Using the JavaScript editor at **makecode.microbit.org**

The different headings show different layers in the software stack

# JavaScript

Taking the code:

basic.showNumber(0)

# Pxt-microbit

This is first translated by pxt-microbit/libs/core/basic void showNumber(int value, int interval = 150) {

<https://github.com/Microsoft/pxt-microbit/blob/master/libs/core/basic.cpp#L19>

(as number does not need scroll) in to: uBit.display.printChar(t.charAt(0), interval \* 5);

uBit is defined in pxt.cpp: MicroBit uBit;

<https://github.com/Microsoft/pxt-microbit/blob/master/libs/core/pxt.cpp#L4>

# microbit

This is passed to the DAL wrapper where the display class is defined: MicroBitDisplay display;

<https://github.com/lancaster-university/microbit/blob/master/inc/MicroBit.h#L111>

# microbit-DAL

This is passed to the DAL where the printChar function is defined:

<https://github.com/lancaster-university/microbit-dal/blob/master/source/drivers/MicroBitDisplay.cpp#L606>

The printChar function calls the printCharAsync function: this->printCharAsync(c, delay);

Which is defined here:

<https://github.com/lancaster-university/microbit-dal/blob/master/source/drivers/MicroBitDisplay.cpp#L481>

The printCharAsync function calls the image.print function: image.print(c, 0, 0);

Which is defined here:

<https://github.com/lancaster-university/microbit-dal/blob/master/source/types/MicroBitImage.cpp#L572>

and calls: this->getBitmap()[y1\*getWidth()+x1] = (v & (0x10 >> col)) ? 255 : 0;

# JavaScript

Taking the code:

basic.showLeds(`

. . . . .

. . . . .

. . # . .

. . . . .

. . . . .

`)

# Pxt-microbit

This is first translated by pxt-microbit/libs/core/basic

<https://github.com/Microsoft/pxt-microbit/blob/master/libs/core/basic.cpp#L41>

in to: uBit.display.print(MicroBitImage(imageBytes(leds)), 0, 0, 0, interval);

uBit is defined in pxt.cpp: MicroBit uBit;

<https://github.com/Microsoft/pxt-microbit/blob/master/libs/core/pxt.cpp#L4>

# microbit

This is passed to the DAL wrapper where the display is defined: MicroBitDisplay display;

<https://github.com/lancaster-university/microbit/blob/master/inc/MicroBit.h#L111>

# microbit-DAL

This is passed to the DAL where the print function is defined:

<https://github.com/lancaster-university/microbit-dal/blob/master/source/drivers/MicroBitDisplay.cpp#L699>

The print function calls the printAsync function: this->printAsync(i, x, y, alpha, delay);

Which calls image.paste: <https://github.com/lancaster-university/microbit-dal/blob/master/source/drivers/MicroBitDisplay.cpp#L567>

Which is defined here: <https://github.com/lancaster-university/microbit-dal/blob/master/source/types/MicroBitImage.cpp#L496>x

# JavaScript

Taking the line of code: *led.plot(x, y)* where x, y are coordinates of an LED

# Pxt-microbit

This is first translated by pxt-microbit/libs/core/led

<https://github.com/Microsoft/pxt-microbit/blob/master/libs/core/led.cpp#L23>

in to: uBit.display.image.setPixelValue(x, y, 0xff);

uBit is defined in pxt.cpp: MicroBit uBit;

<https://github.com/Microsoft/pxt-microbit/blob/master/libs/core/pxt.cpp#L4>

# microbit

This is passed to the DAL wrapper where the display is defined: MicroBitDisplay display;

<https://github.com/lancaster-university/microbit/blob/master/inc/MicroBit.h#L111>

# microbit-DAL

This is passed to the DAL where image is defined: MicroBitImage image;

<https://github.com/lancaster-university/microbit-dal/blob/master/inc/drivers/MicroBitDisplay.h#L238>

The setPixelVlaue function is defined here:

<https://github.com/lancaster-university/microbit-dal/blob/master/source/types/MicroBitImage.cpp#L394>

Which calls: this->getBitmap()[y\*getWidth()+x] = value;