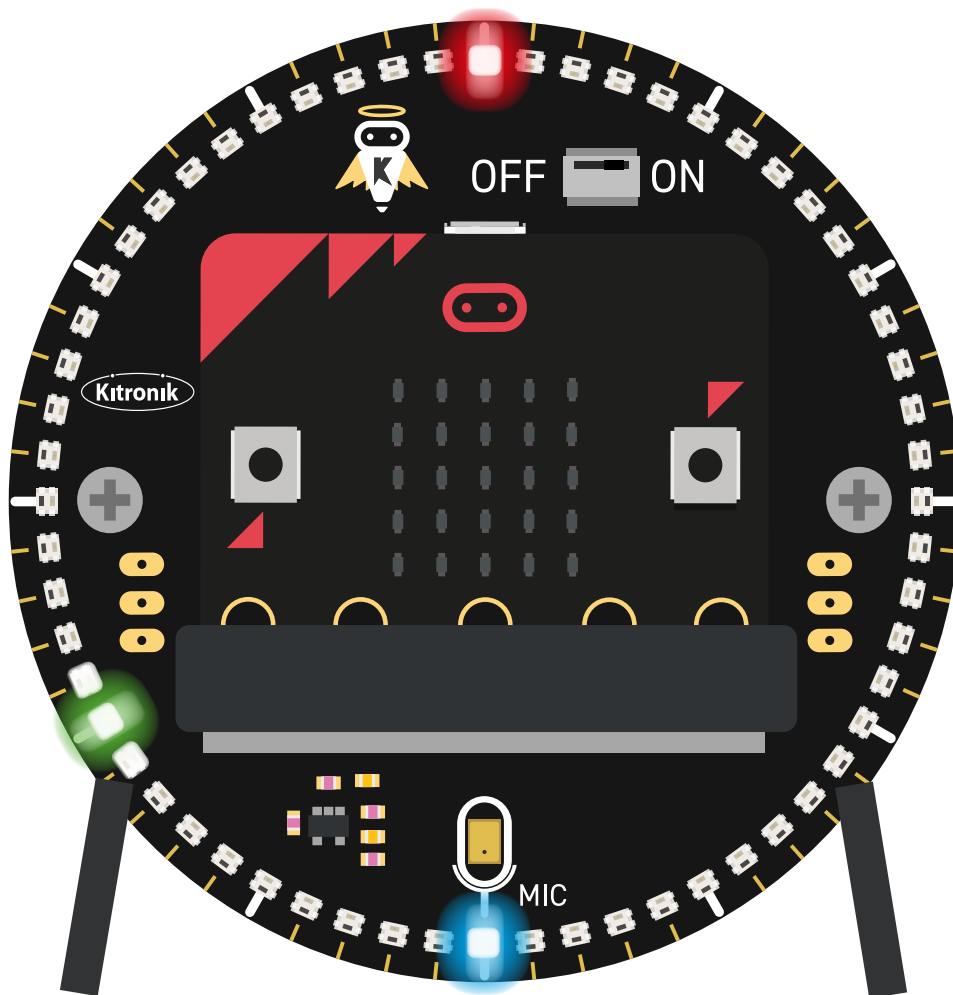


HALO HD ADJUSTABLE CLOCK TUTORIAL

KITRONIK RESOURCES



INTRODUCTION

Learn how to make a clock interface to change and set the time with the Halo HD.

If you have not done the 'Basic Clock Tutorial', it is recommended that you complete it first.

SETTING UP

EQUIPMENT REQUIRED:

- 1 x BBC micro:bit (www.kitronik.co.uk/5613),
- 1 x Halo HD - Alarm Clock Kit (www.kitronik.co.uk/5681)

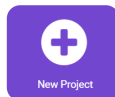
ADDING IN CUSTOM MAKECODE BLOCKS:

We have made custom coding blocks especially for the Halo HD, which helps to make coding super simple within Microsoft MakeCode.

To add these blocks, follow the steps below:

STEP 1: Bring up the MakeCode Block Editor - (makecode.microbit.org).

STEP 2: Click 'New Project'.

