

# Calculator Programs For Stamp Collectors

Mitch Richling

2021-03-19

Author: Mitch Richling  
Updated: 2021-05-13 19:56:12

Copyright 2021 Mitch Richling. All rights reserved.

## Contents

1	Metadata	1
2	Introduction	2
3	The Main Menu	2
4	[CAL]: Image Calibration	2
4.1	Known DPI	2
4.1.1	Examples	2
4.2	Using a target	3
4.2.1	Example	3
4.3	Using a Stamp!	3
4.4	Calibration factor	3
4.4.1	Examples	3
5	[LENG]: Measuring lengths	3
5.1	Horizontal or Vertical Lengths	3
5.1.1	Examples	3
5.2	Generic distances	4
5.2.1	Examples	4
5.3	Philatelic Application: Rotary Press vs. Flat Plate (Take 1)	4
5.3.1	Example 1	4
	Method 1 (Stamp design not aligned)	4
	Method 2 (Stamp design aligned)	5
6	[PERF]: Measuring & Converting Perforations	5
6.0.1	Example: Measuring Standard & Kiusalas Perforations	5
7	[ANGL]: Measuring Angles	6
7.1	Examples	6
7.2	Philatelic Application: Parallel Lines (Take 1)	6
7.2.1	Example	6
8	[DCRD]: Design Coordinates	6
8.1	Example	7
9	[QUAD]: A Philatelic Power Tool	7
9.1	Philatelic Application: Parallel Lines (Take 2)	7
9.2	Philatelic Application: Rotary Press vs. Flat Plate (Take 2)	8
10	Code	8
10.1	Menu Code	8
10.2	Local Functions	10
11	EOF	20

## 1 Metadata

The home for this HTML file is: <https://richmit.github.io/hp42/phil.html>

A PDF version of this file may be found here: <https://richmit.github.io/hp42/phil.pdf>

Files related to this document may be found on github: <https://github.com/richmit/hp42>

Directory contents:

src	-	The org-mode file that generated this HTML document
src_42s	-	Ready to convert source listings for 42s code in this document
docs	-	This html document
bin	-	Importable RAW program files

## 2 Introduction

This application might be best described as an image metrology tool with a focus on philatelic applications. It supports a few standard measurements useful for philately (length & angle) as well as some specialized functionality for philatelists (perforation measurement, kiusalas gauge conversion, Thirkell & design coordinates, design size, & parallel edge checks). While I do most of this kind of work on my laptop with ImageJ (Fiji actually), I frequently use this calculator application with the digital microscope on my "stamp table" or when I'm away from my laptop.

## 3 The Main Menu

MS	ML	Menu	Description	Require Cal?
LBL 85	LBL 92	ANGL	Angle given complex pixel coordinate	Never
LENG:PX>D	LBL 91	LENG:PX>D	Length given real pixel count or complex pixel coordinate	Always
LENG:D>PX	LBL 66	LENG:D>PX	Pixel count given Length or complex coordinate	Always
LENG:□□□□		LENG:□□□□		N/A
LENG:□□□□		LENG:□□□□		N/A
LENG:□□□□		LENG:□□□□		N/A
LENG:CAL	LBL 95	LENG:CAL	Set or display image calibration	Never
QUAD:UL	LBL 57	QUAD:UL		Never
QUAD:UR	LBL 58	QUAD:UR		Never
QUAD:LR	LBL 59	QUAD:LR		Never
QUAD:LL	LBL 60	QUAD:LL		Never
QUAD:□□□□		QUAD:□□□□		N/A
QUAD:CAL	LBL 95	QUAD:CAL	Set or display image calibration	Never
QUAD:∠TB	LBL 62	QUAD:∠TB	Compute angle difference between top an bottom sides	Never
QUAD:∠LR	LBL 63	QUAD:∠LR	Compute angle difference between left an right sides	Never
QUAD:WIDE	LBL 64	QUAD:WIDE	Compute width (left, right, average)	Always
QUAD:TALL	LBL 65	QUAD:TALL	Compute height (top, bottom, average)	Always
QUAD:□□□□		QUAD:□□□□		N/A
QUAD:□□□□		QUAD:□□□□		N/A
PERF:D<>P	LBL 96	PERF:D<>P	Convert between perforations & distance	Only if D=px
PERF:P<>K		PERF:P<>K	Convert between perforations & Kiusalas	Never
PERF:LBL 87	LBL 98	PERF:C??	Set perforation count (C)	Never
PERF:LBL 89	LBL 99	PERF:D??	Cycle distance units: px mm mil um in cm	Never
PERF:LBL 86	LBL 88	PERF:I??	Cycle image length units when D=px: mm mil um in cm	Never
PERF:LBL 67	LBL 95	PERF:CAL	Set or display image calibration when when D=px	Never
DCRD:PX>DC	LBL 75	DCRD:PX>DC	Design Coordinate given complex pixel coordinate	Always
DCRD:DC>PX	LBL 49	DCRD:DC>PX	Complex pixel coordinate given Design Coordinate	Always
DCRD:T>MM		DCRD:T>MM	Thirkell grid (letter+number) to coord in millimeters	Never
DCRD:MM>T		DCRD:MM>T	Coord in millimeters to Thirkell grid (letter+number)	Never
DCRD:ORIG	LBL 90	DCRD:ORIG		N/A
DCRD::CAL	LBL 95	DCRD:CAL	Set or display image calibration	Never
CAL	LBL 95	CAL	Set or display image calibration	Never

