Root Locus 0.2

by Antonio Niño Díaz

1. Introduction

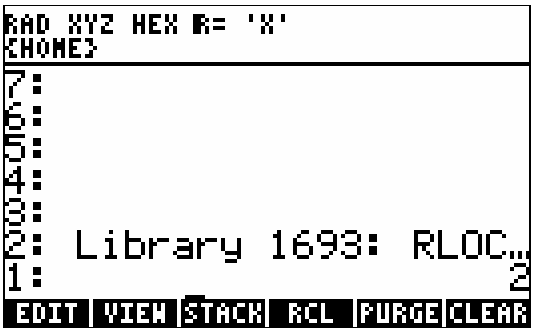
This program allows you to see the root locus of any given transfer function and add a regulator. You can quickly modify the value of 'k' and see if the system with that gain would be stable or unstable. You can also see the poles positions for that gain.

It has been created using HP-GCC 2.0, so it can only run on an HP 50g. It isn't needed to install ARM ToolBox in your calculator, though, the program comes in a way it will execute itself without using it.

IMPORTANT NOTE: This program was created with HP-GCC, that means that if it crashes there is no way to return to the HP menu by software means. In case that happens, just press the reset button (or remove a battery and put it back again after a second). This program has been tested a lot, it shouldn't crash, but it is better to advise, right? You may even lose the contents of the non-flash memory (ports HOME, 0 and 1). You should do backups of your files just in case.

2. Installation

Copy RLOCUS2.LIB to a SD card or transfer it using a USB cable to the calculator. In RPN mode, put it in the stack. Then put the number of the port where you want to install it (0, 1 or 2, 2 preferably) in the stack so you have something like this:

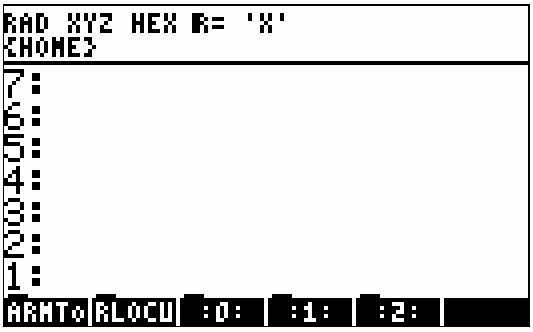


```
RAD XYZ HEX R= 'X'
{HOME}
7:
6:
5:
4:
3:
2: Library 1693: RLOC…
1:                   2
```

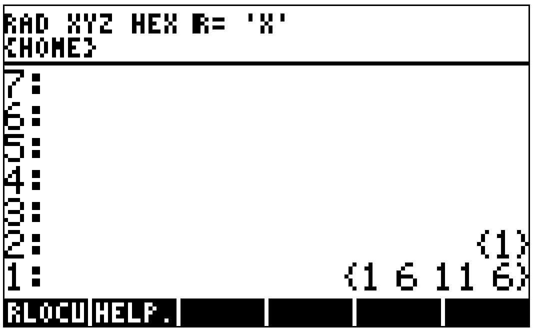
Press 'STO>' to install it, then hold 'ON' and press 'F3' to perform a soft reset. In that way, it will be loaded and you will be able to use it. If there is another library with the same number (1693) it won't be able to install this library. Please, communicate it to me so that I can change the number (or the creator of the other library).

3. Usage

Press 'RSHIFT' and then '2' ('LIB' command), and press 'NEXT' until you see 'RLOCU', like the left picture. Then, enter 2 lists and press 'RLOCU' ('HELP.' is a small text with some instructions about the program).



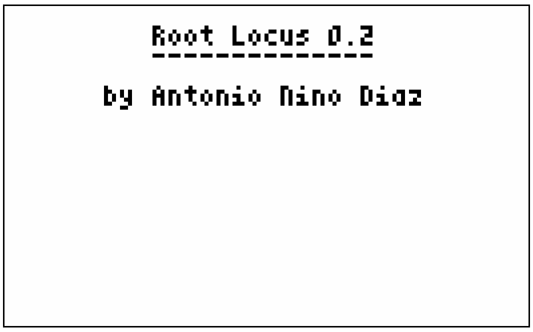
```
RAD XYZ HEX R= 'X'
{HOME}
7:
6:
5:
4:
3:
2:
1:
```



```
RAD XYZ HEX R= 'X'
{HOME}
7:
6:
5:
4:
3:
2:                 {1}
1:          {1 6 11 6}
```

