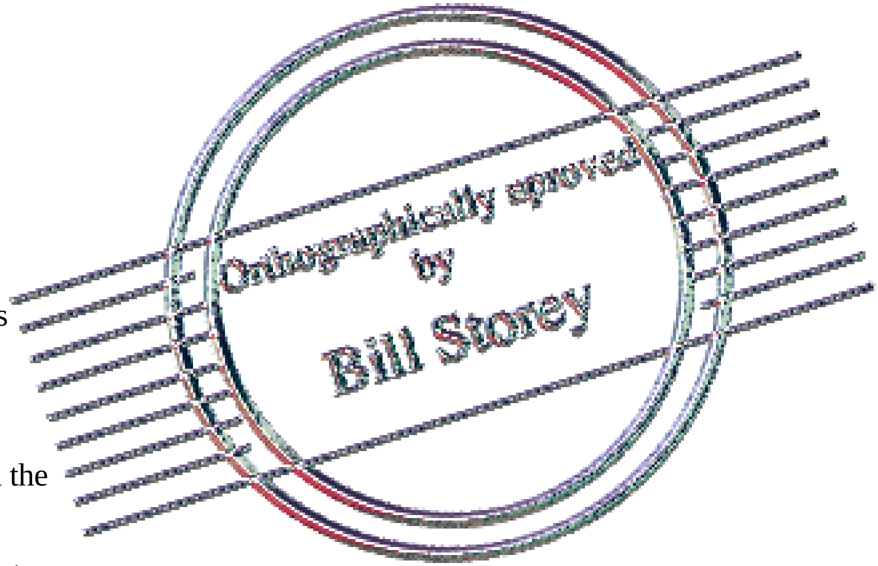
Trigonometry with the HP49G
(and with the HP48)

By Nick Karagiaouroglou



Many, many special thanks to:

Adolph Weidanz	for the endless interest, for all the questions which made me see clearer what I could express better.
Alban Schann	for all the ideas, corrections and the enthusiasm with which he supported me all the time.
Bill Storey	for reading the whole marathon and marking up orthographic errors.
Giorgos Illias	for his suggestion to me, to make an additional trigonometry marathon part for the HP48.
Máximo Castañeda Riloba	for the in depth going comments and ideas about pattern matching and its potential dangers.
Thomas Rast	for keeping on sending me his remarks about errors in mathematical formulae, for his ideas about formuale layout and font usage for program listings and for the humour.
Veli-Pekka Nousiainen	for his remarks on conceptual errors in programs, for his ideas about key pressing conventions and also for the humour.

And also thanks a lot to all the guys out there who kept on powering me up and supported me while I was writing this trigonometric marathon.

Thanks you so much guys, you are magnificent!

Key pressing conventions

→NUM	Right shifted key on the HP49G
ENTER	Unshifted key
HALFTAN	Menu key (Soft key)
ASSUME	Select the command from the command catalog of the HP49G or type it in the command line and enter it
PREV	Left shifted key on the HP48

Trigonometry with the HP49G - Part 1

Hi everybody!

This is the first part of a (hopefully long) news-group-marathon-serial about the trigonometry capabilities of the HP49G and about the mysterious ideas that some strange guys had, a long long time ago in the land which I come from. I mean Pythagoras and co.

I' ll start with the easier things and as the triventure goes on, things will get more complicated. So if many of you out there find this not so useful now, because it is easy, then wait! Heavier things are on the way! Mwaahhahahahaaa!

You already know that a very important relation between the sine and the cosine is:

$$\sin^2(x)+\cos^2(x)=1 \qquad (1)$$

The HP49G command for this relation is TRIG. When it finds the sum of the squares of the sine and the cosine, it converts it always to 1. From this relation we can derive:

$$\sin^2(x)=1-\cos^2(x) \qquad (2)$$

and also

$$\cos^2(x)=1-\sin^2(x) \qquad (3)$$

The command TRIG can also do these replacements. But it does these replacements depending on flag -116 (Prefer SIN or COS). When this flag is set, then the CAS of the HP49G prefers the sine and tries to put as much of it in the result, as possible. So it would do the replacement given with formula (3). When the flag is clear, then the CAS prefers the cosine and tries to put as much of it as possible in the results. So it would do replacement (2). A mnemonic for this behaviour:

Flag -116			
Set	->	Sin	(Two Ss)
Clear-	>	Cos	(Two Cs)

There are two commands, that are related to TRIG. They are: TRIGSIN and TRIGCOS. TRIGSIN always does replacement (3) and *sets* the flag -116. TRIGCOS does replacement (2) and *clears* the flag -116. So after a TRIGSIN the command TRIG will prefer Sines and after a TRIGCOS the command TRIG will prefer Cosines.

Now, you also know that the definition of tangent is:

$$\tan(x)=\frac{\sin(x)}{\cos(x)} \qquad (4)$$

The command for transforming tangents to the quotient of sine and cosine is TAN2SC (TAN to SIN, COS). When you use it, transformation (4) takes place. The inverse transformation can be achieved with the command TRIGTAN, which would return the $\mathrm{TAN}(X)$ if fed with $\frac{\mathrm{SIN}(X)}{\mathrm{COS}(X)}$.

From relation (4) we derive:

$$\tan^2(x)=\frac{\sin^2(x)}{\cos^2(x)} \qquad (5)$$

Now we can replace $\sin^2(x)$ with $1-\cos^2(x)$ on the right hand side to get:

$$\tan^2(x)=\frac{1-\cos^2(x)}{\cos^2(x)} \quad \Leftrightarrow \quad \tan^2(x)=\frac{1}{\cos^2(x)}-1 \quad \Leftrightarrow$$

$$\tan^2(x)+1=\frac{1}{\cos^2(x)} \quad \Leftrightarrow \quad \cos^2(x)=\frac{1}{\tan^2(x)+1} \qquad (6)$$

The command TRIGTAN does also this transformation.

From relation (4) we can also derive:

$$\sin^2(x)=\frac{\tan^2(x)}{\tan^2(x)+1} \qquad (7)$$

Replace in relation (4) $\cos^2(x)$ using relation (3) and you'll see. The wonderful thing is that TRIGTAN can do also transformation (7).

Now enough theory, let's have a party!

<u>Examples:</u>

1) Show that:

$$\mathrm{SIN}(X)^4-\mathrm{COS}(X)^4=\mathrm{SIN}(X)^2-\mathrm{COS}(X)^2$$

In the EQW type:

$$\mathrm{SIN}(X)^4-\mathrm{COS}(X)^4$$

and enter it in stack level 1. Let's try `TRIGTAN`. The result

$$\frac{\mathrm{TAN}(X)^2-1}{\mathrm{TAN}(X)^2+1}$$

looks nice, so let's work with it. We have TAN and we want SIN and COS, so let's use `TAN2SC` . We get

$$\frac{\left(\frac{\mathrm{SIN}(X)}{\mathrm{COS}(X)}\right)^2-1}{\left(\frac{\mathrm{SIN}(X)}{\mathrm{COS}(X)}\right)^2+1}$$

which looks uglier, but if you `EXPAND` this you get:

$$\frac{\mathrm{SIN}(X)^2-\mathrm{COS}(X)^2}{\mathrm{SIN}(X)^2+\mathrm{COS}(X)^2}$$

Now, you may tend to use `TRIG`, to replace the denominator with 1, but this would also replace either the square of the sine or the square of the cosine on the numerator. So take this expression in the EQW, select the denominator and then press `TRIG`, so that the command acts only upon the denominator. Put this now back on the stack. Press `EXPAND` to get rid of the 1 in the denominator and you are the trigo-king of the day.

---

2) Show that:

$$\mathrm{SIN}(X)^4-\mathrm{COS}(X)^4=-\left(2\ \mathrm{COS}(X)^2-1\right)=2\ \mathrm{SIN}(X)^2-1$$

Type:

$$\mathrm{SIN}(X)^4-\mathrm{COS}(X)^4$$

in the EQW and put it on stack. We want first to transform this expression to another, which contains only COS, so let‘s try TRIGCOS. Here you are! Now, press TRIGSIN to get the expression that contains only SIN. Done!

---

3) Show that:

$$\left(\mathrm{SIN}(X)-\mathrm{COS}(X)\right)^2=1-2\ \mathrm{SIN}(X)\ \mathrm{COS}(X)$$

Enter

$$\left(\mathrm{SIN}(X)-\mathrm{COS}(X)\right)^2$$

on stack level 1 and press ENTER to DUPlicate it. (You‘ll see later why the DUPlication.) Let‘s EXPAND this, to see what comes out. You get:

$$\mathrm{SIN}(X)^2-2\ \mathrm{SIN}(X)\ \ \mathrm{COS}(X)+\mathrm{COS}(X)^2$$

This contains the square of the sine plus the square of the cosine, so use TRIG to get them converted to 1. Voila!

But wait! Was the EXPAND really necessary? Press the key ◀ to drop stack level 1 and try TRIG directly. Wow! It works :-)

---

4) Show that:

$$\left(\mathrm{SIN}(X)+\mathrm{COS}(X)\right)^2+\left(\mathrm{SIN}(X)-\mathrm{COS}(X)\right)^2=2$$

Yes, you know what I am going to say. Enter

$$\left(\mathrm{SIN}(X)+\mathrm{COS}(X)\right)^2+\left(\mathrm{SIN}(X)-\mathrm{COS}(X)\right)^2$$

blah, blah, right? Good. Now because in example 3 we saw that TRIG is clever, let‘s use it again. Press TRIG. Did you? What do you have? A nice round 2. :-)

---

5) Show that:

$$\mathrm{SIN}(X)^2\ \mathrm{COS}(Y)^2-\mathrm{COS}(X)^2\ \mathrm{SIN}(Y)^2=\mathrm{SIN}(X)^2-\mathrm{SIN}(X)^2$$

We want a result that only contains SIN, so let‘s try TRIGSIN. Nice, isn‘t it? Oh yes, it is. ;-)

---

6) Show that:

$$\mathrm{COS}(X)^2\ \mathrm{COS}(Y)^2-\mathrm{SIN}(X)^2\ \mathrm{SIN}(Y)^2=$$
$$\mathrm{COS}(X)^2+\mathrm{COS}(Y)^2-1$$

Since we saw that TRIGSIN worked so well in the last example, and we want a result that only has COS, let‘s try the cousin of TRIGSIN, TRIGCOS, that tries to put as much cosine as possible in the result. (I guess that‘s why TRIGCOS is the co(u)sin(e) of TRIGSIN. ;-)) OK, press TRIGCOS, and see the marvel. :-)

---

7) Show that:

$$\frac{COS(X)^2}{TAN(X)^2} = \frac{1}{TAN(X)^2} - COS(X)^2$$

This is a bit tougher, but don't worry. We'll get it soft! Go to the EQW and type:

$$\frac{COS(X)^2}{TAN(X)^2}$$

Don't enter it to the stack, we'll stay in the EQW for a while, since the surroundings are so picturesque there. As you know, from the relation:

$$TAN(X) = \frac{SIN(X)}{COS(X)}$$

we can derive

$$\frac{1}{TAN(X)} = \frac{COS(X)}{SIN(X)}$$

Since we want to have $\frac{1}{TAN(X)^2}$ in the result, we could try to put SIN in the denominator of our expression so that perhaps the cosine in the numerator and the sine in the denominator give:

$$\frac{COS(X)^2}{TAN(X)^2}$$

Select the denominator and apply TRIGSIN to it. Now press ENTER to put the whole expression on the stack and EXPAND it. You get a ratio with a big numerator and a denominator that only has $SIN(X)^2$. The first expression in the numerator is:

$$COS(X)^2 \ SIN(X)^2$$

so this part would give the $-COS(X)^2$. Use FDISTRIB to fully distribute the division over the sum of the numerator. Fine. Now perhaps you think that you only have to use TRIGTAN, to convert $\frac{COS(X)^2}{SIN(X)^2}$ to $\frac{1}{TAN(X)^2}$. But if you apply TRIGTAN to the whole expression, then the part $-COS(X)^2$ will also be converted to an expression that contains $TAN(X)$. So take the expression in the EQW again, and shoot some pictures of that beautiful place, while you select $\frac{COS(X)^2}{SIN(X)^2}$ and press TRIGTAN. Press ENTER to return to the dusty place of the stack again and here you are! A present from the EQW smiles at you in the middle of the dirty suburban stack.

---

8) Show that:

$$TAN(X) + TAN(Y) = TAN(X) \ TAN(Y) \left(\frac{1}{TAN(X)} + \frac{1}{TAN(Y)}\right)$$

This is even tougher. (To me at least.) I only managed to tame it using the peaceful contemplative place of the EQW and CUT, COPY, PASTE. If someone finds an easier way, then please, please, post it. Then I'll take your solution to my cousin Pythagoras and tell him that he was wrong, telling me that the only way to do that, is to use the EQW and some manually done transformations. And who knows, perhaps after some thousands of years, people will talk about the great mathematician and great HP49Gician Vincentoras or Time2Pawgoras.

Let‘s go. Enter the left hand side on the stack. Don‘t ask me why I did that, say just for a try, but using **TAN2SC** is a good start. So use it, now! Did that? OK. Now **EXPAND** this to get everything in only one ratio. The stack contains now:

$$\frac{COS(Y)\ SIN(X) - COS(X)\ SIN(Y)}{COS(Y)\ COS(X)}$$

The denominator of the beast contains $COS(Y)\ COS(X)$, which is a good start if we want to get $TAN(X)\ TAN(Y)$. So, perhaps something good happens, if we multiply the numerator by $SIN(X)\ SIN(Y)$. But if we do that, then we must also multiply the denominator by $SIN(X)\ SIN(Y)$. Take the expression to the EQW. Select the numerator, press ⊠ and type $SIN(X)$, press ⊠ again and type $SIN(Y)$. Now select the product $SIN(X)\ SIN(Y)$ that you just have entered and COPY it. Select the denominator and press ⊠ again. Press PASTE. The EQW contains now:

$$\frac{(COS(Y)\ SIN(X) - COS(X)\ SIN(Y))\ SIN(X)\ SIN(Y)}{COS(Y)\ COS(X)\ SIN(X)\ SIN(Y)}$$

Now we can take the part $COS(Y)\ COS(X)$ of the denominator away and put it in a new denominator of the part $SIN(X)\ SIN(Y)$ of the numerator. In the denominator select $COS(Y)\ COS(X)$ and press CUT. Press ◄ once to get rid of the placeholder left by the CUT operation. Now go to the numerator and select again the part $SIN(X)\ SIN(Y)$ that you have entered a couple of years ago. Press ÷ and then PASTE. You should have now:

$$\frac{(COS(Y)\ SIN(X) - COS(X)\ SIN(Y))\ \frac{SIN(X)\ SIN(Y)}{COS(Y)\ COS(X)}}{SIN(X)\ SIN(Y)}$$

Select the whole sub-expression:

$$\frac{SIN(X)\ SIN(Y)}{COS(Y)\ COS(X)}$$

and press **TRIGTAN**. Now the expression in the EQW is:

$$\frac{(COS(Y)\ SIN(X) - COS(X)\ SIN(Y))\ TAN(Y)\ TAN(X)}{SIN(X)\ SIN(Y)}$$

We have a part of the solution, the part $TAN(Y)\ TAN(X)$. While this part is selected, CUT it. Press ◄ once to get rid of the place holder again. Select the whole remaining expression, press ⊠ and then PASTE to put the sub-expression back. The EQW contains now:

$$\frac{(COS(Y)\ SIN(X) - COS(X)\ SIN(Y))}{SIN(X)\ SIN(Y)}\ TAN(Y)\ TAN(X)$$

Now select the ratio:

$$\frac{(COS(Y)\ SIN(X) - COS(X)\ SIN(Y))}{SIN(X)\ SIN(Y)}$$

and press **PARTFRAC**. The ratio is split in two smaller ratios:

$$\left(\frac{COS(X)}{SIN(X)} - \frac{COS(Y)}{SIN(Y)}\right)\ TAN(Y)\ TAN(X)$$

Select the expression $\frac{COS(X)}{SIN(X)}$ and do a TRIGTAN to it. Select the expression $\frac{COS(Y)}{SIN(Y)}$ and do TRIGTAN again. Now you can press ENTER to put the whole expression back on the stack again, or stay in the EQW and examine what happens if you apply each and every command of the HP49G to the expression.

What? You want more? OK, take this for today and try to solve them alone.

$$\frac{1}{COS(X)^2}+\frac{1}{SIN(X)^2}=\frac{1}{SIN(X)^2\ COS(X)^2}$$

$$(SIN(X)+COS(X))^2-(SIN(X)-COS(X))^2=$$
$$4\ SIN(X)\ COS(X)$$

$$1+TAN(X)^2=\frac{1}{COS(X)^2}$$

$$1+\frac{1}{TAN(X)^2}=\frac{1}{SIN(X)^2}$$

$$\frac{1}{TAN(X)^2}\ COS(X)^2=\frac{1}{TAN(X)^2}-COS(X)^2$$

Don‘t miss the next part of the marathon, where we‘ll be talking about the solutions other beasts, other trigonometric relations and relations of Scotland to Greece, or in other words, ... you will see ;-)

Pythagorian greetings,
Nick.

Hi trig-freaks!

Welcome to our second part of the marathon. A big big „thanks a lot" goes to Thomas Rast for correcting my errata of the first part. You perhaps think that I start with the errata because it is better to correct them before telling more. Well, yes this is one reason. But there is another reason which has to do with monsters. „Errata" resembles the greek word „terata" which means „monsters". Keep on reading to find out what monsters have to do with out marathon. :-)

In the last part we had much fun using the commands TRIG, TRIGSIN, TRIGCOS, TAN2SC, and TRIGTAN. There are a lot more trigonometric commands but let's first do an excursion to a place where there are no similar commands. Our marathonial journey doesn't introduce new built-in trig commands of the 49G today, but we'll see how we can make our own!

You remember that:

$$\cos^2(x) = 1 - \sin^2(x) \qquad (3)$$

which give us a possibility to express the cosine as a function of the sine:

$$\cos(x) = s1\,\sqrt{1-\sin^2(x)} \qquad (8)$$

where $s1$ is an arbitrary sign of $+$ or $-$.

There is no command for this on the 49G . Of course you could first enter $\mathrm{COS}(X)$, square it, use the command TRIGSIN and then take the square root. But this would be cumbersome and also dangerous in expressions with many sines and arbitrary signs of the sines. (Signs of sines... how poetical ;-) What can we do about it?

Well, it seems that if we could somehow substitute $s1\,\sqrt{1-\mathrm{SIN}(X)^2}$ for $\mathrm{COS}(X)$, we would have what we want. Let's try it with a small program:

```
<<
   'COS(X)=s1*√(1-SIN(X)^2)'
   SUBST
>>
```

Store this in $C \rightarrow S$, enter $\mathrm{COS}(X)$ and press the soft key C→S. It takes half a century but at the end we have what we wanted. But wait! What happens if we have $\mathrm{COS}(Y)$ instead of $\mathrm{COS}(X)$? Will the cosine of Y also be substituted with $s1\,\sqrt{1-\mathrm{SIN}(Y)^2}$? Let's try it: Enter $\mathrm{COS}(Y)$ and press C→S. Again after half a century you can see that you waited for nothing. No substitution took place, because the substitution rule was made for X, not for Y. We need a way for doing this for arbitrary names or even expressions like $\mathrm{COS}(a+b)$. So SUBST doesn't fit our needs. Another disadvantage of SUBST for this purpose is, that not only sub-expressions of the form COS(something) will be substituted. Try the following: Enter:

$$\frac{\mathrm{COS}(X)}{\mathrm{SIN}(X)}$$

and press C→S. While the 49G is working (close to a century), you may think that you will get:

$$\frac{s1\,\sqrt{1-\mathrm{SIN}(X)^2}}{\mathrm{SIN}(X)}$$

But you don't get this result. You get an ugly thing with many sub-expressions and wonder how could this ever be calculated. The reason is the way that SUBST works in this case. It seems that it first solves:

$$\mathrm{COS}(X) = s1\sqrt{1-\mathrm{SIN}(X)^2}$$

for X and then uses the found solution to substitute not every occurrence of $\mathrm{COS}(X)$, but every occurrence of X with the found solution.

The following example also shows this. Enter:

$$\mathrm{SIN}(X)+\mathrm{COS}(X)$$

then enter:

$$\mathrm{SIN}(X) = \frac{1}{Y}$$

and then press SUBST. The result is not:

$$\frac{1}{Y}+\mathrm{COS}(X)$$

but:

$$\frac{1}{Y}+\mathrm{COS}\left(\mathrm{ASIN}\left(\frac{1}{Y}\right)\right)$$

This shows that the X in $\mathrm{COS}(X)$ has been replaced by the solution $X = \mathrm{ASIN}\left(\frac{1}{Y}\right)$ of the equation $\mathrm{SIN}(X) = \frac{1}{Y}$.

Here comes one of the secret weapons of the HP49G, one of the most neglected commands, one of my favourites. :-) It is the command MATCH in its two variations ↑ MATCH and ↓ MATCH. This command searches patterns and replaces them, with no further algebraic work. It takes an expression from stack level 2 and a list from stack level 1. The list contains two items. The first is the expression that must be replaced and the second is the expression with which the first expression must be replaced. To see it in work, enter $\mathrm{COS}(X)$, then enter the list:

$$\left\{\mathrm{COS}(X)\quad s1\sqrt{1-\mathrm{SIN}(X)^2}\right\}$$

and then press ↓MATCH. Very quickly you get the desired result and a 1 on stack level 1, which shows that a replacement happened. You would see a 0 there if no replacement were possible. This is an indicator which you can use in your programs, to make decisions what should happen next, if a replacement took place or if it didn‘t. But what can we do if we don‘t have $\mathrm{COS}(X)$ but, say, $\mathrm{COS}(a+b)$ instead? Can we somehow tell the HP49G that the name of the argument of COS doesn‘t matter? Oh yes, we can. Instead of using the list:

$$\left\{\mathrm{COS}(X)\quad s1\sqrt{1-\mathrm{SIN}(X)^2}\right\}$$

we use the list:

$$\left\{\mathrm{COS}(\&a)\quad s1\sqrt{1-\mathrm{SIN}(\&a)^2}\right\}$$

The ampersand before the name a makes this to a special argument. It is not only $\mathrm{COS}(\&a)$ that will be replaced, but also $\mathrm{COS}(X)$, $\mathrm{COS}(Y)$ or even $\mathrm{COS}(a+b)$. Any pattern of the form:

$$\mathrm{COS}(\mathrm{something})$$

will be replaced by the pattern:

$$s1\sqrt{1-\mathrm{SIN}(\mathrm{something})^2}$$

So this is exactly what we need. Note also that this substitution introduces a new variable s1 (the sign) which was not in the original expression. This variable doesn't belong to any replacements that use the ampersand, but we can freely mix up the two types. To see how powerful this command is, use ↓MATCH with the algebraic object SIN(&A + &B) and the list:

{SIN(&A + &B) COS(&B) SIN(&A) + SIN(&B) COS(&A)}

Here we have two variables for pattern replacement. There is no limit to the number of such „general" variable names. So let us make a small program for the replacement of COS(something) with $s1\,\sqrt{1-\mathrm{SIN}(\mathrm{something})^2}$. Enter:

```
<<
  { 'COS(&X)'  's1*√ (1-SIN(&X)^2)' }
  ↑ MATCH  DROP
>>
```

and STOre this in ↑ C → S. Note again that &X is only a place holder for any argument of COS. Let's try it. Enter $\mathrm{COS}(x^2-a)$ and press ↑C→S . Fine!

Now, in the equation:

$$\tan(x) = \frac{\sin(x)}{\cos(x)}$$

we can replace cos(x) with $s1\,\sqrt{1-\sin(x)^2}$ and we have:

$$\tan(x)= \frac{\sin(x)}{s1^*\sqrt{1-\sin^2(x)}} \qquad (9)$$

so that we can also express TAN as a function of SIN. Let's make a new program for this replacement. Enter:

```
<<
   { 'TAN(&X)'  'SIN(&X)/(s1*√ (1-SIN(&X)^2))'  }
   ↑ MATCH  DROP
>>
```

and STOre this in ↑ T → S. Note also that this time the argument &X appears in two places in the replacement. Nice, isn't it? Let's give it a try. Enter COS(X) TAN(X), press ↑C→S and then press ↑T→S to replace both COS and TAN with functions of SIN.

But the pattern matching and replacing commands have even more power. Consider for example the following replacement list:

{ &A +&B   SAB }

which would replace a sum of two arbitrary arguments with the variable SAB. If you apply this replacement to the expression:

SIN(a + b) + SIN(c + d)

then it is not clear what should be replaced. Did you mean the arguments A + B and C + D of the two sines, or did you mean the whole expression, SIN(a + b) + SIN(c + d), which is also a sum of two things? Well, MATCH in its two variations allows any of these cases.

↑ MATCH starts searching from the innermost sub-expressions. In the example above it would return SIN(SAB) + SIN(SAB). The opposite does ↓ MATCH. It starts at the outermost sub-expressions. In the example above it returns SAB.

And what can we do if we want that all occurrences of a pattern at any level in the algebraic object, are to be replaced with some other pattern?

Well, there comes the indicator for a successful matching, that the pattern matching commands also return. Consider the following:

```
<<
   DO
    { replacementList }
    ↑MATCH
   UNTIL NOT
   END
>>
```

Every time ↑ MATCH runs it returns a 1 or a 0. When a replacement takes places a 1 is returned, which NOT makes to a 0, so that the loop runs again. But when no replacement takes place, that means that we can‘t do anything more. ↑ MATCH then returns 0, which NOT turns to 1 that terminates the loop. All possible replacements of the type given by {replacementList} are done and afterwards the program ends. Perfect!

Now, if I tell you that the pattern matching commands have even more power, you‘ll say that I am crazy. But they do. (And I am indeed crazy!) Let‘s find it out using an example. Enter:

$$\cos(x)^4 - \cos(x)^2$$

and press ↑C→S. Press EXPAND to expand this. The result contains many occurrences of s1 which is the arbitrary sign. All occurrences are raised to an even power so they should be replaced by a 1. But the HP49 doesn‘t know that s1 is 1 or −1, and so it can‘t replace all $s1^{evenPower}$ with a 1. We could of course make a replacement program for this, but then we could‘t use it for the cases when s1 is raised to an odd power. So what can we do? There cometh pattern matching again. The replacement list can contain also 3 items. The first two are the items that we already know. The third is a condition. If it evaluates to true, then the replacement will be done. If it evaluates to false, then no replacement will be done. We must check if s1 is raised to an even power, so the object:

$$FP\left(\frac{power}{2}\right)==0.$$

could be the condition. Note that we use here not the normal sign for equals, "=", but the test function "is equal to?" which on the 49G is "==". Note also that we must write the zero in the condition as 0. (real) and not 0 (integer) because it doesn‘t work with 0 as integer. (Though it should, because testing 0.==0 evaluates to true, but that‘s another story.)

Enter the program:

```
<<
  { 's1^&n' 1 'FP(&n/2)==0.' }
  ↑MATCH DROP
>>
```

and STOre this in s1even → 1. Now with the last expression (the one with many s1 occurrences) on stack level 1, press s1even→1 and EXPAND. Nice!

Note also that we used s1 and not &somevar in the replacement, because we don‘t want everything raised to some even power to be replaced by a 1. This limits us to the use of s1 as a sign variable, but if we use this convention throughout all other replacements, then everything will work fine.

If you have time you could also make such replacement programs for the following relations.

$$\sin(x) = s1\sqrt{1-\cos^2(x)} \qquad (10)$$

$$\tan(x) = \frac{s1\sqrt{1-\cos^2(x)}}{\cos(x)} \quad (11)$$

$$\cos(x) = \frac{1}{s1\sqrt{1+\tan^2(x)}} \quad (12)$$

$$\sin(x) = \frac{\tan(x)}{s1\sqrt{1+\tan^2(x)}} \quad (13)$$

and a replacement program that turns $s1^{oddPower}$ to -1 when $s1$ is -1 and to 1 otherwise.

The very interested user could also wrap all these replacement commands to a new library, so that they are available from everywhere. (Though John H. Meyers will say: „Put it in the HOME directory“ where they are also available from everywhere.) Well, do as you like. All ways are open.

It‘s time now to tell the story about the monsters and the relation of Scotland to Greece. You all know about Nessy at Loch Ness, don‘t you? Well, there is a lake at Marathon in Greece, and at the early 80's, Nessy decided to have give extraordinary concerts at the lake of Marathon, where it was much warmer than in cold Scotland. All head bangers went there and had a good time. The first to see Nessy was the witness Charalambos Trabakoulas, a shepherd, who was interviewed after Nessy‘s first appearance:

TV-Man holding the microphone directly at Charalambos‘ mouth: **Mr. Trabakoulas, tell us what you saw!**

Charalambos: **My son, I was over there with the sheep, when I, from eye to eye, directly, with my eyes, I saw the monster!**
TV-Man: **Did you see the whole monster?**

Charalambos: **Oh no, only the head, my son, take that thing away** (the microphone) **and put it where you know.** (What does he mean here?)

TV-Man: **You mean, you saw the monster as it came out of the water of the lake?**

Charalambos: **Oh, no! It had already drunk the whole lake!**

Which is the reason why there is not much water in Athens in summer, and the people don‘t have anything better to do, than finding trigonometric relations!

End for today. Next time we‘ll learn about some new commands. And have some exercises. And have fun. And... talk about the adventures of Charalambos with the extraterrestrials.

Always yours,
greetings from me and Charalambos.
(And of course the sheep... Meeeeehhh Beeeep, Meeeeehhhh, Beeeep!)
Nick.

Trigonometry with the HP49G - Part 3

Hi everybody!

Continuing the marathon with the third part, I see that there is stuff left having to do with substitutions and programming, which only indirectly relates to trigonometry. I think that it would be a pity, not to mention this stuff, so let's start where we ended last time: At the possibilities for replacements and general manipulations of algebraics. It may seem that we lost the path of the trig marathon, but what we see here will be very valuable for later, when we make our way though the jungle of trigonometry. We will return in part 4 to the main route of the trigonometric marathon.

First of all, thanks to VPN for pointing out that making a program for replacement of $s1^{oddPower}$ with a -1 would be incorrect when $s1$ represents a 1.

So we must find a way to distinguish the cases where $s1=1$ from the cases where $s1=-1$. One possible solution would be to take advantage of the fact that the command IFTE is actually a function which can be included in algebraics. So we could substitute $s1$ raised to some odd power with $IFTE(s1==1,1,-1)$. The replacement list should also had a third item that checks if the power is odd, like for example $FP\left(\frac{\&n}{2}\right)\neq 0$.

