

Another Look at Hard Copy Printing

[RodBird](#)

Table of Contents

[Another Look at Hard Copy Printing](#)

[The screen view](#)

[The drawing canvas](#)

[The printing canvas](#)

[Scaling](#)

[Size](#)

[Code sample](#)

This article is about filling the printed page, either by scaling up low resolution graphics or scaling down high resolution graphics. We will use a graphicbox control and the graphics command:

```
#win.graphicbox "print size"
```

You will have a better understanding of the “size” argument after reading this and you will be amazed at what Liberty BASIC can achieve.

In essence, the size argument takes the number of pixels you have drawn and spreads them across the full printed width of a page. If it is set to 100, 100 pixels will fill the printed page, if it is 5000, 5000 pixels will be needed to fill the printed page.

The screen view

On screen the graphicbox control is our view of the drawing. I have used a small graphicbox, you can set the size to whatever suits your application, with or without scrollbars. The reason I have chosen a small graphicbox is that we will be printing graphics far larger than can be viewed, even full screen.

Now, looking through the graphicbox, imagine you see a drawing canvas and behind that a printing canvas. Each has their own size and resolution and both are bigger than the graphicbox. You will draw in one resolution and print in another, “size” will handle the transformation.

The drawing canvas

The drawing canvas that sits behind the graphicbox can be as large as you want. We do not need to define its size it will adopt the size of the drawing. Its resolution is in screen pixels.

Drawing commands will color visible pixels within the graphicbox, outside of those limits drawing is not visible but nevertheless Liberty draws and stores the graphics on the drawing canvas. You may scroll them into view if you switch on graphicbox scrollbars. You may also print the unseen graphics.

The printing canvas

The printing canvas is defined by your paper size and the printing resolution, in Dots Per Inch (DPI), of your printer. We cannot equate pixels with dots easily, the printer will use many dots to create the image of a single pixel, how many, is actually defined by the “size” parameter. The numbers below are based on experimentation.

My typical printer, using A4 paper and printing at 600 DPI is proven to have the capability of printing c5000 pixels across the printed page and c7000 pixels down the printed page in portrait mode. It is capable of higher resolution, especially if I set it to print at 1200 DPI.

Fortunately we don’t need to know the exact limits because the “size” parameter handles the transformation. For example printing 5000 pixels across the page produces a legible “Hello” 1mm wide that requires a magnifying glass to read. 7000 pixels produces a “Hello” that requires a microscope.

Scaling

The drawn pixel is being printed to the paper using a variable number of dots to represent the single pixel image.

Taking the imaginary printing canvas we define how many pixels will be printed width wise by setting the “size” parameter. That sets the number of dots that are applied to each printed pixel. Let’s assume the printer is capable of printing c5000 dots across an A4 page.

If we set “size” to 640 then 8 dots are used per pixel and a 640 pixel drawing will fill the page.

If we set “size” to 1024 then 5 dots are used per pixel and a 1024 pixel drawing will fill the page.

If we set “size” to 5000 then 1 dot is used per pixel and a 5000 pixel drawing will fill the page.

If the drawn image is smaller than “size” it will fill part of the printed page, if it is larger than “size” it will exceed the printed page. So with one simple “size” parameter we are in complete control of the size, placement and resolution of the printed output!

Size

Liberty BASIC 4.04 allows the programmer to specify the size in a number of ways, the choices are

- none - size will adopt your current screen width value in pixels
- VGA - size will be set to 640
- SVGA - size will be set to 800
- XGA - size will be set to 1024
- n - size will be set to n

Code sample

This simple example takes your printer up to a pretty high resolution. It should print pixel perfect graphics on an A4 sheet using a size of 4960. You will be able to beat this but there must be few applications that will need such high resolution printing. Typically you will use less.

```
ImageWidth = 4958
ImageHeight = 7015
midW=int(ImageWidth/2)
midH=int(ImageHeight/2)
nomainwin
WindowWidth = 400
WindowHeight = 400
UpperLeftX = (DisplayWidth-WindowWidth)/2
UpperLeftY = (DisplayHeight-WindowHeight)/2
graphicbox #1.g, 50,50,300,300
open "Graphics Printing" for window_nf as #1
print #1, "trapclose [quit]"
print #1.g,
"down ; place 0 0; color black ; box ";ImageWidth;" ";ImageHeight
print #1.g, "place ";midW;" ";midH
print #1.g, "circle ";midW
print #1.g, "place ";midW;" ";midH
print #1.g, "circle ";midH
print #1.g, "place ";midW;" ";midH
print #1.g, "\Hello"
print #1.g, "flush"
print #1.g, "print 4960"
```

```
wait
```

```
[quit]
```

```
close #1
```

```
end
```