

نظام تحكم لخط انتاج يقوم بفرز الصناديق حسب الحجم

فكرة النظام:

تصميم خط انتاج ذكي قادر على فرز الصناديق بدون تدخل البشر في العملية.

هدف النظام:

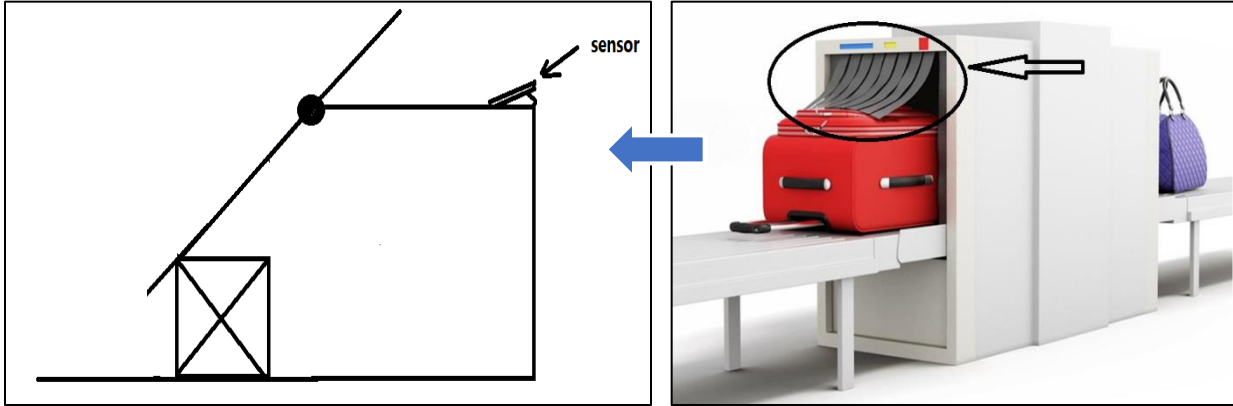
فرز الصناديق التي تمر على خط الانتاج إلى عدة تصنيفات حسب احجامها بحيث يذهب إلى صندوق كل صندوق انتاج فرعية بناء على الحجم

آلية عمل النظام:

سيكون هناك مستشعر موجات صوتية ألتراسونيك يقوم بفرز الصناديق بطريقة معينة وحسب حجم الصندوق ستقوم عدة محركات سيرفو على خط الإنتاج بتغيير زاويتها ليتجه الصندوق الى خط الإنتاج المراد.

خطوات عمل النظام:

أولاً: نقوم بتصميم شكل مقارب للصور التالية

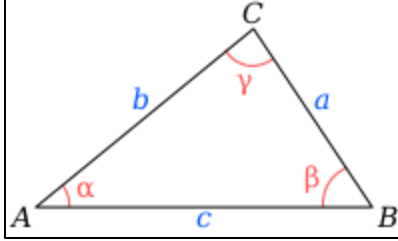


واعترض عن جودة الصور فلأسف لا أجيد تصميم الثري دي ولكن الفكرة مشابهة لحد كبير عدا أن الأشرطة الموجودة بالصورة ستستبدل بذراع طويل يمتد للأعلى مع قاعدة تثبيت بحركة دورانية ويكون الذراع عريض بما يكفي لان يستطيع مستشعر الالتراسونيك التقاط حركته وستكون الفكرة كالتالي:

أولاً يدخل الصندوق عبر الحزام الناقل ويمر بالنظام الذي يحتوي المستشعر بالتالي عند خروج الصندوق سيقوم برفع الذراع والتي سترتفع بزاوية أكبر بناء على حجم الصندوق وبالأعلى سنقوم بتثبيت مستشعر الالتراسونيك وسنثبته على زاوية 15 درجة، وسيقوم المستشعر بحساب المسافة بينه وبين الذراع بحيث كلما قلت المسافة دل هذا على أن حجم الصندوق أكبر وبالتالي يمكننا فرز الصناديق بناء على الحجم بدقة عالية نظراً لدقة المستشعر.

بالنسبة للمعادلة المستخدمة لتحديد حجم الصندوق فهي كالتالي:

أولا المستشعر سيكون بزاوية 15 وسيعطينا القراءة ولنفرضها بالقيمة (أ) ونحن نعلم مسبقاً بالمسافة من المستشعر حتى قاعدة تثبيت الذراع فهذا سيتكون لدينا مثلث بمعلومية ضلعين وزاوية وبالتالي نستطيع استخدام قانون جيب التمام:



حيث في حالتنا فنحن نعرف قيمة الزاوية بيتا والضلعين أي وسي.

$$b^2 = a^2 + c^2 - 2ac \cos \beta$$

وباستعمال القانون نستطيع إيجاد باقي الزوايا.