The replacement program would look like:

```
<<
    { 's1^&n' 'IFTE(s1==1,1,-1) 'FP(&n/2)≠0.' }
    ↑MATCH  DROP
>>
```

If you store this in ↑ s1odd → you can use it together with the program s1even → 1 in another program that replaces $s1$ raised to even *and* to odd powers. This program could be something like:

```
<<
    ↑ s1even → 1
    EXPAND
    ↑ s1odd →
    EXPAND
>>
```

Now, for small expressions the programs are nice but for bigger expressions that also contain many odd powers of $s1$, you can quickly come to very ugly looking results with many IFTE, which don't contribute to the overall readability of the expression. Another problem is that the replacement programs will match anything of the form $s1^n$ but *not* a single $s1$ that isn't raised to some power. (Remember? No algebraic replacement, just patterns.) We could of course check the whole algebraic for existence of $s1$ not raised to any power, but that would be cumbersome. In other words, I am too lazy to do that. ;-)

But let's think again (as VPN says) about this problem. The main advantage of the pattern matsching commands is that is can find and replace patterns. Now, we made the convention that $s1$ is always the name of the arbitrary sign, so we don't need to look for patterns. The command SUBST seems to fit better here, as the code

```
's1=1' SUBST
```

or

```
's1=-1'  SUBST
```

would substitute every occurrence of $s1$ with 1 or -1 respectively, even when $s1$ is not raised to any power at all. But the problem is that we can't do both substitutions with only one SUBST. Remember, when the HP49G wants to tell us that there are more than one possible results, it packs them in a list. Imitating this behaviour we can write a program like:

```
<<
    DUP
   's1=-1' SUBST EXPAND
    SWAP
   's1=1' SUBST EXPAND
    2  →LIST
>>
```

which makes both substitutions and returns the two results in a list. We see here, that the choice of the right tool can make life easier. Since the 49G has so many commands, it is sometimes not easy to decide which one should be used. But using some method for a long time often shows the disadvantages and suggests another method to be used.

Continuing about replacements: As Máximo Castañeda Riloba has pointed out, the replacement of the pattern COS(&X) with $s1\ \sqrt{1-SIN(X)^2}$ will work for an expression that only contains one occurrence of the pattern COS(&X). When we have an expression with multiple occurrences of COS(&X), as in COS(X)+COS(2 X) then each of the COS patterns should be replaced in a way that each sign is independent from the other, because one sign can be 1 or −1, no matter what the other sign is. Putting simply s1 for every replaced pattern makes them both the same. If we only somehow could use a numbering system that distinguishes the signs and writes, say, s1 for the first and s2 for the second, and so on. MATCH can't do that[1] , because one pattern is replaced by one pattern with no numbering or other distinguishing capabilities. Neither SUBST nor | (where) can be used for this purpose. But this doesn't mean that it isn't possible. (A well known phrase when using the HP49G ;-))

We need to write a program that not only checks occurrences of COS (or other trigs) but also keeps track of the value of some iterator variable, which then can be used to make the signs s1, s2, and so on.

[1] Well, take a look at the second part of the Basic Calculus Marathon, to see how pattern matching can be used for this purpose.

A fantastic property of the HP49G is that functions (like COS) are also objects, which can be used not only to perform calculations but also for other purposes. For example enter the list {COS} and then use the command HEAD, which extracts the first element of a list. A naked COS function sits now on stack level 1. It can be used as argument for tests and other things.

Another command made available to the users on the HP49G is the command →LST (Menu 256, second page). This beauty takes an algebraic object, translates it to the RPN command sequence that corresponds to the algebraic, and returns this sequence as a list. Enter for example (COS(X)−1) SIN(X), and press →LST. The result is the list representation of the algebraic object:

{X COS 1 − 2 X   SIN }

The opposite is the command →ALG (Menu 256, second page) which would take this list and build up the original algebraic object.

Having this two things in combination allows as to turn an algebraic to a list, check for occurrences of COS (or any other command/function) keeping track ot the number of the occurrences and replace each occurrence with something that contains a numbered arbitrary sign. For a replacement of COS with $s1\ \sqrt{1-SIN(X)^2}$ where sn is the numbered arbitrary sign, enter the program:

```
<<  →LST
  1
  <<
    IF DUP { COS } HEAD SAME
    THEN DROP { SIN SQ NEG 1 + SQRT } "s" NSUB R→I +
         S~N  { * }  +  +  OBJ→   DROP
    END
  >>
  DOSUBS  →ALG
>>
```

and STOre it in C → S. This program simply checks each object in the list that → LST created. If the object is the function COS, it constructs the list:

$$\{\text{SIN}\quad \text{SQ}\quad \text{NEG}\quad 1\quad +\quad \sqrt{}\quad \text{sn}\quad\}$$

and then explodes it dropping the item count of the list. The program may look like one that uses a local variable procedure but it isn't. The inner program is placed on the stack at run time and is used as an argument to the command DOSUBS.

DOSUBS needs a list at stack level 3, the number of arguments that the program on level 1 needs, and a program on level 1. It applies this program to each group of n arguments of the list in level 3, n being the number of arguments in stack level 2. The result of the program replaces each group of n arguments in the list. NSUB returns the number of the current group of items used as arguments for the program on level 1, as a real number. R → I transforms this real to an integer and adding this integer to the string "s" returns the numbered arbitrary sign as a string, which the command S~N then transforms to a name.

To try it, enter:

$$\text{COS}(X)+\text{SIN}(X)+\text{COS}(X)^2$$

and press C→S. You get:

$$\sqrt{-\text{SIN}(X)^2+1}\;\; s2+\text{SIN}(X)+\left(\sqrt{-\text{SIN}(X)^2+1}\;\; s7\right)^2$$

where s2 and s7 are two distinct arbitrary signs.

You could now be impressed of the ease with which such things are achievable with the HP49G, but there are things even more impressing. Consider for example the algebraic object:

$$\int_0^{\pi} \text{COS}(X)\;\; \text{SIN}(X)dX$$

Applying C → S to this returns the same object unchanged. Why? If you re-enter the algebraic and apply the command → LST to it, the result is:

$$\{0\quad \pi\quad \text{COS}(X)\;\; \text{SIN}(X)\quad X\quad \int\}$$

Checking if each item is the same as COS doesn't work here because the integrant is not transformed into an elementary command sequence that includes COS at some point. Some of the functions that can be in algebraics behave this way, when we apply → LST. One of these functions is the function $\int$. So we could of course think, that we must first find all that functions and do something special if we encounter them. But what if such functions are nested? How many passes should we explicitly program to catch all of those functions?

Fortunately we don't need to do that. Recall that C → S splits an algebraic to its elements, checks each element for a COS function and if it is, it replaces this element. Exactly the same procedure can be used for the case, when one of the elements of the list is itself an algebraic object. Since algebraic objects and built-in functions have different types (9 and 18 respectively), we can do:

```
<<
  → LST
  1
  <<
    IF DUP TYPE 9. ==
    THEN C → S
    END
```

Program continues on next page

```
    IF
      DUP { COS } HEAD SAME
    THEN
      DROP { SIN SQ NEG 1 + √} "s" NSUB
      R→I + S~N  { * } + + OBJ→  DROP
    END
  >>
  DOSUBS
   →ALG
>>
```

STOre this in C → S. Note that the program uses itself recursively (!) to split any number of nested algebraics and replace each occurrence of COS. To test it, enter for example:

$$\int_0^\pi \int_0^\pi COS(X)\ SIN(Y)\,dX\,dY$$

Then press C→S. Isn't that nice? (It is a recursion, so Nick *must* find it beautiful.)

Finishing for today, there is a problem left, that arises because we use numbered arbitrary signs. How could one write a program, that checks for any arbitrary sign, like s1, s2 and so on, and then builds up a list with algebraics in all possible combinations of all arbitrary signs?

That's all for today. Take care and keep on HP49Ging.
'Till next time, recursive greetings,
Nick.

Trigonometry with the HP49G - Part 4

Hi everybody!

In the last two parts of the trigonometry marathon we made a long trip through the replacements jungle. Then came Christmas any the New Year, when Nick was drinking Ouzo with Trabakoulas, thinking about trigonometry and monsters.

Trabakoulas also said that he had an experience with aliens, who came to his house carrying their HP calcs in their heads. I told him that this must have been VPN, but he insisted that they were aliens. They were eager to show him how much trigonometry can be done with the HP49G and they told him to tell me about this lessons. The poor guy now has a big headache applying trigonometry to find positions of his sheep.

Now let‘s return to our main path and examine other trigonometry commands that the HP49G provides. This time we will talk about trigonometric functions of sums of two or more angles. Perhaps you already know about relations like:

$$\cos(x+y)=\cos(x)\ \cos(y)-\sin(x)\ \sin(y)$$

$$\sin(x+y)=\sin(x)\ \cos(y)+\cos(x)\ \sin(y)$$

$$\tan(x+y)=\frac{\tan(x)+\tan(y)}{1-\tan(x)\ \tan(y)}$$

Can the HP49G do these things? As you may have guessed, yes! The command TEXPAND takes the left hand side of these relations and returns the right hand side. Try it. TEXPAND can also transform differences of angles. Enter for example $\mathrm{SIN}(X-Y)$ and press TEXPAND. You get:

$$\mathrm{SIN}(X)\ \mathrm{COS}(Y)-\mathrm{COS}(X)\ \mathrm{SIN}(Y)$$

The command TEXPAND can also be used to show such things like $\mathrm{SIN}(X+\pi)=\mathrm{SIN}(X)$. Enter $\mathrm{SIN}(X+\pi)$, press TEXPAND and then EXPAND to get the final result.

But what about the opposite way? What if you have, say, $\mathrm{COS}(X)\ \mathrm{COS}(Y)-\mathrm{SIN}(X)\ \mathrm{SIN}(Y)$ and want to transform this to $\mathrm{COS}(X+Y)$? Well, then you use the command TCOLLECT. Just try it. Enter $\mathrm{COS}(X)\ \mathrm{COS}(Y)-\mathrm{SIN}(X)\ \mathrm{SIN}(Y)$, press TCOLLECT and enjoy.

There is also the command TLIN, which takes products of trigonometric functions and tries to convert them into expressions with linear trigonometric terms.

Now let‘s have some fun! Take the ouzo bottle out, and help Trabakoulas find his sheep.

1) Simplify the expression:

$$\mathrm{SIN}\left(\frac{\pi}{3}+a\right)+\mathrm{SIN}\left(\frac{\pi}{3}-a\right)$$

This is easy. After you entered the expression press TEXPAND and then EXPAND. The result is:

$$\sqrt{3}\ \mathrm{COS}(a)$$

---

2) Simplify the expression:

$$\mathrm{SIN}(X-Y)\ \mathrm{COS}(X+Y)-\mathrm{COS}(X-Y)\ \mathrm{SIN}(X+Y)$$

Looks like we should use TEXPAND and hope that many sub-expressions cancel out. Let's try it. TEXPAND and then EXPAND returns:

$$-\left(2\ \mathrm{COS}(Y)\ \mathrm{SIN}(Y)\ \mathrm{SIN}(X)^2+2\ \mathrm{COS}(Y)\ \mathrm{COS}(X)^2\ \mathrm{SIN}(X)\right)$$

Not very satisfying. But we can COLLECT common factors and we get:

$$-\left(\left(\mathrm{SIN}(X)^2+\mathrm{COS}(X)^2\right)\ \mathrm{COS}(Y)\ \mathrm{SIN}(Y)\ 2\right)$$

Now we can use TRIG, to turn $\mathrm{SIN}(X)^2+\mathrm{COS}(X)^2$ to 1. The result is now:

$$-\left(2\ \ \mathrm{COS}(Y)\ \ \mathrm{SIN}(Y)\right)$$

Now we have a product of trigonometric expressions. Let's try TLIN to see if it can converted to a linear trigonometric function. Press TLIN and you get $-1\ \mathrm{SIN}(2\ \ Y)$. EXPAND this and you have $-\mathrm{SIN}(2\ \ Y)$

But there is another easier way. Re-enter the expression $\mathrm{SIN}(X-Y)\ \mathrm{COS}(X+Y)-\mathrm{COS}(X-Y)\ \mathrm{SIN}(X+Y)$ and press TCOLLECT. Voila! (TLIN does exactly the same in this case. Try it!)

---

3) Turn $\mathrm{SIN}(X+Y+Z)$ to a sum of products of trigonometric functions.

Enter $\mathrm{SIN}(X+Y+Z)$ and press TEXPAND. (It works with an arbitrary number of angles in the sum.) Use FDISTRIB to completely distribute $\times$ over $+$ and you get:

$$\mathrm{COS}(Y)\ \mathrm{SIN}(X)\ \mathrm{COS}(Z)+\mathrm{SIN}(Y)\ \mathrm{COS}(X)\ \mathrm{COS}(Z)+$$
$$\mathrm{COS}(Y)\ \mathrm{COS}(X)\ \mathrm{SIN}(Z)-\mathrm{SIN}(Y)\ \mathrm{SIN}(X)\ \mathrm{SIN}(Z)$$

(This could be also called, „Turn $\mathrm{SIN}(X+Y+Z)$ to something that is much more complicated")

---

4) Express $\dfrac{1}{\mathrm{TAN}(X+Y)}$ as a function of $\mathrm{TAN}(X)$ and $\mathrm{TAN}(Y)$

Enter $\dfrac{1}{\mathrm{TAN}(X+Y)}$ and press TEXPAND. You get:

$$-\frac{\mathrm{TAN}(Y)\ \ \mathrm{TAN}(X)-1}{\mathrm{TAN}(X)+\mathrm{TAN}(Y)}$$

---

5) Show that:

$$\mathrm{SIN}(X+Y)^2+\mathrm{SIN}(X-Y)^2 =$$
$$2\ \mathrm{SIN}(X)^2\ \mathrm{COS}(Y)^2+2\ \mathrm{SIN}(Y)^2\ \mathrm{COS}(X)^2$$

Just enter the left hand side and use TEXPAND and then EXPAND.

6) Show that:

$$\mathrm{SIN}(X+Y)\ \mathrm{SIN}(X-Y) =$$
$$\mathrm{SIN}(X)^2-\mathrm{SIN}(Y)^2=\mathrm{COS}(Y)^2-\mathrm{COS}(X)^2$$

Enter $\mathrm{SIN}(X+Y)\ \mathrm{SIN}(X-Y)$, press TEXPAND and then EXPAND to get:

$$\mathrm{COS}(Y)^2\ \ \mathrm{SIN}(X)^2-\mathrm{COS}(X)^2\ \ \mathrm{SIN}(Y)^2$$

Now we have squares of SIN and COS. We can use TRIGSIN to turn squares of COS to squares of SIN. Press TRIGSIN to get:

$$\mathrm{SIN}(X)^2-\mathrm{SIN}(Y)^2$$

If you press TRIGCOS you will get:

$$-\left(\mathrm{COS}(X)^2-\mathrm{COS}(Y)^2\right).$$

7) Show that:

$$\frac{2\ \mathrm{SIN}(X+Y)}{\mathrm{COS}(X+Y)+\mathrm{COS}(X-Y)}=\mathrm{TAN}(X)+\mathrm{TAN}(Y)$$

Enter the left hand side of the equation. Press TEXPAND and then EXPAND to obtain:

$$\frac{\mathrm{COS}(Y)\ \mathrm{SIN}(X)+\mathrm{COS}(X)\ \mathrm{SIN}(Y)}{\mathrm{COS}(Y)\ \mathrm{COS}(X)}$$

Use DISTRIB to distribute $/$ over the $+$ and you have:

$$\frac{\mathrm{SIN}(X)}{\mathrm{COS}(X)}+\frac{\mathrm{SIN}(Y)}{\mathrm{COS}(Y)}$$

Now use TRIGTAN to obtain $\mathrm{TAN}(X)+\mathrm{TAN}(Y)$.

8) Show that:

$$\mathrm{COS}(X)+\mathrm{COS}\left(\frac{2\ \pi}{3}+X\right)+\mathrm{COS}\left(\frac{4\ \pi}{3}+X\right)=0$$

TEXPAND and then EXPAND gives you the 0.

9) Show that:

$$\cos(X)^2+\cos\left(\frac{2\,\pi}{3}+X\right)^2+\cos\left(\frac{4\,\pi}{3}-X\right)^2$$

is a constant.

Enter the left hand side, press TEXPAND, then EXPAND to get:

$$\frac{3\;\sin(X)^2+3\;\;\cos(X)^2}{2}$$

Then use TRIG to obtain $\frac{3}{2}$.

---

10) Show that:

$$\tan\left(\frac{\pi}{4}-X\right)=\frac{\sin(X)-\cos(X)}{\sin(X)+\cos(X)}$$

Enter $\tan\left(\frac{\pi}{4}-X\right)$ and press TAN2SC to convert the TAN to a quotient of SIN and COS. Then use TEXPAND and EXPAND to obtain the final result.

---

11) And now a messy one. Show that:

$$\frac{\cos(X+Y)^2+\cos(Y)^2-2\;\cos(X+Y)\;\cos(X)\;\cos(Y)}{\sin(X+Y)^2+\sin(Y)^2-2\;\sin(X+Y)\;\cos(X)\;\cos(Y)}=1$$

Enter the big left hand side. Let's try TEXPAND and then EXPAND. We get an expression with many squares of SIN and COS. We hope that replacing squares of SIN through squares of COS will cancel out many sub-expressions, so we use TRIGCOS. And the answer is: 1

Alternatively we can apply TCOLLECT to the big left hand side. This returns a sum of quotients, which we can EXPAND to obtain the final answer 1. (TCOLLECT needs some seconds to make its job, so be patient and let it finish.)

Another way is to use TLIN to the big left hand side. This also returns a sum of quotients, which we can EXPAND to obtain the 1. (TLIN also needs some seconds to make its job, so again be patient until it finishes.)

Let's stop for today, as ouzo starts influencing me very strongly. Here are some more things, which you could try to solve alone. (After you have survived the ouzo influence, of course ;-))

1) Show that:

$$\cos(X+Y)\;\cos(X-Y)=\cos(X)^2-\sin(Y)^2=$$
$$\cos(Y)^2-\sin(X)^2$$

2) Show that:

$$\frac{\mathrm{SIN}(X-Y)}{\mathrm{SIN}(X)\ \mathrm{SIN}(Y)} = \frac{1}{\mathrm{TAN}(Y)} - \frac{1}{\mathrm{TAN}(X)}$$

3) Show that:

$$\mathrm{COS}(X+Y)^2 + \mathrm{COS}(Y)^2 - 2\ \mathrm{COS}(X+Y)\ \mathrm{COS}(X)\ \mathrm{COS}(Y) = \mathrm{SIN}(X)^2$$

4) Show that the expression:

$$\mathrm{COS}(X)^2 - 2\ \mathrm{COS}(X)\ \mathrm{COS}(A)\ \mathrm{COS}(A-X) + \mathrm{COS}(A-X)^2$$

doesn‘t depend on X.

Just for completeness! Greetings,
Nick.

Trigonometry with the HP49G - Part 5

Hi everybody!

We have seen so far how much power the trigonometric commands of the HP49G provide, but what we have seen isn't even half the available power! Actually Trabakoulas said: "The real power of the HP49G is that it helped me find my sheep, which I was searching a little far away from here, when I saw the aliens. Inform HP that a sheep-finder application for the calc would be a real nice thing, and also that alien trigonometry is identical to ours."

The marathon will continue today with trigonometric functions of products and some additional techniques for working with the trigonometric commands of the HP49G.

Let's take a look to the some formulas of trigonometry, which you may already know.

$$\sin(x)=2\ \sin\left(\frac{x}{2}\right)\ \cos\left(\frac{x}{2}\right) \qquad (14)$$

$$\cos(x)=\cos^2\left(\frac{x}{2}\right)-\sin^2\left(\frac{x}{2}\right) \qquad (15)$$

$$\tan(x)=\frac{2\ \tan\left(\frac{x}{2}\right)}{1-\tan^2\left(\frac{x}{2}\right)} \qquad (16)$$

How could we do the first one? Well, notice that on the right hand side we have trigonometric functions with an argument that is the half of the argument of the trigonometric functions on the left hand side. In such cases it is often a good start, to use HALFTAN and then TAN2SC.

So here we go again. Enter $\mathrm{SIN}(X)$ and press HALFTAN. You get a ratio with TAN functions of $\frac{X}{2}$. Turn the TAN functions to SIN and COS of the same argument using TAN2SC. Press EXPAND to make things a little bit clearer. The denominator of the resulting ratio is the sum of the squares of COS and SIN, so press TRIG to turn it to 1 and get rid of the denominator.

The same way can be used to achieve (15). Enter $\mathrm{COS}(X)$ and press HALFTAN, TAN2SC, EXPAND. If you now press TRIG to make a 1 out of the denominator, then you'll get:

$$-\left(2\ \mathrm{SIN}(X)^2-1\right)$$

This is equivalent to:

$$\mathrm{COS}\left(\frac{X}{2}\right)^2-\mathrm{SIN}\left(\frac{X}{2}\right)^2$$

Obviously TRIG acts also upon the numerator in this case. So instead of pressing TRIG, press ▼, to get the ratio in the EQW, then select the denominator and then press TRIG, so that the command acts upon the denominator only. Now press ENTER and EXPAND to get the desired result.

Let's continue to the third formula. You're lucky! Press HALFTAN and you're done.

The opposite direction is also possible. Enter for example the right hand side of (15) and press TCOLLECT or TLIN. The result is $\mathrm{COS}\left(2\ \frac{X}{2}\right)$ which can be EXPANDed to give $\mathrm{COS}(X)$.

Now enter right hand side of (16) and press TCOLLECT or TLIN. Nothing happens except for some reordering. Notice however that we have an expression with TAN functions of $\frac{X}{2}$, the half of X, and we want an expression that only contains trigonometric functions of X. The command TAN2SC2 seems appropriate here, since it takes trigonometric functions of the half argument and returns trigonometric functions of the argument itself. So let's use it and hope that it does good in this case. Press TAN2SC2. (Be careful because TAN2SC and TAN2SC2 show up identically on the menu keys. TAN2SC2 is the third menu key from the left in the second page of the trigonometry menu 122.) Now press TRIGTAN and you're done.

Let's take a look to some additional examples now.

1) Show that:

$$SIN(3\ X) = 3\ SIN(X) - 4\ SIN(X)^3$$

Enter SIN(3 X), which is the same as SIN(x + 2 X), an expression of the form SIN(A + B). So we expect that TEXPAND could help us here. Press TEXPAND and you get:

$$SIN(X)\ \left(4\ COS(X)^2 - 1\right).$$

We want to have only SIN on the right hand side, so press TRIGSIN and voila!

(TCOLLECT or TLIN applied on $SIN(X)\ \left(4\ COS(X)^2 - 1\right)$ give you SIN(3 X), the expression you started with.)

---

2) Show that:

$$COS(3\ X) = 4\ COS(X)^3 - 3\ COS(X)$$

Enter COS(3 X) and press TEXPAND.

---

3) Show that:

$$\left(COS(X) + SIN(X)\right)^2 = 1 + SIN(2\ X)$$

This is also easy. Just enter the left hand side and press TCOLLECT. The HP49G starts getting bored, so give it something to crunch a bit more.

---

4) Show that:

$$1 + TAN(X) + TAN(2\ X) = \frac{1}{COS(2\ X)}$$

Enter the left hand side of the equation. In this case we want to get an expression with COS of 2 X. So turning TAN(X) to some trigonometric function of 2 X sounds reasonable. Pressing TAN2SC2 while the expression is on the stack is not good because then TAN(2 X) is turned to a function of 4 X. So take the expression to the EQW, select TAN(X) and then press TAN2SC2. Still in the EQW select TAN(2 X) and press TAN2SC. Press ENTER to return to the stack. Press TRIGCOS to turn the expression to $\frac{1}{COS(2\ X)}$.

---

5) Show that:

$$\mathrm{TAN}\left(\frac{\pi}{4}-X\right)^2=\frac{1-\mathrm{SIN}(2\cdot X)}{1+\mathrm{SIN}(2\cdot X)}$$

The argument of TAN on the left hand side is of the general form $A-B$, so TEXPAND seems appropriate as a start. Press TEXPAND and you get a new function with TAN functions of X. Because the result involves SIN functions of $2\cdot X$, try TAN2SC2. This leaves a function with SIN and COS functions of $2\cdot X$. Now press TRIGSIN to get the desired result.

6) Show that:

$$\frac{\mathrm{SIN}(2\cdot X)}{1+\mathrm{COS}(2\cdot X)}\cdot\frac{\mathrm{COS}(X)}{1+\mathrm{SIN}(X)}=\mathrm{TAN}\left(\frac{X}{2}\right)$$

In this case we have $2\cdot X$ and x as arguments for SIN and COS on the left hand side and we want $\frac{X}{2}$ as argument for TAN on the right hand side. We must use HALFTAN twice for the first factor of the left hand side and once for the second factor of the left hand side. To do this we take the left hand side to the EQW, select the first factor and press HALFTAN. The resulting expression looks a bit complicated, but press EXPAND and you get $\mathrm{TAN}(X)$. While $\mathrm{TAN}(X)$ is still selected, press HALFTAN again. Now select the second factor and press HALFTAN. Press ENTER to take the big expression on the stack. This expression contains only $\frac{X}{2}$ as argument for the TAN functions. Press EXPAND and you have what you wanted.

7) Show that:

$$\mathrm{TAN}\left(\frac{\pi}{4}+X\right)-\mathrm{TAN}\left(\frac{\pi}{4}-X\right)=2\cdot\mathrm{TAN}(2\cdot X)$$

Because we again have sums as arguments for TAN functions on the left hand side, press TEXPAND. Then EXPAND the result to make it a little more readable. Now, we want to have $2\cdot X$ as argument for TAN in the final result, so let's try TAN2SC2 and then TRIGTAN, which gives exactly what we wanted to have.

8) Show that:

$$\frac{\mathrm{SIN}(2\cdot X)^2-4\cdot\mathrm{SIN}(X)^2}{\mathrm{SIN}(2\cdot X)^2+4\cdot\mathrm{SIN}(X)^2-4}=\mathrm{TAN}(X)^4$$

It should be clear by now that the arguments $2\cdot X$ of SIN functions on the left hand side should be converted to X first. So take the left hand side on the EQW, select $\mathrm{SIN}(2\cdot X)^2$ on the numerator and press HALFTAN. Then select $\mathrm{SIN}(2\cdot X)^2$ on the denominator and press again HALFTAN. Press ENTER to go to the stack and then press TRIGTAN.

9) Find to what $\mathrm{TAN}(X)$ has to be equal, if the following equation has to be satisfied:

$$4\ \mathrm{SIN}(2\ X)+3\ \mathrm{COS}(2\ X)=3$$

First enter the equation and then press **HALFTAN** to convert all trigonometric functions to TAN functions of the half angle, that is of X. Then enter $\mathrm{TAN}(X)$ and press **SOLVE**. (Don't worry about the denominator, as it never is 0 when X is real.)

---

10) What values must l and m have, if the equation

$$\frac{1}{\mathrm{SIN}(X)}=\frac{l}{\mathrm{TAN}\left(\frac{X}{2}\right)}+\frac{m}{\mathrm{TAN}(X)}$$

must be satisfied for any value of X?

Let's first convert $\mathrm{TAN}\left(\frac{X}{2}\right)$ to trigonometric functions of X. Get the whole equation to the EQW, select $\frac{l}{\mathrm{TAN}\left(\frac{X}{2}\right)}$ on the right hand side and press **TAN2SC2**, which also converts TAN to SIN and COS functions. Select $\frac{m}{\mathrm{TAN}(X)}$ and press **TAN2SC**. Now, select the whole right hand side and press **EXPAND**.

If you have:

$$\frac{(l+m)\ \mathrm{COS}(X)+l}{\mathrm{SIN}(X)}$$

you are on the right path. With this expression selected, press **DISTRIB**. Now you have the equation:

$$\frac{1}{\mathrm{SIN}(X)}=\frac{(l+m)\ \mathrm{COS}(X)}{\mathrm{SIN}(X)}+\frac{l}{\mathrm{SIN}(X)}$$

The expression $\frac{(l+m)\ \mathrm{COS}(X)+l}{\mathrm{SIN}(X)}$ on the right hand side must vanish, because otherwise the right hand side can't be equal to the left hand side for every X value. That means that it must be:

$$l+m=0\ \Leftrightarrow\ m=-l$$

So the equation turns to:

$$\frac{1}{\mathrm{SIN}(X)}=\frac{l}{\mathrm{SIN}(X)}$$

which clearly shows that:

$l=1$ and $m=-1$.

That's all for now, stay tuned for the next parts.
Greetings,
Nick.

Hi everybody!

First of all many thanks to Thomas Rast for posting a correction to an error in the last part. Also many thanks to G. Illias for suggesting to make side notes, if appropriate, to show how such things can be done with the HP48.

If anything has became clear until now, then this must be the fact that there are many too many relations between trigonometric functions, and no general rule for working with all of them. Knowledge of math is important, but getting used to the way that the commands work is also important. As you use your HP49G more and more, you start „knowing in advance“ what the result of some function or command will look like, and you develop a kind of built-in instinct, which helps you to find out, which way you should follow to solve some problem. As Trabakoulas, the father soul of all shepherds says: „Here on the Trigomounts, my son, there is no such thing like a compass that always brings you to your destination. Go with care and ratio and don‘t be afraid to stop and return to your starting point, when you see that some way gets difficult with time.“

In this 6th part of the trigonometry marathon we are going to do some stuff for which the HP49G doesn‘t provide built-in commands. But we will see that nonetheless the HP49G can handle such cases. If I remember well, there has been a discussion here, about such conversions like:

$$\sin(x)+\sin(y)=2\ \sin\left(\frac{x+y}{2}\right)\cos\left(\frac{x-y}{2}\right) \qquad (17)$$

I didn‘t find any single command on the HP49G, which does this conversion from the left to the right. We have to use several commands in combination. Let‘s enter $\mathrm{SIN}(X)+\mathrm{SIN}(Y)$. Note that the result that we want to have contains only trigonometric functions of the *half* of the sum of X and Y. So we hope that we can start with HALFTAN, which converts trigonometric functions to TAN functions of the half angle. Press **HALFTAN**. Now the arguments are all $\frac{X}{2}$ and $\frac{Y}{2}$ but we have TAN instead of SIN and COS functions. Let‘s turn TAN to SIN and COS functions. Press **TAN2SC**. The resulting expression looks a bit of weird. Through repeated attempts and not through „knowing in advance“ I found that a good way to go is the following: Press **COLLECT** to factor the expression. (Takes some time, so be patient.) Take the resulting expression in the EQW and switch to mini font to see more of the expression. The numerator is:

$$\left(\mathrm{COS}\left(\frac{Y}{2}\right)\mathrm{SIN}\left(\frac{X}{2}\right)+\mathrm{COS}\left(\frac{X}{2}\right)\mathrm{SIN}\left(\frac{Y}{2}\right)\right)\left(\mathrm{SIN}\left(\frac{Y}{2}\right)\mathrm{SIN}\left(\frac{X}{2}\right)+\mathrm{COS}\left(\frac{X}{2}\right)\mathrm{COS}\left(\frac{Y}{2}\right)\right)2$$

The denominator is:

$$\left(\mathrm{SIN}\left(\frac{X}{2}\right)^2+\mathrm{COS}\left(\frac{X}{2}\right)^2\right)\left(\mathrm{SIN}\left(\frac{Y}{2}\right)^2+\mathrm{COS}\left(\frac{Y}{2}\right)^2\right)$$

Select the first sub-expression of the numerator:

$$\left(\mathrm{COS}\left(\frac{Y}{2}\right)\mathrm{SIN}\left(\frac{X}{2}\right)+\mathrm{COS}\left(\frac{X}{2}\right)\mathrm{SIN}\left(\frac{Y}{2}\right)\right)$$

and press **TCOLLECT**. This converts the sub-expression to:

$$\mathrm{SIN}\left(\frac{X+Y}{2}\right)$$

Looks like we are on the right way. Select the second sub-expression of the numerator:

$$\left(\mathrm{SIN}\left(\frac{Y}{2}\right)\ \mathrm{SIN}\left(\frac{X}{2}\right)+\mathrm{COS}\left(\frac{X}{2}\right)\ \mathrm{COS}\left(\frac{Y}{2}\right)\right)$$

and press **TCOLLECT** again. Fine, we have:

$$\mathrm{COS}\left(\frac{X-Y}{2}\right)$$

Select the whole denominator:

$$\left(\sin^2\left(\frac{x}{2}\right)+\cos^2\left(\frac{x}{2}\right)\right)\left(\sin^2\left(\frac{y}{2}\right)+\cos^2\left(\frac{y}{2}\right)\right)$$

and press **TCOLLECT** again. This returns a nice round 1. Press **ENTER** and then **EXPAND** to get rid of this 1 in the denominator. Voila!