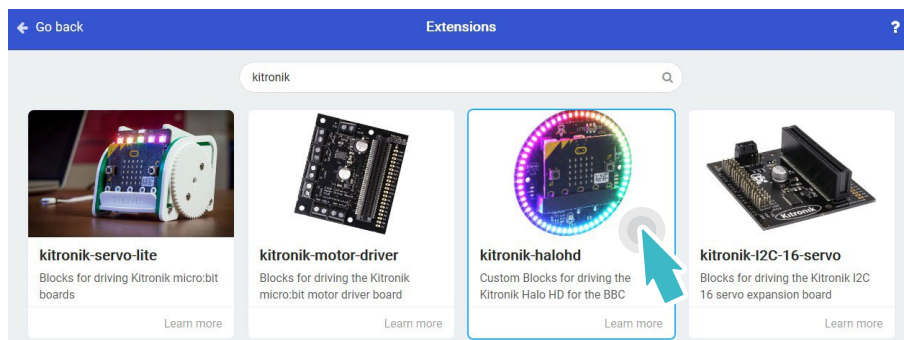
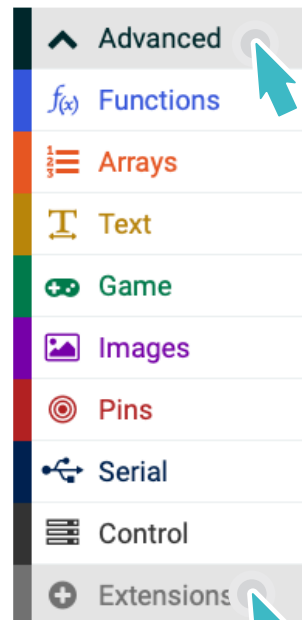


STEP 3: In the toolbox towards the left of the screen, select the 'Advanced' section. Additional block categories will appear below.

STEP 4: Select 'Extensions'.

STEP 5: In the pop up's search bar type 'Kitronik'.

STEP 6: Locate & select the 'kitronik-halohd' box.



THE TUTORIAL

RECREATE BASIC CLOCK CODE

STEP 1: From the first tutorial, we had the code for displaying the time. Start by recreating that program, but remove the 'set time' block from the 'on button pressed' block.



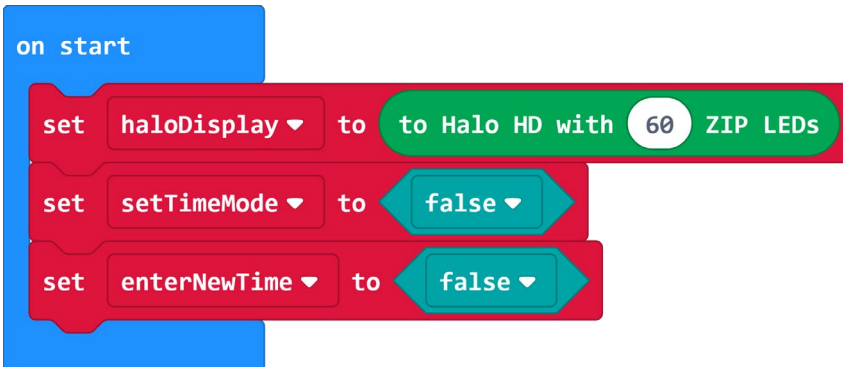
```
on start
  set haloDisplay to to Halo HD with 60 ZIP LEDs

forever
  haloDisplay clear
  haloDisplay set ZIP LED Read seconds for ZIP display to red
  haloDisplay set ZIP LED Read minutes for ZIP display to green
  haloDisplay set ZIP LED Read hours for ZIP display to blue
  haloDisplay show

on button A+B pressed
```

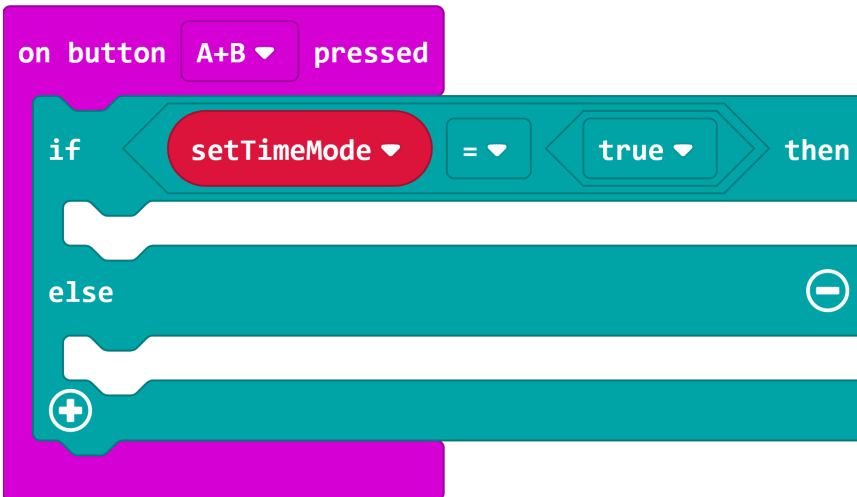
CHANGING MINUTES

STEP 1: Next, two variables are required for the interface for setting the time. Create one variable called 'setTimeMode' and another called 'enterNewTime'. Add them to the 'on start' block and set them to 'false' using the block in the 'logic' section.



```
on start
  set haloDisplay to to Halo HD with 60 ZIP LEDs
  set setTimeMode to false
  set enterNewTime to false
```