Notes:

- Please ignore the first two columns in the table – they are used to auto-generate the menu code for the application.
- Image calibration is not required all the time – see the last column of the table
- In examples I will generally leave off key strokes to get to the correct menu. i.e. I'll say [PX>D] instead of [LENG] [PX>D]
- Parameter setting functions don't drop the stack. Hitting [SHIFT] before one of these keys will report the current value.
- Functions like [PX>DC] that require a complex argument will do a [COMPLEX] when given real arguments.

## 4 [CAL]: Image Calibration

Note: The [CAL] function is located on most menu pages so it is always handy.

Image coordinates are measured in pixels. The upper left point is pixel (0,0). X values increase as we move to the right. Y values increase as we move down. This is the most common coordinate system used in computer graphics.

This application works with images with square pixels only. When using a scanner make sure the X & Y DPI settings are the same!

Calibration is retained between runs of the application. If the application is started up without calibration data, then a default will be set (2400 DIP scan with units in mm resulting in a calibration factor of 94.4881889764).

### 4.1 Known DPI

When using a scanner, the scanner DPI settings may be used to roughly calibrate an image.

#### 4.1.1 Examples

- 2400 DPI scans and using inches as the unit: 2400 [ENTER] 1 [CAL]
- 2400 DPI scans and using mm as the unit: 2400 [ENTER] 25.4 [CAL]
- 1000 DPcm scans using mm as the unit: 1000 [ENTER] 10 [CAL]

## 4.2 Using a target

Including a calibration target in the image field is the most common way to calibrate microscope images. For high magnification expensive, etched glass reticle are generally used. For the lower magnifications normally used by stamp collectors, a simple scale may be included in the image frame. I simply lay the stamps on top of a plastic ruler.

When using a calibration target it is rare that the target is precisely lined up horizontally or vertically, so providing a simple pixel count for its length is impossible. Instead we provide the pixel coordinates of two points on the target.

### 4.2.1 Example

We have a 1mm ruler in the frame. The 0 mark of the ruler is at pixel coordinates (123, 456) and the 3 mark of the ruler is at pixel coordinates (567, 345). If we wish to work directly in mm, then we do this:

- Enter the 0 coordinate: 123 [ENTER] 456 [COMPLEX] – assuming "RECT" mode
- Enter the 1 coordinate: 234 [ENTER] 345 [COMPLEX]
- Enter physical distance: 3
- Press the CAL menu key: [CAL]

If we wanted to work in cm instead of mm, we would use a distance of 0.3 instead.

If we wanted to work in mil instead of mm, we would use a distance of 118.11 instead

## 4.3 Using a Stamp!

The techniques documented here can be very useful, but must be used with care. Stamp paper can warp and deform – depending on printing process, storage conditions, etc. . .

Suppose you have an image of a stamp (perhaps from an auction website) that should have an design width of 22mm. You determine the upper left of the design is at pixel coordinates (300, 100), and the upper right of the design is at pixel coordinates (536, 101). Also suppose you wish to work in inches (you will see why in a moment):

```
322 [ENTER] 125 [COMPLEX] 582 [ENTER] 126 [COMPLEX] [-] 22 [ENTER] 25.4 [÷] [CAL]
```

Now notice the value on the screen is 300.1840. Many stamp images posted on auction sites are scanned at 300 DPI, and the fact this is so close to 300 tells us that we probably have a 300 DPI scan. We can now use 300 as our calibration data, or use the computed value.

You can use this technique with any feature of a stamp with a known measurement – not just design size. For example, some souvenir sheets have very precise paper sizes allowing the width or height of the paper to be used for calibration.

The perforations may also be used as a calibration aid. For example if we know the stamp in question has a perforation of 70 on the Kiusalas gauge, then we know 10 perforations should measure very close to 16mm.

## 4.4 Calibration factor

The "calibration factor", returned by [SHIFT] [CAL], is the current image calibration data in units of pixels/length.

### 4.4.1 Examples

- 2400 DPI scans and using inches as the unit: 2400
- 2400 DPI scans and using mm as the unit: 94.4881889764
- 1000 DPcm scans using mm as the unit: 100

## 5 [LENG]: Measuring lengths

### 5.1 Horizontal or Vertical Lengths

For horizontal & vertical lengths in the image, one simply needs to provided a pixel count as a real number.