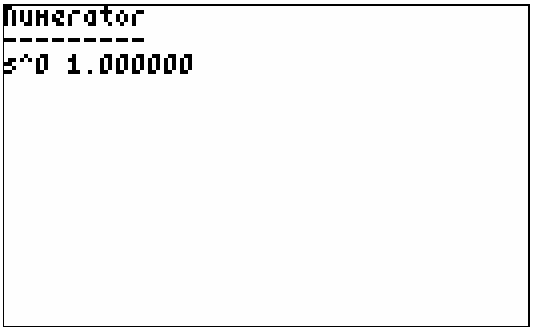
You need at least 50 KB free in port HOME to be able to run it, it will return an error if not. You can also insert lists as strings, like: "{ 1 6 11 6 }". The numbers don't have to be integers, you can type numbers like '-12.3E-4', but if you use numbers too high (or too small) it will overflow (or underflow) the C variables. This program uses 'double' data type, which range is ±1.7E±308 (15 significant digits). You shouldn't try to use too high numbers, it will probably fail. This program should be able to handle up to 10th degree polynomials, but it is quite slow for such complex polynomials, so I wouldn't use more than 5th degree. The list in level 2 (the first one) is the numerator of the transfer function and the one in level 1 (the second one) is the denominator. For example, {1 2 3} = s^2 + 2*s + 3. The transfer function of the sample screen is: G(s)=1/(s^3 + 6*s^2 + 11*s + 6).

When you press 'RLOCU' the calculator will show this screen:

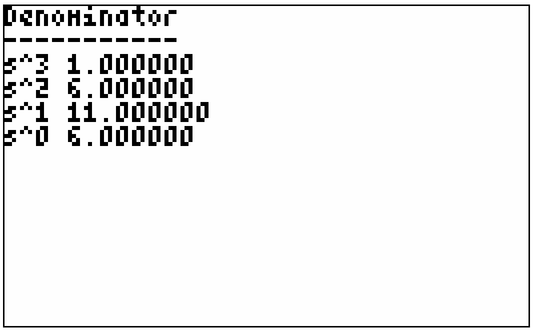


```
Root Locus 0.2
--------------

by Antonio Nino Diaz
```

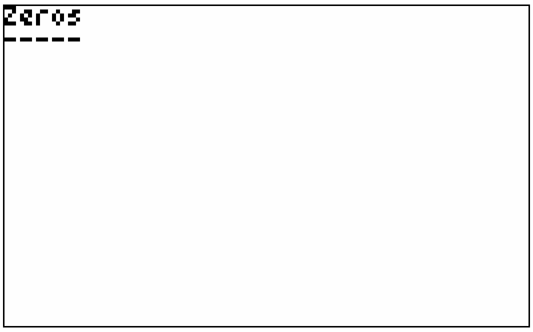
You can press 'ON' to skip the next four screens, or any other key to go to the next one:



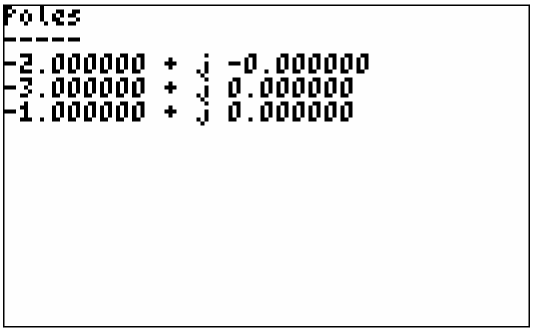
```
Numerator
---------
s^0 1.000000
```



```
Denominator
-----------
s^3 1.000000
s^2 6.000000
s^1 11.000000
s^0 6.000000
```