The summary of what we have done: HALFTAN, TAN2SC, COLLECT, TCOLLECT applied to the first and second sub-expression of the numerator, TCOLLECT applied to the denominator and EXPAND.

You can use the same method also for:

$$\sin(x)-\sin(y)=2\ \sin\left(\frac{x-y}{2}\right)\ \cos\left(\frac{x+y}{2}\right) \qquad (18)$$

$$\cos(x)+\cos(y)=-2\ \cos\left(\frac{x+y}{2}\right)\ \cos\left(\frac{x-y}{2}\right) \qquad (19)$$

$$\cos(x)-\cos(y)=2\ \sin\left(\frac{x+y}{2}\right)\ \sin\left(\frac{x-y}{2}\right) \qquad (20)$$

A similar transformation is:

$$\frac{\sin(x)-\sin(y)}{\sin(x)+\sin(y)}=\frac{\tan\left(\frac{x-y}{2}\right)}{\tan\left(\frac{x+y}{2}\right)} \qquad (21)$$

Apply the method used to show (17) separately on the numerator and the denominator. Take the left hand side in the EQW, select the numerator, apply the above method, then select the denominator and apply the method again. Then you get:

$$\frac{\mathrm{SIN}\left(\frac{X-Y}{2}\right)\ \mathrm{COS}\left(\frac{X+Y}{2}\right)}{\mathrm{COS}\left(\frac{X-Y}{2}\right)\ \mathrm{SIN}\left(\frac{X+Y}{2}\right)}$$

Press **ENTER** to put this expression on the stack. Use **TRIGTAN** to convert SIN and COS to TAN functions of the same argument, and you're ready.

But now the question is: „Do I have to do all this every time I want to do such a conversion?“ Well, no! It would be cumbersome and not so easy, because we applied some trigonometric functions separately on *parts* of our expressions. If we wanted to do exactly the same programmatically, then we would have to use commands that split our expressions, check what the sub-expressions are and what the functions are that combine the sub-expressions, check arguments, and so on. Because we don't want to write a new CAS on top of the existing CAS, we choose an easier way: MATCH (Again, ;-))

For example, we can use MATCH with the list:

$$\left\{ \mathrm{SIN}(\&A)+\mathrm{SIN}(\&B) \quad 2\ \mathrm{SIN}\left(\frac{\&A+\&B}{2}\right)\ \mathrm{COS}\left(\frac{\&A-\&B}{2}\right) \right\}$$

for the conversion (17). But then we have 2 problems:

1) How can we be sure that this would work, also for not factored expressions, like for example $\mathrm{SIN}(X)^2-\mathrm{SIN}(Y)^2$?

2) How can we repeatedly match, until all matching has been done?

The answer to the first question seems to be to use COLLECT, so that the necessary factoring for the following MATCH is achieved. I don't know if this works perfectly, but I didn't have any case where it didn't up to now.

The answer to the second question is, to use MATCH in a loop, until nothing more can be matched. The following (not so) small program does this for the conversions (17), (18), (19) and (20):

```
<<
   COLLECT

   WHILE
    { 'SIN(&A)+SIN(&B)' '2*SIN((&A+&B)/2)*COS((&A-
&B)/2)' }
    ↓MATCH
   REPEAT
   END

   WHILE
    { 'SIN(&A)-SIN(&B)' '2*SIN((&A-
&B)/2)*COS((&A*&B)/2)' }
    ↓MATCH
   REPEAT
   END

   WHILE
   { 'COS(&A)+COS(&B)' '2*COS((&A+&B)/2)*COS((&A-&B)/2)'
   }
    ↓MATCH
   REPEAT
   END

   WHILE
    { 'COS(&A)-COS(&B)' '2*SIN((&A+&B)/2)*SIN((&A-
&B)/2)' }
    ↓MATCH
   REPEAT
   END
>>
```

You can add more WHILE − REPEAT − END loops for other similar conversions if you like. The loop uses the 1 or 0 returned by ↓MATCH, to check if it some matching has been done or not. If something matched, it repeats. If nothing matched it exits. There also a funny thing in this loop. Remember that the general form of such loops is:

```
WHILE
   test-clause
REPEAT
   body
END
```

But in this case we have done something like:

```
WHILE
   body-and-test-clause
REPEAT
END
```

The actions to be repeated is also where the test-clause resides, between the WHILE and the REPEAT statement, because

↓ MATCH returns both the results of the work *and* a true (1) or false (0). Nice demonstration of the flexibility of the HP49G, isn‘t it? Somehow it reminds me of C, where such „compacting“ of test-clauses and work in a single line are also possible.

Let‘s now move on to:

$$\tan(x)+\tan(y)=\frac{\sin(x+y)}{\cos(x)\ \cos(y)}$$

Enter the left hand side, $\mathrm{TAN}(X)+\mathrm{TAN}(Y)$, and press **TAN2SC**, to convert TAN to SIN and COS functions that appear on the right hand side. Press **EXPAND** and you get:

$$\frac{\mathrm{COS}(Y)\ \mathrm{SIN}(X)+\mathrm{COS}(X)\ \mathrm{SIN}(Y)}{\mathrm{COS}(X)\ \mathrm{COS}(Y)}$$

The denominator already looks like what we want. Press ▼ to get the expression in the EQW, select the numerator and press **TCOLLECT**. Voila!

And now for some examples:

1) Turn $\mathrm{SIN}(X)^2-\mathrm{SIN}(Y)^2$ to a product of trigonometric functions of $\frac{X}{2}$ and $\frac{Y}{2}$.

The method described above works here. We first **COLLECT** to turn the expression to:

$$(\mathrm{SIN}(X)-\mathrm{SIN}(Y))\ (\mathrm{SIN}(X)+\mathrm{SIN}(Y))$$

Then we use separately for $\mathrm{SIN}(X)-\mathrm{SIN}(Y)$ and $\mathrm{SIN}(X)+\mathrm{SIN}(Y)$:

a) **HALFTAN**
b) **TAN2SC**
c) **COLLECT**
d) **TCOLLECT** applied to the first and second sub-expression of the numerator
e) **TCOLLECT** applied to the denominator and **EXPAND**.

Or we just use the program from above.

---

2) Turn

$$\frac{\mathrm{SIN}(X)^2-\mathrm{SIN}(Y)^2}{(\mathrm{COS}(X)+\mathrm{COS}(Y))^2}$$

to a product of trigonometric functions of $\frac{X}{2}$ and $\frac{Y}{2}$.

We first **COLLECT** to turn the expression to:

$$\frac{(\mathrm{SIN}(X)-\mathrm{SIN}(Y))\ (\mathrm{SIN}(X)+\mathrm{SIN}(Y))}{(\mathrm{COS}(X)+\mathrm{COS}(Y))^2}$$

Then we use the above method separately for $\mathrm{SIN}(X)-\mathrm{SIN}(Y)$ and $\mathrm{SIN}(X)+\mathrm{SIN}(Y)$ on the numerator and for the denominator $(\mathrm{COS}(X)+\mathrm{COS}(Y))^2$d. At the end we also use **TRIGTAN** and we get:

$$\mathrm{TAN}\left(\frac{X+Y}{2}\right)\ \mathrm{TAN}\left(\frac{X-Y}{2}\right)$$

Or we just use the program from above, followed by a **TRIGTAN**.

---

3) Show that:

$$\frac{COS(2\cdot A)-COS(4\cdot A)}{COS(4\cdot A)+COS(2\cdot A)} = TAN(A)\cdot TAN(3\cdot A)$$

Again, enter the lest hand side and press HALFTAN, TAN2SC and COLLECT on the whole expression. Then TCOLLECT each factor of the numerator and the denominator separately. Then apply TRIGTAN on the whole expression. Or use the program and then TRIGTAN.

---

4) Convert the expression:

$$1+SIN(X)+COS(X)+SIN(X)\cdot COS(X)$$

to a product.

Enter the expression. Press HALFTAN, TAN2SC and COLLECT. After this you have:

$$\frac{\left(SIN\left(\frac{X}{2}\right)+COS\left(\frac{X}{2}\right)\right)^2\cdot COS\left(\frac{X}{2}\right)^2\cdot 2}{\left(SIN\left(\frac{X}{2}\right)^2+COS\left(\frac{X}{2}\right)^2\right)^2}$$

Get the expression in the EQW and apply TRIG on the denominator, to replace it with a 1. Now select the sub-expression:

$$SIN\left(\frac{X}{2}\right)+COS\left(\frac{X}{2}\right)$$

of the numerator and press TCOLLECT. Press ENTER and then COLLECT to get:

$$COS\left(\frac{X}{2}\right)^2\cdot cos\left(\frac{x}{2}-\frac{\pi}{4}\right)\cdot 4$$

In the last example the step TCOLLECT did another trigonometric transformation:

$$a\cdot sin(x)+b\cdot cos(x)=\sqrt{a^2+b^2}\cdot cos\left(x+arctan\left(\frac{b}{a}\right)-\frac{\pi}{2}\right)$$

If you enter the left hand side and press TCOLLECT then you get that. But this works only in real mode. In complex mode pressing TCOLLECT doesn‘t do anything. Also, if a and b are expressions with trigonometric functions themselves, then you get different results, depending on what exactly a and b look like. So if you want this type of conversion to be performed independently of what a and b look like, you should write a small program (perhaps using MATCH ;-)) to always get the desired result.

That‘s all for today. Of course if some genius out there finds a better/faster method, then please tell us, so that we don‘t raise the consumation of coffee to unbelievable degrees, waiting for the HP49G to finish some calculation. Having said that and after all complaining about the slowness of the HP49G, how much time would such things take, if we were supposed to do them by hand?

Greetings,
Nick.

Trigonometry with the HP49G - Part 7

Hi everybody!

We are at the seventh part of our trigo marathon already, if I didn‘t make any mistakes with counting. In this seventh part we‘ll take a look at the inverse functions of sin, cos and tan. We‘ll also take a look at Trabakoulas‘ time travel.

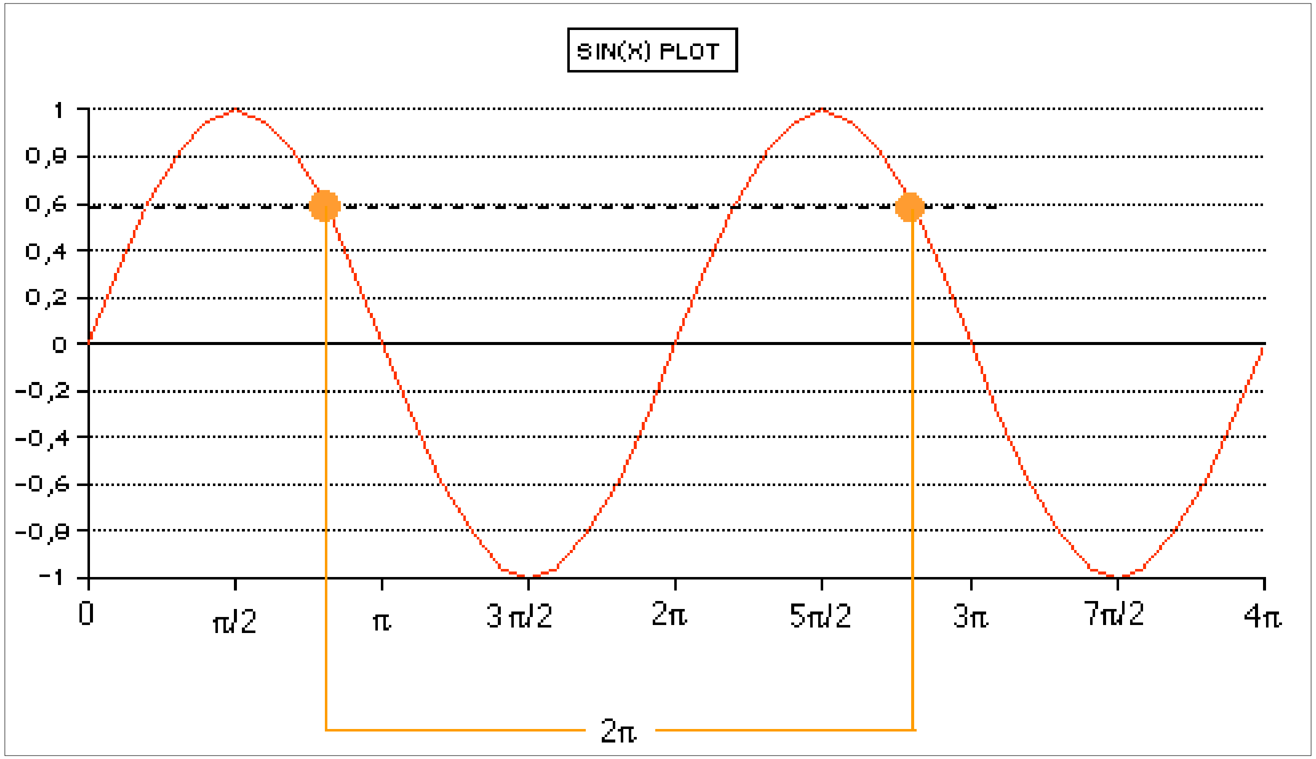
You of course know that trigonometric functions are periodic. They behave like doing after a while what they already have done. For example the sin function keeps repeating itself, as it oscillates between 1 and −1.

The distance between two x-coordinates which $SIN(X)$ sends to the same y-coordinate is the period of the function. (Actually this is not quite correct, but it suffices for now.) For $SIN(X)$ the period is 2π. If we express this algebraically, then we have:

$$SIN(X+2\pi)=SIN(X)$$

You can do this on your HP49G: Enter $SIN(X+2\pi)$, then press TEXPAND and EXPAND. The result is $SIN(X)$. Or enter $SIN(X+4\pi)$ and press TEXPAND and EXPAND. Same result!

The period of $COS(X)$ is also 2π. That means, if you have some quantity and take its cosine, then adding 2π to this quantity and taking the cosine, returns the same number.



The period of $TAN(X)$ is π.

Now, why is all this important? Imagine that you have for example $SIN(X)$ and you want the quantity X. There are more than one X which have the known sine. In fact there is an infinite number of such X.

This was what Trabakoulas found difficult to understand. The aliens told him all about periods and the like and he applied this to time coordinate. He found that if he starts at t_0 assuming a $\sin(\omega\,(t_0+t))$ function with frequency ω for his picture of the world, then when $t=\frac{2\ n1\ \pi}{\omega}$ has passed by, he actually has reached the point where he started, though he never went backwards in time. But this contradicts his observation of steadily getting older. ;-)

The functions that return an angle when fed a value of a trigonometric function of that angle are the inverse trigonometric functions. The inverse function to $\mathrm{SIN}(X)$ is on the HP49G $\mathrm{ASIN}(X)$. It finds an angle, or arc, which has a given sine. If you solve an equation like:

$$\mathrm{SIN}(X) = a$$

for X, then the HP49G answers with:

$$\{x = -((2\cdot n1-1)\cdot\pi) + \mathrm{ASIN}(a) \quad x = 2\cdot n1\cdot\pi + \mathrm{ASIN}(a)\}$$

Both solutions contain an arbitrary integer $n1$. The infinite number of values that this integer can have, shows that there is also an infinite number of angles x, that have the given sine a. It would be nice if we could show that the found solutions really have the sine a. So press SIN while the solutions list is on stack level 1. Now you have:

$$\left\{\begin{array}{l}\mathrm{SIN}(X) = \mathrm{SIN}(-((2\cdot n1-1)\cdot\pi) + \mathrm{ASIN}(a))\\ \mathrm{SIN}(X) = \mathrm{SIN}(2\cdot n1\cdot\pi + \mathrm{ASIN}(a))\end{array}\right\}$$

Both right hand sides can be TEXPANDed, so press TEXPAND. Now press OBJ→, and then ◀ to explode the solutions list. On stack level 1 you have:

$$\mathrm{SIN}(X) = \sqrt{-(a^2-1)}\cdot\mathrm{SIN}(2\cdot n1\cdot\pi) + a\cdot\mathrm{COS}(2\cdot n1\cdot\pi)$$

Here comes again VPN's idea and wish for INTEGERASSUME. The HP49G *returns* solutions that contain some arbitrary integer, but afterwards it *doesn't know* what variables are assumed to be integers. If we had this, then the above formula would simplify to a because $\mathrm{SIN}(2\cdot n1\cdot\pi) = 0$ and $\mathrm{COS}(2\cdot n1\cdot\pi) = 1$ when $n1$ is integer. The same is true for the other solution. We can of course make a program that replaces $\mathrm{SIN}(2\cdot n1\cdot\pi)$ with 0 and $\mathrm{COS}(2\cdot n1\cdot\pi)$ with 1, but having such a feature like INTEGERASSUME would be better.

Anyway, the other inverse trigonometric functions are ACOS and ATAN. There are also commands that convert between them. These are:

ACOS2S - convert $\arccos(x)$ to $\frac{\pi}{2} - \arcsin(x)$

ASIN2C - convert $\arcsin(x)$ to $\frac{\pi}{2} - \arccos(x)$

ASIN2T - convert $\arcsin(x)$ to $\arctan\left(\frac{x}{\sqrt{1-x^2}}\right)$

ATAN2S - convert $\arctan(x)$ to $\arcsin\left(\frac{x}{\sqrt{1-x^2}}\right)$

For example, if you have $\mathrm{ATAN}(X)$ and want to convert to ACOS, then you press ATAN2S and then ASIN2C. The result is:

$$\frac{\pi}{2} - \mathrm{ACOS}\left(\frac{X}{\sqrt{X^2+1}}\right)$$

The HP49G has some automatic simplifications when it deals with inverse trigonometric functions. For example, enter $\mathrm{ASIN}(X)$ and then press COS. The result is $\sqrt{-(X^2-1)}$. Try also other combinations. Enter an inverse trigonometric function (ASIN, ACOS, ATAN) and then press the key of a trigonometric function (SIN, COS, TAN) in any combination you like. Look how the HP49G gets rid of the inverse trigonometric functions, returning expressions with no trigonometric functions at all.

Let‘s do some examples:

1) Show that $\mathrm{ASIN}(X)+\mathrm{ACOS}(X)$ is a constant.

Enter the expression and press ACOS2S and then EXPAND, to get $\frac{\pi}{2}$.

---

2) Show that $\mathrm{SIN}(\mathrm{ACOS}(X))+\mathrm{COS}(\mathrm{ASIN}(X))=1-X^2$

Enter the left hand side and press TEXPAND, EXPAND. Or enter the left hand side and press EVAL.

---

3) Show that the expression

$$\mathrm{SIN}\left(\mathrm{ATAN}(X)+\mathrm{ATAN}\left(\frac{1}{X}\right)\right)$$

can be used as a kind of definition of the function SIGN.

Enter the expression and press TEXPAND. Then press EXPAND. The result is $\frac{|X|}{X}$ which can be thought as a definition for SIGN. (What happens at $X=0$?)

---

4) Given an angle a, find all angles that have a sine equal to $-\mathrm{SIN}(a)$.

The equation $\mathrm{SIN}(X)=-\mathrm{SIN}(a)$ must be solved for X. So enter the equation, enter X and then press SOLVE.

Example (4) shows where we are going to go in the next part. Yes, you guessed right: Trigonometric equations and their solutions! So if you like, send me any trigonometric equation that you find hard/impossible to solve with the HP49G, and I‘ll try to solve them and take them in the next part.

Greetings,
Nick.

Trigonometry with the HP49G - Part 8

Hi everybody!

In the previous parts of the trigonometry marathon we learned many things about the trigonometric and some algebraic capabilities of the HP49G. Things that will be very useful for what we are going to do in this part: Solve trigonometric equations.

First of all, let it be said, *there is no general method that will solve all trigonometric equations*. But there are some groups of trigonometric equations. Any equation that belong to such a group can be solved using the same method. Of course the method for solving an equation that belongs to one group will be different from the method for solving an equation that belongs to another group.

But we can make a program, that checks to which group such an equation belongs, and then acts accordingly. The general requirements for this program will be:

- Recognise the group that such an equation belongs to, no matter how the equation is written.
- Let as much as possible be done by the built in CAS.
- When an equation doesn't belong to the groups that we examine here, pass it to the built in SOLVE.

The commented code that represents the thoughts/ideas here, is at the end of this part. It is written solely in UserRPL. You can download it from www.hpcalc.org or enter yourself it in your HP49G.

So let's start!

Group 1.
A very easy kind of trigonometric equation is:

$$a \cdot \mathrm{trigFunction}(x) = b$$

where trig function can be SIN, COS or TAN. As you might have expected the HP49G can solve such equations right out of the box.

Example:

Solve the equation $\mathrm{COS}(X) = \frac{2}{3}$ for X

Simply enter the equation, enter X and then press **SOLVE**.

The result is:

$$\left\{ X = -\left(2 \cdot n1 \cdot \pi + \mathrm{ACOS}\left(\frac{2}{3}\right)\right) \quad X = 2 \cdot n1 \cdot \pi + \mathrm{ACOS}\left(\frac{2}{3}\right) \right\}$$

So for this group, we don't need to program anything. :-)

Group 2.

$$\mathrm{trigFunction}(f(x)) = \mathrm{trigFunction}(g(x))$$

where trigfunction can be again SIN, COS or TAN (but the same for both sides of the equation) and $f(x)$ and $g(x)$ are two distinct functions of x. Let's try to solve such an equation:

Enter:

$$\mathrm{SIN}\left(3 \cdot X + \frac{\pi}{4}\right) = \mathrm{SIN}\left(2 \cdot X - \frac{\pi}{3}\right)$$

then enter X and if you are brave enough then press **SOLVE**. The HP49G needs an eternity to return some result. Actually I never was patient enough to let it finish this calculation. So perhaps it can solve such equations, but the time that it needs to do so is not acceptable. It is a bit strange that the HP49G can solve much more difficult looking equations easily, and at the same time it seems to hang with such easy things.

So let‘s help it. Press ON to interrupt the calculation, if you were brave enough to start it. The equation:

$$\mathrm{SIN}\left(3\cdot X+\frac{\pi}{4}\right)=\mathrm{SIN}\left(2\cdot X-\frac{\pi}{3}\right)$$

tells us, that X is such that the arcs of:

$$3\cdot X+\frac{\pi}{4}$$

and of:

$$2\cdot X-\frac{\pi}{3}$$

have equal sines. Remember the property of arc sine in part 7? There are infinite arcs whose sines are equal to:

$$\mathrm{SIN}\left(3\cdot X+\frac{\pi}{4}\right)$$

They are:

$$\cdots,\ -4\cdot\pi+3\cdot X+\frac{\pi}{4},\ -2\cdot\pi+3\cdot X+\frac{\pi}{4},\ 3\cdot X+\frac{\pi}{4},\ 2\cdot\pi+3\cdot X+\frac{\pi}{4},\ 4\cdot\pi+3\cdot X+\frac{\pi}{4},\ \cdots$$

and also

$$\cdots,\ 5\cdot\pi+3\cdot X+\frac{\pi}{4},\ 3\cdot\pi+3\cdot X+\frac{\pi}{4},\ \pi-3\cdot X+\frac{\pi}{4},\ -\pi-3\cdot X+\frac{\pi}{4},\ -3\cdot\pi+3\cdot X+\frac{\pi}{4},\ \cdots$$

The two sets of solutions are represented by:

$$2\cdot k_1\cdot\pi+3\cdot X+\frac{\pi}{4}$$

and by:

$$-((2\cdot k_1-1)\cdot\pi)-3\cdot X+\frac{\pi}{4}-$$

where k_1 is an arbitrary integer. The last 2 formulae are all arcs whose sines are equal to the sines of:

$$3\cdot X+\frac{\pi}{4}$$

The same way all arcs whose sines are equal to the sines of:

$$2\cdot X-\frac{\pi}{3}$$

are given through:

$$2\cdot l_1\cdot\pi+2\cdot X-\frac{\pi}{3}$$

and:

$$-((2\cdot l_1-1)\cdot\pi)-2\cdot X-\frac{\pi}{3}$$

where l_1 is another arbitrary integer.

Now, we want to find X such that:

$$\mathrm{SIN}\left(3\cdot X+\frac{\pi}{4}\right)=\mathrm{SIN}\left(2\cdot X-\frac{\pi}{3}\right)$$

which means that the arcs having these sines must also be equal. That leads us to 4 equations:

1) $2\cdot k_1\cdot\pi+3\cdot X+\frac{\pi}{4}=2\cdot l_1\cdot\pi+2\cdot X-\frac{\pi}{3}$

2) $-((2\cdot k_1-1)\cdot\pi)-3\cdot X+\frac{\pi}{4}=-((2\cdot l_1-1)\cdot\pi)-2\cdot X-\frac{\pi}{3}$

3) $2\cdot k_1\cdot\pi+3\cdot X+\frac{\pi}{4}=-((2\cdot l_1-1)\cdot\pi)-2\cdot X-\frac{\pi}{3}$

4) $-((2\cdot k_1-1)\cdot\pi)-3\cdot X+\frac{\pi}{4}=2\cdot l_1\cdot\pi+2\cdot X-\frac{\pi}{3}$

From the first, we derive:

$$2\cdot k_1\cdot\pi+3\cdot X+\frac{\pi}{4}=2\cdot l_1\cdot\pi+2\cdot X-\frac{\pi}{3}\Leftrightarrow$$

$$3\cdot X+\frac{\pi}{4}=2\cdot l_1\cdot\pi-2\cdot k_1\cdot\pi+2\cdot X-\frac{\pi}{3}\Leftrightarrow$$

$$3\cdot X+\frac{\pi}{4}=2\cdot(l_1-k_1)\cdot\pi+2\cdot X-\frac{\pi}{3}$$

Because l_1 and k_1 are integers, l_1-k_1 is also an integer, which we can give the name n_1. So we have:

$$3\cdot X+\frac{\pi}{4}=2\cdot n_1\cdot\pi+2\cdot X-\frac{\pi}{3}$$

We turned the trigonometric equation to an equation without any trigonometric functions, which the HP49G can easily solve.

The second equation gives us the same set of solutions as the first one.

From the third equation we derive:

$$2\cdot k_1\cdot\pi+3\cdot X+\frac{\pi}{4}=-((2\cdot l_1-1)\cdot\pi)-2\cdot X-\frac{\pi}{3}\Leftrightarrow$$

$$3\cdot X+\frac{\pi}{4}=-2\cdot k_1\cdot\pi-((2\cdot l_1-1)\cdot\pi)-2\cdot X-\frac{\pi}{3}\Leftrightarrow$$

$$3\cdot X+\frac{\pi}{4}=-((2\cdot(k_1+l_1)-1)\cdot\pi)-2\cdot X-\frac{\pi}{3}$$

Because k_1 and l_1 are integers, k_1+l_1 is also an integer, which we can give the name m_1. So we have:

$$3\cdot X+\frac{\pi}{4}=-((2\cdot m_1-1)\cdot\pi)-2\cdot X-\frac{\pi}{3}$$

The last equation can be simplified a little bit further if we consider m_1, which is an integer ...,-3, -2, -1, 0, 1, 2, 3, ...

The quantity $-((2\cdot m_1-1)\cdot\pi)$ is then:

$$\cdots,\ 7\cdot\pi,\ 5\cdot\pi,\ 3\cdot\pi,\ \pi,\ -\pi,\ -3\cdot\pi,\ -5\cdot\pi,\ \cdots$$

That means, the quantity $-((2\cdot m_1-1)\cdot\pi)$ is π multiplied by an odd integer. This can be also represented by $(2 \cdot n_1 + 1) \cdot \pi$.

So the last equation turns to:

$$3 \cdot X + \frac{\pi}{4} = (2 \cdot n_1 + 1) \cdot \pi - 2 \cdot X - \frac{\pi}{3}$$

That means, that if we could check that the equation belongs to the group $SIN(A) = SIN(B)$, we could replace it with the two equations $A = 2 \cdot n_1 \cdot \pi + B$ and $A = (2 \cdot n_1 + 1) \cdot \pi - B$ and solve these two new equations for X, instead of the original equation.

How can we check that an equation belongs to the group $SIN(A) = SIN(B)$? We could for example MATCH $SIN(A) = SIN(B)$ with $A = 2 \cdot n_1 \cdot \pi + B$, check the flag that MATCH returns and act accordingly. But consider an equation of the form $SIN(A) = -SIN(B)$. This also belongs to the same group because $-SIN(B) = SIN(-B)$ and so the equation becomes $SIN(A) = SIN(-B)$. If we use MATCH to replace $SIN(A) = SIN(B)$, then cases like $SIN(A) = -SIN(B)$, or $-SIN(A) = SIN(B)$, or $-SIN(A) = -SIN(B)$, and also $SIN(A) + SIN(B) = 0$ and so on will not be MATCHed. Furthermore when the equation to solve is not factored but contains $SIN(A) - SIN(B)$ as a factor, then we of course can't make a MATCH.

What can we do? An easy solution is to first find all factors of the equation, build an equation with each factor equal to 0 and build a list with all these equations.

If our factoring part of the program also finds a -1 and $-SIN(A) + SIN(B)$ as factors of $-(SIN(A) - SIN(B))$, then the possible variations of such equations are:

$SIN(A) - SIN(B) = 0$
$SIN(A) + SIN(B) = 0$

The built-in command FACTORS does this, but we will not use it here because it factors too much for our purposes. You'll see later on why it doesn't exactly fit here. The built in command COLLECT does factoring that better suits our needs, but it doesn't return a list of all factors. But we can use COLLECT and a little bit programming to get all factors from COLLECT not in an algebraic object but in a list. The code that does this is commented at the end of this part of the marathon. It takes an algebraic object and returns the factors that COLLECT finds in a list. Note that it simply rejects any denominator from a factor, which can be dangerous if the denominator is 0 for the solutions of the numerator.