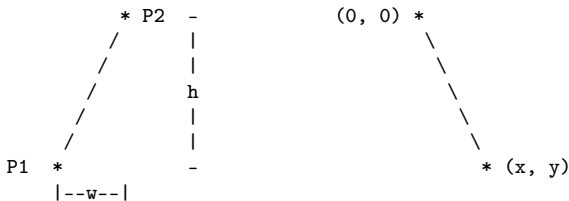
For stamps it can be handy to rotate the image so that interesting lengths are at 0 or 90 degrees. For example, when working with US Washington-Franklin issues it is a good idea to line up the design frame with the horizontal.

#### 5.1.1 Examples

- 2400 DPI image working in inches:
  - 1200 [PX>D] → 0.5
  - 2400 [PX>D] → 1.0
- The same image working in mm:
  - 1200 [PX>D] → 12.7
  - 2400 [PX>D] → 25.4

## 5.2 Generic distances

To measure the length of a non-horizontal/vertical line, we provide a complex number to the [PX>D] function. One might think of the coordinates as specifying a line from the origin, upper left image pixel at (0, 0), to the given coordinates. Alternately one might think of the coordinates as the width & height of the line. The signs of the coordinates are ignored – i.e. the absolute value of each coordinate is used.



To measure the length of a line segment given by two points, we simply enter both points as complex numbers and subtract them. We then give the difference to the [PX>D] function.

### 5.2.1 Examples

- For a 2400 DPI image working in mm (2400 [ENTER] 25.4 [CAL]):
  - 123 [ENTER] 456 [COMPLEX] 234 [ENTER] 345 [COMPLEX] [-] [PX>D] → 1.7300 mm
- For a 2400 DPI image working in inches (2400 [ENTER] 1 [CAL]):
  - 1 [ENTER] 1 [COMPLEX] 100 [ENTER] 100 [COMPLEX] [-] [PX>D] → 0.0583 in
- To compute the length in raw pixels, set cal to 1 (1 [ENTER] [CAL]):
  - 1 [ENTER] 1 [COMPLEX] 100 [ENTER] 100 [COMPLEX] [PX>D] → 140.0 PX

## 5.3 Philatelic Application: Rotary Press vs. Flat Plate (Take 1)

For an alternative approach see: Philatelic Application: Rotary Press vs. Flat Plate (Take 2)

### 5.3.1 Example 1

Scenario: We have a Washington-Franklin, and we would like to know if it was printed on a rotary press. We have a 2400 DPI scan.

For reference, here are the measurements for the various printing options:

Press	Width	Height
Flat	18.5-19mm	22mm
Rot Vert	18.5-19mm	22.5-23mm
Rot Horz	19.5-20mm	22mm

Method 1 (Stamp design not aligned) We begin by locating the coordinates for three frame corners:

Corner	Coordinates
Upper left	(150, 160)
Upper right	(1903, 161)
Lower left	(149, 2239)

- Calibration. If we have not already calibrated, we need to do that first
  - 2400 25.4 [CAL]
- Now we compute the width of our stamp design
  - 150 [ENTER] 160 [COMPLEX] [ENTER] [ENTER] 1903 [ENTER] 161 [COMPLEX] [-] [PX>D] → 18.55 mm
  - Note we duplicated the first coordinate so we can use it later...
- Now we compute the height of our stamp design
  - [X<>Y] 149 [ENTER] 2239 [COMPLEX] [-] [PX>D] → 22.00 mm

Our stamp is 18.55mm x 22.00mm – and so it must be a flat plate stamp.

Method 2 (Stamp design aligned) In this case the lower and upper frame lines of the design are perfectly horizontal. In this case we only need two points.

We begin by locating the coordinates for three frame corners (UL & LR, or UR & LL):

Corner	Coordinates
Upper left	(150, 160)
Lower right	(1902, 2240)

- Calibration. If we have not already calibrated, we need to do that first
  - 2400 25.4 [CAL]
- Now we compute the width of our stamp design
  - 150 [ENTER] 160 [COMPLEX] 1902 [ENTER] 2240 [COMPLEX] [-] [COMPLEX] [PX>D] → 22.01mm
- Now we compute the height of our stamp design
  - [Rv] [Rv] [PX>D] → 18.54mm

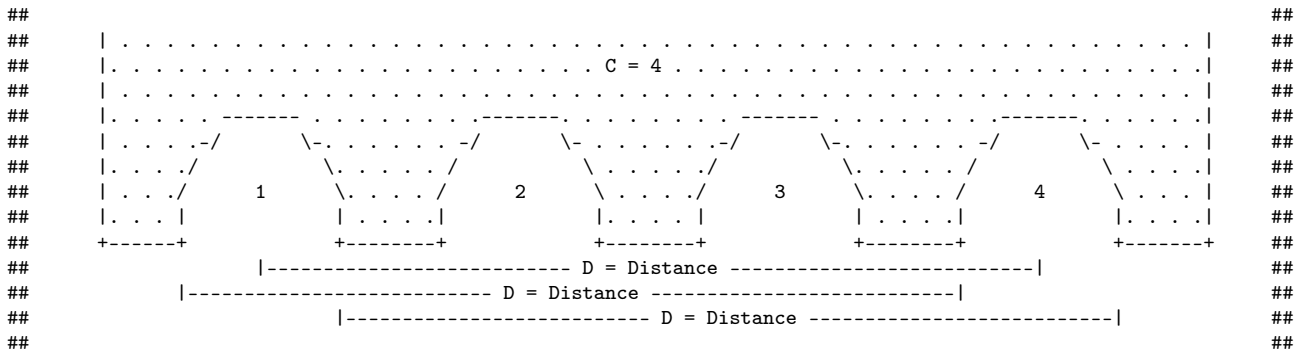
Our stamp is 18.54mm x 22.01mm – and so it must be a flat plate stamp.