سوف نقوم بعد ذلك بافتراض عدة قيم للقيمة (أ) – والتي تمثل المسافة التي حصلنا عليها من المستشعر – لكي نجد القيمة المناسبة لفرز الصناديق.

أولاً لنفرض أن مساحة النظام الذي تدخل عبره الصناديق عبارة عن 60 سم في 60 سم وسنفرض المسافة بين المستشعر وقاعدة تثبيت الذراع = 60 سم وزاوية تثبيت المستشعر تساوي 15.

بناء على هذه الافتراضات فإن أعلى قيمة للمستشعر (أ) ستكون 62 سم عندما يكون الذراع قائم الزاوية

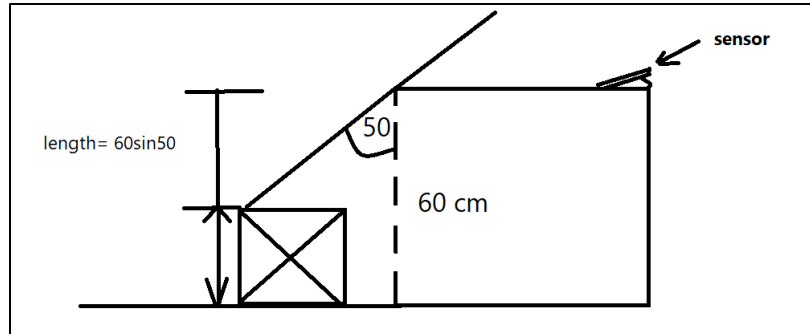
وبالتالي يمكننا أن نفرز الاحجام بناء على هذه المعلومة فمثلاً:

1- لو فرضنا قيمة (أ) تساوي 30 سم أي أن المستشعر حسب المسافة بينه وبين الذراع 30 سم وبما أن المسافة حتى قاعدة التثبيت = 60 سم فسنستعمل قانون جيب التمام لحساب الزاوية العلوية والتي في حالتنا ستكون (14 درجة) بينما زاوية الذراع السفلية ستكون متممة للزاوية العلوية بالتالي ستكون 76 درجة أي أننا نستطيع القول أن الصندوق ذو حجم كبير بناء على الزاوية التي تحركت بها الذراع

2- لنفرض القيمة (أ) تساوي 45 سم حينها ستكون قيمة الزاوية العلوية 35 والزاوية السفلية ستكون 55 أي أن الصندوق تقريبا متوسط الحجم

3- لنفرض القيمة (أ) تساوي 55 سم حينها ستكون قيمة الزاوية العلوية 62 والزاوية السفلية ستكون 28 أي أن الصندوق تقريبا صغير الحجم

ويمكن فرز الصناديق بدقة أكثر فمثلا لو كنت تعرف الزاوية يمكنك استنتاج حجم الصندوق الفعلي ببعض الحسابات فمثلا كانت الزاوية السفلية تساوي 50 درجة وبمعلومية المسافة من الحزام الناقل وحتى مثبت الذراع والتي تساوي 60 سم وبما أن المثلث قائم الزاوية يمكن إيجاد الزاوية الأخيرة وبالتالي حساب كامل الاضلاع



بعد إيجاد القيم المناسبة للمستشعر نبدأ ببرمجة الكود على الاردوينو كالتالي:

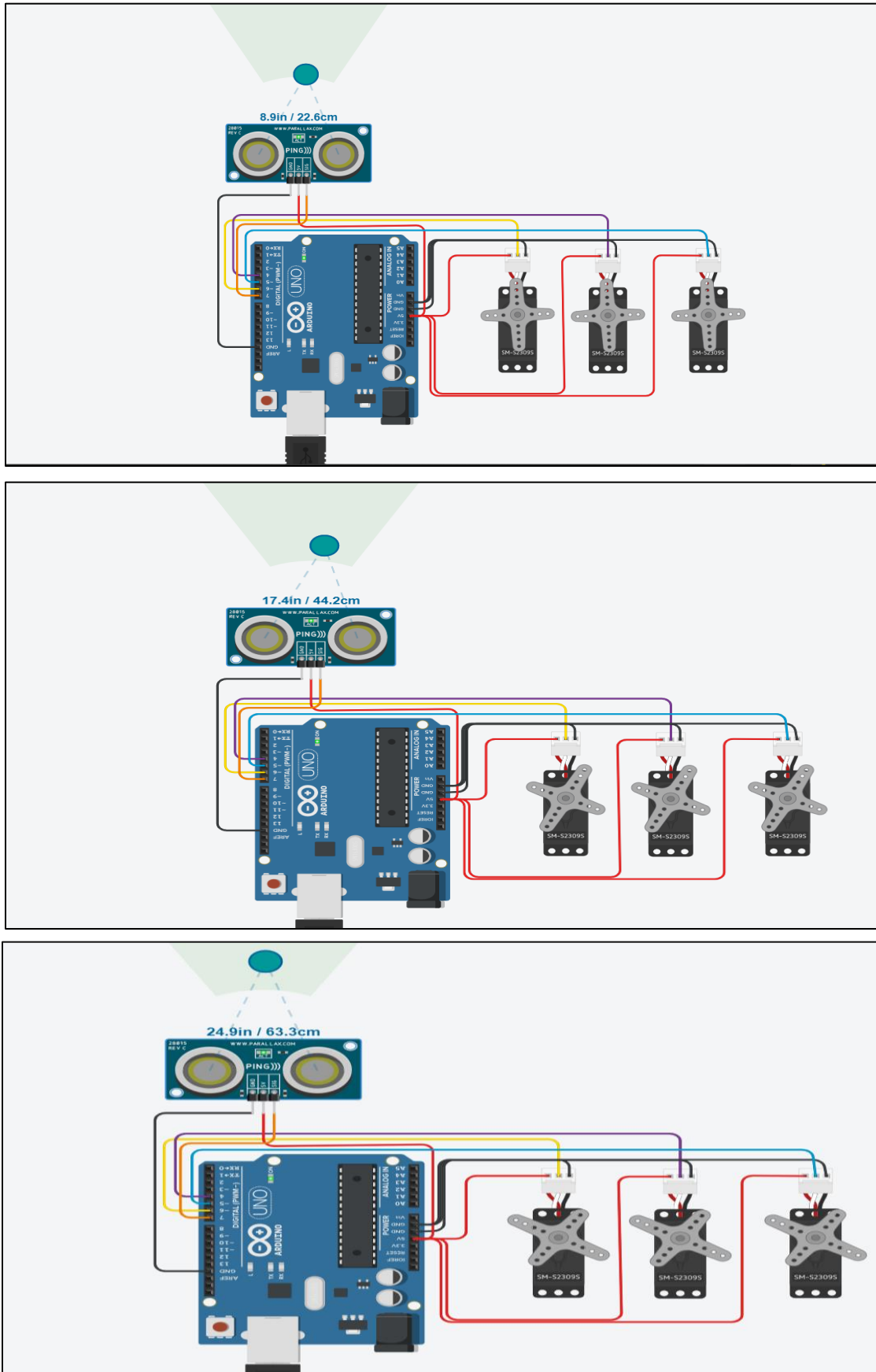
```

Text
1 #include <Servo.h>
2 Servo motor1 ;
3 Servo motor2 ;
4 Servo motor3 ;
5 int x=0;
6
7 void setup() {
8   motor1.attach(6);
9   motor2.attach(5);
10  motor3.attach(4);
11  motor1.write(90);
12  motor2.write(90);
13  motor3.write(90);
14  Serial.begin(9600);
15  delay(2000);
16
17  void loop() {
18    pinMode(7, OUTPUT);
19    digitalWrite(7,0);
20    delay(50);
21    digitalWrite(7,1);
22    delayMicroseconds(2);
23    pinMode(7, INPUT);
24    x=pulseIn(7,1);
25    Serial.println(x);
26    if (x<1700) {
27      motor1.write(90);
28      motor2.write(90);
29      motor3.write(90);
30    } else if (x<2650){
31      motor1.write(120);
32      motor2.write(120);
33      motor3.write(120);
34    } else if (x>2650){
35      motor1.write(30);
36      motor2.write(30);
37      motor3.write(30);
38    }
39  }

```

Serial Monitor

وأخيراً نقوم بمحاكاة ذلك الكود على موقع التينكر كاد:



Neat Hango-Uusam

Simulator time: 00:02:12

All changes saved

Code Stop Simulation Export Share

Text 1 (Arduino Uno R3)

```

1 #include <Servo.h>
2 Servo motor1 ;
3 Servo motor2 ;
4 Servo motor3 ;
5 int x=0;
6
7 void setup() {
8   motor1.attach(6);
9   motor2.attach(5);
10  motor3.attach(4);
11  motor1.write(90);
12  motor2.write(90);
13  motor3.write(90);
14  Serial.begin(9600);
15  delay(2000);
16
17 void loop() {
18   pinMode(7, OUTPUT);
19   digitalWrite(7,0);
20   delay(50);
21   digitalWrite(7,1);
22   delayMicroseconds(2);
23   pinMode(7, INPUT);
24   x=pulseIn(7,1);
25   Serial.println(x);
26   if (x<1700) {
27     motor1.write(90);
28     motor2.write(90);
29     motor3.write(90);}
30   else if (x<2650){
31     motor1.write(120);
32     motor2.write(120);
33     motor3.write(120);}
34   else if (x>2650){
35     motor1.write(30);
36     motor2.write(30);
37     motor3.write(30);}

```

Serial Monitor

