOR\_1<=heattouch && i==0) //increasing/decreasing heat

 {

 PlayTone(800,100);

 OnFor(OUT\_A, motorswitcher); //turns NiCr off

 Wait(100);

 until(SENSOR\_1>=heattouch); //until it is released

 PlayTone(800,100);

 OnFor(OUT\_C, motorswitcher); //completes buzzer circuit

 On OUT\_B; //rattle

 Wait(30);

 repeat (40)

 {

 Toggle OUT\_B; //shakes rattle for 12.3 seconds total

 Wait(30);

 }

 Off(OUT\_B);

 i=1;

 Float(OUT\_A && OUT\_B && OUT\_C);

 }

 }

}

task RCX4()

{

 while(true)

 {

 if(SENSOR\_1<=lacetouch && i==0) //lace tying trigger

 {

 Toggle(OUT\_A);

 Toggle(OUT\_B);

 OnFor(OUT\_A, laceknotmotor); //intial pulling

 OnFor(OUT\_B, pneumaticmotor); //removes the pin

 OnFor(OUT\_A, laceknotmotor2); //finishes the knot

 i=1;

 Float(OUT\_A && OUT\_B);

 }

 }

}

task Identify()

{

 Wait(100);

 repeat(RCX)

 {

 PlayTone(800,20);

 Wait(35);

 }

}