## 6 [PERF]: Measuring & Converting Perforations

Perforations are usually measured in units of perfs/2cm; however, other units are used in some specialized areas of philately. For example, the Kiusalas gauge measures perforations in units of mil/perf – a more natural measure for USPS bureau issues. This application provides tools to convert between standard perforations and Kiusalas perforations ([P<>K]).

Also provided is a way to approximate standard perforation measurement by measuring the distance (D) spanned by a number of perforations (C).

Below is an illustration of how to measure distance and count perforations. First pick a recognizable feature in your starting and ending perforations, and measure the distance between them. The first distance in the illustration measures the distance between "perf bottoms". Take careful note of how we count – the number of "bottoms" in this case.



Distance (D), perforation count (C), traditional perforations (P) measured in perfs/2cm, and Kiusalas perforations (K) measured in mil/perf are related:

$$K = \frac{5000D}{127(C-1)}$$

$$P = \frac{20(C-1)}{D}$$

$$P = \frac{100000}{127K}$$

### 6.0.1 Example: Measuring Standard & Kiusalas Perforations

Scenario: We have a Washington-Franklin that should be perforated Kiusalas 70, and we wish to verify perfs. We have a 2400 DPI scan.

We find the coordinate of the first perf bottom at (150, 160) and 10 perfs over we have another bottom at (1662, 161).

- Calibration. If we have not already calibrated, we need to do that first
  - 2400 25.4 [CAL]
- Now we compute our perfs
  - 150 [ENTER] 160 [COMPLEX] 1662 [ENTER] 161 [COMPLEX] [D<>P] → 11.24859
- Next we compute the Kiusalas value
  - [P<>K] → 70.0000

And it looks like our stamp really is perforated Kiusalas 70!

## 7 [ANGL]: Measuring Angles

Angles are measured with respect the horizontal – the negative of the argument of the complex number representing the pixel coordinate. We use the negative because pixel y-coordinates go up in the down direction.

TIP: **[ANGL]** requires a complex number. If you give it something else it will call **COMPLEX** to convert the lower two stack levels into a complex number. This saves the user two key presses – i.e. you don't have to hit **[SHIFT] [COMPLEX]** to convert to complex before you use **[ANGL]**.

## 7.1 Examples

- 123 [ENTER] 123 [COMPLEX] [ANGLE]  $\rightarrow -45$  – assuming degree angle mode

## 7.2 Philatelic Application: Parallel Lines (Take 1)

For an alternative approach see: Philatelic Application: Parallel Lines (Take 2).

## Why?

- Verify that the two cut edges of a US coil stamp are parallel
- Verify that perforations are parallel
- Verify that the cut edge of a souvenir sheet is parallel with an engraved margin line
- Verify that design edges are parallel – to make sure an image has no perspective distortion

### 7.2.1 Example

Figure 1 illustrates four types of connections between two nodes, P1 and P2, and two other nodes, P3 and P4. The connections are labeled as follows:

- (a) P1 \* \* P3 (vertical line) P2 \* \* P4 (vertical line)
- (b) P1 \* \* P3 (vertical line) P2 \* \* P4 (vertical line) OR
- (c) P1 \* \* P3 (vertical line) P2 \* \* P4 (vertical line) OR
- (d) P1 \* \* P3 (vertical line) P2 \* \* P4 (vertical line) OR

- Angle 1: P1.x [ENTER] P1.y [COMPLEX] P2.x [ENTER] P2.y [COMPLEX] [-] [ANGLE] [STO 00]
- Angle 2: P3.x [ENTER] P3.y [COMPLEX] P4.x [ENTER] P4.y [COMPLEX] [-] [ANGLE] [STO 00]
- Difference: [RCL- 00] [ABS]

The result will be the absolute angle by which the two lines differ – i.e. if we have parallel lines it should be 0.

