

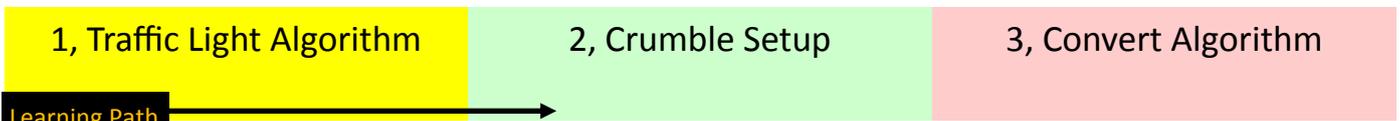
# Crumble Traffic Lights

## Computer Science Concepts

- All programming is an algorithm turned into code
- An Algorithm is a step by step sequences of instructions (or rules) to do something

**Program Aim** Think through traffic light sequence algorithm, independently setup crumble and convert algorithm into crumble code

3a, Partially Sighted Crossing      3b, Level Crossing Lights



## Year Group

-I have taught this successfully in Year 5 & 6. Whilst the programming is not complex the addition of control kit and the traffic light order adds complexity.

## Time Needed

-I have taught this in Year 6 in an hour where they have had lots of previous programming experience but little control programming but an hour and 30 minutes will ensure all pupils finish all the main learning path activities.

**Resources**  
 Scratch Traffic Lights or YouTube Traffic Light Video inside YouTube Looping Website, Algorithm Sheet, Power Maker Card, Traffic Light Maker Card, Crumble board, Traffic light crumb, USB cable, Four crocodile clip cables **Optional:- For 3a Buzzer Crumb, two extra crocodile cables, Buzzer Maker Card for 3b sparkle, Three extra crocodile cables, Sparkle Maker Card, Power Pack, Batteries**

## 1, Traffic Light Algorithm

Recap that algorithms can be a step by step sequence of instructions to achieve something. Explain that they are going to work out the precise algorithm for a sequence of traffic lights and once they have completed it they can turn their algorithm into code that works on a set of real lights. Hold up the traffic light crumb by 4Tronix (don't connect it as puzzling that out later is part of the challenge)

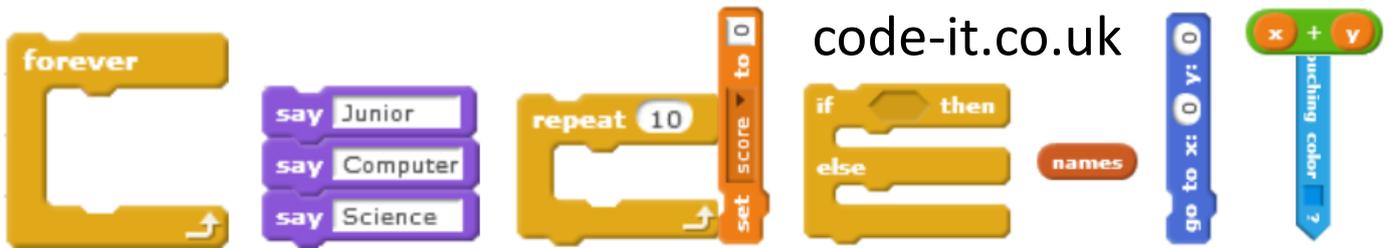
Hand out the algorithm sheet and go through the Belisha Beacon example. Emphasise how the algorithm needed a wait so there is enough time to see the light and enough time for it to remain off.

Go through the algorithm commands they can use on the top right of the sheet. If they have done a lot of Scratch explain that you don't want Scratch code.

Put up the Scratch traffic light making sure it is in full screen mode so the code cannot be seen. Explain how the traffic light changes down then up. Draw out that these are different. Can they spot why? Answer Red and Amber are on together when changing down.