If an equation contains factors of the form $SIN(A) - SIN(B)$ or $SIN(A) + SIN(B)$, the code will also return the corresponding equations in a list. The two forms of course can't be MATCHed both at once. But we can make an additional pre-check and convert one possible forms of an equation to the other form, and then replace it with the list:

$$\{A = 2 \cdot n_1 \cdot \pi + B \quad A = (2 \cdot n_1 + 1) \cdot \pi - B\}$$

which can be solved easily. So we choose $SIN(A) - SIN(B) = 0$ to represent both possible forms of such equations and need to make an additional check that converts $SIN(A) + SIN(B) = 0$ to $SIN(A) - SIN(-B) = 0$.

But there is an additional thing, to take care of. Suppose that we want to solve for X. The equation can in general contain factors that don't depend on X, like for example 2, $var_1 - var_2$ and so on. These factors will later lead to equations like $2 = 0$ or $var_1 - var_2 = 0$, which can't be solved for X. We must filter out such factors. This is done with a subsequent procedure which filters out all factors that don't depend on the variable for which we want to solve.

Exactly the same way, we can include code in the program, that solves equations like $COS(A) = COS(B)$. With the same considerations as above, we can replace equations of the form $\mathrm{COS}(A)+\mathrm{COS}(B)=0$ with $\mathrm{COS}(A)-\mathrm{COS}(B-\pi)=0$ because $\mathrm{COS}(B)=-\mathrm{COS}(B-\pi)$, then replace the created equation of the form $\mathrm{COS}(A)-\mathrm{COS}(B)=0$ with:

$$\{A = 2\,n_1\,\pi + B \quad A = 2\,n_1\,\pi - B\}.$$

For $\mathrm{TAN}(A)=\mathrm{TAN}(B)$ we can replace equations of the form $\mathrm{TAN}(A)+\mathrm{TAN}(B)=0$ with $\mathrm{TAN}(A)-\mathrm{TAN}(-B)=0$ because $\mathrm{TAN}(A)=-\mathrm{TAN}(-B)$, then replace the so created equation of the form $\mathrm{TAN}(A)-\mathrm{TAN}(B)=0$ with $A = n_1\,\pi + B$.

We are at the end of the second group. Let‘s move on to the third group.

Group 3.

This group contains equations of the form

$$\mathrm{trigFunction1}(f(x)) = \mathrm{trigFunction2}(g(x))$$

where trigFunction1, trigFunction2 are two different trigonometric functions and $f(x)$, $g(x)$ are two different terms that contain the variable to solve for. Examples would be:

$$\mathrm{COS}(X-\pi) = \mathrm{SIN}\left(2\,X+\frac{\pi}{3}\right),\quad \mathrm{TAN}(X) = \mathrm{SIN}\left(\frac{X}{2}+\pi\right)$$

and so on.

For equations that only contain SIN and COS but no TAN, it is easier. Take for example $\mathrm{COS}(A)=\mathrm{SIN}(B)$. Since it is:

$$\mathrm{SIN}(B)=\mathrm{COS}\left(B-\frac{\pi}{2}\right)$$

we can replace SIN(B) with $\mathrm{COS}\left(B-\frac{\pi}{2}\right)$ and then use the code that we already have written for the cases $\mathrm{COS}(A)-\mathrm{COS}(B)=0$. For example an equation of the form $\mathrm{SIN}(A)+\mathrm{COS}(B)=0$ will be MATCHed to:

$$\mathrm{COS}\left(A-\frac{\pi}{2}\right)+\mathrm{COS}(B)=0$$

through the extra code. The resulting equation

$$\mathrm{COS}\left(A-\frac{\pi}{2}\right)+\mathrm{COS}(B)=0$$

will be MATCHed to:

$$\mathrm{COS}\left(A-\frac{\pi}{2}\right)-\mathrm{COS}(B-\pi)=0$$

through the subsequent MATCH for which we already wrote the code. And the equation:

$$\mathrm{COS}\left(A-\frac{\pi}{2}\right)-\mathrm{COS}(B-\pi)=0$$

will be MATCHed to the equation list:

$$\left\{A-\frac{\pi}{2} = 2\,n_1\,\pi + B - \pi \quad A-\frac{\pi}{2} = 2\,n_1\,\pi - (B-\pi)\right\}$$

also with code which is already written.

In the same group we have also equations that contain TAN, like $\mathrm{SIN(A)=TAN(B)}$. For such equations, I didn't find a general programmable method (yet). It seems that the way to solve them, if there is a way, varies to much from case to case.

Group 4.
Let's move on to the fourth group. This group has trigonometric equations that are algebraic in a trigonometric function. An example would be: $a\ \mathrm{SIN}(X)^2 + b\ \mathrm{SIN}(X) + c = 0$. If we substitute Y for $\mathrm{SIN}(X)$, then the equation becomes $a\ Y^2 + b\ Y + c = 0$, which we can solve for Y. Then we have the two solutions of the quadratic equation $Y = \mathrm{solution}_1$ and $Y = \mathrm{solution}_2$. We make the back-substitution $Y = \mathrm{SIN}(X)$ we then we have $\mathrm{SIN}(X) = \mathrm{solution}_1$ and $\mathrm{SIN}(X) = \mathrm{solution}_2$, which can be solved easily. The HP49G can solve such equations without help. And because the original equation with the variable to solve for is passed to SOLVE, when none of the previous MATCHes did anything, we don't need to program additional code for such cases. Almost every algebraic expression with trigonometric functions that can be factored, can be solved this way.

Group 5.
The fifth group contains equations of the form $f(\sin(x),\cos(x),\tan(x)) = 0$. A general way to work with such equations is to convert every trigonometric function that appears in the equation to a function of $\mathrm{TAN}\left(\frac{X}{2}\right)$ with the command HALFTAN. So we will have an equation where only terms with $\mathrm{TAN}\left(\frac{X}{2}\right)$. This can then be solved by the HP49G through factorisation. Many of the resulting equations contain big powers of $\mathrm{TAN}\left(\frac{X}{2}\right)$ and so can't be factored analytically. But if you have set the flag -109 for numerical factorisation, then the HP49G returns numerical solutions.

The nice thing is that the HP49G does solve such equations without help, when the arguments of SIN, COS and TAN, are all simply X. The not so nice thing is that the HP49G gets more and more problems when the arguments are the same for all trigonometric functions, but they are more complicated than simply X. For example it solves $\mathrm{SIN}(X)+\mathrm{COS}(X)+\mathrm{TAN}(X)$ but $\mathrm{SIN}(X+1)+\mathrm{COS}(X+1)+\mathrm{TAN}(X+1)$ causes much more problems.

So we need to program additional code which checks to see if there are at least two different trigonometric functions, and if the arguments of them are all the same, then substitutes, say Y for this argument, transforms all trigonometric functions to $\mathrm{TAN}\left(\frac{X}{2}\right)$-functions, solves for Y, then substitutes back $Y = \mathrm{argument}$ and then solves for X.

Group 6.
The sixth group contains equations of the form $a\ \sin^2(x) + b\ \cos^2(x) + c\ \sin(x)\ \cos(x) + d = 0$. The HP49G often runs into troubles, when trying to solve such equations. It seems that often the factorisation of such equations results in factors which are difficult to solve for the CAS. That's why we didn't use FACTORS at the start. The equation could be factored in such a way, that neither the form of the equation can be easily recognised, nor the resulting equations of the form $\mathrm{factor} = 0$ can be solved. First the idea for recognising such a form. Use FDISTRIB to remove all groupings of terms. Then MATCH $\mathrm{COS}(X)^2$, $\mathrm{SIN}(X)^2$ and $\mathrm{COS}(X)\ \mathrm{SIN}(X)$ to 0. EXPANDing will then return an equation of the form $d = 0$. Check to see if d depends on the variable to solve for. If not, then we have an equation that belongs to group 6.

Now how to transform it to something easy to solve: Subtract $d = 0$ from the equation and then add $d\ \left(\mathrm{SIN(X)}^2 + \mathrm{COS(X)}^2\right)$ to the left hand side. Since $\mathrm{SIN(X)}^2 + \mathrm{COS(X)}^2 = 1$ we have added and subtracted d, so the equation remains the same. But now it is in the form:

$$a\ \sin^2(x) + b\ \cos^2(x) + c\ \sin(x)\ \cos(x) + d\ \left(\sin^2(x) + \cos^2(x)\right) = 0$$

or

$$(a+d)\ \sin^2(x) + (b+d)\ \cos^2(x) + c\ \sin(x)\ \cos(x) = 0$$

that means in the general form:

$$A\ X_1^{\ 2} + B\ X_2^{\ 2} + C\ X_1\ X_2 = 0$$

where $A = a + d$, $B = b + d$, $C = c$, $X_1 = \sin(x)$ and $X_2 = \cos(x)$.

Equations of the form $A\ X_1^{\ 2} + B\ X_2^{\ 2} + C\ X_1\ X_2 = 0$ can always be factored to:

$$\frac{\left(2\ A\ X_1 + C\ X_2 + X_2\ \sqrt{-\left(4\ B\ A - C^2\right)}\right)\ \left(2\ A\ X_1 + C\ X_2 - X_2\ \sqrt{-\left(4\ B\ A - C^2\right)}\right)}{4\ A}$$

(Try it yourself).

The factored form of such equations can be easily solved by the CAS. If we consider that $X_1 = \sin(x)$, $X_2 = \cos(x)$ then we can divide by $\cos(x)$ and get and equation of the form $a\ \ \tan(x) + b = 0$ for every factor. (Caution! we must check if the solutions are such that also $\cos(x) = 0$)

So we use FACTORS to find the two factors, divide each factor by $\cos(x)$ and then use TRIGTAN to change $\frac{\sin(x)}{\cos(x)}$ to $\tan(x)$. We have now two equations with only one occurrence of $\tan(x)$, which can both be solved by the built-in CAS.

Before we go to the program that does all this, some things must be said. First of all, the program is far from being perfect, if there is such a thing like a perfect program. I tried to make it as general as possible, but there will be always cases, where it doesn‘t give solutions, or where it even crashes. We could add for example code for argument checking, or code for solving additional cases, or code for fixing bugs, which can appear when some equation behaves in such a way, that it leads to errors. Also we could add code for checking the behaviour of solutions, when some denominators are simply thrown away. And many many other things.

And at the end, after so much blah blah, here it is. TRISOL, the program for solving trigonometric equations. (No, it is not TRIstan and ISOLde, it is TRIgonometric SOLve ;-))

```
%%HP: T(3)A(R)F(.);
\<< OVER

@Make a list with the
@COLLECTed equation

"Finding factors
" 1
DISP COLLECT 1 \->LIST

@Put this list in local variable factors
@Next routine returns all factors that
@the previous COLLECT found in a list.

\-> factors
  \<< 20 SF           @Set flag 20
    WHILE 20 FS?      @While flag 20 is set
    REPEAT 20 CF      @repeat
                      @clear flag 20
```

```
factors 1            @Put factors list
                     @and a 1 on the stack
                     @DOSUBS procedure starts here
      \<< \-> fact   @store current factor in
                     @local variable fact
        \<<
          IF fact    @if current factor is alg
TYPE 9 ==
          THEN       @then
            CASE fact    @case factor is a product
OBJ\-> { * } SWAP POS
              THEN
DROP 20 SF               @then drop the argument
                         @count. Set flag 20
              END
DROPN fact OBJ\->        @drop as many objects as
                         @the argument count
{ NEG} SWAP POS          @case the factor is negated
              THEN       @then drop argument count
DROP -1 20 SF            @return -1 and set flag 20
              END
DROPN fact OBJ\-> { / }  @drop as many objects as
                         @the argument count
SWAP POS                 @case factor is a quotient
              THEN
DROP2 20 SF              @drop argument count and
              END        @denominator. Set flag 20
DROPN fact
            END          @default case: drop as many
                         @objects as the argument count.
                         @return current factor
          ELSE fact @else current factor is
                    @name or number. Simply return it
          END
        \>>
      \>> DOSUBS         @DOSUBS procedure ends here
'factors' STO    @Store factors list in local factors
    END factors          @End of WHILE-REPEAT loop
  \>> NOVAL NOVAL RCLF   @Return factors list and
                         @two NOVAL and the current
                         @flags. The NOVALs are the
                         @initial contents of the
```

```
                         @locals subs and subvar
                         @which we will use later on.
@Store arguments in local variables.
@ eq:     The unchanged equation.
@ var:    The variable to solve for
@ feq:    List with all factors
@ subs:   Variable will be used if
@         substitutions must be done
@ subvar: Name of the substitution variable
@ flags:  The user flags


\-> eq var feq subs
subvar flags
  \<<

@The factors that do not depend on
@the variable to solve for will
@be filtered out from feq


"Filtering factors
"
1 DISP feq 1     @Put feq and a 1 on the stack
                 @DOSUBS procedure starts here

    \<< NSUB R\->I 2 DISP   @display current
                            @factor count
      IF DUP TYPE 9 \=/     @If current factor
OVER TYPE 6 \=/ AND         @is not algebraic
                            @and not name
      THEN DROP             @then drop it
      ELSE                  @else
        IF LNAME DUP          @if it doesn't
{ } SAME                      @contain any names
        THEN DROP2            @then drop it
        ELSE                  @else
          IF AXL var            @if it doesn't
POS NOT                         @contain the
                                @variable to
                                @solve for
```

```
          THEN DROP             @then drop it
          ELSE 0 =              @else
                                @else build up
                                @the equation
                                @factor=0
          END
        END
      END
    \>>       @DOSUBS procedure ends here
DOSUBS 'feq'       @Store the resulting
STO                @list of factors in
                   @local variable feq

feq 1      @return filtered factors list and 1

               @DOSUBS procedure starts here
    \<<

               @MATCH each factor to a standard
               @form that will be used with MATCH
               @later.

"Standardizing
"
NSUB R\->I + 1 DISP

@MATCH forms with SIN and COS
@(Third group -> second group)

@MATCH sin(a)+cos(b)=0 to cos(a-Pi/2)+cos(b)=0
{ '
SIN(&A)+COS(&B)=0' '
COS(&A-\pi/2)+COS(&B)=0
' } \|vMATCH DROP

@MATCH cos(a)+sin(b)=0 to cos(a)+cos(b-Pi/2)=0
{ '
COS(&A)+SIN(&B)=0' '
COS(&A)+COS(&B-\pi/2)=0
' } \|vMATCH DROP

@MATCH sin(a)-cos(b)=0 to cos(a-Pi/2)-cos(b)=0
{ '
SIN(&A)-COS(&B)=0' '
COS(&A-\pi/2)-COS(&B)=0
' } \|vMATCH DROP

@MATCH cos(a)-sin(b)=0 to cos(a)-cos(b-Pi/2)=0
{ '
COS(&A)-SIN(&B)=0' '
COS(&A)-COS(&B-\pi/2)=0
' } \|vMATCH DROP


@MATCH forms with SIN only or COS only
@(Find form that represents all variations of
@an equation of the second group)

@MATCH sin(a)+sin(b)=0 to sin(a)-sin(-b)=0
{ '
SIN(&A)+SIN(&B)=0' '
SIN(&A)-SIN(-&B)=0' }
\|vMATCH DROP


@MATCH cos(a)+cos(b)=0 to cos(a)-cos(b-pi)=0
{ 'COS(&A
)+COS(&B)=0' 'COS(&A)
-COS(&B-\pi)=0' }
\|vMATCH DROP

@MATCH tan(a)+tan(b)=0 to tan(a)-tan(-b)=0
{ 'TAN(&A
)+TAN(&B)=0' 'TAN(&A)
-TAN(-&B)=0' } \|vMATCH
DROP

\-> eqfact          @Store current equation
                    @factor=0 in local variable


@The next local variables procedure finds
@the type of the current equation factor=0.
@If the equation belongs to any of the groups
```

```
@that were considered in this part, then it
@puts an equivalent equation in eqfact, which
@can be solved much easier by the CAS

      \<<
"Finding type of eq
"
NSUB R\->I + 1 DISP
        CASE     @Second group of equations
eqfact           @Case sin(a)-sin(b)=0
{
'SIN(&A)-SIN(&B)=0' '
&A=2*\pi*n1+&B' }
\|vMATCH
          THEN     @then put a list with
                   @A=2*pi*n1+B and
                   @A=(2*n1+1)*pi-B
                   @in local eqfact
"Type sin(a)-sin(b)=0
"
1 DISP eqfact { 'SIN(
&A)-SIN(&B)=0' '&A=(2
*n1+1)*\pi-&B' } \|vMATCH
DROP 2 \->LIST 'eqfact'
STO
          END DROP
                      @Case cos(a)-cos(b)=0
eqfact { 'COS(&A)-COS
(&B)=0' '&A=2*\pi*n1+&B
' } \|vMATCH
          THEN     @then put a list with
                   @A=2*pi*n1+B and
                   @A=2*pi*n1-B
                   @in local eqfact

"Type cos(a)-cos(b)=0
"
1 DISP eqfact { 'COS(
&A)-COS(&B)=0' '&A=2*
\pi*n1-&B' } \|vMATCH
DROP 2 \->LIST 'eqfact'
STO
          END DROP
eqfact             @Case tan(a)-tan(b)=0
{ 'TAN(&A)-TAN
(&B)=0' '&A=\pi*n1+&B'
} \|vMATCH
          THEN     @then put
                   @A=pi*n1+B
                   @in local eqfact
"Type tan(a)-tan(b)=0
"
1 DISP 'eqfact' STO
          END DROP
eqfact FDISTRIB      @Sixth group of equations
                   @Put the fully distributed
                   @form of eqfact in eqfact
'eqfact' STO eqfact        @Case eqfact contains
var SIN 2 ^ { 0 } +        @at least two of the
\|vMATCH SWAP var COS 2    @forms a*SIN(X)^2
^ { 0 } + \|vMATCH SWAP    @b*COS(X)^2 c*SIN(X)*COS(X)
var COS var SIN *          @and a form that does not
{ 0
} + \|vMATCH SWAP '&A'     @contain the variable to
var COS * var SIN *        @solve for
{
0 } + \|vMATCH SWAP 5
ROLLD OR 3 \->LIST
\GSLIST 2 \>= SWAP EXPAND
LNAME
          IF DUP { }
SAME
          THEN DROP 0
          ELSE AXL
var POS
          END NOT ROT
AND
          THEN              @then replace the constant
                            @term with itself
"Type asin\178x+bcos\178x+
csinxcosx"                  @multiplied with
1 DISP eqfact OVER -        @SIN(X)^2+COS(X)^2
SWAP var SIN 2 ^ var        @and factor the
COS 2 ^ + * + FACTORS       @resulting eqfact
```

```
1
"Filtering factors,
building equations"
1 DISP

              @Another DOSUBS starts here
              @to filter out factors that
              @do not contain the variable
              @to solve for.

            \<< NSUB
R\->I 2 DISP
              IF DUP
TYPE 9 \=/ OVER TYPE 6
\=/ AND
              THEN
DROP
              ELSE
                IF
LNAME DUP { } SAME
                THEN
DROP2
                ELSE
IF AXL var POS NOT
THEN DROP
ELSE 0 =
END
                END
              END
            \>>  @DOSUBS procedure ends here
DOSUBS
1             @Next DOSUBS procedure
              @also checks to see if the
              @equation contains SIN and COS.
              @If it does, then it divides
              @by COS(X) and then applies
              @TRIGTAN to the factor.
              @It builds the equivalent
              @equations of the form
              @a*TAN(X)+b=0, which can
              @be solved easily by the CAS
            \<<
```

```
              IF DUP
\->LST DUP { SIN } HEAD
POS 1 \>= SWAP { COS }
HEAD POS 1 \>= AND
              THEN
var COS / DISTRIB
TRIGTAN
              END
            \>> DOSUBS
              @DOSUBS procedure ends here

'eqfact' STO
          END DROP @Fifth group of equations
                          @Find all arguments of
                          @trigonometric functions.

eqlst trigarg
eqfact \->LST             @Convert algebraic to list
{ } \->
eqlst trigarg
          \<< eqlst 1    @Return the list
                         @of the algebraic
                        @DOSUBS procedure starts here
            \<<
              IF DUP   @If the current object is
{ SIN COS TAN } SWAP   @SIN COS or TAN
POS
              THEN
OVER EXPAND 'trigarg'  @Then expand the previous
STO+                   @object and add it to the list
                       @trigarg
              END
EVAL                   @evaluate current object
            \>>
                       @DOSUBS procedure ends here
DOSUBS
DROP

trigarg eqlst
          \>> DUP { TAN      @If there are at least
} HEAD POS 1 \>= OVER
{ SIN } HEAD POS 1 \>=       @two of the functions
```

```
ROT { COS } HEAD POS         @SIN COS TAN
1 \>= 3 \->LIST \GSLIST 2 \>=
          THEN DUP 2         @and if all arguments
            \<< SAME         @of these functions
            \>> DOSUBS       @are the same
            IF { 1 }
+ \PILIST
            THEN             @then
"Type f(sinx,cosx,tanx)
"
1 DISP eq HALFTAN            @turn all trigonometric
EXPAND FACTOR -105 CF        @functions to tan(arg/2)
OVER HEAD 2 / EXPAND
'TempSolVar' 2 \->LIST      @replace arg/2 with
\|^MATCH DROP 'eqfact'      @TempSolVar
STO 'TempSolVar' SWAP       @and store the back
HEAD 2 / EXPAND =           @substitution formula
'subs' STO var              @in local subs.
'subvar' STO
'TempSolVar' 'var'
STO
            ELSE DROP
            END
          END DROP     @Equation belongs to
                       @none of the above groups
"CAS Type
" 1 DISP
        END eqfact     @solve
var "Solving
" NSUB
R\->I + 1 DISP SOLVE
        IF subs NOVAL  @If subs contains something
\=/                    @different than NOVAL
        THEN           @then
          IF DUP { }     @if the solutions list
\=/                      @isn't empty
          THEN           @then
"Back substitution,
and solution"
1 DISP subs SUBST       @perform back substitution
subvar SOLVE            @and solve for original
          END subvar    @variable
'var' STO NOVAL         @restore var, subs, subvar and
'subs' STO NOVAL        @flags to their initial values
'subvar' STO
        END flags       @restore flags
STOF
      \>>
    \>> DOSUBS          @Do for each equation of the
  \>>                   @form factor=0
\>>
```

If you must solve trigonometric equations, then give it a try. Next time we'll be solving some examples with it.

Boy! I'm so tired, I see only SIN COS and TAN. Must go sleep now. (John, you got me! ;-)) No wake up till next part, keep tuned!

Solved Greetings(x)= 0
Nick.

P.S. Dreaming of the Meta-CAS that runs on the CAS that runs on the OS. Zzzzzzzzzzzzzzzz......

Trigonometry with the HP49G - Part 9

Hi everybody!

In this part we are going to solve many trigonometric equations. Some of them really weirdos. Some of them so strange that even the CAS and TRISOL together can‘t figure out how they return solutions. And we are going to see how the story of the universe is similar to the story of software.

You know of course that the Big Bang theory is the widely accepted theory about the birth of the universe. But did you know about the Big Bug theory? No? Oh, this is going to be the widely accepted theory about the birth of the software universe. Especially for the trigonometry software universe, there are reasons to believe that at some time in the past there was a huge Bug, a singularity, which we call TRISOL. Our knowledge about what was before TRISOL is quite limited, as God (Mr. Parisse ;-)) won‘t tell us much about the mysteries of the CAS. (Cosmic Algebra Superstring ;-))

But Big Bugs tend to evolve with time and sometimes, quite unexpectedly, they may contain usable code. Pattern formation out of the chaos, so to speak. ;-) So it happened with TRISOL. Trabakoulas the shepherd has edited some parts, after he came to me and told me that he lost some sheep because he used Big Bug TRISOL to find their positions. He left me in peace only after I had promised to pay for the lost sheep. Boy, why are these animals so expensive? ;-) But he gave me the re-edited and re-commented code, TRISOL the second, which we are going to use for solving a bunch of equations. It is at the end of the previous chapter, and at www.hpcalc.org together with this document.

After you downloaded it to your HP49G, switch to complex exact mode, set flag -109 for numeric factorisation, and here we go.

1) Let‘s start with an easy one. Solve $3\ \sqrt{2}\ \mathrm{COS}\left(X+\frac{\pi}{3}\right)=\frac{2}{3}$

Enter the equation, enter X and press TRISOL. It finds the solutions:

$$X=\frac{\left((6\ n_1-1)\ \pi+3\ \mathrm{ACOS}\left(\frac{\sqrt{2}}{9}\right)\right)}{3\ a}$$

and

$$X=\frac{-\left((6\ n_1+1)\ \pi+3\ \mathrm{ACOS}\left(\frac{\sqrt{2}}{9}\right)\right)}{3\ a}$$

in 37.6 seconds. If you solve the same equation for X with the command SOLVE, you get the same solutions in 17.4 seconds. It is clear that TRISOL has a big overhead trying to determine what kind of equation this is. Let‘s follow the fate of the equation as it passes through the processing teeth of TRISOL. First it is COLLECTEDed to:

$$\frac{\left(9\ \mathrm{COS}\left(X+\frac{\pi}{3}\right)-\sqrt{2}\right)\ \sqrt{2}}{3}$$

Then the denominator is dropped and the numerator gets converted to its factors list:

$$\left\{9\ \mathrm{COS}\left(X+\frac{\pi}{3}\right)-\sqrt{2}\quad \sqrt{2}\right\}$$

The factor $\sqrt{2}$ is filtered out, because it can‘t give us any solutions. The remaining factors are used to build a list of equations:

$$\left\{ 9\ \mathrm{COS}\left(a\ X + \frac{\pi}{3}\right) - \sqrt{2} = 0 \right\}$$

Each equation in the list (only one in this case) is checked for belonging to one of the special groups that TRISOL processes further. The checks that are done to see if it belongs to the group $a\ \sin^2(x) + b\ \cos^2(x) + c\ \sin(x)\ \cos(x) + d = 0$ are negative.

Because the equation is found to be none of the special kinds, TRISOL faithfully passes it to the built-in SOLVE, which does its work very well in this case.

---

2) Solve $\mathrm{SIN}\left(3\ X + \frac{\pi}{4}\right) = \mathrm{SIN}\left(2\ X - \frac{\pi}{3}\right)$

Using TRISOL you get the two solutions

$$X = \frac{(24\ n_1 - 7)\ \pi}{12}$$

and

$$X = \frac{(24\ n_1 - 13)\ \pi}{60}$$

in 17.2 seconds.

If you use SOLVE for this equation you get

$$X = 2\ n_1\ \pi - 2\ \mathrm{ATAN}\left(\frac{35\ \sqrt{2} - \left(28\ \sqrt{3} - 1\right)}{\left(64\ \sqrt{3} - 155\right)\ \sqrt{2} + 127\ \sqrt{3} - 156}\right)$$

in 330.9 seconds

Let‘s first see if the solutions are the same. If you apply →NUM on the expression

$$\mathrm{ATAN}\left(\frac{35\ \sqrt{2} - \left(28\ \sqrt{3} - 1\right)}{\left(64\ \sqrt{3} - 155\right)\ \sqrt{2} + 127\ \sqrt{3} - 156}\right)$$

to turn it to a number, and the apply XQ to this number, you get $\frac{7\ \pi}{24}$. The EXPANDed solution that SOLVE returned is then:

$$X = \frac{(24\ n_1 - 7)\ \pi}{12}$$

which is exactly the same as the first of the 2 solutions that TRISOL returned. If we make a sequence of such solutions entering:

$$X = \frac{(24\ n_1 - 7)\ \pi}{12}\ \ -3\ \ 3\ \ 1\ \ \mathrm{SEQ}$$

then we get the list of solutions with n_1 from -3 to 3:

$$\left\{ X = \frac{-79\ \pi}{12}\quad X = \frac{-55\ \pi}{12}\quad X = \frac{-31\ \pi}{12}\quad X = \frac{-7\ \pi}{12}\quad X = \frac{17\ \pi}{12}\quad X = \frac{41\ \pi}{12}\quad X = \frac{65\ \pi}{12} \right\}$$

SUBSTituting these solutions for X in the original equation and EXPANDing the resulting equations always returns $1 = 1$, which shows that the found solutions are OK.

But what about the second solution that TRISOL found? Let‘s make a sequence again. Entering

$$X = \frac{(24\, n_1 - 7)\, \pi}{12} \quad -3\ \ 3\ \ 1\ \ \mathrm{SEQ}$$

returns:

$$\left\{ X = \frac{-59\ \pi}{60} \quad X = \frac{-7\ \pi}{12} \quad X = \frac{-11\ \pi}{60} \quad X = \frac{13\ \pi}{60} \quad X = \frac{37\ \pi}{60} \quad X = \frac{61\ \pi}{60} \quad X = \frac{17\ \pi}{12} \right\}$$

First of all we see here that there are some solutions in this set, which are also in the first. But most of them are new. If you SUBSTitute these solutions in the original equation and EXPAND then you get a list of equations:

$$\left\{ -\mathrm{SIN}\left(\frac{27\ \mathrm{Pi}}{10}\right) = -\mathrm{SIN}\left(\frac{23\ \mathrm{Pi}}{2}\right) \quad 1=1 \quad -\mathrm{SIN}\left(\frac{3\ \mathrm{Pi}}{10}\right) = \mathrm{SIN}\left(\frac{7\ \mathrm{Pi}}{2}\right) \quad \cdots \quad 1=1 \right\}$$

Most of them seem to be wrong, but if you apply $\boxed{\to \mathrm{NUM}}$ on the left and right hand side you see that they are correct solutions. You can also apply $\boxed{\to \mathrm{NUM}}$ to the whole equation at once. A result of 0 (or about 0) shows that the equation holds.

So TRISOL gave us solutions that the built-in SOLVE didn‘t find! What does this tell us? Even when a set of solutions is found, never be sure that there are no more than those that you see on the screen!

Let‘s now again follow what happens to the equation when TRISOL starts crunching on it. First it gets COLLECTed to:

$$\mathrm{SIN}\left(3\ X + \frac{\pi}{4}\right) - \mathrm{SIN}\left(2\ X - \frac{\pi}{3}\right).$$

Then the list

$$\left\{ \mathrm{SIN}\left(3\ X + \frac{\pi}{4}\right) - \mathrm{SIN}\left(2\ X - \frac{\pi}{3}\right) = 0 \right\}$$

is made, and the contained equation is compared to certain patterns, in order to MATCH it with some standard form, that will be used later to find if the equation belongs to a special group. This step leaves the equation untouched. The check for special groups find out that this equation belongs to the group $\mathrm{SIN}(A) - \mathrm{SIN}(B) = 0$ and so transform the equation to the list of two equations:

$$\left\{ 3\ X + \frac{\pi}{4} = 2\ \pi\ n_1 + 2\ X - \frac{\pi}{3} \quad 3\ X + \frac{\pi}{4} = (2\ n_1 + 1)\ \pi - \left(2\ X - \frac{\pi}{3}\right) \right\}$$

This list is then passed to SOLVE which finds the two solutions.