STEP 2: The code needs to be told that we are in "setTime" mode. To do this, place an 'if else' statement in the 'on button A+B pressed' to check if the variable 'setTimeMode' equals 'true'.



```
on button A+B pressed
  if setTimeMode = true then
  else
```

HALO HD - ADJUSTABLE CLOCK TUTORIAL

STEP 3: Each section of the 'if else' statement needs to set one of our new variables to be 'true'. The code needs to be:

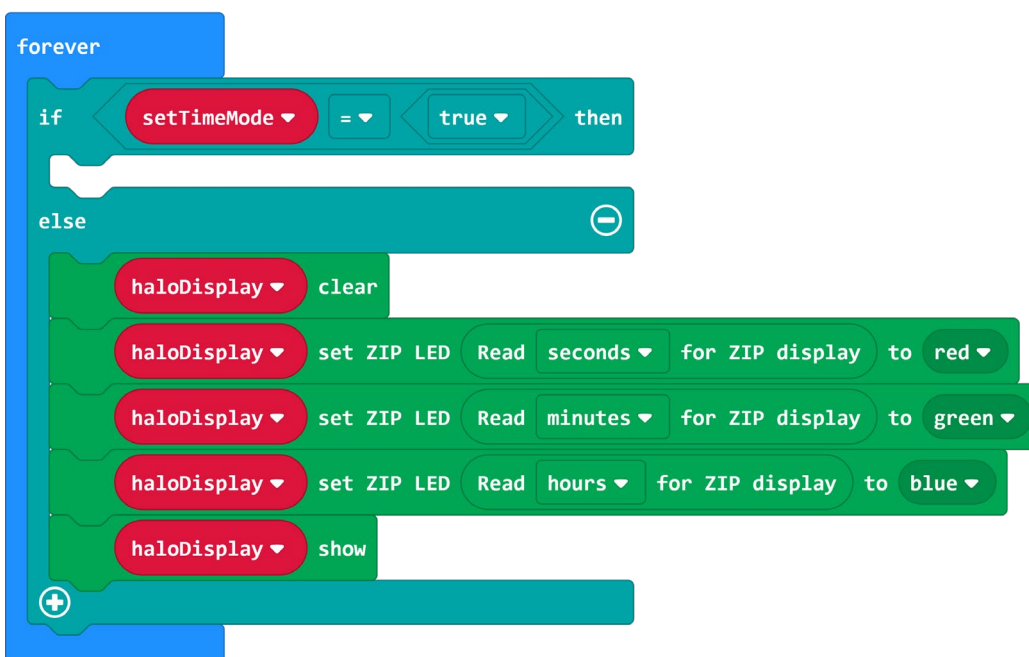
if 'setTimeMode' is 'true', set 'enterNewTime' to 'true', else 'setTimeMode' to 'true'.

So, if the program is in the "setTime" mode, the 'A+B' button press will confirm and enter the new time settings.



```
on button A+B pressed
  if setTimeMode = true then
    set enterNewTime to true
  else
    set setTimeMode to true
```

STEP 4: Going back to the 'forever' loop, we need to add an 'if else' statement to check if we are in "setTime" mode. The 'if' statement needs to check whether the 'setTimeMode' variable equals 'true', 'else' run the code for displaying the time.