8 [DCRD]: Design Coordinates

Design Coordinates are used to locate points on a stamp referenced to the design. A few different schemes are in use with the "Thirkell" philatelic position finder probably the most popular. Most of these systems work the same way. They use the upper left corner of the stamp design as the origin of a coordinate system – i.e. they measure distance to the right and down from the upper left corner of the design. The most common unit of measurement is millimeters; however, a few also add a grid system on top. For example the Thirkell uses a 3mm grid.

```

+-----+
(      * P3      )
(  0 *-----+  )
(  |           |  )
(  |           |  ) - Design coordinates are used locate points on the stamp
(  |           |  ) - for a point (x,y) is (x-0_x, y-0_y)
(  |   * P1   |  ) - The most common units are mm
(  |           |  ) - Axis orientation: Coordinate values get bigger to the right or down
(  |           |  ) - These kinds of coordinates are frequently used to locate plate flaws
(  |           |  )
(  |           |  ) - Special cases
(  +-----+  ) - Points above or to the left of 0 will have negative coordinates -- Ex: P3
(      P2 *    ) - Points need not be inside the design -- Ex: P2 & P3
+-----+

```

When working with design coordinates you must first tell the application the coordinates of the origin by entering a complex number representing the pixel coordinates of the upper left corner of the design and hitting the **[ORIG]** button. If you shift press this button then it will display the current origin.

Now you can convert pixel coordinates to and from design coordinates.

Two functions are also available to convert standard millimeter design coordinates to and from Thirkell coordinates: `[MM>T]` & `[T>MM]`. Note that Thirkell coordinates are reversed – i.e. the first coordinate (the letter) is on the vertical axis and the second coordinate (the integer) is on the horizontal axis. These functions work in millimeters directly, and do not require the image to be calibrated; however, if you wish to use these functions in concert with `[PX>DC]` & `[DC>PX]` then you must calibrate your image and work in millimeters!

TIP: [MM>T], [PX>DC] & [DC>PX] require a complex number. If you provide something else they will call **COMPLEX** to convert the lower two stack levels into a complex number. This saves the user two key presses – i.e. you don't have to hit [SHIFT] [COMPLEX] to convert to complex before you use [ANGL].

## 8.1 Example

Scenario: You wish to report a new plate flaw to the nice people that make the Zumstein catalog using the Thirkell grid. We have a 2400 DPI scan.

First we locate the upper left hand corner of the design and our plate flaw:

Item	Pixel Coordinates
Upper left corner of design	(150, 160)
Center of our plate flaw	(713, 956)

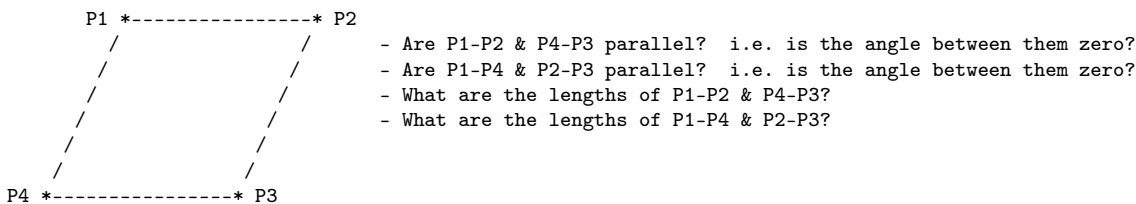
- Calibration. If we have not already calibrated, we need to do that first
  - 2400 25.4 [CAL]
- Next we enter the coordinate menu
  - [DCRD]
- We enter the origin point
  - 150 [ENTER] 160 [ORIG]
- Now we enter the coordinates of our flaw
  - 713 [ENTER] 956 [PX>DC] → 5.9584 + 8.424333i mm
- Finally we convert this to Thirkell
  - [MM>T] → Thirkell: 2C

## 9 [QUAD]: A Philatelic Power Tool

This is a handy tool that automates some common computations:

- Compute width & height of a rectangular stamp design
- Determine if two lines are parallel
- Determine if four points Determine a parallelogram – i.e. are the opposite sides parallel

It is called "QUAD" because it works with four points which will form the corners of a quadrilateral – also "QUAD" avoid confusion with the built in command RECT. Below is a picture of the geometric situation.



When using the QUAD tool, the first step is to input the corner coordinates via [UL], [UR], [LR], & [LL]. In this context:

Menu	Corner	Point Number
UL	Upper Left	Point P1
UR	Upper Right	Point P2
LR	Lower Right	Point P3
LL	Lower Left	Point P4

TIP: The data entry keys ([UL], [UR], [LR], & [LL]) each require a complex number. If you provide something else they will call COMPLEX to convert the lower two stack levels into a complex number. This saves the user two key presses – i.e. you don't have to hit [SHIFT] [COMPLEX] to convert to complex before you use [ANGL].

### 9.1 Philatelic Application: Parallel Lines (Take 2)

For an alternative approach see: Philatelic Application: Parallel Lines (Take 1).

A typical US Washington-Franklin vertical coil has a paper width of about 21.5mm. Because of the way the stamps were cut into coils, the flat sides should be absolutely parallel.

Scenario: We have a Washington-Franklin vertical coil, and wish to verify that the flat edges are parallel. We have a 2400 DPI scan.