Is there any other way to solve this equation? Well, yes theoretically. We can use TEXPAND, to get trigonometric functions that all have X as arguments. We then have an expression containing sines and cosines of X. We can turn $\mathrm{COS}(X)$ and $\mathrm{SIN}(X)$ to TAN of $\frac{X}{2}$ using HALFTAN. We can the COLLECT to cancel some terms and factors. So we have an equation which only contains $\mathrm{TAN}\left(\frac{X}{2}\right)$ as trigonometric function, but what an expression this is! If we use this way for the equation here we get a huge ratio of two polynomials in $\mathrm{TAN}\left(\frac{X}{2}\right)$. The numerator is factored in a product of a polynomial in

$TAN\left(\frac{X}{2}\right)$ of degree 2, and a polynomial in $TAN\left(\frac{X}{2}\right)$ of degree 4. The coefficients of the powers of $TAN\left(\frac{X}{2}\right)$ are often rather big integers.

The HP49G can solve the polynomial of degree 2 in $TAN\left(\frac{X}{2}\right)$ in this case, and returns the first solution set that TRISOL also found. But it can‘t solve the polynomial of degree 4 in $TAN\left(\frac{X}{2}\right)$ and so it doesn‘t find the second set of solutions this way. It should find this set of solutions with numeric factorisation, because the equation to solve is a polynomial, the flag for numeric factorisation was set, and the equation has no other symbolic parameters. But it doesn‘t! So for me this is reason to believe that this is the reason why SOLVE finds the first but not the second solution set. Internally it seems to be trying to do what was described above, or at least something similar.

This second equation can only be solved if you apply →NUM to the coefficients of the powers of $TAN\left(\frac{X}{2}\right)$ and the use SOLVE, which is not very understandable for me, because as already said, the flag for numeric factorisation was set, so the HP49G should *automatically* do this.

Note that the (theoretical) method TEXPAND, HALFTAN, COLLECT, SOLVE works theoretically only with integer multiples of X, that is X, $2\ X$ and so on but not $\frac{3\ X}{2}$ or $\frac{X}{3}$ as arguments of SIN, COS, TAN. This because it is only then when TEXPAND returns trigonometric functions only of X and nothing else.

---

3) Solve $COS\left(7\ X+\frac{\pi}{9}\right)=COS\left(6\ X-\frac{4\ \pi}{45}\right)$

TRISOL returns

$$\left\{X=\frac{-(10\ \pi n_1+\pi)}{5}\quad X=\frac{90\ \pi n_1-\pi}{585}\right\}$$

in 21.1 seconds. Essentially it does the same as in example 2: It COLLECTs, it transforms to a standard form, it checks for special groups, it finds the group $COS(A)-COS(B)=0$ and it builds up the equation list for SOLVE.

If you try to solve this with SOLVE, the it takes 438.7 seconds and again returns only one set of solutions, namely the set:

$$X=\frac{(10\ n_1-1)\ \pi}{5}$$

This set is the same like the first that TRISOL returns though it looks different. To see this, you can make a sequence of both sets for n_1, say from -3 to 3 and reverse one of them.

---

4) Solve $SIN(X)=COS\left(2\ X-\frac{\pi}{4}\right)$

TRISOL returns:

$$\left\{X=\frac{8\ \pi n_1+3\ \pi}{4}\quad X=\frac{8\ \pi n_1+3\ \pi}{12}\right\}$$

in 17.9 seconds. The equation is first COLLECTed, then its factors are found and returned as equations of the form factor = 0 in a list.

Here there is of course only one factor, and so the list is:

$$\left\{ \mathrm{SIN}(X)-\mathrm{COS}\left(2\ X-\frac{\pi}{4}\right)=0 \right\}$$

The equation is then found to be of the general form $\mathrm{SIN}(A)-\mathrm{COS}(B)=0$ and it is MATCHed to:

$$\mathrm{COS}\left(X-\frac{\pi}{2}\right)-\mathrm{COS}\left(2\ x-\frac{\pi}{4}\right)=0$$

Then the new equation is found to be of the form $\mathrm{COS}(A)-\mathrm{COS}(B)=0$ and is MATCHed to the equation list:

$$\left\{ X-\frac{\pi}{2}=2\ \pi n_1+2\ X-\frac{\pi}{4} \quad X-\frac{\pi}{2}=2\ \pi n_1-\left(2\ X-\frac{\pi}{4}\right) \right\}$$

This equation list is passed to SOLVE.

Solving the original equation with SOLVE returns an empty list in 36.2 seconds, which means that you very quickly get the result, that there are no solutions! But the solutions that TRISOL finds are valid, you can try them out!

Why does SOLVE fail that way? Try TEXPAND, HALFTAN, COLLECT on the equation:

$$\mathrm{SIN}(X)-\mathrm{COS}\left(2\ X-\frac{\pi}{4}\right)=0$$

You get a ratio who‘s numerator is a polynomial in $\mathrm{TAN}\left(\frac{X}{2}\right)$ of degree 4. Again the same problem as in 2 and 3, only that this time there is no polynomial of degree up to 2, which can be solved at any case. So no solution is found.

You could also try to solve this another way: Apply TEXPAND and then COLLECT to the whole equation. You get a factored form with numerator:

$$\left(\sqrt{2}\ \mathrm{COS}(X)-1\right)\left(\sqrt{2}\ \mathrm{SIN}(X)+\sqrt{2}\ \mathrm{COS}(X)+1\right)\sqrt{2}$$

The first factor $\sqrt{2}\ \mathrm{COS}(X)-1$ should be easy to solve with SOLVE and the second $\sqrt{2}\ \mathrm{SIN}(X)+\sqrt{2}\ \mathrm{COS}(X)+1$ also doesn‘t seem to be very difficult. But SOLVE still returns an empty list! Only when you manually take the factors apart and SOLVE separately $\sqrt{2}\ \mathrm{COS}(X)-1$ and $\sqrt{2}\ \mathrm{SIN}(X)+\sqrt{2}\ \mathrm{COS}(X)+1$ for X, you get solutions. Now, why the HP49G can‘t solve the product when each of the factors is an equation that it can solve? This is a question that I unfortunately can‘t answer up to now. Perhaps it has to do with the fact that the CAS doesn't use TEXPAND in this case?

Even stranger: when you start in approximate mode, SOLVE returns the solutions in 18.6 seconds. But if the factors can be solved in exact mode then this should not be necessary. And if automatic switch to approximate mode is enabled, then at least the numeric solutions should be found.

---

5) Let‘s try the factors of 4 as equations for themselves. Solve $\sqrt{2}\ \mathrm{COS}(X)-1$

TRISOL returns the solutions in 22 seconds while SOLVE returns the same solutions in 8.4 seconds. In this case again, the overhead of TRISOL makes the difference in time.

---

6) Solve $\sqrt{2}\ \mathrm{SIN}(X)+\sqrt{2}\ \mathrm{COS}(X)+1$ (second factor of 4)

TRISOL returns the solutions in 63.2 seconds. SOLVE returns them in 24.4 seconds. The solutions look different but they are mathematically equal. TRISOL finds out that the equation belongs to the group $f(\sin(x),\cos(x),\tan(x))=0$ and so uses HALFTAN to build an equation $g\left(\mathrm{TAN}\left(\frac{X}{2}\right)\right)=0$. The function $g\left(\mathrm{TAN}\left(\frac{X}{2}\right)\right)$ is a polynomial of degree 2 in $\mathrm{TAN}\left(\frac{X}{2}\right)$ and so can be easily solved by the HP49G.

In both examples 5 and 6 TRISOL needs about 2.6 times longer that SOLVE needs to solve this.

7) Solve $\mathrm{TAN}\left(\frac{X}{2}+\frac{\pi}{3}\right)=\dfrac{1}{\mathrm{TAN}\left(\frac{X}{2}\right)}$

TRISOL returns

$$\left\{X=2\ n_1\ \pi-2\ \mathrm{ATAN}\left(\sqrt{3}-2\right)\quad X=2\ n_1\ \pi-2\ \mathrm{ATAN}\left(\sqrt{3}+2\right)\right\}$$

in 39.1 seconds. SOLVE returns exactly the same solutions in 17.9 seconds, that is TRISOL takes about 2.2 times longer.

TRISOL COLLECTs the equation and keeps the numerator $\mathrm{TAN}\left(\frac{X}{2}+\frac{\pi}{3}\right)\ \tan\left(\frac{x}{2}\right)-1$ which it passes to SOLVE.

8) Solve $\mathrm{SIN}\left(X^2-3\ x+1\right)=\mathrm{SIN}(4\ X-2)$

TRISOL returns the solutions:

$$X=\frac{7+8\ \sqrt{8\ \pi n_1+37}}{2}$$

$$X=-\frac{-7+8\ \sqrt{8\ \pi n_1+37}}{2}$$

$$X=\frac{-1+8\ \sqrt{8\ \pi n_1+4\ \pi+5}}{2}$$

$$X=-\frac{-1+8\ \sqrt{8\ \pi n_1+4\ \pi+5}}{2}$$

in about 23.2 seconds. Again it finds that the equation is of the special type $\mathrm{SIN}(A)-\mathrm{SIN}(B)=0$ and builds up a list of two equations. Because the argument $X^2-3\ X+1$ of SIN at the left hand side is a quadratic in X, each of the equations of this list gives two solutions for a total of 4 solutions.

SOLVE seems to gasp a lot, if you feed it with this equation. It works and works and works, and after 10 minutes (!) it errors „Not reducible to a rational expression“. If you use TEXPAND you can see that the resulting equation also contains trigonometric terms like $\mathrm{COS}\left(X^2\right)$ which take the possibility away to build up an equation of the form $f\left(\mathrm{TAN}\left(\frac{X}{2}\right)\right)=0$.

9) Solve $\mathrm{SIN}(X+a)=\mathrm{COS}(3\ X+b)$ for X.
Taken from examinations of the year 1934 at the Pilot School (School of Icarus) in Greece.

TRISOL returns in 19.6 seconds

$$\left\{ X=\frac{4\ \pi n_1-(\pi-(2\ a-2\ b))}{4} \quad X=\frac{4\ \pi n_1+\pi-(2\ a-2\ b)}{8} \right\}$$

It works like in example 4.

SOLVE returns in 134.7 seconds a more complicated form of the solutions. (Actually much too complicated for my gusto, but it is correct.)

Now think how the poor guys there at the School of Icarus have solved this equation, without an HP in their hands. I think not the way the HP49G solves it. ;-)

---

10) Solve $\mathrm{TAN}\left(\frac{X+a}{X-a}\right)=\mathrm{TAN}\left(\frac{X+b}{X-b}\right)$ for X.

TRISOL returns in 54.6 seconds

$$X=\frac{\left((\pi\ a+\pi\ b)\ n_1+2\ a-2\ b-\sqrt{(\pi^2\ a^2-2\ \pi^2\ b\ a+\pi^2\ b^2)\ n_1^{\,2}+(4\ \pi a^2-4\ \pi b^2)\ n_1+4\ a^2-8\ b\ a+4\ b^2}\right)}{2\ \pi n_1}$$

and

$$X=\frac{\left((\pi\ a+\pi\ b)\ n_1+2\ a-2\ b+\sqrt{(\pi^2\ a^2-2\ \pi^2\ b\ a+\pi^2\ b^2)\ n_1^{\,2}+(4\ \pi a^2-4\ \pi b^2)\ n_1+4\ a^2-8\ b\ a+4\ b^2}\right)}{2\ \pi n_1}$$

It finds that the equation belongs to the group TAN(A)= TAN(B) and builds up the equation

$$\frac{X+a}{X-a}=\pi\ n_1+\frac{X+b}{X-b}$$

It passes then this equation to SOLVE.

I didn‘t have the patience to let SOLVE finish this calculation because after 20 minutes it was still trying to find a solution. So perhaps it does find it, perhaps it doesn‘t. But even if it finds a solution, it is not a good idea to use it for this case.

If you try to TEXPAND and COLLECT this equation, then you see that there are TAN functions with many different arguments, like $\frac{X}{X-A}$, $\frac{A}{X-A}$ and so on, so HALFTAN wouldn‘t create a polynomial in $\mathrm{TAN}\left(\frac{arg}{2}\right)$ where arg is always the same.

---

11) Solve $\cos\left(7\cdot X+\frac{\pi}{7}\right)^2=\cos\left(2\cdot\pi+\frac{\pi}{3}\right)$

TRISOL returns in 38.5 seconds

$$\left\{X=\frac{42\cdot\pi\cdot n_1+46\cdot\pi}{147}\quad X=\frac{42\cdot\pi\cdot n_1-52\cdot\pi}{147}\quad X=\frac{42\cdot\pi\cdot n_1-31\cdot\pi}{147}\quad X=\frac{42\cdot\pi\cdot n_1+25\cdot\pi}{147}\right\}$$

The equation is of the form $\cos(A)^2=\cos(B)^2$. TRISOL first COLLECTS it to:

$$\left(\cos\left(7\cdot X+\frac{\pi}{7}\right)-\cos\left(2\cdot\pi+\frac{\pi}{3}\right)\right)\cdot\left(\cos\left(7\cdot X+\frac{\pi}{7}\right)+\cos\left(2\cdot\pi+\frac{\pi}{3}\right)\right)$$

Then it builds the equation list:

$$\left\{\cos\left(7\cdot X+\frac{\pi}{7}\right)-\cos\left(2\cdot\pi+\frac{\pi}{3}\right)=0\quad \cos\left(7\cdot X+\frac{\pi}{7}\right)+\cos\left(2\cdot\pi+\frac{\pi}{3}\right)=0\right\}$$

The first equation in this list belongs to the special group $\cos(A)-\cos(B)=0$. So TRISOL builds up the equation list

$$\left\{7\cdot X+\frac{\pi}{7}=2\cdot\pi\cdot n_1+2\cdot\pi+\frac{\pi}{3}\quad 7\cdot X+\frac{\pi}{7}=2\cdot\pi\cdot n_1-\left(2\cdot\pi+\frac{\pi}{3}\right)\right\}$$

and passes these equations to SOLVE. The second equation in the list is first MATCHed to

$$\cos\left(7\cdot X+\frac{\pi}{7}\right)-\cos\left(2\cdot\pi+\frac{\pi}{3}-\pi\right)=0$$

then recognised as one of the form $\cos(A)-\cos(B)=0$ and then the equation list is built up:

$$\left\{7\cdot X+\frac{\pi}{7}=2\cdot\pi\cdot n_1+2\cdot\pi+\frac{\pi}{3}-\pi\quad 7\cdot X+\frac{\pi}{7}=2\cdot\pi\cdot n_1-\left(2\cdot\pi+\frac{\pi}{3}-\pi\right)\right\}$$

which SOLVE solves afterwards.

SOLVE needs only 23.8 seconds and returns solutions involving $\operatorname{ACOS}\left(\cos\left(2\cdot\pi+\frac{\pi}{3}\right)\right)$ which EXPANDed is $\frac{2\cdot\pi}{3}$. You can prove that the solutions of TRISOL and SOLVE are both mathematically correct.

---

12) Solve $4\cdot\sin(X)^2-3\cdot\sin(X)-1=0$

TRISOL first factors to $(\sin(X)-1)\cdot(4\cdot\sin(X)+1)$ and then solves the two equations $\sin(X)-1=0$ and $4\cdot\sin(X)+1=0$. It returns the solutions in 34 seconds.

SOLVE only needs 13 seconds to return the same solutions.

---

13) Solve $4\cdot\cos(X)^2-2\cdot\left(\sqrt{2}+1\right)\cdot\cos(X)+\sqrt{2}=0$

TRISOL needs 45.6 seconds to return the solutions while SOLVE needs only 14.4 seconds for the same solutions.

---

14) Solve $4\cdot\mathrm{COS}(X)^2 - 2\cdot\left(\sqrt{2}+1\right)\cdot\mathrm{SIN}(X)+\sqrt{2}=0$

TRISOL returns a list with 4 numeric solutions after 88.8 seconds. It first tries to COLLECT which doesn't do anything here, and then recognises the equation as one of the special group $f(\sin(x),\cos(x),\tan(x))=0$. It uses HALFTAN to turn it to $f\left(\tan\left(\frac{x}{2}\right)\right)=0$, with $f\left(\tan\left(\frac{x}{2}\right)\right)$ a ratio of polynomials in $\tan\left(\frac{x}{2}\right)$. The degree of the numerator is 4. This polynomial ratio is passed to FACTOR which factors it by switching to numeric mode. Then the equations list with equations of the form $\mathrm{factor}=0$ is built up. The substitution $\frac{X}{2}=Y$ is made for each of these equations and the list is given to SOLVE. When the solutions are returned, the back substitution is made and the equations are solved for X. The solutions can be converted to symbolic solutions by applying →NUM to the resulting sub expressions $2\cdot\mathrm{ATAN}(\mathrm{arg})$ of the solutions, and then XQ to the whole solution.

SOLVE returns an empty list in 31.7 seconds if you start with exact mode. It returns the numeric solutions in 26.8 seconds if you start at approximate mode. So the flag for automatic switch to approx. mode, doesn't seem to help much here.

---

15) Solve $3\cdot\mathrm{TAN}(X)^2 - 4\cdot\sqrt{3}\cdot\mathrm{TAN}(X)+3=0$

TRISOL returns the solutions in 38.7 seconds. It first COLLECTs the equation and makes the equation list:

$$\left\{3\cdot\mathrm{TAN}(X)-\sqrt{3}=0 \quad \mathrm{TAN}(X)-\sqrt{3}=0\right\}$$

It then passes this to SOLVE.

SOLVE returns more complicated looking solutions in 12.7 seconds. The solutions contain ATAN functions with many square roots, but applying EXPAND to the solutions makes them like the solutions that TRISOL returns.

---

16) Solve $2\cdot\mathrm{SIN}(X)=3\cdot\mathrm{TAN}(X)$

TRISOL solves this in 50.7 seconds. It first uses HALFTAN and FACTOR and then passes the resulting factored equation

$$2\cdot\mathrm{TAN}\left(\frac{X}{2}\right)\cdot\left(5\cdot\mathrm{TAN}\left(\frac{X}{2}\right)+i\cdot\sqrt{5}\right)\cdot\left(5\cdot\mathrm{TAN}\left(\frac{X}{2}\right)+i\cdot\sqrt{5}\right)=0$$

to SOLVE. Note that two of the three solutions are complex.

SOLVE returns the same solutions in 21.1 seconds.

---

17) Solve $a\cdot\mathrm{SIN}(X)=b\cdot\mathrm{TAN}(X)$

TRISOL needs 67.8 seconds to find the solutions. It works here like in 16. SOLVE errors out with „Not reducible to a rational expression“ after about 20 seconds.

---

18) Solve $a+\text{TAN}(X)=b+\dfrac{1}{\text{TAN}(X)}$

TRISOL returns the solutions in 37.5 seconds. It COLLECTs and builds the equation $\text{TAN}(X)^2+(a+b)\ \text{TAN}(X)-1=0$ which is then passed to SOLVE.

SOLVE needs 16 seconds to solve the equation.

---

19) Solve $\text{SIN}(X)\ \text{COS}(X)+\text{SIN}(X)-\text{COS}(X)-1=0$

TRISOL returns

$$\left\{X=-\frac{4\ \pi n_1-\pi}{2} \quad X=\frac{4\ \pi n_1+\pi}{2}\right\}$$

in 33.5 seconds. It first COLLECTs the equation and builds up the list

$$\{\text{COS}(X)+1=0 \quad \text{SIN}(X)-1=0\}$$

then passes the list to SOLVE.

SOLVE returns

$$X=\frac{4\ \pi\ n_1+\pi}{2}$$

in 11.6 seconds. It looks like it has lost some solutions but it hasn‘t. Make a sequence of solutions for both results to prove that.

---

20) Solve $\text{COS}(X)=2\ \text{SIN}\left(\dfrac{X}{2}\right)$

Both TRISOL and SOLVE error „Not reducible to a rational expression“ But this is not true. If we apply HALFTAN to COS(X) twice, and to $\text{SIN}\left(\dfrac{X}{2}\right)$ once, then we have a rational expression, a ratio of polynomials in $\text{TAN}\left(\dfrac{X}{4}\right)$. This should be solvable, at least with numeric factoring.

If you feed TRISOL with this polynomial in $\text{TAN}\left(\dfrac{X}{4}\right)$, then it returns the numeric solutions in 59.4 seconds. SOLVE needs 21.1 seconds.

The question here is, how to add code to TRISOL to handle such cases. It should first find all trigonometric functions, check if all arguments for these functions are of the form $\dfrac{\text{numerator}}{n\ 2^m}$ where numerator and n are the same for all arguments, and then apply HALFTAN to each trigonometric function the appropriate number of times, so that an equation of the form $f\left(\text{TAN}\left(\dfrac{\text{numerator}}{k}\right)\right)=0$ appears, where numerator and k are the same for all arguments of TAN. It is possible to do that, but I don‘t know if it is also reasonable. (Well, I must confess that I would like to do that, if only for the fun of it. :-))

---

21) Solve $1+\mathrm{COS}(2\cdot X)=6\cdot\mathrm{SIN}(X)^2$

TRISOL returns the solutions in 43.3 seconds. SOLVE needs 28.5 seconds. TRISOL doesn't find this equation to belong to any of the special groups, so it simply passes it to SOLVE hoping for the best. So the time difference is only for the overhead of checking for such special cases.

---

22) Solve $2\cdot\mathrm{SIN}(X)^2+\mathrm{SIN}(2\cdot X)^2=3$

TRISOL needs 56.5 seconds and works here like in 21. SOLVE needs for the same solutions 39.5 seconds.

---

23) Solve $\left(\sqrt{2}+1\right)\cdot\mathrm{SIN}(X)^2+\left(\sqrt{2}-1\right)\cdot\mathrm{COS}(X)^2+\mathrm{SIN}(2\cdot X)=\sqrt{2}$

TRISOL needs 63.2 seconds to find the solutions. SOLVE finds the same solutions in 31.1 seconds. TRISOL works here like in 21.

---

24) Solve $\mathrm{COS}(X)=\dfrac{2\cdot\mathrm{TAN}(X)}{1+\mathrm{TAN}(X)^2}$

Taken from exams at the greek military school.

TRISOL finds the equation to belong to the special group $f(\sin(x),\cos(x),\tan(x))=0$. It uses HALFTAN EXPAND FACTOR to build a polynomial in $\mathrm{TAN}\left(\frac{X}{2}\right)$ and then passes the polynomial to SOLVE. It needs 70.9 seconds to complete.

SOLVE finds the solutions in 25.5 seconds.

Applying HALFTAN and FACTOR to the original equation, we get:

$$\frac{-\left(\left(\mathrm{TAN}\left(\frac{X}{2}\right)+1\right)\cdot\left(\mathrm{TAN}\left(\frac{X}{2}\right)-1\right)\cdot\left(\mathrm{TAN}\left(\frac{X}{2}\right)-\left(2-\sqrt{3}\right)\right)\cdot\left(\mathrm{TAN}\left(\frac{X}{2}\right)-\left(2+\sqrt{3}\right)\right)\right)}{\left(\mathrm{TAN}\left(\frac{X}{2}\right)^2+1\right)^2}$$

Some of the solutions of this equation are a bit hard to understand. If we simply put the solutions of the form

$$X=\frac{4\cdot n_1\cdot\pi-\pi}{2}$$

and

$$X=\frac{4\cdot n_1\cdot\pi-\pi}{2}$$

for some integer values of n_1 back to the original equation, then the right hand side becomes $\frac{+\infty}{\infty}$, so we must calculate the limit of the right hand side for X approaching $\frac{4\cdot n_1\cdot\pi-\pi}{2}$ or $\frac{4\cdot n_1\cdot\pi-\pi}{2}$. The result is 0 for any integer value of n_1, like the result for the left hand side, so the solutions are correct.

---

25) Solve $\dfrac{3\ \mathrm{TAN}(X)+\dfrac{2}{\mathrm{TAN}(X)}}{3\ \mathrm{TAN}(X)+\dfrac{5}{\mathrm{TAN}(X)}}=\dfrac{2}{3}$

TRISOL needs 24.8 seconds to find the solutions while SOLVE needs 11.9 seconds. TRISOL COLLECTs, drops the denominator and passes $3\ \mathrm{TAN}(X)^2-4=0$ to SOLVE.

---

26) Solve $\mathrm{SIN}(X)+\mathrm{SIN}(3\ X)=2\ \mathrm{SIN}(2\ X)$

TRISOL needs 45 seconds to find the 5 solutions which SOLVE returns in 27.6 seconds. TRISOL simply passes the equation to SOLVE in this case.

---

27) Solve $\mathrm{SIN}(X)+\mathrm{SIN}(2\ X)+\mathrm{SIN}(3\ X)+\mathrm{SIN}(4\ X)=0$

TRISOL works like in 26 and finds 7 solutions in 84.8 seconds. SOLVE finds the same solutions in 65.2 seconds.

---

28) Solve $\mathrm{COS}(X)-\mathrm{COS}(2\ X)+\mathrm{SIN}(3\ X)=0$

Both find the same solutions. TRISOL in 59.8 seconds and SOLVE in 39.1 seconds. TRISOL does here the same like in 26.

---

29) Solve $2\ \mathrm{SIN}(X)^2+2\ \mathrm{SIN}(X)\ \mathrm{COS}(X)-1=0$

TRISOL finds that the equation belongs to the special group $a\ \sin^2(x)+b\ \sin(x)\ \cos(x)+c\ \cos^2(x)+d=0$. It transforms this equation to:

$$2\ \mathrm{SIN}(X)^2+2\ \mathrm{SIN}(X)\ \mathrm{COS}(X)-1\left(\mathrm{SIN}(X)^2+\mathrm{COS}(X)^2\right)=0 \Leftrightarrow$$

$$\mathrm{SIN}(X)^2+2\ \mathrm{SIN}(X)\ \mathrm{COS}(X)-\mathrm{COS}(X)^2=0$$

then factors it and builds up the equations list

$$\left\{\mathrm{SIN}(X)+\left(1+\sqrt{2}\right)\ \mathrm{COS}(X)=0\quad \mathrm{SIN}(X)-\left(-1+\sqrt{2}\right)\ \mathrm{COS}(X)=0\right\}$$

to SOLVE. It takes 39.9 seconds to find the solutions.

SOLVE needs 21.1 seconds but finds the same solutions in numerical form.

---

30) Solve $2\ \mathrm{SIN}(X)^2+2\ \mathrm{SIN}(X)\ \mathrm{COS}(X)+a\ \mathrm{COS}(X)^2-b=0$

TRISOL works like in 29. It returns the solutions in 77.3 seconds.

SOLVE on the other hand returns an empty list in 24.9 seconds.

---

31) Solve $2\ \mathrm{SIN}(X)^2+4\ \mathrm{SIN}(X)\ \mathrm{COS}(X)+5\ \mathrm{COS}(X)_2=3$

TRISOL finds the solutions in 46.5 seconds while SOLVE needs 29.4 seconds but returns numerical solutions. TRISOL works here like in 29.

32) Solve $5\ \mathrm{SIN}(X)^2 - 3\ \mathrm{SIN}(X)\ \mathrm{COS}(X) - 2\ \mathrm{COS}(X)^2 = 0$

TRISOL returns the solutions in 80.5 seconds. It factors the equation to:

$$(\mathrm{SIN}(X) - \mathrm{COS}(X))\ (5\ \mathrm{SIN}(X) + 2\ \mathrm{COS}(X)) = 0$$

It then MATCHes $\mathrm{SIN}(X) - \mathrm{COS}(X) = 0$ to:

$$\mathrm{COS}\left(X - \frac{\pi}{2}\right) - \mathrm{COS}(X) = 0$$

and then finds that this belongs to the special group $\mathrm{COS}(a) - \mathrm{COS}(b) = 0$. The second equation $5\ \mathrm{SIN}(X) + 2\ \mathrm{COS}(X) = 0$ is found to be of the form $f(\sin(x),\cos(x),\tan(x)) = 0$ and so HALFTAN EXPAND is used to find a polynomial in $\mathrm{TAN}\left(\frac{X}{2}\right)$ which is then passed to SOLVE.

SOLVE finds the solutions in 28.5 seconds.

---

33) Solve $\mathrm{SIN}(X) + 2\ \mathrm{COS}(X) = \frac{1}{\mathrm{COS}(X)}$

TRISOL needs 43.5 seconds for this. SOLVE does it in 25.4 seconds but returns numeric results. TRISOL works here like in 32.

---

34) Solve $a\ \mathrm{SIN}(X) + 2\ \mathrm{COS}(X) = \frac{1}{\mathrm{COS}(X)}$

TRISOL finds the solutions in 51.8 seconds and works here like in 32. SOLVE returns an empty list in 13.8 seconds.

---

35) Solve $2\ \mathrm{COS}(X) + \mathrm{SIN}(3\ X) = 1$

TRISOL simply passes this to SOLVE. It returns the solutions in 65.1 seconds.

SOLVE needs for the same solutions only 47.3 seconds.

Both return numeric solutions.

---

36)Solve $\mathrm{SIN}(3\ X) = 8\ \mathrm{SIN}(X)^3$

TRISOL works like in 35 and returns the solutions in 52 seconds.

SOLVE needs only 35.7 seconds.

---

37) Solve $\frac{\mathrm{COS}(X)}{\mathrm{COS}(a - X)} = m$

TRISOL works like in 35 and returns the solutions in 72.5 seconds.

SOLVE needs only 55 seconds.

---

38) Solve $\mathrm{SIN}(\pi\ \mathrm{COS}(X)) = \mathrm{COS}(\pi\ \mathrm{SIN}(X))$

Taken from exams at the greek Polytechnics. What weird exams are those in Greece, I'm telling you! ;-)

TRISOL needs 72.4 seconds to return the solutions. It finds the equation to be of the form $\sin(a)-\cos(b)=0$, MATCHes it to

$$\mathrm{COS}\left(\pi\ \mathrm{COS}(X)-\frac{\pi}{2}\right)-\mathrm{COS}(\pi\ \mathrm{SIN}(X))=0$$

and then finds that it belongs to the group $\cos(a)-\cos(b)=0$. It builds up the equation list

$$\left\{\pi\ \mathrm{COS}(X)-\frac{\pi}{2}=2\ \pi n_1+\pi\ \mathrm{SIN}(X)\quad \pi\ \mathrm{COS}(X)-\frac{\pi}{2}=2\ \pi n_1-\pi\ \mathrm{SIN}(X)\right\}$$

and passes this list to SOLVE.

SOLVE returns, well I don't know because I interrupted it after about 2 minutes.

---

39) Solve $\dfrac{\mathrm{TAN}(X+a)}{\mathrm{TAN}(X-a)}=m$

TRISOL does it in 58.3 seconds while SOLVE needs only 38.7 seconds.

---

40) Solve $\dfrac{\mathrm{TAN}\left(\frac{\pi}{3}-X\right)}{\mathrm{COS}(X)^2}=\dfrac{\mathrm{TAN}(X)}{\mathrm{COS}\left(\frac{\pi}{3}-x\right)^2}$

TRISOL needs 120.1 seconds. SOLVE needs only 94.7 seconds.

---

41) Solve $\mathrm{TAN}\left(\dfrac{1}{\mathrm{TAN}(X)}\right)=\dfrac{1}{\mathrm{TAN}(\mathrm{TAN}(X))}$

Both error with „Not reducible to a rational expression" But the equation can be solved. Use TAN2SC, EXPAND, COLLECT. Take the resulting expression to the EQW, select the numerator and TCOLLECT. The numerator goes to

$$-1\ \cos\left(\frac{\mathrm{SIN}(X)^2+\mathrm{COS}(X)^2}{\mathrm{COS}(X)\ \mathrm{SIN}(X)}\right)$$

Select the sub-expression $\mathrm{SIN}(X)^2+\mathrm{COS}(X)^2$ and press TRIG to convert it to a 1. Now the whole expression is:

$$\frac{-1\ \mathrm{COS}\left(\dfrac{1}{\mathrm{COS}(X)\ \mathrm{SIN}(X)}\right)}{\mathrm{COS}(\mathrm{COS}(X)\ \mathrm{SIN}(X))\ \mathrm{SIN}(\mathrm{COS}(X)\ \mathrm{SIN}(X))}=0$$

Press ENTER to put this expression to the stack and SOLVE for X. You get the results 180.5 seconds.

If you feed TRISOL with the above equation you get an empty list after 56.3 seconds and this is very very surprising if you think about what

TRISOL does in this case. It COLLECTs and throws away the denominator, so that

$$-1\ \mathrm{COS}\left(\frac{1}{\mathrm{COS}(X)\ \mathrm{SIN}(X)}\right)=0$$

remains as the equation to solve. Then it checks for special groups, and finds that this equation doesn‘t belong to any of these groups. So it passes the remaining equation to SOLVE.

Now SOLVE takes over and *can‘t solve*

$$-1\ \mathrm{COS}\left(\frac{1}{\mathrm{COS}(X)\ \mathrm{SIN}(X)}\right)=0$$

though it *can solve*

$$\frac{-1\ \mathrm{COS}\left(\frac{1}{\mathrm{COS}(X)\ \mathrm{SIN}(X)}\right)}{\mathrm{COS}\left(\mathrm{COS}(X)\ \mathrm{SIN}(X)\right)\ \mathrm{SIN}\left(\mathrm{COS}(X)\ \mathrm{SIN}(X)\right)}=0$$

that is the same equation with a denominator! Why? Dunno, but it is kind of amusing.

So if you had to solve the equation

$$-1\ \mathrm{COS}\left(\frac{1}{\mathrm{COS}(X)\ \mathrm{SIN}(X)}\right)=0$$

you should first MATCH the expression

$$\frac{1}{\mathrm{COS}(X)\ \mathrm{SIN}(X)}$$

to Y, solve for Y, then substitute

$$Y=\frac{1}{\mathrm{COS}(X)\ \mathrm{SIN}(X)}$$

back to the solutions and solve again for X.

---

42) Solve

$$\mathrm{SIN}(|X|)+\mathrm{COS}(|X|)+\mathrm{TAN}(|X|)+\frac{1}{\mathrm{SIN}(|X|)}+\frac{1}{\mathrm{COS}(|X|)}+\frac{1}{\mathrm{TAN}(|X|)}+3=0$$

Taken from the exams at the Greek Polytechnics 1947. I told you, the exams are really weird there. ;-)

TRISOL needs 312.6 seconds and SOLVE needs 109.7 seconds. (Complex mode, X is assumed to be real.)

We have 42 examples, and if I remember well this number has to do something with the question about the universe, us and everything else. So I think I better stop here. We don‘t want to know more than this universe tells us, do we?

Only a small word about TRISOL. It isn‘t meant to replace SOLVE. It is only a try, a very imperfect try, to automate what you do when you don‘t get an answer with SOLVE right away. I hope you enjoyed the bugs, the corrections, the ideas behind it. And I hope that you change it and tailor it to best fit your needs. What I find big fun, when trying to do such things like solving similar equations, is that I can‘t always explain to myself how I do it, in order to sit down and write a program that does the same. I mean, look at

$$\mathrm{SIN}(X)+\mathrm{COS}\left(\frac{X}{4}\right)-\mathrm{TAN}\left(\frac{X}{2}\right)=0$$

You can see immediately that you must apply HALFTAN once to $\cos\left(\frac{X}{4}\right)$, twice to $\mathrm{TAN}\left(\frac{X}{2}\right)$, and 3 times to $\mathrm{SIN}(X)$ in order to turn this equation to a ratio of polynomials in $\mathrm{TAN}\left(\frac{X}{8}\right)$. Now writing a program that does the same is like transferring a thought to the calc or to any other programmable machine. But to do that you must first think about things that happen automatically in mind. When you write such a program that does the same as you do, it doesn‘t of course mean that your thoughts work in exactly the same way. You don‘t find logarithms of the denominators of X, $\frac{X}{2}$ and $\frac{X}{4}$, and you don‘t divide them with $\ln(2)$ in mind. *You simply see* that $\mathrm{TAN}\left(\frac{X}{2}\right)$ should be transformed twice with HALFTAN, while $\mathrm{SIN}(X)$ should be transformed 3 times. But what hides behind this „you simply see“? There must be a relation, some vague kind of similarity between this „you simply see“ and the program. A relation that can be described on the level of bits and bytes (?), or a relation of what the currents in my brain and my calc produce when they flow, which at the end is the polynomial in $\mathrm{TAN}\left(\frac{X}{8}\right)$ in this case.

Well that‘s all for today. Keep tuned and solve them all!
(TRI)SOLved greetings,
Nick.

Trigonometry with the HP49G - Part 10

Hi all!

It is the tenth and last part of the Trigonometry Marathon and we already have seen a lot of things. But there is still stuff waiting to be discovered. Would you ever think that the little HP49G is such a big place if you take a look from the inside? ;-)

Until now we stayed in the real domain. Today we will dare a small jump into the complex. (As if it weren‘t complex enough already..;-))

So get your backpack and VPN don‘t forget your swiss army knife and here we go, our trip into the complex begins.

You may already know that there are some relations between trigonometric functions and complex exponentials. Since the Complex Marathon starts right after the end of the Trigonometry Marathon, I think it is better to leave the derivation of these relations on the HP49G for the first part of the Complex Marathon. For now it is enough to show what can be done with these relations:

1) $e^{i\,x} = \cos(x) + i\ \sin(x)$
2) $e^{-i\,x} = \cos(x) - i\ \sin(x)$

First of all, the HP49G can do this. The command SINCOS takes complex exponentials and returns them as trigonometric functions. It is the first command on the second page of the TRIG menu.

We assume here that X and Y are real and also that Z is complex. Enter X ADDTOREAL then Y ADDTOREAL and then Z UNASSUME so that these assumptions are done. Also switch the HP49G to complex rigourous mode.

Now, enter $e^{i\,X+\frac{\pi}{2}}$ and press SINCOS. The HP49G returns

$$e^{\frac{\pi}{2}}\ \left(\mathrm{COS}(X) + i\ \mathrm{SIN}(X)\right)$$

Enter $e^{X+i\,Y}$, press SINCOS, and you get

$$e^{X}\ \left(\mathrm{COS}(Y) + i\ \mathrm{SIN}(Y)\right)$$

But enter $e^{i\,Z}$, press SINCOS, and the result is

$$e^{-\mathrm{IM}(Z)}\ \left(\mathrm{COS}\left(\mathrm{RE}(Z)\right) + i\ \mathrm{SIN}\left(\mathrm{RE}(Z)\right)\right)$$

Why the difference? Well, X and Y are assumed to be real, so the HP49G knows for example that the real part of X is X and the imaginary part of X is 0. But if Z is complex, and nothing else is known about it, then the HP49G writes leaves RE(Z) and IM(Z) unevaluated, to denote the real and imaginary part of Z.

If we add the relations (1) and (2) we get:

3) $\cos(x) = \dfrac{e^{i\,x}}{2} + \dfrac{e^{-i\,x}}{2}$

If we subtract (2) from (1) we get:

4) $\sin(x) = i\ \dfrac{e^{-i\,x}}{2} - i\ \dfrac{e^{i\,x}}{2}$

The command for converting trigonometric functions to complex exponentials is EXPLN. It is the first command on the menu EXP&LN.

Enter SIN(X), press EXPLN, and you get the result

$$\frac{e^{i\,X} - \dfrac{1}{e^{i\,X}}}{2\ i}$$

Though it is already readable enough on the HP49G, let‘s make it looking more familiar. Press **LIN** (third command on the menu EXP&LN) to get

$$i\,\frac{e^{-ix}}{2}-i\,\frac{e^{ix}}{2}$$

LIN tries to make real or complex exponentials linear. (EXPAND doesn‘t fit here, because it brings the two terms e^{iX} and $\frac{1}{e^{iX}}$ over a common denominator and so returns a more complex looking expression.)

Now, with

$$i\,\frac{e^{-ix}}{2}-i\,\frac{e^{ix}}{2}$$

on stack level 1 press **DUPDUP** to make two copies and then press **RE** to get the real part of the expression. Press **EXPAND** or **COLLECT** and you see SIN(X) again. This is correct, because we started with a real thing, that is SIN(X), and so even turning it to a complex exponential, it still remains real. You‘ll see how important this can be later on, in this part. Press now ▶ to bring one of the copies that you have made on stack level 1 and press **IM** to get the imaginary part of the expression. Press **EXPAND** and you see that the imaginary part is 0 as it must be. (Since we started with the real SIN(X), we expect the HP49G to return 0.)

Instead of using **RE** and **IM**, you can also use **SINCOS**. Press **ROT** to bring the second copy on stack level 1, Press **SINCOS** and then **EXPAND**. The result is again SIN(X).

If you enter SIN(Z) and press **EXPLN**, then you get

$$\frac{e^{iZ}-\frac{1}{e^{iZ}}}{2\,i}$$

Perhaps you wonder why there are no RE(Z) and IM(Z) in this case. Well, if there were such expressions, they would appear as

$$\frac{e^{i\,(RE(Z)+i\,IM(Z))}-\frac{1}{e^{i\,(RE(Z)+i\,IM(Z))}}}{2\,i}$$

that is in a form that is equivalent to Z itself, because every complex number Z is the same as $RE(Z)+i\;IM(Z)$.

Not only the trigonometric functions can be converted to complex exponentials/logarithms but also the inverse trigonometric functions. Enter for example ACOS(X) and press **EXPLN**. The result is

$$\frac{LN\left(e^{\frac{LN\left(X^2-1\right)}{2}}+X\right)}{i}$$

If you don‘t like the representation $e^{\frac{LN\left(X^2-1\right)}{2}}$, then press **EXP2POW** to convert this to

$$\frac{LN\left(X+\sqrt{X^2-1}\right)}{i}$$

There are some things that should be mentioned about this result.

The first is, that the HP49G can‘t get REal and IMaginary parts of this expression. So if we want to do that, we must do something ourselves. Though this will be covered better at the complex marathon, let it be said here, that the argument of the LN can be written as a complex number of the form $r\ e^{i\theta}$, where r is the magnitude and θ the angle of the complex number. Thus we have $\mathrm{LN}\left(r\ e^{i\theta}\right) = \mathrm{LN}(r) + i\theta$. The REal and IMaginary parts of this are easy to calculate (if we assume that θ is the angle of the principal value).

The second thing is that the HP49G doesn‘t consider assumptions about variables when it evaluates or expands $\mathrm{LN}\left(X + \sqrt{X^2 - 1}\right)$. If you make the assumption $X \geq 1$, and then try to find the REal part of this expression, the HP49G doesn‘t return the expression itself, but simply writes $\mathrm{RE}\left(\mathrm{LN}\left(X + \sqrt{X^2 - 1}\right)\right)$. The same with the IMaginary part. It doesn‘t return 0 but $\mathrm{IM}\left(\mathrm{LN}\left(X + \sqrt{X^2 - 1}\right)\right)$. If you have only $X + \sqrt{X^2 - 1}$ as argument, and have made the assumption $X \geq 1$, then the HP49G returns $X + \sqrt{X^2 - 1}$ as the REal part of the expression and 0 as the IMaginary part.

The third thing is that, if you start at complex mode with $\mathrm{ACOS}(X)$ and you press EXPLN, you get

$$\frac{\mathrm{LN}\left(X + \sqrt{X^2 - 1}\right)}{i}$$

no matter what assumptions you have made for X. But if you are at real mode, enter for example $\mathrm{ASIN}(X)$ and press EXPLN, then you get a huge expression:

$$\frac{i\ \mathrm{LN}\left(\left(e^{\frac{\mathrm{RE}(\mathrm{LN}(X^2-1))}{2}}\right)^2 \mathrm{SIN}\left(\frac{\mathrm{IM}(\mathrm{LN}(X^2-1))}{2}\right)^2 + \left(e^{\frac{\mathrm{RE}(\mathrm{LN}(X^2-1))}{2}}\right)^2 \mathrm{COS}\left(\frac{\mathrm{IM}(\mathrm{LN}(X^2-1))}{2}\right)^2 + 2\ X\ e^{\frac{\mathrm{RE}(\mathrm{LN}(X^2-1))}{2}}\ \mathrm{COS}\left(\frac{\mathrm{IM}(\mathrm{LN}(X^2-1))}{2}\right) + X^2\right)}{2} + \frac{\pi}{2}$$

If you start at real mode and you have previously assumed that for example $X \geq 1$ then these assumptions *are* taken into consideration . I think that the HP49G in complex mode considers expressions like for example $\mathrm{LN}\left(X^2 + 1\right)$ to be general complex expressions and so it doesn‘t care to show explicitly what is the real or the imaginary part. But in real mode, it explicitly shows real and imaginary parts, as well as the CAS allows and tries to return results according to the assumptions you have made. A bit more on assumptions. Lets say that you make the assumption $X \geq 1$. Then things like $\mathrm{IM}\left(\sqrt{X^2 - 1}\right)$ are correctly evaluated to 0. Does this means that the HP49G considers this assumption? Well, it does in many other cases, but not for this one, though it looks like it did. Lets assume $-1 \leq X \leq 1$ and find then the imaginary part of $\sqrt{X^2 - 1}$. How can we make the assumption $X \geq -1\ \ \mathrm{AND}\ \ X \leq 1$? Entering this expression and then using ASSUME, results in an error. But you can enter $X \geq -1$, press ASSUME, then enter $X \leq 1$ and press ASSUME. If you now take a look at the list REALASSUME, you see that $X \geq -1\ \ \mathrm{AND}\ \ X \leq 1$ is in the list. Quite hard to understand why the HP49G doesn‘t let you do it directly with $X \geq -1$ AND $X \leq 1$ ASSUME and wants you to use $X \geq -1$ ASSUME and then $X \leq 1$ ASSUME instead. But it has its reasons. If you do $X \leq 2$ ASSUME and then $X \leq 1$ ASSUME, then the HP49G only writes $X \leq 1$ in the list REALASSUME because it correctly finds out that $X \leq 2$ AND $X \leq 1$ is equivalent to $X \leq 1$!

And this though $X \leq 2$ AND $X \leq 1$ *can‘t* be simplified with EXPAND or EVAL on the stack! Could it be that this is the built-in back door for simplifying logical expressions? (And also the back door for another marathon? ;-)) Back to our imaginary part of $\sqrt{X^2-1}$. With the assumption $X \geq -1$ AND $X \leq 1$ the expression $\mathrm{IM}\left(\sqrt{X^2-1}\right)$ should be evaluated to $\sqrt{1-X^2}$ and $\mathrm{RE}\left(\sqrt{X^2-1}\right)$ should be evaluated to 0.

But it doesn‘t! If you make this assumption, enter $\sqrt{X^2-1}$ and press RE then the result is $\sqrt{X^2-1}$ and the result of $\sqrt{X^2-1}$ IM is 0, which is not correct, considering that $-1 \leq X \leq 1$. More about the influence of the many operation modes and assumptions to the calculations will be in the complex marathon.

The fourth thing is that the expression

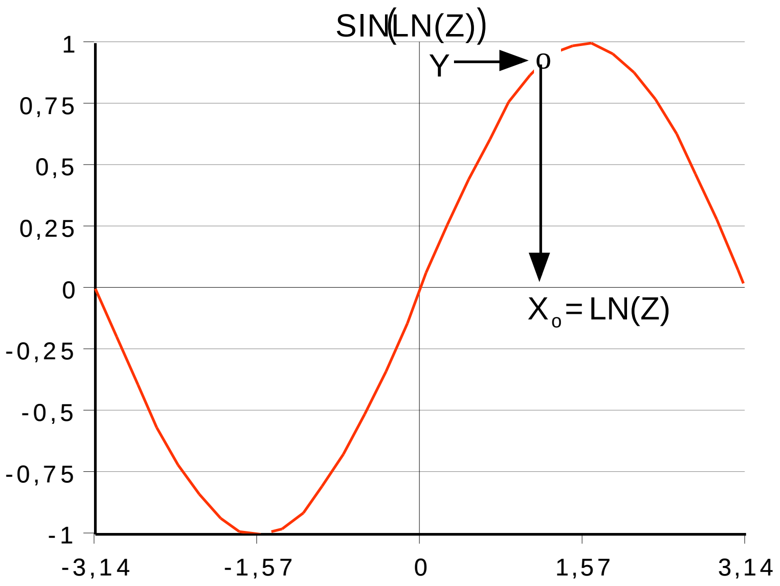
$$\frac{\mathrm{LN}\left(X+\sqrt{X^2-1}\right)}{i}$$

cannot be reconverted to $\mathrm{ACOS}(X)$ using SINCOS. So it looks like a one way ticket from inverse trigonometric functions to complex logarithms. Let‘s try to find if and how the conversion from logarithms to inverse trigonometric functions can be made. Let‘s say we have $\mathrm{LN}(Z)$ and want to convert it to $\mathrm{ASIN}(W)$, where W is some function of Y. We want to find what W looks like. So enter $\mathrm{LN}(Z)=\mathrm{ASIN}(W)$ and solve this for W. The result is $W=\mathrm{SIN}(\mathrm{LN}(Z))$. Does this mean that whenever you have $\mathrm{LN}(Z)$ you can convert it to $\mathrm{ASIN}(\mathrm{SIN}(\mathrm{LN}(Z)))$? Well, unfortunately not exactly. The reason is the ASIN which can send one argument to more than one result. Because of this property, this function (and all other inverse trigonometric functions) are programmed so that they return the *principal value* of all different possible values, which goes from $-\frac{\pi}{2}$ to $\frac{\pi}{2}$. $\mathrm{ASIN}(\mathrm{SIN}(X))$ *not* necessarily equal to X. To understand this better, do the following:

Define $\mathrm{F}(X)=\mathrm{LN}(X)-\mathrm{ASIN}(\mathrm{SIN}(\mathrm{LN}(X)))$.

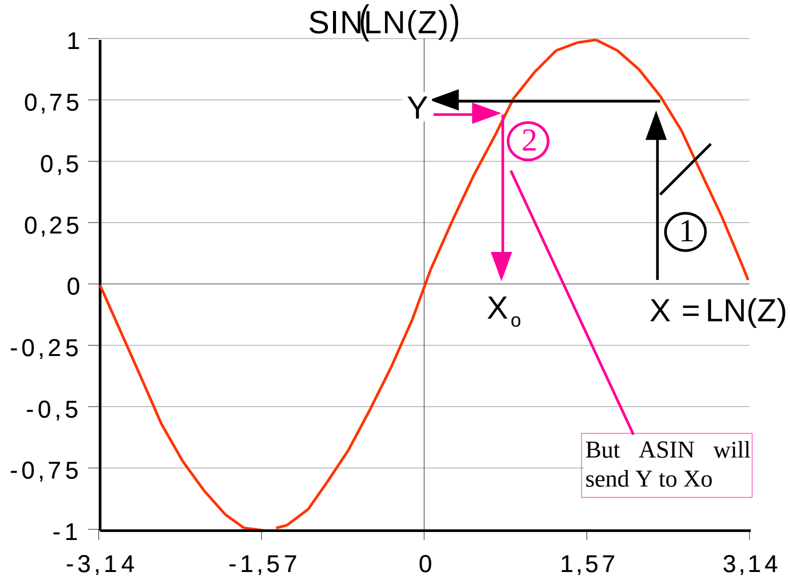
Then enter .5 and press F. The result is 0, which shows that in this case $\mathrm{LN}(X)=\mathrm{ASIN}(\mathrm{SIN}(\mathrm{LN}(X)))$. The function returns always 0 for arguments between 0.20787957635 and 4.81047738099. But try to calculate $\mathrm{F}(5.)$ and suddenly you get 0.07728317126. What is going on here? What are the strange numbers 0.20787957635 and 4.81047738099? I got a real headache thinking about the reason and was about to throw this HP49G away. That‘s why the 10th part of the trigonometry marathon had such a delay. But then Trabakoulas came and helped again. He told me to make a plot of the $\mathrm{SIN}(\mathrm{LN}(Z))$ against $\mathrm{LN}(Z)$ to understand why the HP49G behaves this way. (Turn page to read his explanations.)

When $LN(Z)$ is less than $\frac{\pi}{2}$ and greater than $-\frac{\pi}{2}$, then a given value for the sine is sent by $ASIN$ to the principal value X_o. (You start at the Y-Axis at Y, go horizontally until you meet the sine curve at point o, then go down vertically to the X-Axis until you meet the point X_o.)



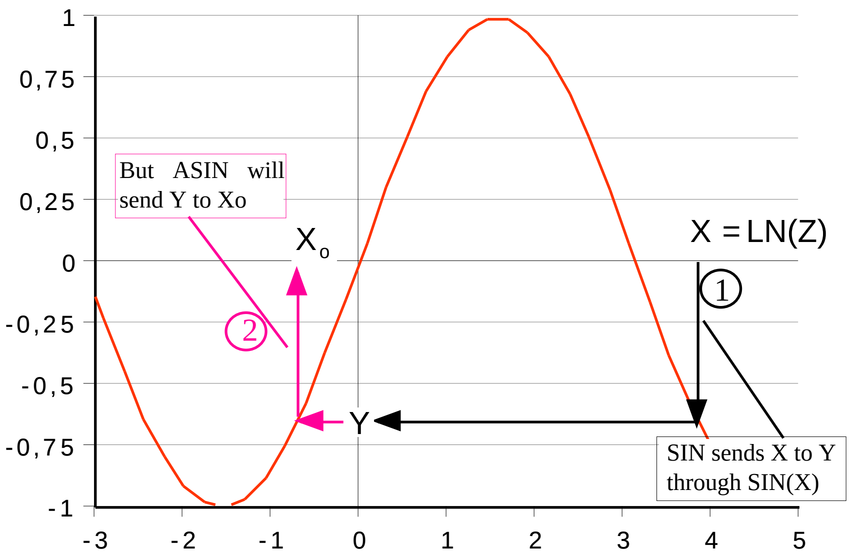
But when $LN(Z)$ is for example greater than $\frac{\pi}{2}$ then the SIN function sends $LN(Z)$ to $SIN(LN(Z)) = Y$, (from X go up until you meet the SIN curve and then to the left until you meet $Y = SIN(LN(Z))$ at the Y-axis). But then $ASIN$ sends Y to X_o and *not* to X. (From Y at the Y-Axis go to the right until you meet the curve $SIN(X)$ and then down until you reach X_o.)

We see that the arguments that „belong“ to the principal values are



those between $-\frac{\pi}{2}$ and $\frac{\pi}{2}$. But in this case the argument of SIN is not X but $LN(X)$. That means that these arguments $LN(X)$ go from $-\frac{\pi}{2}$ to $\frac{\pi}{2}$. And that means that X itself goes from $e^{-\frac{\pi}{2}}$=0.20787957635 to $e^{\frac{\pi}{2}}$=4.81047738099. That is where the strange numbers come from.

And it can get even crazier! See for example how an argument greater than π, can be sent to the other side (the spy who jumped over), to a negative X_0 value. You start at X, go down until you cut the sine curve, then go to the left until you are on the Y-axis at point Y. (The border to the other side ;-)) Then ASIN (the master executor) sends you leftwards to the sine curve because that branch is nearest, and when you go up again, to meet the X-axis, you realise that you are not where you started but at Xo<0. (The other spies cheated you, should we plan vengeance?)



And to make things „better" ;-), the same happens at the other side, when LN(Z) is negative. The situation is mirrored there.

We see that converting from LN to ASIN (or any other inverse trigonometric function) is not an easy thing to do.

Now, the benefit of converting trigonometric functions to complex exponentials (and/or logarithms) is that many things become possible, which can't be done otherwise. Consider for example $\sum_{n=0}^{N} \mathrm{SIN}(n\ X)$.

The HP49G can't return a result for this symbolic sum. And even if you have a numeric N, it takes a lot of time to return a result, when N goes to bigger values. But this sum can be calculated using conversion to complex exponentials. Let's do that. Enter $\sum_{n=0}^{N} \mathrm{SIN}(n\ X)$. Take the sum to the EQW and select SIN(n X)). Press now EXPLN. The expression SIN(n X) is converted to

$$\frac{e^{i\,n\,X} - \frac{1}{e^{i\,n\,X}}}{2\ i}$$

Press ENTER to take the sum to the stack. Now, before going any further, enter 'N' ADDTOREAL to tell the HP49G that the N of the sum is a real. Now press EXPAND and after some seconds you have the symbolic result for the symbolic sum! It is in complex form, but the imaginary part of it is 0, as it must be because we started from the real expression SIN(n X). Press ENTER to make a copy of this result and then press IM. It takes a while, but then a result with trigonometric functions is returned. Press TCOLLECT to simplify this result to 0. Press ◄ to get rid of the 0 and then RE to calculate the real part of the sum. Then press TCOLLECT, EXPAND to get

$$\frac{\mathrm{SIN}(X\ N + X) - (\mathrm{SIN}(X\ N) + \mathrm{SIN}(X))}{2\ \mathrm{COS}(X) - 2}$$

This result is the sum of SIN(n X) with n from 0 to N. It is valid for every N. We just have jumped to complex hyper space (without a sheep on our back ;-)), made things that are impossible in our real space, and then returned with the result. You can use this result to DEFINE user functions that calculate such a sum, with n from 0 to, say 1000, instead of waiting until the HP49G builds $\mathrm{SIN}(0\ \ X)+\mathrm{SIN}(1\ \ X)+\quad+\ \mathrm{SIN}(1000\ \ X)$. The same way you can calculate $\sum_{n=n_0}^{N}\mathrm{SIN}(n\ \ X)$ or $\sum_{n=0}^{N}\mathrm{COS}(n\ \ X)$ and so on.

But wait a minute. The sum of $\mathrm{SIN}(n\ X)$ with n from 0 to N is a finite quantity for any X when N is a finite number. But our result for this sum contains $2\ \ \mathrm{COS}(X)-2$ in the denominator. This is equal to 0, when X is $2\quad \pi n$. Does this mean that then the sum is the infinite and we have made a mistake? No, because the numerator of the result is then also 0, which tells us that we have $\frac{0}{0}$ and so must work with limits. Because N in our result is an integer and because the HP49G still doesn‘t have INTEGERASSUME, lets put an integer value for N in our result. Press ENTER to make a copy of the result for the sum, and then enter 'N =5' and press SUBST. Then enter 'X = 2 ı' and press lim. The HP49G returns 0 which is correct. That means that the expression

$$\frac{\mathrm{SIN}(X\ \ N+X)-\left(\mathrm{SIN}(X\ \ N)+\mathrm{SIN}(X)\right)}{2\ \ \mathrm{COS}(X)-2}$$

is 0 for $N=5$ and $X = 2\ \ ı$. You can try also other combinations of values for N and X, like $N=4$ and $X = 2\ \ ı$, $N=4$ and $X = 6\ \ ı$ and so on. This is also a nice way to demonstrate the following fact: Because the sum $\mathrm{SIN}(0\ \ X)+\mathrm{SIN}(1\ \ X)+\quad+\ \mathrm{SIN}(1000\ \ X)$ has no singularities when N is finite, so does also its equivalent form

$$\frac{\mathrm{SIN}(X\ \ N+X)-\left(\mathrm{SIN}(X\ \ N)+\mathrm{SIN}(X)\right)}{2\ \ \mathrm{COS}(X)-2}$$

The value of this expression for $2\quad \pi n$ *can be defined to be* the limit for $X = 2\quad \pi n$. It is not only that we can go infinitely near the point $2\quad \pi n$ to have a defined result for

$$\frac{\mathrm{SIN}(X\ \ N+X)-\left(\mathrm{SIN}(X\ \ N)+\mathrm{SIN}(X)\right)}{2\ \ \mathrm{COS}(X)-2}$$

but that we can also use that result as the value of the expression at that point. This result, *that we define*, exists also at the point $X = 2\quad \pi n$, because if it wouldn‘t, then also the sum $\mathrm{SIN}(0\ \ X)+\mathrm{SIN}(1\ \ X)+\quad+\ \mathrm{SIN}(1000\ \ X)$ should have an undefined value for $X = 2\quad \pi n$, which is absurd! The same holds for every other expression (like $\frac{\mathrm{SIN}(X)}{X}$ when $X\rightarrow 0$) if the limit exists.

Let‘s move on to other conversions. The HP49G has also the hyperbolic functions SINH, COSH, TANH, ASINH, ACOSH, and ATANH built-in. The command EXPLN also converts such functions to complex exponentials. For example, enter COSH(X) and press EXPLN to convert this to

$$\frac{e^X+\frac{1}{e^X}}{2}$$

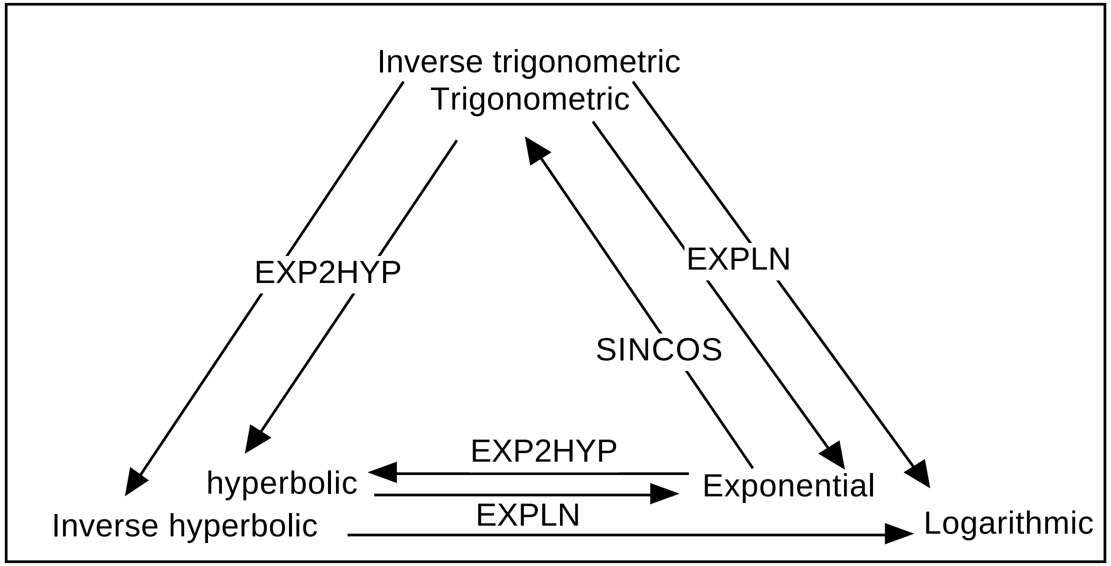
If you don‘t like this form (like I do) press LIN to get

$$\frac{1}{2}\ e^X+\frac{1}{2}\ e^{-X}$$

The opposite can be done with the command EXP2HYP, which converts exponentials to hyperbolics. For example enter e^X and press EXP2HYP. The result is SINH(X)+ COSH(X). The trigonometric functions SIN, COS, TAN, ASIN, ACOS, ATAN can also be converted to hyperbolic functions with EXP2HYP. Enter TAN(X) and press EXP2HYP to get

$$\frac{\mathrm{SINH}(2\cdot i\cdot X)+\mathrm{COSH}(2\cdot i\cdot X)}{i\cdot \mathrm{SINH}(2\cdot i\cdot X)+i\cdot \mathrm{COSH}(2\cdot i\cdot X)+i}.$$

And here is a picture with all built-in conversions:



Let's do some examples now. (Complex rigourous mode, X, Y and N are in REALASSUME)

1) Show that $\mathrm{COSH}(X)^2-\mathrm{SINH}(X)^2=1$

Enter $\mathrm{COSH}^2(X)-\mathrm{SINH}^2(X)$. Press EXPLN and EXPAND to get a nice round 1.