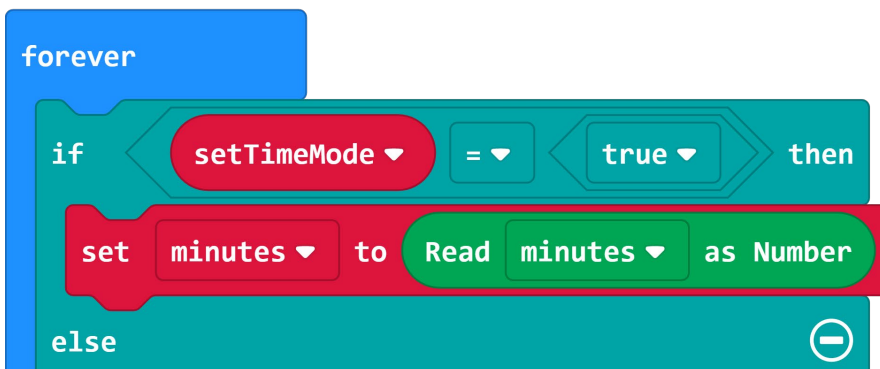


```
forever
  if setTimeMode = true then
  else
    haloDisplay clear
    haloDisplay set ZIP LED Read seconds for ZIP display to red
    haloDisplay set ZIP LED Read minutes for ZIP display to green
    haloDisplay set ZIP LED Read hours for ZIP display to blue
    haloDisplay show
```

HALO HD - ADJUSTABLE CLOCK TUTORIAL

STEP 5: Next, we need to create two more variables: 'minutes' and 'hours'. These will be the values for setting the new time.

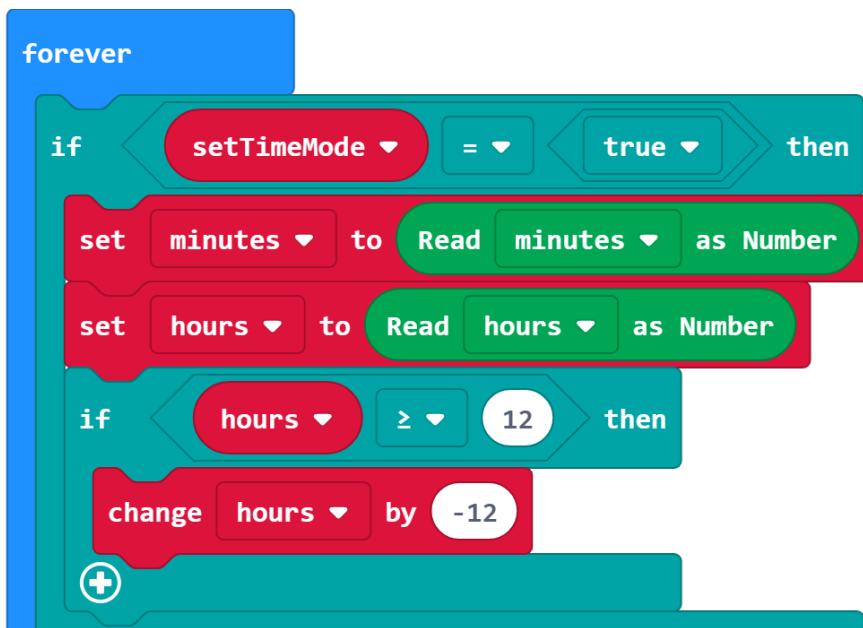
STEP 6: The new variable 'minutes' needs to be set to the current minute reading on the clock. Inside the Clock section of the Halo HD extension, there is a 'read time as number' block. Set 'minutes' to be equal to this and select 'minutes' from the drop-down. Place the 'set minutes' block into the first slot of the 'if else' statement in the 'forever' loop.



STEP 7: See if you can create the blocks that read and set the hours...

HINT: It's the same blocks, but with different selections on the drop-downs.

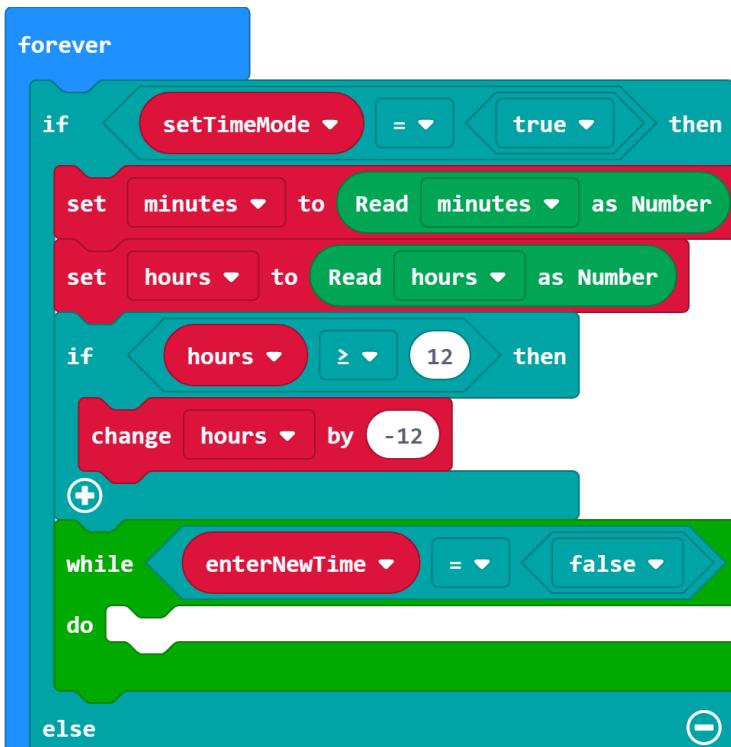
STEP 8: As the RTC is setup in 24 hour mode, we also need to add in a quick check at this point to make sure we don't try and display 24 hour time on a 12 hour clock! Use an 'if' statement to check whether the read hours is greater than or equal to 12, and if it is, take 12 from it's value.