Our first step is to identify two points in each edge. The points should be separated as far apart as possible. Suppose this results in the following:

Corner	Coordinates
Upper left	(150, 160)
Upper right	(2182, 122)
Lower right	(2181, 2250)
Lower left	(149, 2132)

Note that we didn't try to line the points up horizontally – our upper left point is at 160 while our upper right point is at 122!

- Calibration. If we have not already calibrated, we need to do that first
  - 2400 25.4 [CAL]
- Now we enter the corner data
  - [QUAD] 150 [ENTER] 160 [UL] 2182 [ENTER] 122 [UR] 2181 [ENTER] 2250 [LR] 149 2132 [LL]
- Now we can compute the LR Angle:
  - [ $\angle$ LR]  $\rightarrow$  0.0021 – assuming DEG angle mode

## 9.2 Philatelic Application: Rotary Press vs. Flat Plate (Take 2)

For an alternative approach see: Philatelic Application: Rotary Press vs. Flat Plate (Take 1)

Scenario: We have a Washington-Franklin, and we would like to know if it was printed on a rotary press. We have a 2400 DPI scan. For reference, here are the measurements for the various printing options:

Press	Width	Height
Flat	18.5-19mm	22mm
Rot Vert	18.5-19mm	22.5-23mm
Rot Horz	19.5-20mm	22mm

We begin by locating the coordinates for four frame corners:

Corner	Coordinates
Upper left	(150, 160)
Upper right	(1903, 161)
Lower right	(1902, 2240)
Lower left	(149, 2239)

- Calibration. If we have not already calibrated, we need to do that first
  - 2400 25.4 [CAL]
- Now we enter the corner data
  - [QUAD] 150 [ENTER] 160 [UL] 1903 [ENTER] 161 [UR] 1902 [ENTER] 2240 [LR] 149 2239 [LL]
- Now we can compute the width:
  - [WIDE] → 18.553 mm
- And the height:
  - [TALL] → 22.00 mm

Our stamp is 18.553mm x 22.00mm – and so it must be a flat plate stamp.

## 10 Code

## 10.1 Menu Code

```
(MJR-generate-42-menu-code "PHIL" 0 tbl 0 1 'stay 'up #'MJR-custom-gen-lab #'MJR-custom-gen-sub)
```

```

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@      (PHIL)
@@@ DSC: Auto-generated menu program
LBL "PHIL"
    LBL 01                @@@ Page 1 of menu PHIL
    CLMENU
    XEQ 85
    KEY 1 XEQ 92
    "LENG"
    KEY 2 GTO 02
    "QUAD"
    KEY 3 GTO 03
    "PERF"
    KEY 4 GTO 04
    "DCRD"
    KEY 5 GTO 05
    "CAL"
    KEY 6 XEQ 95
    KEY 9 GTO 00
    MENU
    STOP
    GTO 01
    LBL 02                @@@ Page 1 of menu LENG
    CLMENU

```



```

"PX>D"
KEY 1 XEQ 91
"D>PX"
KEY 2 XEQ 66
"CAL"
KEY 6 XEQ 95
KEY 9 GTO 01
MENU
STOP
GTO 02
LBL 03          @@@@ Page 1 of menu QUAD
CLMENU
"UL"
KEY 1 XEQ 57
"UR"
KEY 2 XEQ 58
"LR"
KEY 3 XEQ 59
"LL"
KEY 4 XEQ 60
"CAL"
KEY 6 XEQ 95
KEY 7 GTO 06
KEY 8 GTO 06
KEY 9 GTO 01
MENU
STOP
GTO 03
LBL 06          @@@@ Page 2 of menu QUAD
CLMENU
"∠TB"
KEY 1 XEQ 62
"∠LR"
KEY 2 XEQ 63
"WIDE"
KEY 3 XEQ 64
"TALL"
KEY 4 XEQ 65
KEY 7 GTO 03
KEY 8 GTO 03
KEY 9 GTO 01
MENU
STOP
GTO 06
LBL 04          @@@@ Page 1 of menu PERF
CLMENU
"D<>P"
KEY 1 XEQ 96
"P<>K"
KEY 2 XEQ "P<>K"
XEQ 87
KEY 3 XEQ 98
XEQ 89
KEY 4 XEQ 99
XEQ 86
KEY 5 XEQ 88
XEQ 67
KEY 6 XEQ 95
KEY 9 GTO 01
MENU
STOP
GTO 04
LBL 05          @@@@ Page 1 of menu DCRD
CLMENU
"PX>DC"
KEY 1 XEQ 75
"DC>PX"
KEY 2 XEQ 49
"T>MM"
KEY 3 XEQ "T>MM"
"MM>T"
KEY 4 XEQ "MM>T"
"ORIG"
KEY 5 XEQ 90
KEY 9 GTO 01
MENU

