

I detta projekt kommer du att undersöka temperaturmätningar med hjälp av en temperatursensor och sedan bygga en digital stämningssring som ändrar färgen på TI-Innovator-hubbens röd/grön/blåa (RGB) lysdiod (LED) när temperaturen på ditt finger ändras. Du kan även visa stämningssnivån på din räknare! Först kommer du att arbeta igenom en rad "utmaningar" som kommer att bygga upp dina färdigheter och kunskaper för att till slut uppnå det slutliga målet att skapa en digital stämningssring.



Bakgrund:

Stämningssringen uppfanns av Joshua Reynolds. De var mycket populära på 1970-talet och de förekommer fortfarande idag. Stenen i ringen förmodas ändra färg beroende på stämningen hos bäraren av ringen. "Stenen" hos ringen är i verkligheten ett ihåligt glasskal som innehåller *termotropiska* flytande kristaller. Moderna stämningssringar är gjorda av en platt plastremsa som innehåller flytande kristaller som sätts in i det ihåliga glaset och monteras inom infattningen hos ringen. Kristallerna reagerar på förändringar i temperatur genom vridningar på ett regelbundet sätt. Denna vridning förändrar deras molekylära geometri, som förändrar våglängder av ljus som reflekteras från kristallerna. Våglängder av ljus är ett annat sätt att säga färg, så när temperaturen hos de flytande kristallerna förändras, så förändras färgen som reflekteras från "stenen". Alltså ändrar ringen färg efter handtemperaturen hos bäraren.

Utrustning och materiel:

- Temperaturgivare: TI-Innovator™ Hub (sketch vs. 1.3)
- TI-Nspire™ CX (OS version. 4.5 eller senare)
- Färgad piprensare (Chenille): "Stämning" som ska mätas!

Utmaningar:

1. Använd Ange färg för att utforska med hjälp av lysdioden Color. Försök att hitta RGB-värdena för alla färger i stämningssdiagrammet, t.ex. sända "SET COLOR 255 128 0" blir gult.
2. Använd DispAt-kommandot att visa ditt namn på flera platser på skärmen.
3. Använd en For..EndFor-loop att visa talen 1 till 10.
4. Anslut en temperaturgivare till TI-Innovator-hubben och visa temperaturen på räknaren.
5. Använd en slinga för att läsa och visa temperatur.
6. Valfritt Använd en loop för att läsa, visa och logga in i arrayList för tid och temperatur. Plotta resultatet.
7. Använd en While.. EndWhile-loop tillsammans med getKey () kommando för att övervaka temperaturen och fatta ett beslut om att visa ett meddelande, "varmt" eller "kallt". Ändra sedan ditt program för att inkludera en tredje nivå mellan varmt och kallt, "Skönt".

Avslutande Utmaning:

Bygg en stämningssring för att upprepade gånger läsa av temperaturgivaren, bestämma stämningen hos personen, visa temperaturvärde och visa stämningen.

Exempel på fragment av TI-BASIC-kod för TI-Nspire CX:

Kodfragment för att ställa in RGB-lysdioden till blått på hubben:

```
Send "SET COLOR 0 0 255"
```

Kodfragment för att ställa in RGB-lysdioden till gult på hubben:

```
Send "SET COLOR 255 128 0"
```

Kodfragment för att visa ett meddelande på räknaren:

```
DispAt 5, "JAG ÄR GLAD"
```

Kodfragment för att ansluta och läsa temperaturgivaren:

```
Send "CONNECT TEMPERATURE 1 TO IN 1"
```

```
Send "READ TEMPERATURE 1"
```

```
Get t
```

```
DispAt 4, "TEMPERATURE IS ", t
```

Kodfragment för att läsa och visa temperaturgivaren varje sekund tills Esc-tangenten trycks ned:

```
key:= " "
```

```
While key≠"esc"
```

```
key:=getKey()
```

```
Send "READ TEMPERATURE 1"
```

```
Get t
```

```
DispAt 4, "TEMPERATUREN ÄR", t, "°C"
```

```
Wait 1
```

```
EndWhile
```

Kodfragment för att läsa temperatur var 5:e sekund i 1 minut och lagra tid i en lista som heter tid och temperatur i en lista som heter Temp. Dessa listor kan plottas i diagram på en Data-& Statistik-sida.

```
time:={}
temp:={}
For n,1,120
time[n]:=n*.5
Send "READ TEMPERATURE 1"
Get t
temp[n]:=t
DispAt 4, "TEMPERATUREN ÄR", t
Wait .5
EndFor
```

Kodfragment för att fatta beslut med programmeringsstrukturen IF-Elseif-Then-Endif:

```
Send "READ TEMPERATURE 1"
Get t
If t<24 Then
Send "SET COLOR 255 0 0"
Endif
If t≥24 and t<27 Then
Send "SET COLOR 0 255 0"
Endif
If t≥27 Then
Send "SET COLOR 0 0 255"
Endif
```

Sensor och Hub koppling:

