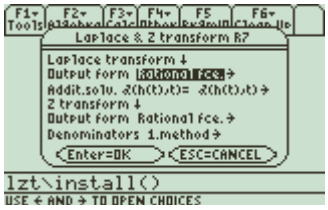
1. Installation

- Unpack a file *lzt.zip*
- Send a file *lztR7.89g* to your calculator though graph-link cable and software. In the calc a folder *lzt* is appearing.
- Unpack a file *kerno.zip* under *distributed_files* and send *KerNO.89z* to the calc. If your calculator is HW2 or HW3 version send also *hw3patch.89z.* The HW should be determined by pressing F1 and A.

- Install KERNO – On calculator type *kerno()* and press EXE. If HW2 or HW3 version install hw3patch before – type *hw3patch().*
- Install LZT – Type *lzt\install().*

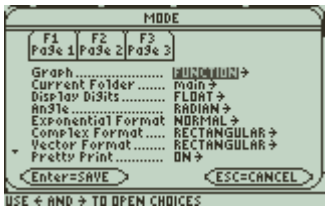


Choose whether an answer expressions shall be of common denominator form at all cost (*Rational fce.*) or of partial fractions form (Partial frac.).

The Installation process creates a custom menu. It can be switched between custom and main menu by pressing 2nd and CUSTOM.

Some items of mode options are changing within the installation.

ANGLE is set to RADIAN
COMPLEX FORMAT is set to RECTANGULAR
EXACT/APPROX is set to AUTO

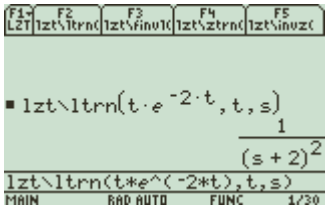


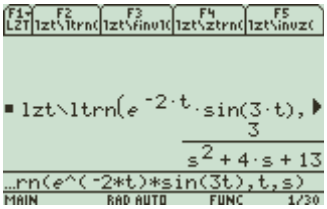
Keep these options when you are using LZT.

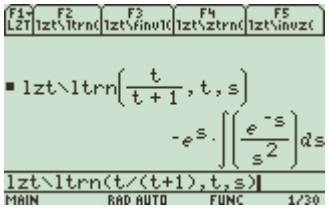
2. Direct Laplace transformation

lzt \ ltrn (f (t) , t , s)

f (t)	a time dependent function you want to transform
t	an independent continuous time variable
s	an independent complex variable of Laplace transform





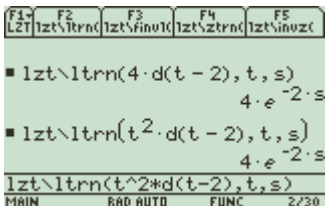


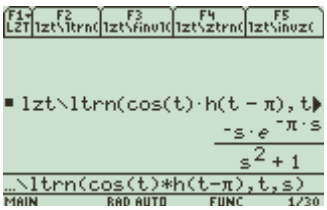
Unit step, Dirac pulse

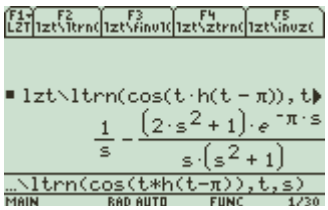
h (t – t₀)	an unit step, *h (t – t₀) = 0 when t < t₀, else = 1*
d (t – t₀)	a dirac pulse in time *t₀*

possibly

h (p(t)), d (p(t)) where *p(t)* is a polynomial function of real roots



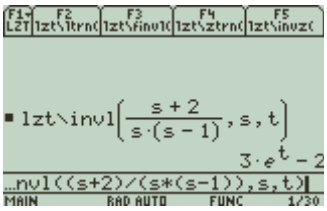




3. Inverse Laplace transformation

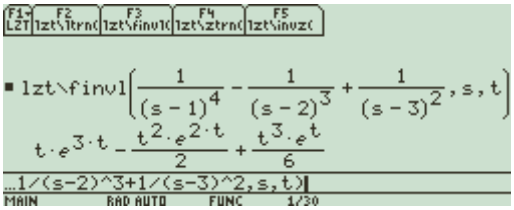
lzt \ invl (F (s) , s , t)

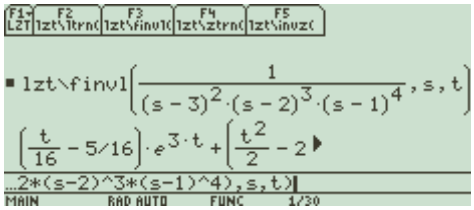
F (s)	Laplace transform of *f (t)*
s	the independent complex variable of Laplace transform
t	the independent continuous time variable



Inverse transformation of expressions with several multiple roots or fractions

lzt \ finvl (F (s) , s , t)



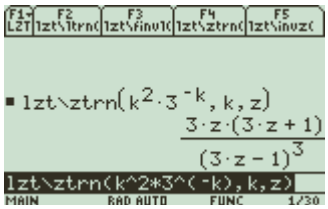


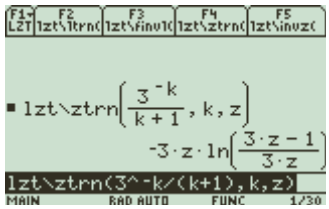
In such a case will *finvl* evaluate a result many times faster than *invl*. When the evaluation is taking a long time then break it (press ON) and try it the other way.

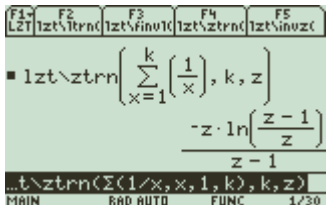
4. Direct Z transformation

lzt \ ztrn (f (k) , k , z)

f (k)	a discrete function you want to transform
k	an independent discrete variable
z	an independent complex variable of Z transform





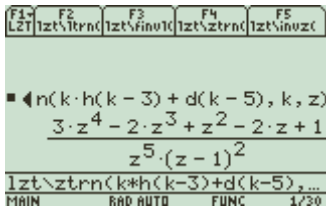


Unit step, Dirac pulse

h (k – k₀)	the unit step, *h (k – k₀) = 0 when k < k₀, else = 1*
d (k – k₀)	the dirac pulse, *d (k – k₀) = 1 when k = k₀, else = 0*

possibly

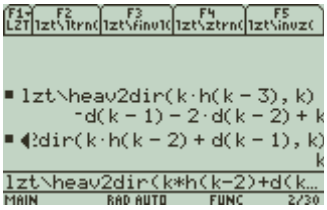
h (p(k)), d (p(k)) where *p(k)* is a polynomial function of real roots



Conversion of Unit steps to discrete Dirac pulses

lzt \ heav2dir (f (k) , k)

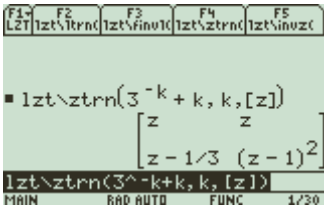
f (k) a discrete function containing one or more different unit steps



This function may simplify an answer of ***invz*** tool, when it contains more than one unit step or the unit step and dirac pulses at once.

Answer with separated numerators and denominators of partial fractions

lzt \ ztrn (f (k) , k , [z])

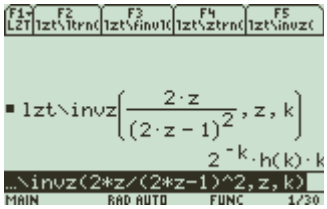


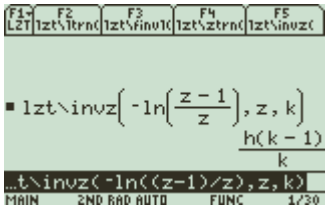
The result is represented by matrix. First row contains numerators and second row contains denominators. Each column represents a partial fraction and whole result is given by sum of all partial fractions. This is a way how to prevent TI89 software from an attempt to simplify the result expression in each step which may be too slow in some cases.