```

## 10.2 Local Functions

[illegible][illegible]

~~~~~

[illegible]

10

ARCL ST X  
AVIEW  
RTN

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

@@@ NAM: NUM-P 61  
LBL 61  
FUNC 01  
L4STK  
0  
LSTO "\_TMPC"  
SF 25  
RCL "PhilP1"  
FS?C 25  
ISG "\_TMPC"  
NOP  
SF 25  
RCL "PhilP2"  
FS?C 25  
ISG "\_TMPC"  
NOP  
SF 25  
RCL "PhilP3"  
FS?C 25  
ISG "\_TMPC"  
NOP  
SF 25  
RCL "PhilP4"  
FS?C 25  
ISG "\_TMPC"  
NOP  
RCL "\_TMPC"  
RTN

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

@@@ DSC: Action for  $\angle$ TB  
@@@ NAM: TB\_ANGLE 62  
LBL 62  
FUNC 01  
L4STK  
XEQ 61  
4  
X $\neq$ Y?  
GTO 56  
RCL "PhilP2"  
RCL- "PhilP1"  
XEQ 37 @@@@ ANGLE  
RCL "PhilP3"  
RCL- "PhilP4"  
XEQ 37 @@@@ ANGLE  
-  
RTN

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

@@@ DSC: Action for  $\angle$ LR  
@@@ NAM: LR\_ANGLE 63  
LBL 63  
FUNC 01  
L4STK  
XEQ 61  
4  
X $\neq$ Y?  
GTO 56  
RCL "PhilP4"  
RCL- "PhilP1"  
XEQ 37 @@@@ ANGLE  
RCL "PhilP3"  
RCL- "PhilP2"  
XEQ 37 @@@@ ANGLE  
-  
RTN

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

@@@ DSC: Action for WIDE  
@@@ NAM: LR\_ANGLE 64  
@@@ OUT: Z: Top Width

```

@@@@ OUT: Y: Bottom Width
@@@@ OUT: X: Average Width
LBL 64
FUNC 03
L4STK
XEQ 61
4
X≠Y?
GTO 56
RCL "PhilP2"
RCL- "PhilP1"
XEQ 91 @@@@ Convert PX to length
RCL "PhilP3"
RCL- "PhilP4"
XEQ 91 @@@@ Convert PX to length
RCL ST Y
RCL ST Y
+
2
÷
RTN

```

```

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@ DSC: Action for TALL
@@@@ NAM: LR_ANGLE 65
@@@@ OUT: Z: LEFT
@@@@ OUT: Y: RIGHT
@@@@ OUT: X: Average Width
LBL 65
FUNC 03
L4STK
XEQ 61
4
X≠Y?
GTO 56
RCL "PhilP4"
RCL- "PhilP1"
XEQ 91 @@@@ Convert PX to length
RCL "PhilP3"
RCL- "PhilP2"
XEQ 91 @@@@ Convert PX to length
RCL ST Y
RCL ST Y
+
2
÷
RTN

```

```

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@ DSC: ERR: Enter Data First
@@@@ FAQ: NOT A FUNCTION. GTO Target
@@@@ NAM: EEDF 56
LBL 56
"ERR: Enter"
└─" Corners"
AVIEW
RTN

```

```

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@ NAM: DCRD:ORIG 90
LBL 90
FUNC 00
L4STK
REAL?
COMPLEX
FC? 64
STO "PhilCo"
XEQ 45 @@@@ GET "PhilCo"
"Orig: "
ARCL ST X
AVIEW
RTN

```

```

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@ DSC: ACTION for DCRD:PX>DC
@@@@ NAM: DCRD:PX>DC 75

```





```

└" mm"
AVIEW
RTN
LBL 47
"Enter Thir"
└"kell; R/S"
AON
STOP
AOFF
GTO 40
RTN

```

```

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@ DSC: ANGL Label 85
LBL 85
FUNC 00
L4STK
RECT @@@@ We require RECT mode!
"ANGL"
XEQ 79 @@@@ GET "PhilIU"
XEQ 77 @@@@ GET "PhilDU"
XEQ 78 @@@@ GET "PhilPC"
XEQ 76 @@@@ GET "PhilIC"
RTN

```

```

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@ DSC: CAL in PERF Label 67
LBL 67
FUNC 00
L4STK
XEQ 77 @@@@ GET "PhilDU"
X≠0? @@@@ IF-BOOL
RTNNO
"CAL"
RTNYES

```

```

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@ DSC: I: Label 86
LBL 86
FUNC 00
L4STK
CLA
XEQ 77 @@@@ GET "PhilDU"
X≠0? @@@@ IF-BOOL
RTNNO
"I:"
XEQ 79 @@@@ GET "PhilIU"
70
+
XEQ IND ST X
R↓
RTNYES

```

```

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@ DSC: D: Label 89
LBL 89
FUNC 00
L4STK
"D:"
XEQ 77 @@@@ GET "PhilDU"
50
+
XEQ IND ST X
R↓
RTN

```

```

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@ DSC: C: Label 87
LBL 87
FUNC 00
L4STK
"C:"
XEQ 78 @@@@ GET "PhilPC"
AIP
R↓
RTN

```