Traffic Light Crumb by 4Tronix



# Crumble Traffic Lights Page 2

## 1, Traffic Light Algorithm Continued

When pupils think they are finished check their algorithm with the example at the end of this planning. If it is correct or very close (timings are out a little) wait until you have the next pupil who is correct and pair them up to move onto Crumble Setup and Crumble Programming.

Having taught this in three schools and six classes (to date) I have found there are always some pupils who ignore what is happening in the Scratch model or video and create what they think is happening. Showing them what is happening in the model and praise the parts they have got right this helps them to have the confidence to fix it.

Don't be tempted to do it for them as you are only developing dependence on you. I have scribed for particularly slow writers as the thinking is more important than being able to write it down in this activity.

## 2, Crumble Setup

It helps that you will be busy for a while checking algorithms so pupils will have to get on independently just using the crumble kit, maker cards and a computer with the free crumble software installed.

Typical errors involve

- Not wiring up the kit correctly (point them to the pictures on the maker cards)
- Not plugging the USB into the right computer (You may laugh but I have seen it happen more than once)
- Failing to realise that the picture of the crumble on the card needs to be black chip up so wiring it up to the wrong side.

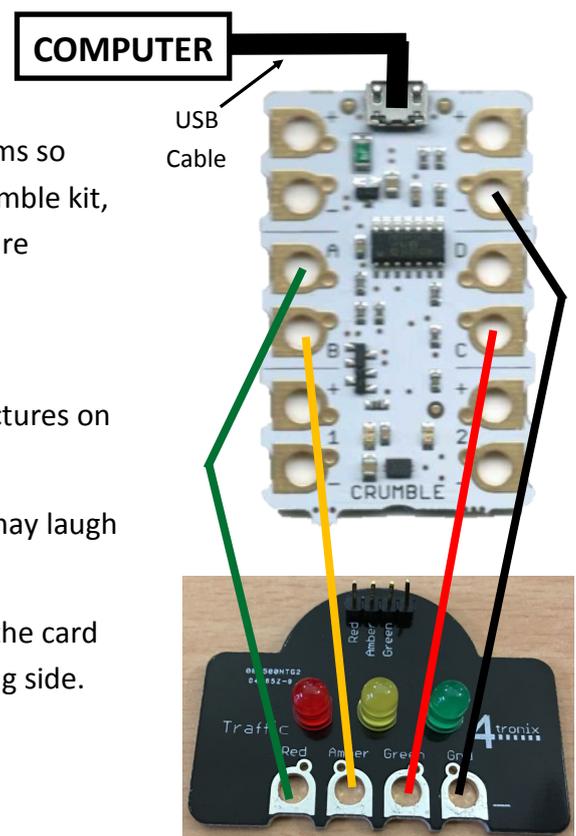
More time means less pressure to get it right first time.

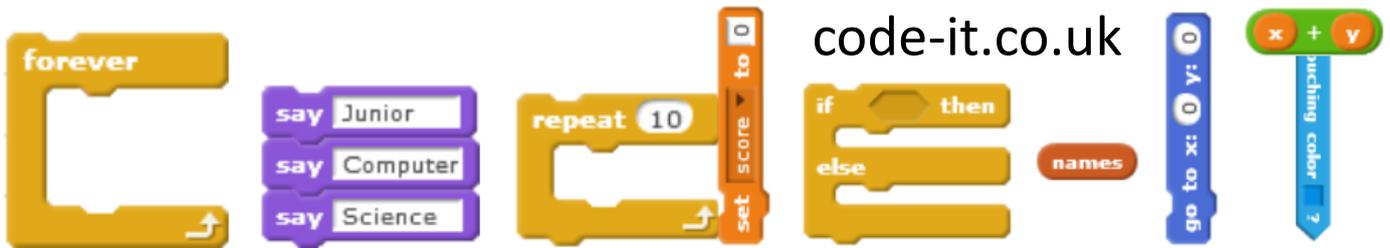
One ground (Gnd) wire completes the circuit for all three bulbs

The bulbs get their power from the small amount of electricity passed through the USB cable.

If you want the traffic lights to continue working unplugged from the PC then add a battery pack.

The program is sent to the black chip on the board. Previous programs may still be on there.





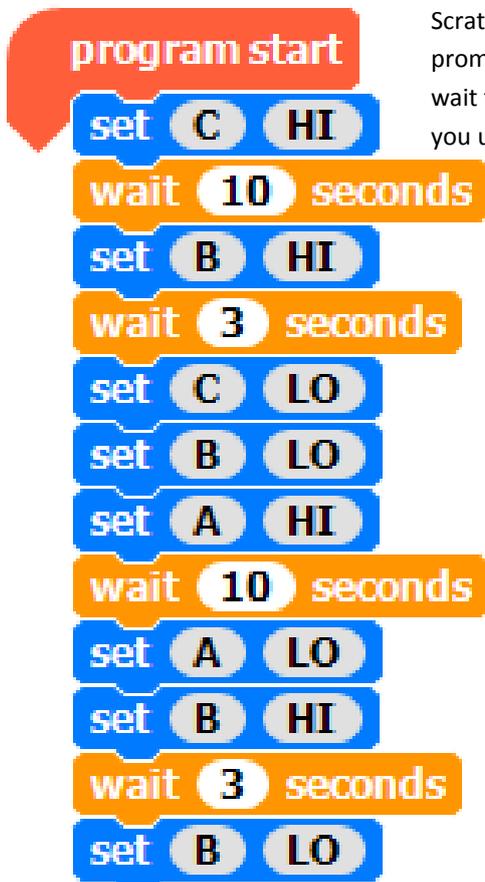
## Crumble Traffic Lights Page 3

### 3, Convert Algorithm into Program

The programming content is mainly sequencing with a forever loop once they have the sequence order correct.

Typical Bugs involve

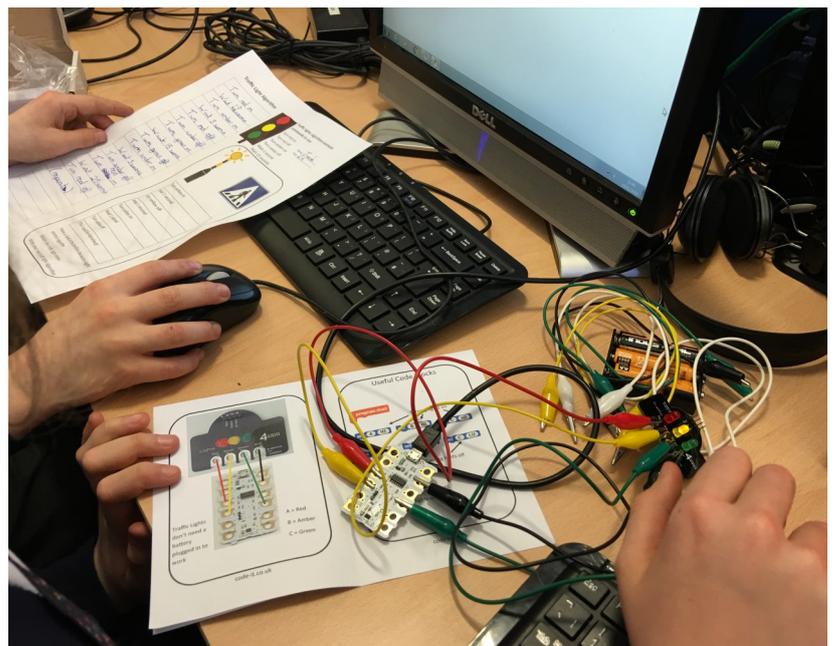
- Not reading the traffic light card to see what programming elements can be used to turn the light on and off. This is easily remedied by getting them to read the card.
- If they have used Scratch they may think that they can create parallel blocks of code that run at the same time. Currently Crumble only works with one script. Informing them that they must keep all code in one block fixes this.
- If pupils have wired their lights to different Import/Output letters than those on the card they will need to take this into account when programming. This is not necessarily a bug.