HALO HD - ADJUSTABLE CLOCK TUTORIAL

STEP 9: Now that we have the current minutes and hours stored as variables, we can change their values and use them to set a new time.

Whilst we are setting a new time, the Halo HD needs to display the values we are changing, until we confirm the new settings. To do this, add a 'while' loop after the 'set hours' block. The loop should continue while the variable 'enterNewTime' is 'false'.



```

forever
  if setTimeMode = true then
    set minutes to Read minutes as Number
    set hours to Read hours as Number
    if hours ≥ 12 then
      change hours by -12
    while enterNewTime = false
  else
  
```

STEP 10: In the loop, we need to display the time values we are changing. The code is very similar to our clock display code. See if you can remember how we did that, and then modify the code to display our variables.



```

while enterNewTime = false
do
  haloDisplay clear
  haloDisplay set ZIP LED minutes to green
  haloDisplay set ZIP LED hours to blue
  haloDisplay show
else
  
```

HALO HD - ADJUSTABLE CLOCK TUTORIAL

STEP 11: Now there needs to be a way to increment the 'minutes' value. We will do this with the 'on button A pressed' block. Make it change the 'minutes' variable by 1 each time.

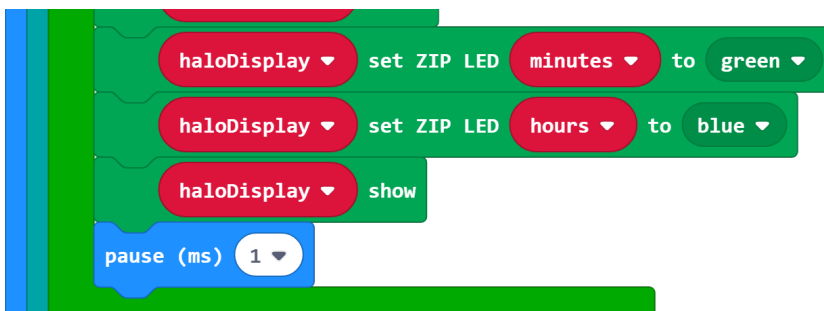


STEP 12: Only increasing the 'minutes' by 1 could make setting the time a very long process; it would be nice to have a much quicker way.

Using the 'on button B pressed' block, see if you can make the 'minutes' variable change by 10.



STEP 13: To make sure the micro:bit has time to check for the button presses, a short 'pause' of 1ms needs to be added to the end of the 'while' loop.



STEP 14: If you have a micro:bit connected, click 'Download' to transfer your code.

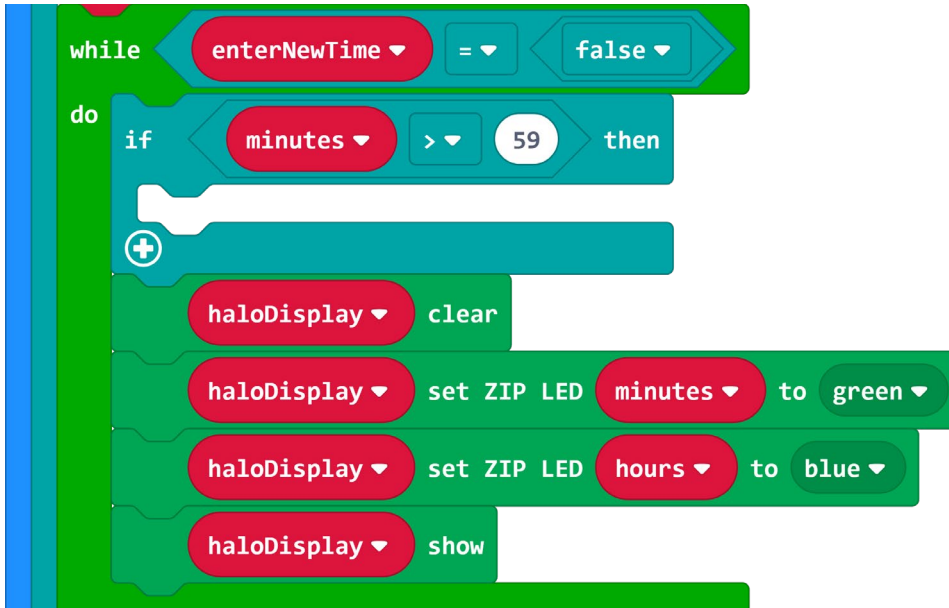
Test the code by pressing 'A+B' to get into time setting mode and using the 'A' and 'B' buttons to change the 'minutes' value.

HINT: Test what happens when the 'minutes' value is increased beyond the next hour.

HALO HD - ADJUSTABLE CLOCK TUTORIAL

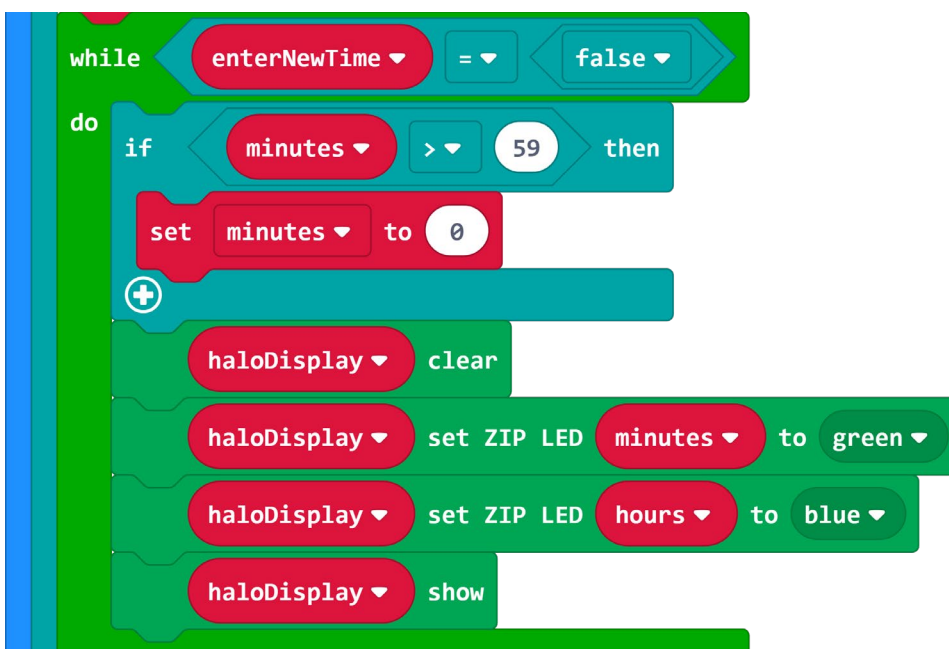
SETTING THE TIME

STEP 1: Did you set the minutes past the hour? You might have noticed that the LED disappears. This is because the value of 'minutes' becomes greater than the number of LEDs on the Halo HD. To correct this, we can use an 'if' statement to check if 'minutes' is greater than... which number do you think? Place this 'if' statement at the start of the 'while' loop.



```
while enterNewTime = false
do
  if minutes > 59 then
    haloDisplay clear
    haloDisplay set ZIP LED minutes to green
    haloDisplay set ZIP LED hours to blue
    haloDisplay show
```