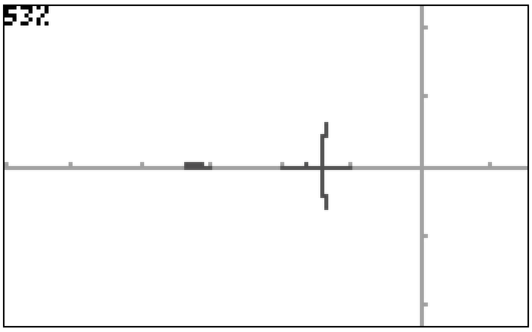


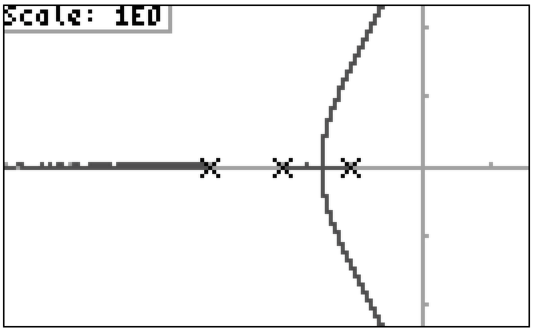
```
Zeros
-----
```



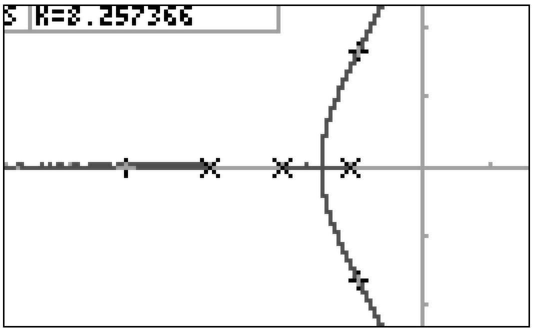
```
Poles
-----
-2.000000 + j -0.000000
-3.000000 + j 0.000000
-1.000000 + j 0.000000
```

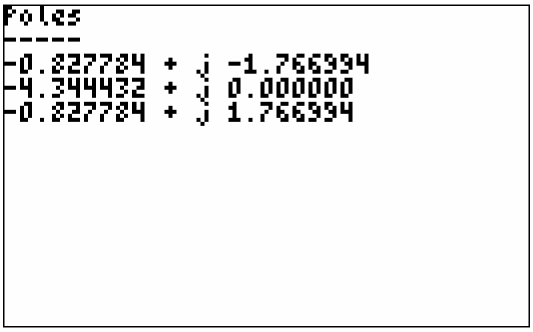
Once you finish with them (you can press 'ON' at any screen to skip them all) you will get to the graphic screen. You can just wait until it finishes or press 'ENTER' to make it 10 times faster (although it will lose quality). You can press 'ON' to cancel. Once it finishes it will show the scale of the axises in the upper left corner of the screen. The scale is calculated automatically. While drawing, the battery usage is higher, so it may be a good idea to make it go faster.





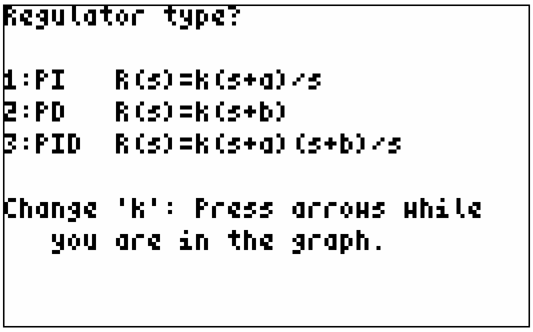
Now you can press 'ON' to exit the program, 'ALPHA' to add a regulator or the arrows (up/down - fast, right/left-slow) to change the value of 'k'. If you press any arrow you will be able to move through the graphic, and you will see in the upper left of the screen a box with a letter ('S' or 'U' for stable or unstable) that tells you the stability of the system for that value of 'k', and the current value of 'k'. If you press 'ENTER' it will tell you the current positions of the poles. If you press 'BACKSPACE' it will hide the poles and show the scale again.



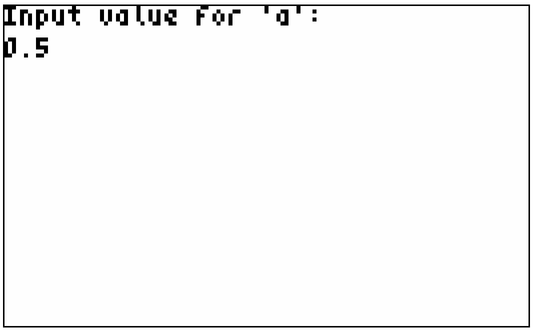


```
Poles
-----
-0.827784 + j -1.766994
-4.344432 + j 0.000000
-0.827784 + j 1.766994
```

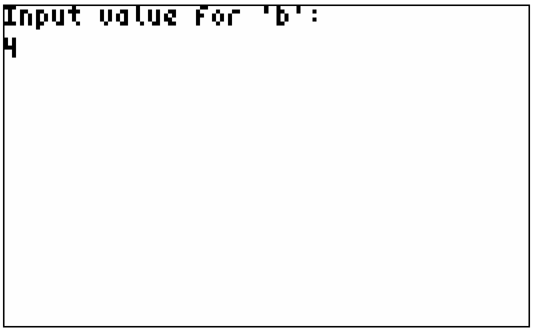
To add a regulator, just press ALPHA while in the graphic screen. You will be asked for the type of regulator (you can't choose a P regulator because you can do that by changing the value of 'k' just like explained before). Just press the key of the number for the regulator you want to add. Then you will be asked for its parameters (one for PI and PD regulators, two for PID regulator). You have to write the numbers like '-1.2E-3', '.5', '12222', etc, and then press 'ENTER'. If the regulator has been successfully added, it will show a confirmation message:



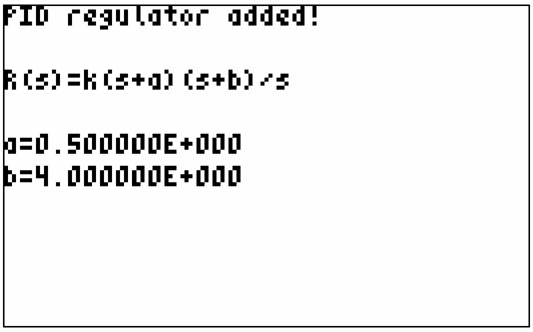
```
Regulator type?

1:PI   R(s)=k(s+a)/s
2:PD   R(s)=k(s+b)
3:PID  R(s)=k(s+a)(s+b)/s

Change 'k': Press arrows while
   you are in the graph.
```



```
Input value for 'a':
0.5
```



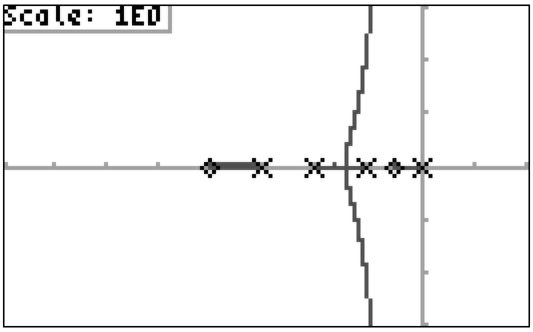
```
Input value for 'b':
4
```

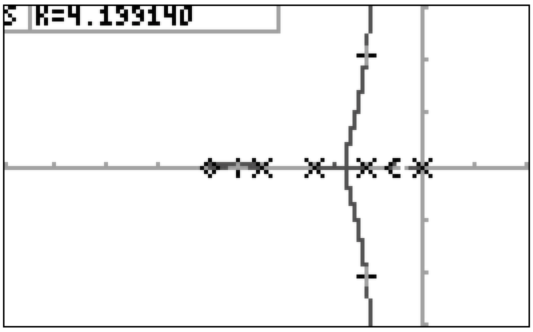


```
PID regulator added!

R(s)=k(s+a)(s+b)/s

a=0.500000E+000
b=4.000000E+000
```

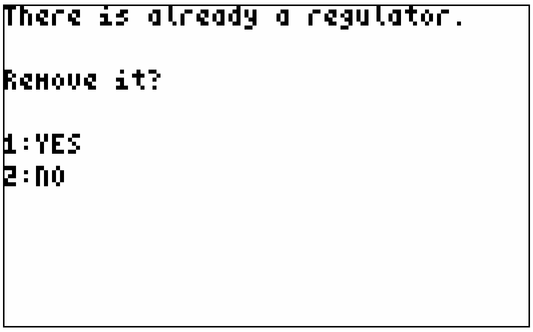
After the regulator has been added, it will draw the new root locus, and you will be able to do the same things as to the other graphic (change the value for 'k', press 'ENTER' to show the poles for that value, press 'ON' to exit...).

<u>The value for 'k' is the real value for the regulator</u>, not just the value for the root locus drawing.





If you want to change the regulator (no, you can't add another one without removing the first one) just press 'ALPHA' and it will ask you to confirm. Press 1 to remove it or 2 to cancel.



```
There is already a regulator.

Remove it?

1:YES
2:NO
```

If you remove it it won't have to redraw the original root locus, it has been saved. It only has to draw again when you add another regulator.

4. Credits

This program was created with:

ARM ToolBox 3.12 by Claudio Lapilli

HP-GCC 2.0, Copyright (c) 2004-2007 HP-GCC Development Team

HPAPINE by Khanh-Dang NGUYEN THU-LAM

Polynomial Root Finders, Copyright (c) 1992, 1993, 1994 LNT, University of Erlangen Nuernberg, FRG and Rice University, Houston, TX

5. Contact me

My email is antonio_nd@hotmail.com. Please, tell me if you find any error so that I can fix it (or at least try to fix it). I'm a student, so I may not have time to do it (or knowledge, I'm studying Industrial Engineering, not Informatics Engineering). Anyway, the source code is available, it comes with the compiled library. If you want to modify it, go ahead. This program is licensed under the GPL license, so you will have to keep the program open-source. You can visit my web to check for new versions:

http://antoniond.drunkencoders.com/
http://antoniond_blog.drunkencoders.com/

6. License