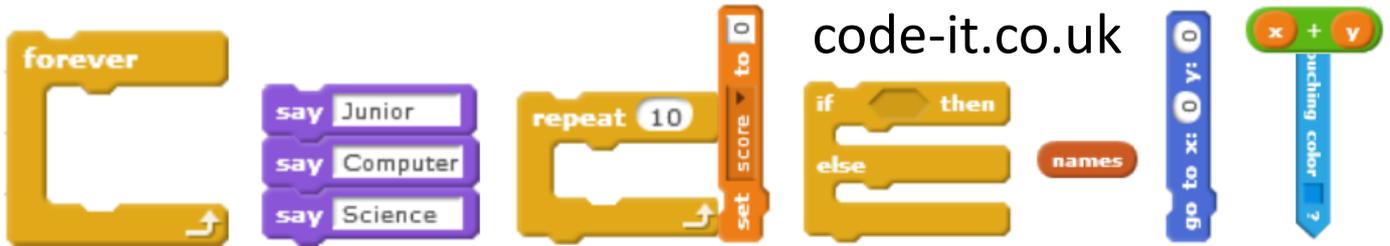
Successful code built using the Scratch Traffic light algorithm prompt will look like this. The wait times will be different if you use the YouTube video

#### Input Outputs

Lights are **Outputs** as they *put out* information that can be used by others outside the digital device.



Year 5 pupils at Ringwood Junior School  
converting their algorithm into code



# Crumble Traffic Lights Page 4

## 3a Partially Sighted Crossing

Ask pupils who have created a successful traffic light program how a partially sighted person, who can't see the lights would know when to cross the road.

Once they have figured out that they can use a noise of some sort show them the buzzer crumb. Challenge them to mark on their algorithm when the buzzer would go on and off.

Check that this is as opposite

Then challenge them to use the buzzer crumb maker card to turn this algorithm into code.

## 3b Level Crossing Lights

(if pupils have added the buzzer this will need to be disconnected to complete this extension activity)

If the traffic light was used to stop cars at a train level crossing when would the foot traffic warning lights start and finish. You could always show pupils a video of a train crossing to help them design their adapted algorithm and program.

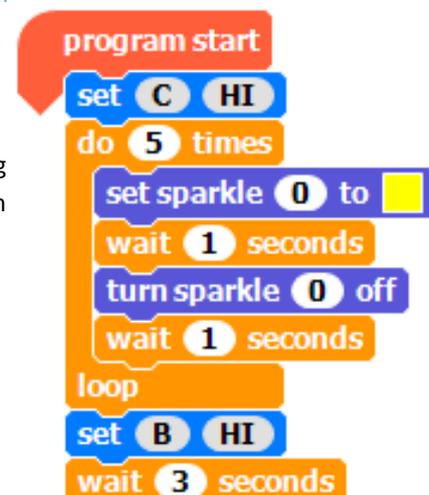
The warning lights would come on in the same places as the buzzer example opposite.

Pupils can then use the crumble and power maker cards to program the sparkles to flash. They will need to build this into one code block within the traffic lights program.

## Traffic Light Algorithm

Turn red on	Turn buzzer on
Wait 10 seconds	
Turn amber on	Turn buzzer on order can vary
Wait 3 seconds	
Turn red off	
Turn amber off	Order not important
Turn green on	
Wait 10 seconds	
Turn amber on	
Turn green off	
Wait 3 seconds	
Turn amber off	
(Repeat all forever)	

One possible way of creating a flashing light within the traffic light program



### Linking to DT

A much better way of doing this would be to combine this module with DT. Build a traffic light holder to mount the light element. Pupils could also add a button for pedestrians to trigger the lights to change.