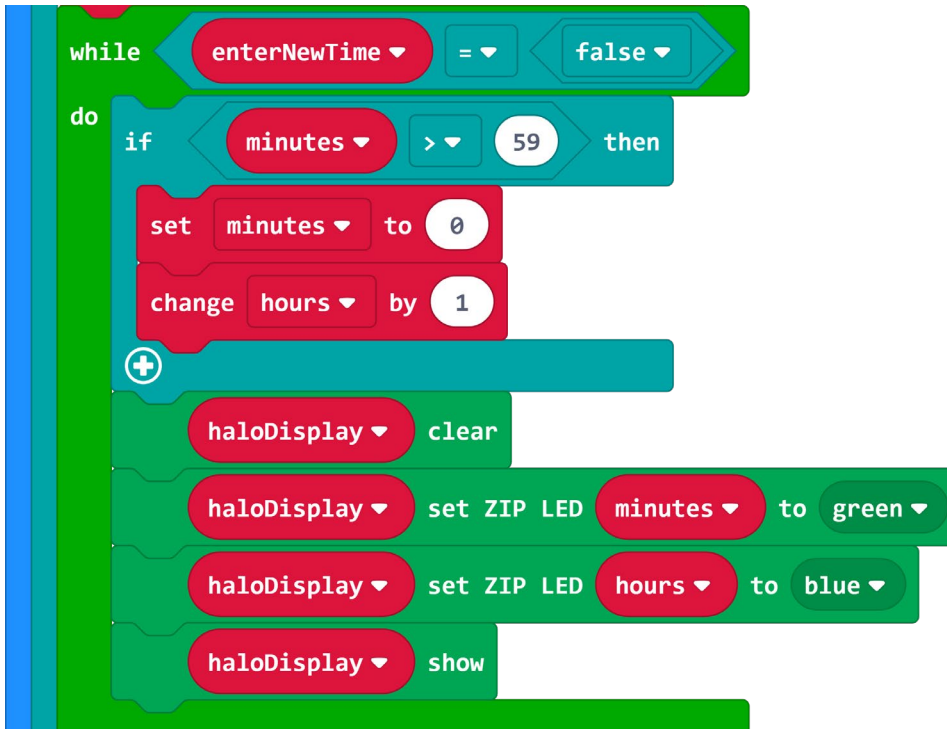
STEP 2: Now we have the check in place, the 'minutes' variable needs to be set back to 0 when the condition is met. Add the appropriate block inside the 'if' statement.



```
while enterNewTime = false
do
  if minutes > 59 then
    set minutes to 0
    haloDisplay clear
    haloDisplay set ZIP LED minutes to green
    haloDisplay set ZIP LED hours to blue
    haloDisplay show
```

HALO HD - ADJUSTABLE CLOCK TUTORIAL

STEP 3: Since 'minutes' has gone past the hour and been set back to 0, that means that the hour must have changed as well. Add a block to the 'if' statement to change the 'hours' variable by 1.



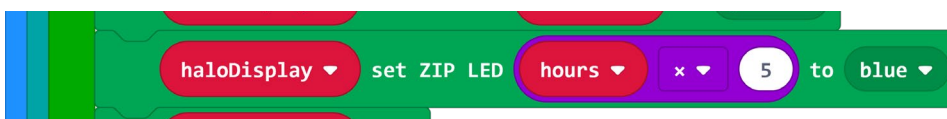
```

while enterNewTime = false
do
  if minutes > 59 then
    set minutes to 0
    change hours by 1
  +
  haloDisplay clear
  haloDisplay set ZIP LED minutes to green
  haloDisplay set ZIP LED hours to blue
  haloDisplay show
  
```

STEP 4: If you have a micro:bit connected, click 'Download' to transfer your code. Again, test the code by going into time setting mode and increasing the value of 'minutes'.

HINT: Test what happens now when we increase to the next hour.

STEP 5: At the moment, when we increase the 'minutes' and cause the 'hours' to move forward, the 'hours' only move on by a single ZIP LED. This is not correct, as the hour marks on the Halo HD are actually every 5 ZIP LEDs. This is easy to fix. In the block where we set the ZIP LED for the 'hours', change it so it reads:



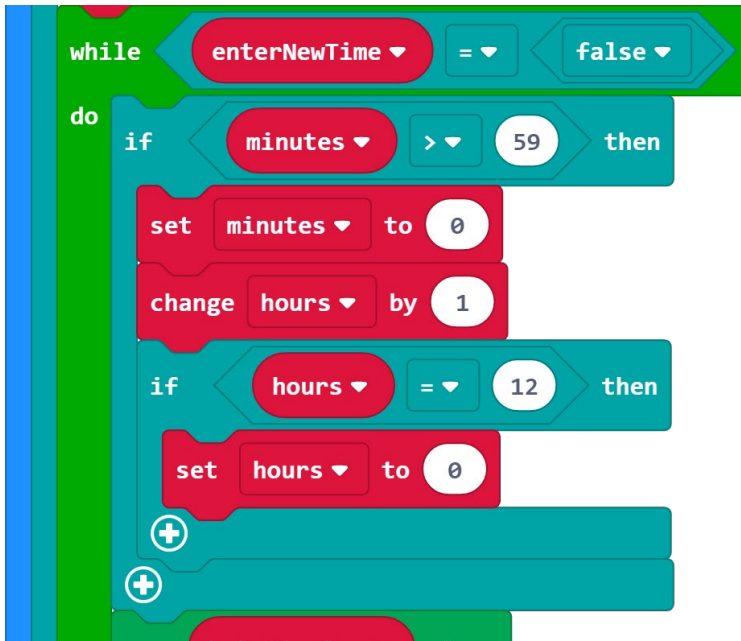
```

haloDisplay set ZIP LED hours x 5 to blue
  