5. Inverse Z transformation

lzt \ invz (F (z) , z , k)

F (z)	Z transform of *f (k)*
z	the independent complex variable of Z transform
k	the independent discrete variable





finvz works just as ***finvl***. See the section 3.

6. FAQ

Q: What to do when an error "Invalid program reference" occurred

A: TI-89 operation system v2.0 or higher contains a restriction that disallow assembly programs return an answer. LZT use assembly programs. There are patches and kernels that break this and others restrictions. In a folder "\distributed files" are two. First is KERNO by Greg Dietsche and second is PREOS by Patrick Pelissier. You must install one to LZT works. See the documentation included. There are always new versions of the operation system and now is here a new version of TI-89 named Titanium. May become that LZT with these versions of kernels will not work. In such a case look for new version or other patch that break the restriction of the operation system. Here are some of places where to look.

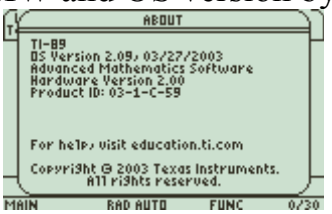
www.ticalc.org
http://www.ticalc.org/pub/89/asm/shells/date.html
http://calc.gregd.org/
http://www.tigen.org/kevin.kofler/ti89prog/

7. Help to solve compatibility issues

As you know LZT has some compatibility issues on latest HW and OS version. Causes “Invalid program reference” exception and others. These issues are caused by exploiting assembly programs since the OS v.2 and higher restricts assembly program to return an expression. There are patches breaking SW restrictions and HW differences. But because I haven't owned all HW versions of TI-89, TI-92+, Voyage and Titanium of course. The truth is I've had only HW v2.0 so I'm not able to check every possible combinations of HW and OS versions to determine how to fix the issue on a particular HW and SW version. On this subject I'm asking you for help other users. If your HW or OS version is 2.00 or higher at least then please be so kind and let me now the way you fix the restriction.

If you've decided to help then please

check your HW and OS version by pressing F1 and A.



Send a message to this address lzt.ti89@gmail.com. The address has been specially dedicated to this purpose now. Into the message subject please mention a type of your calculator, HW version and OS version. Into the message body please mention what have you done the lzt works.

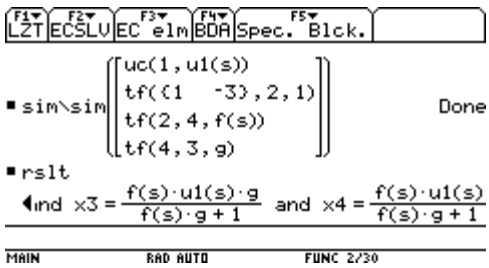
e.g.

Subject: **TI-89, HW version 3.00, OS version 3.05**
I've installed hw3patch103 and preos104 and then lzt works or I've installed kerno31 and lzt still asserts “Invalid program reference” exception.

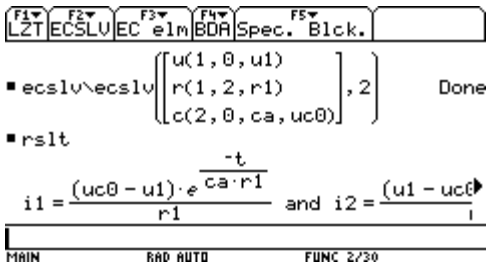
I'm going to include instal instructions for the particular combinations of HW and OS versions into this document on the basis of information from you.

8. Shells for Laplace and Z transform

If you deal with automation or simulation of dynamic systems then may be useful for you a program for calculation block diagram algebra**.** In a zip-archive "\LZT_shell\bda.zip" is the program named **Block diagram algebra** simulates like MATLAB Simulink but symbolically. It doesn't have a GUI, the diagram description is entered by a command line.



Electric circuit solver is a program for symbolic simulation of electric circuits, specially suited for dynamic process in RLC circuits. The circuit description is entered by the command line. Install it from "\LZT_shell\ecslv.zip".



You should know that this programs needs Laplace and Z transformation to be installed otherwise they can't fully works.