2) Express $e^{\mathrm{ACOS}(X)}$ without using any exponential, trigonometric or hyperbolic functions.

Enter $e^{\mathrm{ACOS}(X)}$, press EXPLN and EXPAND. Result:

$$\frac{1}{\left(X+\sqrt{X^2-1}\right)^i}$$

3) Express $\mathrm{SIN}(i\cdot X)$ as a hyperbolic function.

Enter $\mathrm{SIN}(i\cdot X)$ and press EXPLN to get

$i\cdot \mathrm{SINH}(X)$

4) Turn $\mathrm{COS}(X+i\cdot Y)$ to an expression that consists of functions that have either X or Y but not both X and Y as arguments.

Enter $\mathrm{COS}(X+i\cdot Y)$. Since we want to have X or Y alone as arguments, press TEXPAND to expand the expression to sums of products:

$\mathrm{COS}(i\cdot Y)\cdot \mathrm{COS}(X)-\mathrm{SIN}(i\cdot Y)\cdot \mathrm{SIN}(X)$

Press ▼ to get this to the EQW. Select $\mathrm{COS}(i\cdot Y)$ and press EXP2HYP. Now select $\mathrm{SIN}(i\cdot Y)$ and press again EXP2HYP. Press ENTER to put the result

$\mathrm{COSH}(Y)\cdot \mathrm{COS}(X)-i\cdot \mathrm{SINH}(Y)\cdot \mathrm{SIN}(X)$

to the stack.

5) Find the real and imaginary parts of $\mathrm{COS}(X+i\ Y)$.

Enter $\mathrm{COS}(X+i\ Y)$ and press ENTER to make a copy of this expression at stack level 2. Press RE. You get

$$\frac{e^{-Y}\ \mathrm{COS}(X)+e^{X}\ \mathrm{COS}(X)}{2}$$

which is the real part. Is this equal to $\mathrm{COSH}(Y)\ \mathrm{COS}(X)$, the real part of the expression from the last example? Let‘s see. Press COLLECT to convert the expression to

$$\frac{\left(e^{-Y}+e^{Y}\right)\ \mathrm{SIN}(X)}{2}$$

Now take this to the EQW, select $e^{-Y}+e^{Y}$ and press EXP2HYP. The result is

$$\frac{2\ \mathrm{COSH}(Y)\ \mathrm{COS}(X)}{2}$$

Press ENTER and then EXPAND to get $\mathrm{COSH}(Y)\ \mathrm{COS}(X)$. Now press ▶ to take $\mathrm{COS}(X+i\ Y)$ to stack level 1. Press IM to find the imaginary part. Press COLLECT and take the resulting expression to the EQW. Select $e^{-Y}-e^{Y}$ and press EXP2HYP. Press ENTER and then EXPAND to see that this is equal to the imaginary part of the previous example.

6) Show that if X, Y are real, then $\mathrm{SIN}(X+i\ Y)\ \mathrm{SIN}(X-i\ Y)$ is also real.

Enter $\mathrm{SIN}(X+i\ Y)\ \mathrm{SIN}(X-i\ Y)$ and press TEXPAND. Take the result to the EQW. Select and apply EXP2HYP to all occurrences of $\mathrm{COS}(i\ Y)$ and $\mathrm{SIN}(i\ Y)$. Press ENTER and EXPAND to simplify the expression to

$$\mathrm{COSH}(X)^2\ \mathrm{SIN}(X)^2+\mathrm{SINH}(X)^2\ \mathrm{COS}(X)^2$$

7) Find the symbolic sum $\sum_{n=0}^{N}\mathrm{COSH}(n\ X)$.

Enter

$$\sum_{n=0}^{N}\mathrm{COSH}(n\ X)$$

take this to the EQW, select $\mathrm{COSH}(n\ X)$ and press EXPLN. Press ENTER and the EXPAND to find the symbolic sum. You can use PARTFRAC to split this a sum of smaller quotients. Take the result to the EQW and apply EXPAND to each quotient. The result is then

$$\frac{e^{X\ N+X}}{2\ e^{X}-2}+\frac{e^{X}}{\left(2\ e^{X}-2\right)\ e^{X\ N+X}}+\frac{1}{2}.$$

8) Convert $e^{X+i\ Y}$ to an expression with trigonometric functions.

Enter $e^{X+i\ Y}$ and press SINCOS.

9) Solve $\mathrm{COSH}(X)+e^{X}=0$ for X.

If you try to solve this with the built-in SOLVE then you get the error „Not reducible to a rational expression“. But if you enter $\mathrm{COSH}(X)+e^{X}=0$, press EXPLN and then solve this for X, you get a list with 2 solutions. After you EXPAND the arguments of the logarithm functions, the solutions are

$$X = 2\ i\ n2 + LN\left(\frac{i\ \sqrt{3}}{3}\right)$$

and

$$X = 2\ i\ n2 + LN\left(-\frac{i\ \sqrt{3}}{3}\right)$$

Now, what do you think? Should TRISOL make its evolutionary way to HYPEXTRISOL? ;-)

10) Convert $\dfrac{(e^X)^6 + (e^X)^5 + e^X + 1}{2\ (e^X)^3}$ to a more simple expression that contains hyperbolic functions of multiples of X.

Enter

$$\frac{(e^X)^6 + (e^X)^5 + e^X + 1}{2\ (e^X)^3}$$

Since we want multiples of X like $2\ X$, $3\ X$ and so on, as arguments of hyperbolic functions, it seems reasonable to use LIN, to turn things like $(e^X)^n$ to things like $e^{n\ X}$. Press **LIN**. Now press **EXP2HYP**. The result is $SINH(3\ X) + COSH(2\ X)$.

In all these examples I have used very very often VPN's program for STARTEQW, modified to contain all commands like EXP2HYP, EXP2POW etc., that you find only through menu hunt otherwise. (Hello J.H.Meyers ;-)). But of course it's up to you how to use the commands. If you prefere menus, the use menus. If you prefer typing and entering the commands, then do it that way. There is no "ultimative way" to use the HP49G. Just follow your own gusto.

Ending this last part of the Trigonometry Marathon, I want to say thanks to all people who commended, corrected and asked. This was one of the main powers that kept me on working. Trabakoulas also wants to thank you all, for helping find all his sheep. (Except the one at the ski jump, of course ;-))

Before putting the COLLECTed PDF parts of this marathon to hpcalc, I'll add a part with trigonometric/hyperbolic/exponential conversions on the HP48. But I'll not post this part here. Also I'll put the newest version of TRISOL to hpcalc.

Next marathon will be the Complex Marathon (is that VPN screaming? ;-)), where Kojak will SOLVE complex cases with the joint forces of TRISOL and COMSOL. Or was it COMTRISOL? Or HYPEXCOMTRISOL? Well, we will see. Also Trabakoulas will be running on ice making complex jumps with sheep.
Thanks a lot for your interest and keep tuned.
Hyperbolic greetings,
Nick.

Trigonometry with the HP48 - Additional Part 11

This additional part of the trigonometry marathon will be dedicated to the users of the HP48. This calculator doesn't have out of the box the big variety of commands available to the HP49G, but nonetheless there are some things that are possible with the built-in commands only. Of course it is possible to install ERABLE and do much more, but then the biggest part of the marathon up to now applies also to the HP48. So let's go and see what is possible using only the built-in functions.

We have seen that on the HP49G the command EXPLN converts trigonometric functions to complex exponentials. On the HP48 there is the *operation* → DEF which does this. Unfortunately this operation is only available in the EQW. You can't use it in programs or elsewhere. (Perhaps some guru out there could tell us, if there is a SYSEVAL that can perform this operation on an algebraic on the stack.) Let's test this operation. Enter 'SIN(X)' on the stack and press ▼ once, to take the expression to the EQW. The EQW starts in scroll mode, so press the key CANCEL (that means the key ON) once to exit this mode and enter edit-mode. Press ◄ to select the SIN function. With this function selected press the menu key RULES. This brings a menu with operations that can be performed. The first is → DEF. Press the menu key →DEF. After a while the HP48 shows the result:

$$\frac{EXP(X\ i)-EXP(-X\ i)}{2\ i}$$

This can be used to derive such things like $\sin^2(x)+\cos^2(x)=1$. Let's see how this can be achieved. Enter 'SIN(X)^2+COS(X)^2' and take this to the EQW. Select SIN, press RULES and then →DEF. When the HP48 is ready select the function COS, press again RULES and then →DEF. Now the expression

$$\left(\frac{EXP(X\ i)-EXP(-X\ i)}{2\ i}\right)^2+\left(\frac{EXP(X\ i)+EXP(-X\ i)}{2}\right)^2$$

is on the EQW. Press ENTER to take this to the stack. Now press SYMBOLIC, to get the symbolic menu. Press EXPAN twice to expand the expression to:

$$\left(\frac{EXP(X\ i)}{2\ i}-\frac{EXP(-X\ i)}{2\ i}\right)\left(\frac{EXP(X\ i)}{2\ i}-\frac{EXP(-X\ i)}{2\ i}\right)+\left(\frac{EXP(X\ i)}{2}+\frac{EXP(-X\ i)}{2}\right)\left(\frac{EXP(X\ i)}{2}+\frac{EXP(-X\ i)}{2}\right)$$

Now press again ▼, CANCEL and then select multiplication sign between

$$\left(\frac{EXP(X\ i)}{2\ i}-\frac{EXP(-X\ i)}{2\ i}\right)$$

and

$$\left(\frac{EXP(X\ i)}{2\ i}-\frac{EXP(-X\ i)}{2\ i}\right).$$

Press RULES and then ←D to distribute the multiplication to the left. The result is:

$$\left(\frac{EXP(X\ i)}{2\ i}\right)\left(\frac{EXP(X\ i)}{2\ i}-\frac{EXP(-X\ i)}{2\ i}\right)-\frac{EXP(-X\ i)}{2\ i}\left(\frac{EXP(X\ i)}{2\ i}-\frac{EXP(-X\ i)}{2\ i}\right)+\left(\frac{EXP(X\ i)}{2}+\frac{EXP(-X\ i)}{2}\right)\left(\frac{EXP(X\ i)}{2}+\frac{EXP(-X\ i)}{2}\right)$$

Select the multiplication sign between

$$\left(\frac{EXP(X\ i)}{2\ i}\right)$$

and

$$\left(\frac{EXP(X\ i)}{2\ i}-\frac{EXP(-X\ i)}{2\ i}\right).$$

Press **RULES** and then press **D→** to distribute to the right. The result now is:

$$\frac{EXP(X\ i)}{2\ i}\ \frac{EXP(X\ i)}{2\ i}-\frac{EXP(X\ i)}{2\ i}\ \frac{EXP(-X\ i)}{2\ i}-\frac{EXP(-X\ i)}{2\ i}\left(\frac{EXP(X\ i)}{2\ i}-\frac{EXP(-X\ i)}{2\ i}\right)+\left(\frac{EXP(X\ i)}{2}+\frac{EXP(-X\ i)}{2}\right)\left(\frac{EXP(X\ i)}{2}+\frac{EXP(-X\ i)}{2}\right)$$

Select multiplication sign between

$$\frac{EXP(X\ i)}{2\ i}$$

and

$$\frac{EXP(X\ i)}{2\ i}$$

and press **RULES**. Press **PREV** to get the last side of the RULES menu. Press the menu key **COLCT**. Now you have:

$$.25\ EXP(X\ i)^2\ i^{-2}-\frac{EXP(X\ i)}{2\ i}\ \frac{EXP(-X\ i)}{2\ i}-\frac{EXP(-X\ i)}{2\ i}\left(\frac{EXP(X\ i)}{2\ i}-\frac{EXP(-X\ i)}{2\ i}\right)+\left(\frac{EXP(X\ i)}{2}+\frac{EXP(-X\ i)}{2}\right)\left(\frac{EXP(X\ i)}{2}+\frac{EXP(-X\ i)}{2}\right)$$

Now multiplication between

$$\frac{EXP(X\ i)}{2\ i}$$

and

$$\frac{EXP(-X\ i)}{2\ i}$$

and from the menu RULES press the menu key **COLCT** again to get:

$$.25\ EXP(X\ i)^2\ i^{2}-.25\ EXP(X\ i)\ EXP(-X\ i)\ i^{-2}-\frac{EXP(-X\ i)}{2\ i}\left(\frac{EXP(X\ i)}{2\ i}-\frac{EXP(-X\ i)}{2\ i}\right)+\left(\frac{EXP(X\ i)}{2}+\frac{EXP(-X\ i)}{2}\right)\left(\frac{EXP(X\ i)}{2}+\frac{EXP(-X\ i)}{2}\right)$$

Now select the third multiplication sign (counting from the left) of the sub-expression $-.25\ EXP(X\ i)\ EXP(-X\ i)\ i^{-2}$ and from the menu RULES press **A→** to associate the two exponentials in parentheses. The sub-expression is now $-.25\ (EXP(X\ i)\ EXP(-X\ i))\ i^{-2}$. Select the multiplication sign between the exponentials and from the menu

RULES press **←M** to merge the two exponentials. Now the sub-expression is $-.25\ \mathrm{EXP}(-(X\ i)+X\ i)\ i^{-2}$. Select the exponential of this sub-expression. From the menu RULES press **COLCT**. The sub expression is now: $-.25\ 1\ i^{-2}$. Select the second multiplication sign and again press **COLCT**. Now this sub-expression is $-.25\ i^{-2}$.

Now in the sub-expression

$$\frac{\mathrm{EXP}(-X\ i)}{2\ i}\left(\frac{\mathrm{EXP}(X\ i)}{2\ i}-\frac{\mathrm{EXP}(-X\ i)}{2\ i}\right)$$

select the multiplication sign between

$$\frac{\mathrm{EXP}(-X\ i)}{2\ i}$$

and

$$\left(\frac{\mathrm{EXP}(X\ i)}{2\ i}-\frac{\mathrm{EXP}(-X\ i)}{2\ i}\right)$$

and press again **D→**. Working like with the first sub-expression you can bring this to the form:

$$.25\ i^{-2}-.25\ \mathrm{EXP}(-X\ i)^2\ i^{-2}$$

so that the whole expression is:

$$.25\ \mathrm{EXP}(X\ i)^2\ i^{-2}-.25\ i^{-2}-\left(.25\ i^{-2}-.25\ \mathrm{EXP}(-X\ i)^2\ i^{-2}\right)+\left(\frac{\mathrm{EXP}(X\ i)}{2}+\frac{\mathrm{EXP}(-X\ i)}{2}\right)\left(\frac{\mathrm{EXP}(X\ i)}{2}+\frac{\mathrm{EXP}(-X\ i)}{2}\right)$$

Now select the first minus sign in the sub-expression $25\ i^{-2}-\left(.25\ i^{-2}-.25\ \mathrm{EXP}(-X\ i)^2\ i^{-2}\right)$ and from the menu RULES press **(←** to include the expression $-.25\ i^{-2}$ in the parentheses. The expression is now:

$$.25\ \mathrm{EXP}(X\ i)^2\ i^{-2}-\left(.25\ i^{-2}+.25\ i^{-2}-.25\ \mathrm{EXP}(-X\ i)^2\ i^{-2}\right)+\left(\frac{\mathrm{EXP}(X\ i)}{2}+\frac{\mathrm{EXP}(-X\ i)}{2}\right)\left(\frac{\mathrm{EXP}(X\ i)}{2}+\frac{\mathrm{EXP}(-X\ i)}{2}\right)$$

Select the first plus sign of the sub-expression $\left(.25\ i^{-2}+.25\ i^{-2}-.25\ \mathrm{EXP}(-X\ i)^2\ i^{-2}\right)$ and from the RULES menu press **COLCT**. Now you have:

$$.25\ \mathrm{EXP}(X\ i)^2\ i^{-2}-\left(.5\ i^{-2}-.25\ \mathrm{EXP}(-X\ i)^2\ i^{-2}\right)+\left(\frac{\mathrm{EXP}(X\ i)}{2}+\frac{\mathrm{EXP}(-X\ i)}{2}\right)\left(\frac{\mathrm{EXP}(X\ i)}{2}+\frac{\mathrm{EXP}(-X\ i)}{2}\right)$$

Repeat the whole procedure for the sub-expression

$$\left(\frac{\mathrm{EXP}(X\ i)}{2}+\frac{\mathrm{EXP}(-X\ i)}{2}\right)\left(\frac{\mathrm{EXP}(X\ i)}{2}+\frac{\mathrm{EXP}(-X\ i)}{2}\right).$$

The results of the manipulations are:

$$.25\ \mathrm{EXP}(X\ i)^2\ i^{-2}-\left(.5\ i^{-2}-.25\ \mathrm{EXP}(-X\ i)^2\ i^{-2}\right)+$$

1) $\frac{\mathrm{EXP}(X\ i)}{2}\left(\frac{\mathrm{EXP}(X\ i)}{2}+\frac{\mathrm{EXP}(-X\ i)}{2}\right)+$

$\frac{\mathrm{EXP}(-X\ i)}{2}\left(\frac{\mathrm{EXP}(X\ i)}{2}+\frac{\mathrm{EXP}(-X\ i)}{2}\right)$

$$.25\ \mathrm{EXP}(X\ i)^2\ i^{-2} - \left(.5\ i^{-2} - .25\ \mathrm{EXP}(-X\ i)^2\ i^{-2}\right) +$$

2)
$$\frac{\mathrm{EXP}(X\ i)}{2}\ \frac{\mathrm{EXP}(X\ i)}{2} + \frac{\mathrm{EXP}(X\ i)}{2}\ \frac{\mathrm{EXP}(-X\ i)}{2}$$
$$\frac{\mathrm{EXP}(-X\ i)}{2}\ \frac{\mathrm{EXP}(X\ i)}{2} + \frac{\mathrm{EXP}(-X\ i)}{2}\ \frac{\mathrm{EXP}(-X\ i)}{2}$$

3)
$$.25\ \mathrm{EXP}(X\ i)^2\ i^{-2} - \left(.5\ i^{-2} - .25\ \mathrm{EXP}(-X\ i)^2\ i^{-2}\right) +$$
$$.25\ \mathrm{EXP}(X\ i)^2 + .25 + .25 + .25\ \mathrm{EXP}(-X\ i)^2$$

4)
$$.25\ \mathrm{EXP}(X\ i)^2\ i^{-2} - \left(.5\ i^{-2} - .25\ \mathrm{EXP}(-X\ i)^2\ i^{-2}\right) +$$
$$.25\ \mathrm{EXP}(X\ i)^2 + .5 + .25\ \mathrm{EXP}(-X\ i)^2$$

Now, you may think that the rest is easy, but it isn't. This has to do with the fact that i^{-2} can't be easily converted to a -1 on the HP48. We have a little more to do. Select the exponentiation sign of the first i^{-2} and from the menu RULES press the key **1/()** to convert i^{-2} to $\mathrm{INV}(i^2)$. Now select the exponentiation sign of i^2 in $\mathrm{INV}(i^2)$ and from the RULES menu press **COLCT** to convert this to $\mathrm{INV}(-1)$. Convert all appearances of i^{-2} to $\mathrm{INV}(-1)$ this way. Now the expression is:

$$.25\ \mathrm{EXP}(X\ i)^2\ \mathrm{INV}(-1) -$$
$$\left(.5\ \mathrm{INV}(-1) - .25\ \mathrm{EXP}(-X\ i)^2\ \mathrm{INV}(-1)\right) +$$
$$.25\ \mathrm{EXP}(X\ i)^2 + .5 + .25\ \mathrm{EXP}(-X\ i)^2$$

Press **ENTER** to put this edited expression on the stack. Press **SYMBOLIC** for the symbolic menu and from this menu press **COLCT**.

The result is a 1.

If despite all this work you still don't want to install ERABLE, then you must belong to the hard(est) core of the users that want to control each electron that passes through the registers of the processor. ;-)

And if despite all this work you still complain that the HP49G is slow, then you must be very young and didn't have any experience with the HP48. ;-)

The operation that does the opposite of $\to\mathrm{DEF}$ is $\to\mathrm{TRG}$. It transforms exponentials to trigonometric functions. Enter for example $\mathrm{EXP}(i\ X)$ and take it to the EQW. Select the EXP function, press **RULES** and then press **→TRG**. The result is

$$\mathrm{COS}\left(\frac{X}{i}\right) + \mathrm{SIN}\left(\frac{X}{i}\right)\ i$$

which you can **COLCT** to $\mathrm{SIN}(X)\ i + \mathrm{COS}(X)$.

Let's have an example. (What a patient guy I am ;-)) We want to transform $\mathrm{SIN}(X)\ \mathrm{COS}(X)$ to $\dfrac{\mathrm{SIN}(2\ x)}{2}$. Go to the EQW, enter $\mathrm{SIN}(X)\ \mathrm{COS}(X)$ and then press **◀** to go to edit mode. Now select the SIN function and from the menu RULES press the key **→DEF**. Do the same with the function COS. Now you have:

$$\left(\frac{\mathrm{EXP}(X\ i) - \mathrm{EXP}(-X\ i)}{2\ i}\right)\left(\frac{\mathrm{EXP}(X\ i) + \mathrm{EXP}(-X\ i)}{2}\right)$$

Now, select the division line of the first ratio and from the RULES menu press **←D** to distribute the division by $2\ i$. The expression now is:

$$\left(\frac{EXP(X\ i)}{2\ i} - \frac{-EXP(-X\ i)}{2\ i}\right)\left(\frac{EXP(X\ i)+EXP(-X\ i)}{2}\right)$$

Select the division line of the second ratio and from the RULES menu press ←D again to distribute the division by 2. The expression now is:

$$\left(\frac{EXP(X\ i)}{2\ i} - \frac{-EXP(-X\ i)}{2\ i}\right)\left(\frac{EXP(X\ i)}{2} + \frac{+EXP(-X\ i)}{2}\right)$$

Next, select the multiplication sign between the two big parentheses and from the RULES menu press ←D again. The expression is now:

$$\frac{EXP(X\ i)}{2\ i}\left(\frac{EXP(X\ i)}{2} + \frac{+EXP(-X\ i)}{2}\right) - \frac{-EXP(-X\ i)}{2\ i}\left(\frac{EXP(X\ i)}{2} + \frac{+EXP(-X\ i)}{2}\right)$$

Now select the first multiplication sign and from the RULES menu press D→ to convert the expression to:

$$\frac{EXP(X\ i)}{2\ i}\,\frac{EXP(X\ i)}{2} + \frac{EXP(X\ i)}{2\ i}\,\frac{+EXP(-X\ i)}{2} - \frac{-EXP(-X\ i)}{2\ i}\left(\frac{EXP(X\ i)}{2} + \frac{+EXP(-X\ i)}{2}\right)$$

Select again the first multiplication sign and from the RULES menu press COLCT:

$$\frac{.25\ EXP(X\ i)^{2}}{i} + \frac{.25\ EXP(X\ i)\ EXP(-X\ i)}{2\ i} - \frac{-EXP(-X\ i)}{2\ i}\left(\frac{EXP(X\ i)}{2} + \frac{EXP(-X\ i)}{2}\right)$$

Now repeat the last steps for the sub-expression

$$\frac{-EXP(-X\ i)}{2\ i}\left(\frac{EXP(X\ i)}{2} + \frac{+EXP(-X\ i)}{2}\right)$$

That is, select the multiplication sign and press D→ from the rule menu. Then COLCT the two resulting sub-terms so that you have:

$$\frac{.25\ EXP(X\ i)^{2}}{i} + \frac{.25\ EXP(X\ i)\ EXP(-X\ i)}{2\ i} - \left(\frac{.25\ EXP(-X\ i)\ EXP(X\ i)}{i} + \frac{.25\ EXP(-(X\ i))^{2}}{i}\right)$$

Press ENTER to put the expression on the stack. From the SYMBOLIC menu press COLCT. This returns:

$$-\frac{.25\ EXP(-(X\ i))^{2}}{i} + \frac{.25\ EXP(X\ i)^{2}}{i}$$

Press ▼ to take the expression to the EQW, and CANCEL to exit scroll mode. Select the exponentiation sign of the first exponential and from the menu RULES press E(). This linearizes the exponential. Do the same with the second exponential, so that you have:

$$-\frac{.25\cdot EXP(-(X\cdot i)\cdot 2)}{i}+\frac{.25\cdot EXP(X\cdot i\cdot 2)}{i}$$

Select now the first exponential, press **RULES** and then **→TRG**, to convert the exponential to trigonometric functions. Do the same with the second exponential. Now you have:

$$-\frac{.25\cdot\left(COS\left(\frac{-(X\cdot i)\cdot 2}{i}\right)+SIN\left(\frac{-(X\cdot i)\cdot 2}{i}\right)\cdot i\right)}{i}+\frac{.25\cdot\left(COS\left(\frac{X\cdot i\cdot 2}{i}\right)+SIN\left(\frac{X\cdot i\cdot 2}{i}\right)\cdot i\right)}{i}$$

Press **ENTER** to exit the EQW and to go to the stack with the edited expression. It looks as if you could COLCT terms but you must first use EXPA. From the menu SYMBOLIC press **EXPA** five times to convert the expression to

$$\frac{-0.25\cdot COS\left(\frac{-(X\cdot i)\cdot 2}{i}\right)}{i}+\frac{-.25\cdot SIN\left(\frac{-(X\cdot i)\cdot 2}{i}\right)\cdot i}{i}+\frac{.25\cdot COS\left(\frac{X\cdot i\cdot 2}{i}\right)}{i}+\frac{.25\cdot SIN\left(\frac{X\cdot i\cdot 2}{i}\right)\cdot i}{i}$$

Now press **COLCT** once. This converts the expression to:

$$-(.25\cdot -SIN(2\cdot X))-\frac{0.25\cdot COS(2\cdot X)}{i}+.25\cdot SIN(2\cdot X)+\frac{.25\cdot COS(2\cdot X)}{i}$$

Press **COLCT** again to get the result: $.5\cdot SIN(2\cdot X)$.

The method can be used also for other expressions. We use **→DEF** to turn trigonometric functions to complex exponentials. Then we distribute using **D→** and **←D**. We can linearize the exponentials with **E()**. And at the end we turn the complex exponentials to trigonometric functions using **→TRG**. Between these steps, we can use **(←**, **→)** or **←M**, **M→** or **[()]** to group terms, **E^** to bring products of exponentials $EXP(a)\cdot EXP(b)$ to the form $EXP(a+b)$, and also **COLCT** to collect like terms. At the end we use **EXPA** as many times as needed, so that the following **COLCT** can perform a collection of like terms and throw away terms that cancel each other. Note the importance of the operation **E^** which linearizes the product $EXP(a)\cdot EXP(b)$. It is this step, which causes the creation of functions with the combined arguments $EXP(a+b)$.

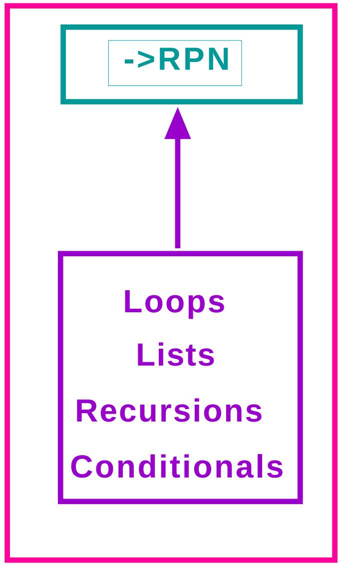
Another non-programmable operation that is available in the menu RULES is the operation TRG. This operation makes conversions like $SIN(x+y)=SIN(x)\cdot COS(y)+COS(x)\cdot SIN(y)$. Let's try it. Go to the EQW and enter $COS(X-Y)$. Press **◀** and then select the function COS. Press the menu key **RULES** and then **TRG***. The result is $COS(X)\cdot COS(Y)+SIN(X)\cdot SIN(Y)$.

The same operation can also be used for transformations of trigonometric functions of multiples of an angle, like for example $SIN(2\cdot X)$, though a little bit additional work is needed. Let's do that. In the EQW enter $SIN(2\cdot X)$ and select SIN. From the RULES menu press **TRG***. The HP48 gives a short insulting beep and does nothing. But we can transform $2\cdot X$ to $X+X$. Select the factor 2 of $2\cdot X$, go to the RULES menu and press **+1-1** to add and subtract 1 to 2. Now the expression is $SIN((2+1-1)\cdot X)$. Select the minus sign and from the RULES menu press **←T** to transfer the -1 before the $+1$. The expression is now $SIN((2+-1+1)\cdot X)$. Select the plus sign that is before the −1 and from the RULES menu (second page) press COLCT. Now the expression is SIN((1+1) X). Select the multiplication sign and from the RULES menu press ←D to distribute the multiplication over the sum. the expression becomes SIN(1 X+1 X). Now select the SIN function and from the RULES menu press TRG*. The EQW now contains SIN(1 X) COS(1 X)+COS(1 X) SIN(1 X). Press ENTER to go to the stack and from the menu SYMBOLIC press COLCT to get the result 2 COS(X) SIN(X).

As you can see, many of the trigonometric conversions are possible, but the available operations cannot be used from the stack or in programs. Also, these operations are much more "elementary" than those of the HP49G, which means that you have to press many more keys, "many more" being an euphimism. If you combine this with the slower execution of the HP48, then you see directly that you must have patience. But on the other hand, I wish I had some of these elementary operations on the HP49G, like for example +1-1, to turn a simple 2 into a 1+1, for which I still don't know if it can be done on the HP49G. This would be nice for such things like for example SIN(4 X) which I could transform to SIN(3 X+X) or SIN(2 X+2 X) according to my needs. The first could be converted to SIN(3 X) COS(X)+COS(3 X) SIN(X) and the second to 2 COS(2 X) SIN(2 X). Both expressions are mathematically equal, but many times one just fits better than the other. A set of such elementary commands for the HP49G would give us a much more detailed control of the way that the calculator does its work.