```

HALO HD - ADJUSTABLE CLOCK TUTORIAL

STEP 6: Just as the 'minutes' need resetting when they pass the 59th minute, the 'hours' needs resetting when they pass the 11th hour.

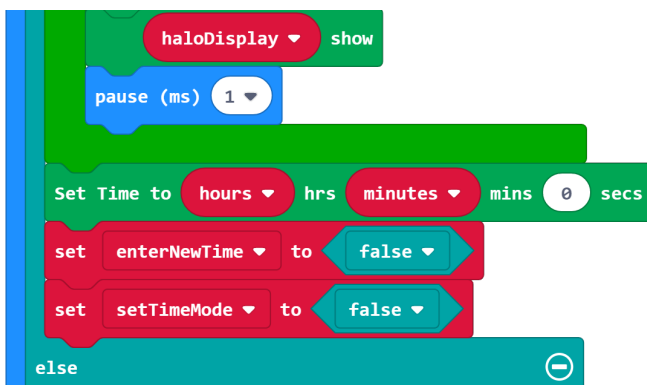
See if you can add the code to reset the 'hours' back to 0.



STEP 7: Once we have reached the time we want to set, pressing buttons 'A+B' will exit the 'while' loop. Now we need to save this new time to the Halo HD.

Inside the Clock section of the Halo HD extension, there is a 'set time' block. Add this after the 'while' loop and insert the 'hours' and 'minutes' variables into the appropriate fields.

Finally, we just need to set 'enterNewTime' and 'setTimeMode' back to false (this is to make sure the code does not loop back around into "setTime" mode again and displays the new time instead).



STEP 8: CODING COMPLETE! Let's click 'Download' and transfer your code to the Halo HD and enjoy a colourful clock with time setting functionality. (See next page for complete code).

HALO HD - ADJUSTABLE CLOCK TUTORIAL

```

forever
  if setTimeMode = true then
    set minutes to Read minutes as Number
    set hours to Read hours as Number
    if hours >= 12 then
      change hours by -12
    +
  while enterNewTime = false
    do
      if minutes > 59 then
        set minutes to 0
        change hours by 1
        if hours = 12 then
          set hours to 0
        +
      haloDisplay clear
      haloDisplay set ZIP LED minutes to green
      haloDisplay set ZIP LED hours x 5 to blue
      haloDisplay show
      pause (ms) 1
    +
  Set Time to hours hrs minutes mins 0 secs
  set enterNewTime to false
  set setTimeMode to false
else -
  haloDisplay clear
  haloDisplay set ZIP LED Read seconds for ZIP display to red
  haloDisplay set ZIP LED Read minutes for ZIP display to green
  haloDisplay set ZIP LED Read hours for ZIP display to blue
  haloDisplay show
  +

on start
  set haloDisplay to to Halo HD with 60 ZIP LEDs
  set setTimeMode to false
  set enterNewTime to false

on button A+B pressed
  if setTimeMode = true then
    set enterNewTime to true
  else -
    set setTimeMode to true
  +

on button A pressed
  change minutes by 1

on button B pressed
  change minutes by 10
  
```

For any further queries or support, please visit the Kitronik website: www.kitronik.co.uk/5672

Or get in touch:

Telephone +44 (0) 115 970 4243

Sales email: sales@kitronik.co.uk

Tech Support email: support@kitronik.co.uk

Web: www.kitronik.co.uk



[kitronik.co.uk/twitter](https://twitter.com/kitronik)



[kitronik.co.uk/facebook](https://www.facebook.com/kitronik)



[kitronik.co.uk/youtube](https://www.youtube.com/kitronik)



[kitronik.co.uk/instagram](https://www.instagram.com/kitronik)



Designed & manufactured
in the UK by 