```

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@ DSC: Action for menu key ANGLE
@@@ NAM: ANGLE 92
LBL 92
L4STK
REAL?
COMPLEX
XEQ 37 @@@@ CPLX-F00
RTN

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@ NAM: CPLX-ANGLE 37
@@@ DSC: CPLX-ANGLE complex input only
LBL 37
FUNC 11
L4STK
FC? 73 @@ IF-RECT
GTO 42
GTO 31
LBL 42 @@ IF-THEN RECT MODE
POLAR
COMPLEX
RECT
GTO 44
LBL 31 @@ IF-ELSE POLAR MODE
POLAR
COMPLEX
RECT
LBL 44 @@ IF-END
X<>Y
R↓
RTN

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@ DSC: Action for menu key CAL
@@@ NAM: CAL 95
LBL 95
FUNC 00          @### REQ:free42>=2.5.24
L4STK            @### REQ:free42>=3.0
FS? 64
GTO 29
X<>Y
ABS
X<>Y
÷
STO "PhilIC"
LBL 29
RCL "PhilIC"
"Cal: "
ARCL ST X
AVIEW
RTN

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@ DSC: Action for menu key D→P
@@@ NAM: D→P 96
LBL 96
FUNC 11          @### REQ:free42>=2.5.24
L4STK            @### REQ:free42>=3.0
XEQ 39 @@@@ Convert distance to mm
XEQ 78 @@@@ GET "PhilPC"
1
-
X<>Y
÷
20
×
RTN

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@ DSC: Action for menu key P<>K
LBL "P<>K"
FUNC 11          @### REQ:free42>=2.5.24
L4STK            @### REQ:free42>=3.0
127

```



```

*
100000
X<>Y
÷
RTN
RTN

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@ DSC: Action for menu key C:
@@@@ NAM: C: 98
LBL 98
FUNC 00          @### REQ:free42>=2.5.24
L4STK           @### REQ:free42>=3.0
STO "PhilPC"
RTN

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@ DSC: Action for menu key D:UNIT
@@@@ NAM: D: 99
LBL 99
FUNC 00          @### REQ:free42>=2.5.24
L4STK           @### REQ:free42>=3.0
XEQ 77 @@@@ GET "PhilDU"
1
+
6
MOD
STO "PhilDU"
R↓
RTN

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@ DSC: Action for menu key I:UNIT
@@@@ NAM: I: 88
LBL 88
FUNC 00          @### REQ:free42>=2.5.24
L4STK           @### REQ:free42>=3.0
XEQ 79 @@@@ GET "PhilIU"
1
+
5
MOD
STO "PhilIU"
R↓
RTN

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@ DSC: Convert distance to mm
LBL 39
FUNC 01          @### REQ:free42>=2.5.24
L4STK           @### REQ:free42>=3.0
XEQ 77 @@@@ GET "PhilDU"
X=0?           @@@@ IF-BOOL
GTO 34
GTO 35
LBL 34           @@@@ IF-THEN (pixels)
R↓
XEQ 91 @@@@ Convert PX to I units
XEQ 79 @@@@ GET "PhilIU"
80
+
XEQ IND ST X
GTO 36
LBL 35           @@@@ IF-ELSE (not-pixels)
80
+
1
-
XEQ IND ST X
LBL 36           @@@@ IF-END
RTN

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@ DSC: Action for menu key PX>D
@@@@ NAM: PX>D 91
LBL 91

```

```

FUNC 11          @### REQ:free42>=2.5.24
L4STK           @### REQ:free42>=3.0
ABS
XEQ 76 @@@@ GET "PhilIC"
÷
RTN

```

```

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@ DSC: Action for menu key D>PX
@@@ NAM: D>PX 66
LBL 66
FUNC 11          @### REQ:free42>=2.5.24
L4STK           @### REQ:free42>=3.0
ABS
XEQ 76 @@@@ GET "PhilIC"
×
RTN

```

```

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@ DSC: Names for D units
LBL 50
"|-px"
RTN
LBL 51
"|-mm"
RTN
LBL 52
"|-mil"
RTN
LBL 53
"|-μm"
RTN
LBL 54
"|-in"
RTN
LBL 55
"|-cm"
RTN

```

```

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@ DSC: @@@@ Names for I units
LBL 70
"|-mm"
RTN
LBL 71
"|-mil"
RTN
LBL 72
"|-μm"
RTN
LBL 73
"|-in"
RTN
LBL 74
"|-cm"
RTN

```

```

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@ DSC: Conversion to mm.
LBL 80          @@@@ "|-mm"
R↓
RTN
LBL 81          @@@@ "|-mil"
R↓
25.4
×
1000
÷
RTN
LBL 82          @@@@ "|-μm"
R↓
1e3
÷
RTN
LBL 83          @@@@ "|-in"
R↓

```

[illegible]

RTN

[illegible]

END

11 EOF