Now, let's ask ourselves, if it is possible to program the HP48 using *only* UserRPL, so that many trigonometric conversions that are possible in the EQW, become also possible on the stack?. If we succeed making such programs, then we would have some benefits like for example, using these programs in other programs, avoiding work in the slow EQW, and last but not least taking a look at the way that elementary commands are combined to give more complex commands which can be further combined and so on. Like constructing a house out of a set of a few pieces. We will watch the construction of a set of trigonometric commands using pictures.

First of all, let's make a program that converts an algebraic object to a a list of RPL objects. The list should be constructed in a way, that evaluating the objects one after the other, the original algebraic object appears again. You should have reason to be glad, because HP already put such a program in the calculator. Just type TEACH and press ENTER. A new directory EXAMPLES with examples is generated in HOME. Go to that directory. Inside EXAMPLES there is another directory named PRGS. Go to PRGS. In the menu VAR you see that there is a program named →RPN. This is the program that we need. Recall →RPN and go to HOME again. Create a directory named TRIGO (or anything else you like) and go to this directory. The program should be on stack level 1. Enter '→RPN' and press STO. This is one of our basic programs. We will need it occasionally, for example to prove that some sub-expression is or is not in an algebraic object. This program uses repeated OBJ→ in a loop to split an algebraic object to the objects of which it consists. It uses conditionals to check if an object of the original algebraic is itself an algebraic, in which case it just calls itself and passes itself this sub-algebraic, or to just puts the current object in the result list, if it is not an algebraic. Our set of trigonometric commands now consists of one basic command, namely the program →RPN.



The next basic thing that we need, is a program that completely expands an algebraic. The programmable command EXPAN which the HP48 provides, does only one expansion and then stops. Programming a new command for complete expansion just uses the built-in command EXPAN repeatedly, until the algebraic object doesn't change any more. Here we go:

```
<< 0 -> iter
      << "Expanding..." 1 DISP
         DO
            "Pass " 'iter' INCR +
            2 DISP
            DUP EXPAN
            DUP ROT
         UNTIL
            SAME
         END
      >>
>>
```

Store this program in EXPAND in the same directory you have already stored → RPN. The program takes an algebraic and returns its completely expanded form. It also shows which pass is being performed during execution. Let's have an example. Enter

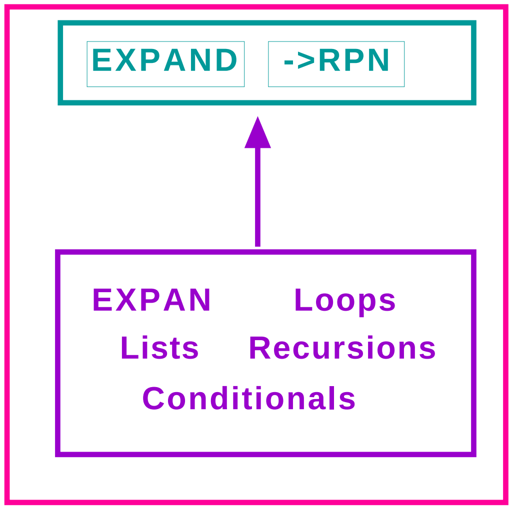
EXP(X i − Y)

and press ENTER. Go to the menu SYMBOLIC and press EXPA. The result is:

$$\frac{EXP(X\ i)}{EXP(Y)}.$$

Now press ▶, go to the menu VAR and press EXPAND. The result is:

$$\frac{EXP(X)^{i}}{EXP(Y)}.$$



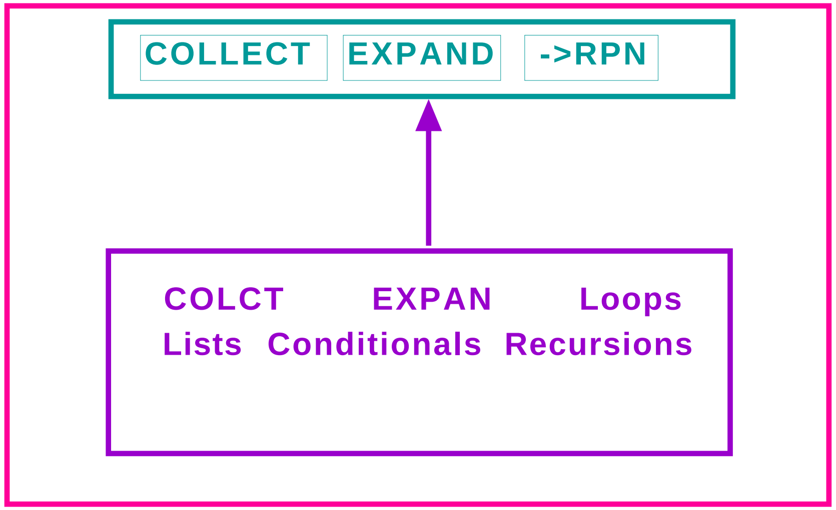
We will use this command very often in the following programs for trigonometric conversions. Our set of commands now consists of 2 commands.

Now that we have EXPAND, the opposite comes into mind. A program that does complete collection of like terms. The manual of the HP48 says that the built-in COLCT does this, but this is not always true. There are cases, where further collecting is possible, but COLCT stops at an intermediate point. It is easy to make such a program, now that we have EXPAND. We just edit EXPAND and replace the command EXPAN with COLCT:

```
<< 0 -> iter
      << "Collecting..." 1 DISP
         DO
            "Pass " 'iter' INCR +
            2 DISP
            DUP COLCT
            DUP ROT
         UNTIL
            SAME
         END
      >>
>>
```

Enter the program and store it in COLLECT. Now our set looks like the first picture on the next page.

The next command that we will program, will be TR → EXP for conversion of trigonometric functions to complex exponentials. We use MATCH repeatedly until no more matching can be done. The program listing is on the next page.




```
<< "Converting SIN->EXP...
" 1 DISP
   DO
      { 'SIN(&A)' '(EXP(&A*i)-
        EXP(-(&A*i)))/(2*i)' }
      ↑MATCH
   UNTIL
      NOT
   END
   "Converting COS->EXP...
" 1 DISP
   DO
      { 'COS(&A)' '(EXP(&A*i)+
        EXP(-(&A*i)))/2' }
      ↑MATCH
 UNTIL
      NOT
 END
 "Converting TAN->EXP...
" 1 DISP
  DO
       { 'TAN(&A)' '(EXP(&A*i*2)-1)/
         ((EXP(&A*i*2)+1)*i)' }
       ↑MATCH
  UNTIL
       NOT
  END
>>
```

Store this program in TR → EXP. Try to convert some trigonometric functions to complex exponentials. Enter for example SIN(X) COS(X) and press TR→EXP. The result is:

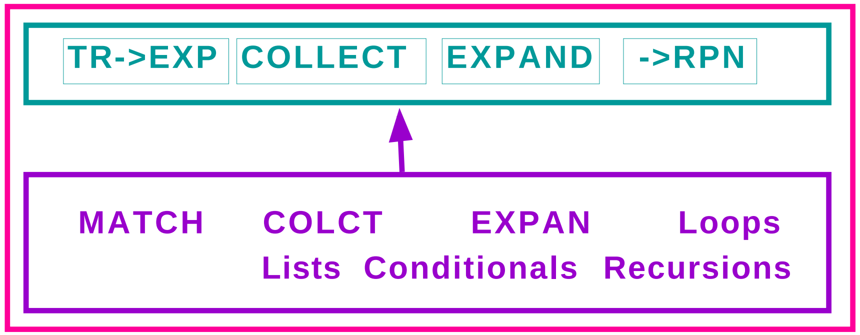
$$\left(\frac{EXP(X\ i)-EXP(-(X\ i))}{2\ i}\right)\left(\frac{EXP(X\ i)+EXP(-(X\ i))}{2}\right)$$

We can apply our EXPAND and then our COLLECT to this result. We then get:

$$-\left(\frac{.25\ EXP(-X)^{2i}}{i}\right)+\frac{.25\ EXP(X)^{2i}}{i}$$

We see that COLLECT doesn't collect the powers of exponentials. This is because the command COLCT doesn't do this. We will see what we can do about it later on.

Our set has now already four commands and looks like:

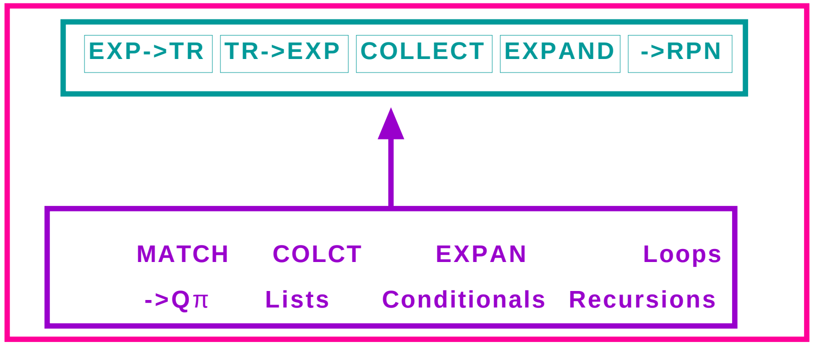


Now that we have TR → EXP we should also have the opposite, a program for transforming exponentials to trigonometric functions. We use the same method as for EXP → TR.

```
<< "Converting EXP->SINCOS...
" 1 DISP
   DO
      { 'EXP(&A)' 'COS(&A/i)+
        SIN(&A/i)*i' }
      ↑MATCH
   UNTIL
      NOT
   END
>>
```

Store this in EXP → TR. Test it by entering EXP(−i X) and press EXP→TR. The result is $\mathrm{COS}\left(\frac{X\ i}{i}\right)+\mathrm{SIN}\left(\frac{X\ i}{i}\right)\ i$. The two last commands don't return their result completely expanded, but this is not what they are made for. You can use EXPAND and COLLECT after EXP → TR or TR → EXP to do that. The commands TR → EXP and EXP → TR will be used for the construction of further commands, and so they are made as elementary as possible.

Our command set now consists of 5 commands:



Let's start now making such commands like TRIGLIN for linearizing products or powers of trigonometric functions. The recipe is mainly to first completely expand the trigonometric functions, so that powers like $\mathrm{SIN}(X)^2$ are converted to SIN(X) SIN(X). Then we can use MATCH repeatedly, to replace such products with their linearized form. When we are done with replacements, we can re-COLLECT everything. In order to also catch TAN(X) we can convert it to $\frac{\mathrm{SIN}(X)}{\mathrm{COS}(X)}$ at the start of the program. Because the need for this conversion is likely to return later, we make an extra small program that does this, so that we can use it a lot in other programs.

```
<< "Converting TAN->SINCOS
" 1 DISP
    DO
        {'TAN(&X)' 'SIN(&X)/COS(&X)'}
        ↑MATCH
  UNTIL
    NOT
  END
>>
```

Stre this program in TAN2SC. Now enter:

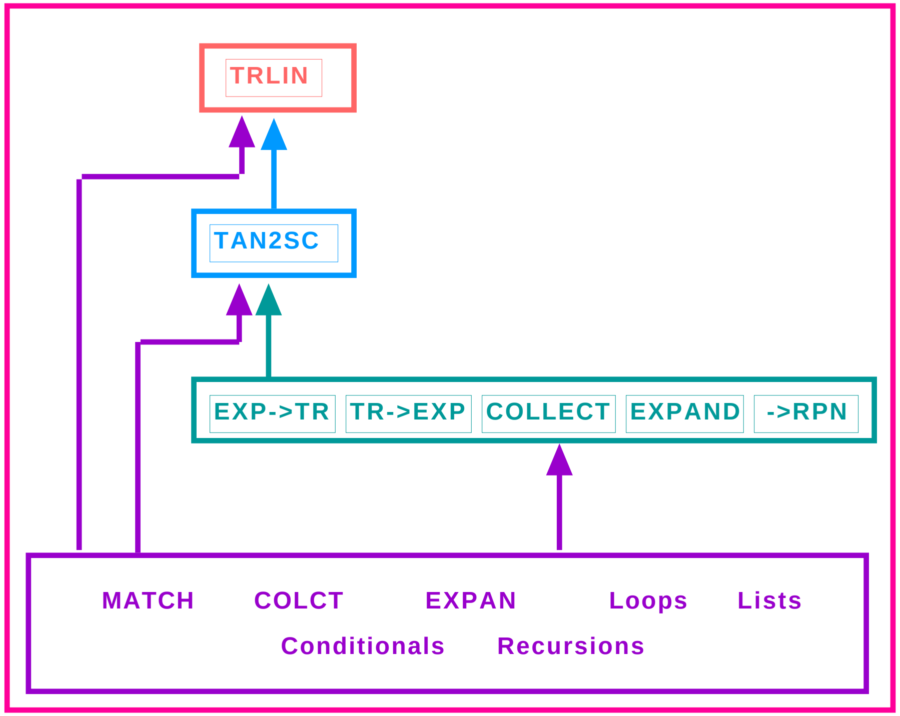
```
<< 0 ->iter
    <<  TAN2SC
        DO
          COLLECT EXPAND
          COLLECT EXPAND
          "Linearizing TRIG...
" 1 DISP
          "PASS " 'iter' INCR +
          2 DISP
          {'SIN(&X)*SIN(&Y)'
           '.5*COS(&X-&Y)-.5*COS(&X+&Y)'}
          ↑MATCH SWAP
          {'COS(&X)*COS(&Y)'
           '.5*COS(&X-&Y)+.5*COS(&X+&Y)'}
```

```
          ↑MATCH ROT OR SWAP
          {'SIN(&X)*COS(&Y)'
           '.5*SIN(&X-&Y)+.5*SIN(&X+&Y)'}
          ↑MATCH ROT OR SWAP
          {'COS(&X)*SIN(&Y)'
           '-.5*SIN(&X-&Y)+.5*SIN(&X+&Y)'}
          ↑MATCH ROT OR SWAP
          {'&A*SIN(&X)*SIN(&Y)'
           '&A*(.5*COS(&X-&Y)-.5*COS(&X+&Y))'}
          ↑MATCH ROT OR SWAP
          {'&A*COS(&X)*COS(&Y)'
           '&A*(.5*COS(&X-&Y)+.5*COS(&X+&Y))'}
          ↑MATCH ROT OR SWAP
          {'&A*SIN(&X)*COS(&Y)'
           '&A*(.5*SIN(&X-&Y)+.5*SIN(&X+&Y))'}
          ↑MATCH ROT OR SWAP
          {'&A*COS(&X)*SIN(&Y)'
           '&A*(-.5*SIN(&X-&Y)+.5*SIN(&X+&Y))'}
          ↑MATCH ROT OR
        UNTIL
          NOT
         END
         COLLECT
      >>
>>
```

Store this in TRLIN. You may wonder why the the sequence COLLECT EXPAND is called twice at the start of the program. Well, even COLLECT and EXPAND don't do their work completely sometimes. I have experimented with them and I I found out that calling them twice is enough. Should you find cases where even calling them twice is not enough, then Nick must think again. ;-)

You also may wonder why we MATCH every pattern twice, like SIN(&X) SIN(&X) *and* &A SIN(&X) SIN(&X), when SIN(&X) SIN(&X) appears in &A SIN(&X) SIN(&X) as a pattern. The answer is that ↑ MATCH doesn't see that. So we have to do the matching explicitly also for &A SIN(&X) SIN(&X). This makes execution time longer, but it covers all (?) cases, so that it is worthwhile to do.

Let's take a look again at our growing command set, and the dependencies of the commands.



Couldn't we do a similar program for linearizing exponentials? Well, yes, we could for example collect all products, quotients and powers of exponentials, but COLLECT doesn't fit here because COLCT can't collect such things. So we need three additional small programs that collect expressions of the form $(e^a)^b$ to $e^{a\ b}$, expressions of the form $\frac{e^a}{e^b}$ to e^{a-b} and expressions of the form $e^a\ \ e^b$ to e^{a+b}. We don't put all this functionality in one program, because we will need *only some but not all* of this functionality in other programs later. So we are going to program each conversion separately. Let's do the first conversion:

```
<< 0 -> iter
  << "Collecting (e^a)^b...
" 1 DISP
      DO
       "PASS " 'iter' INCR
       + 2 DISP
            { 'EXP(&A)^&B'
         'EXP(&A*&B)' }
             ↑MATCH
      UNTIL
             NOT
      END
   >>
>>
```

Store this in EXPPOWCLCT.

Now the second conversion:

```
<< 0 -> iter
  << "e^a/e^b -> e^a*e^-b...
" 1 DISP
    DO
       "PASS " 'iter' INCR
       + 2 DISP
        { 'EXP(&A)/EXP(&B)'
          'EXP(&A)*EXP(-*&B)' }
        ↑MATCH SWAP
        { '&A*EXP(&B)/EXP(&C)'
           '&A*EXP(&B)*EXP(-*&C)' }
        ↑MATCH ROT OR SWAP
        { 'EXP(&B)/(&A*EXP(&C))'
          'EXP(&B)*EXP(-*&C)/&A' }
         ↑MATCH ROT OR SWAP
        { '&A*EXP(&B)/(&C*EXP(&D))'
           '&A*EXP(&B)*EXP(-*&D)/&C' }
          ↑MATCH ROT OR
      UNTIL
          NOT
      END
   >>
>>
```

Store this EXPRAT2PROD. Again we match $\text{EXP}(\&A)\ \text{EXP}(\&B)$) in all possible forms because otherwise the command ↑MATCH wouldn't match a $\text{EXP}(X)\ \text{EXP}(Y)$ to a $\text{EXP}(X+Y)$ using the pattern $\text{EXP}(\&A)\ \text{EXP}(\&B)$.

The next utility program:

```
<< 0 -> iter
  << "Collecting e^a*e^b...
" 1 DISP
      DO
       "PASS " 'iter' INCR
       + 2 DISP
        { 'EXP(&A)*EXP(&B)'
         'EXP(&A+&B)' }
        ↑MATCH
        { '&C*EXP(&A)*EXP(&B)'
         '&C*EXP(&A+&B)' }
        ↑MATCH ROT OR
      UNTIL
         NOT
      END
   >>
>>
```

Store this in EXPPRODCLCT.

We now have enough, to be able to construct a new command that completely linearizes exponentials. Assume that we have some expression with exponentials like for example

$$\mathrm{EXP}(-X)^{i}\ \left(\mathrm{EXP}(X\ 2)-\mathrm{EXP}(X\ i)^{2}\right)$$

We can use EXPAND to completely expand it to

$$\mathrm{EXP}(-X)^{i}\ \left(\mathrm{EXP}(X)\ \mathrm{EXP}(X)\right)-\mathrm{EXP}(-X)^{i}\ \left(\mathrm{EXP}(X)^{i}\ \mathrm{EXP}(X)^{i}\right)$$

Then, using repeatedly EXPPOWCLCT we transform it to

$$-\left(\mathrm{EXP}(-X\ i)\ \mathrm{EXP}(X\ (2\ i))\right)+\mathrm{EXP}(-X\ i)\ \mathrm{EXP}(X\ 2)$$

Next we use EXPPRODCLCT to transform it to

$$\mathrm{EXP}(-X\ i)\ \left(\mathrm{EXP}(X)\ \mathrm{EXP}(X)\right)-\mathrm{EXP}(-X\ i)\ \left(\mathrm{EXP}(X\ i)\ \mathrm{EXP}(X\ i)\right)$$

We then use EXPRAT2PROD but this has no effect since there are no quotients of exponentials. Then comes EXPPRODCLCT which turns the expression to

$$\mathrm{EXP}\left(-(X\ i)+(X+X)\right)-\mathrm{EXP}(-X\ i+(X\ i+X\ i))$$

And finally we use COLLECT again to get the linearized form

$$\mathrm{EXP}\left(-(X\ i)+2\ X\right)-\mathrm{EXP}(X\ i)$$

Programming this function is really easy. We just use already programmed commands:

```
<<
   EXPAND
   DO
     DUP EXPPOWCOLCT EXPRAT2PROD
     EXPPRODCLCT COLLECT DUP ROT
   UNTIL SAME
   END
>>
```

STOre this in EXPLIN.

Now let's try our new programs, TRLIN and EXPLIN.

Enter $\mathrm{SIN}(X)\ \mathrm{COS}(X)$ and press TRLIN. In about 12 seconds you get $.5\ \mathrm{SIN}(2\ X)$. Enter $\mathrm{COS}(X)^2$ and press TRLIN again. In about 14 seconds you get $.5+.5\ \mathrm{COS}(2\ X)$. Enter $\mathrm{SIN}(X)\ \mathrm{COS}(Y)$ and press TRLIN. You get $.5\ \mathrm{SIN}(X-Y)+.5\ \mathrm{SIN}(X+Y)$ in about 16 seconds.

Now enter

$$\frac{\mathrm{EXP}(X)^{2}\ \mathrm{EXP}(2\ Y+X)}{\mathrm{EXP}(X)}$$

Press EXPLIN. The result $\mathrm{EXP}(2\ X+2\ Y)$ is returned in about 27 seconds. But there is still a problem. Enter $\mathrm{SIN}(X)^2$ and press TR→EXP. You get

$$\left(\frac{\mathrm{EXP}(X\ i)-\mathrm{EXP}\left(-(X\ i)\right)}{2\ i}\right)^{2}$$

Now press EXPLIN to linearize this. The program needs about 98 seconds to return the expression

$$.25\ \mathrm{EXP}\left(-(2\ X\ i)\right)\ i^{-2}+.25\ \mathrm{EXP}\left(2\ X\ i\right)\ i^{-2}-.5\ i^{-2}$$

As you can see the program COLLECT, didn't collect i^{-2} to -1. This is because the underlying COLCT can't do this. We need an additional utility that does this. How can we program this? We can replace i^n with its numeric equivalent $(0,1)^n$ and then evaluate the expression. Powers of the imaginary unit i, are then evaluated to their numeric equivalents. Then the built-in command ->Qπ can be used to turn the complex numbers to symbolics again, and COLLECT can be used to simplify things.

Enter the program:

```
<< 0 ->iter
   << "Collecting i^n...
" 1 DISP
      DO
        "Pass " 'iter' INCR +
        2 DISP { 'i^&n' '(0,1)^&n' }
        ↑MATCH
      UNTIL
         NOT
      END
      EVAL ->Qπ
      0 'iter' STO
      "Converting (1,0)->1...
" 1 DISP
      DO
        "Pass " 'iter' INCR +
        2 DISP { (1,0) 1 } ↑MATCH
      UNTIL
         NOT
      END
      0 'iter' STO
      "Converting (0,1)->i...
" 1 DISP
      DO
        "Pass " 'iter' INCR +
        2 DISP { (1,0) 1 } ↑MATCH
      UNTIL
         NOT
      END
    >>
>>
```

and store it in iCLCT. (Its iMac, iSearch, iRon time, so why not iCLCT? ;-)) But why do we have to MATCH also (1,0) to 1 and (0,1) to i? The (unexpected) answer is that ->Qπ will turn for example (0,1) to −i, but not (1,0) to 1 and not (0,1) to i. So for these two cases we must do that explicitly in our program. Now edit EXPLIN. At the end of the program add iCLCT and COLLECT, so that EXPLIN now is:

```
<<
      EXPAND
      DO
         DUP
         EXPPOWCOLCT
         EXPRAT2PROD
         EXPPRODCLCT
         COLLECT DUP ROT
    UNTIL
         SAME
    END
    iCLCT COLLECT
>>
```

Let's try the last example again. Enter $SIN(X)^2$ and press TR→EXP. You get

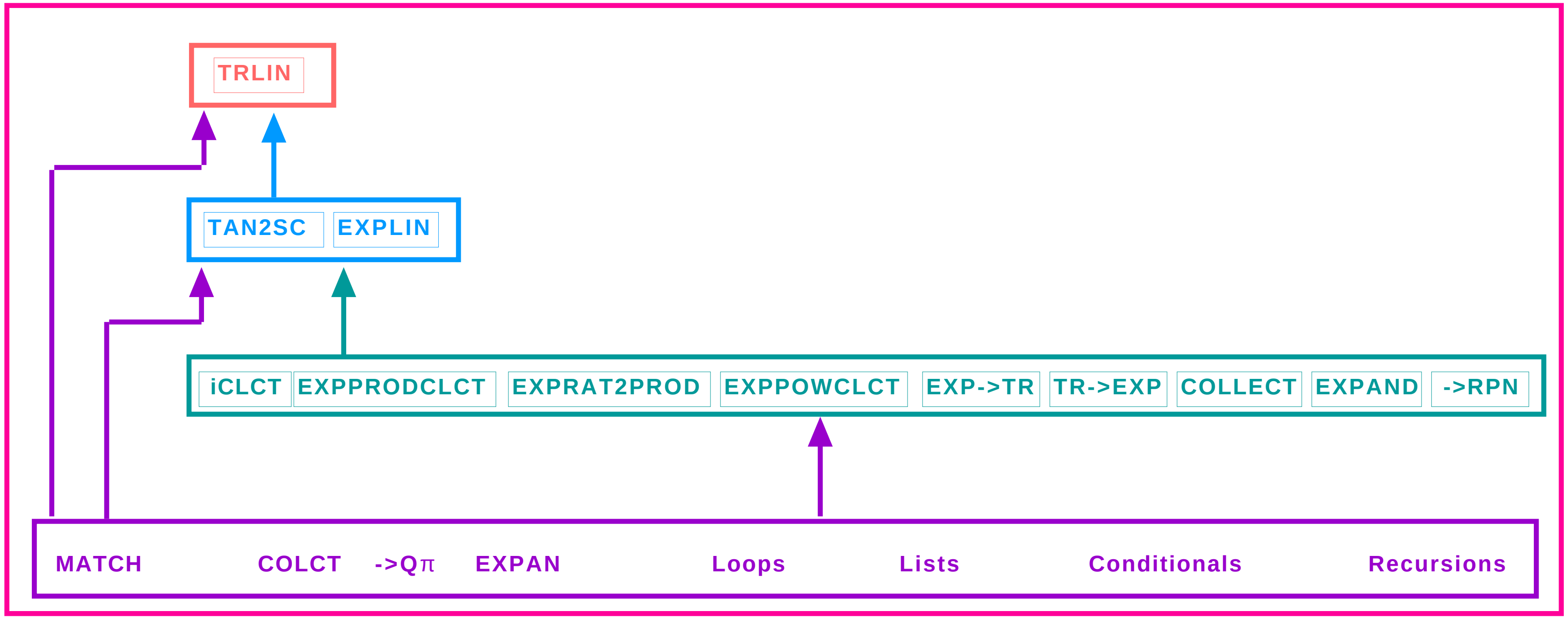
$$\left(\frac{EXP(X\ i)-EXP(-(X\ i))}{2\ i}\right)^2$$

Press EXPLIN. The result

$$.5-.25\ EXP(-(2\ X\ i))-.25\ EXP(2\ X\ i)$$

is shown in about 114 seconds.

Let's take a look again at our command set which has grown taller and wider. (Picture in next page.)



Well, now comes TREXPAND, which proves to be difficult to achieve with MATCH in loop. If someone manages to do that, then please, please post it!

But that doesn't mean that it can't be done with other means. We already have TR → EXP which converts trigonometric functions to exponentials. We can use it, then EXPAND the resulting expression and use EXPPOWCLCT, EXPRAT2PROD and EXP → TR again to turn the exponentials to products. The first version of TREXPAND looks like:

```
<<
   EXPAND COLLECT
   EXPAND COLLECT
   TR->EXP EXPAND
   EXPPOWCLCT
EXPRAT2PROD
   EXP->TR
   COLLECT EXPAND
   COLLECT
>>
```

The double execution of EXPAND COLLECT at the start of the program is because of the same reasons as on page 11-11. At the end of the program we use the sequence COLLECT EXPAND COLLECT because after EXP → TR has finished, the arguments of the trigonometric functions are sometimes complicated. We use COLLECT them to simplify them, making the following EXPAND somehow easier for the HP48. We then COLLECT to let some terms vanish.

But this version has a problem. It will not expand things like SIN(2 X) because none of the used commands can expand EXP(2 X) to EXP(X + X). We must add code that does this. To do this, we must check if some arguments of the trigonometric functions are of the form n X, where n is an integer. If we find such an argument we can replace it with the sum X + X + … + X and then repeat the procedure of trigonometric expansion. We can be sure that after the first part of the program has finished (this is the program TREXPAND as it is now), no sums will appear as arguments to trigonometric functions, because sums will be already (trigonometrically) expanded.

First we need a function that turns expressions of the form n X to X + X + … + X.

```
<<
  EXPAND COLLECT
  EXPAND COLLECT
  "Converting ALG->RPN...
" 1 DISP
  ->RPN
  -> alglst
  <<
    alglst 1
    <<
      "Searching TRIG. args...
Object " NSUB 1 DISP
      IF
        {SIN COS TAN}
        OVER POS
      THEN
        SWAP PROD->SUM
        SWAP
      END
      EVAL
    >>
    DOSUBS EVAL
  >>
>>
```

Stire this in FINDTRIGARGS. (I can't think of a better name.) The program uses → RPN to turn the algebraic to a RPN list. The following DOSUBS procedure checks if the next object is a trigonometric function SIN, COS or TAN. If it isn't then it simply evaluates the object to build the algebraic step by step again. If it is, then it uses the function PROD → SUM (which we didn't write yet) to turn arguments of trigonometric functions that are of the form n X to sums and then evaluates the object to build the algebraic with the altered arguments of trigonometric functions.

Now we need PROD → SUM, a function for conversion of n X to X + X + … + X, where n is an integer.

```
<<
  "Checking TRIG. arg...
" 1 DISP
  IF DUP TYPE 9 ==
  THEN
    DUP ->RPN DUP HEAD OVER 3 GET
    3 PICK 2 GET 4 ROLL 4 OVER SIZE SUB +
    -> oldarg factor oper rest
    <<
      IF factor TYPE NOT
      THEN
        IF
          factor FP NOT { * } oper POS AND
        THEN
          rest EVAL 'rest' STO rest 1 factor 1 -
          START
            rest +
          NEXT
        ELSE
          oldarg
        END
      ELSE
        oldarg
      END
    >>
  END
>>
```

Store this in PROD → SUM. Perhaps you would like to use DBUG to see how it works. ;-)

So the program TREXPAND now becomes:

```
<<
   EXPAND COLLECT
   EXPAND COLLECT
   TR->EXP EXPAND
   EXPPOWCLCT
   EXPRAT2PROD
   EXP->TR
   COLLECT EXPAND
   COLLECT
   IF
      DUP FINDTRIGARG
      DUP ROT SAME NOT
   THEN
      TR->EXP EXPAND
      EXPPOWCLCT EXPRAT2PROD
      COLLECT EXP->TR COLLECT
      iCLCT COLLECT EXPAND
      COLLECT
   END
>>
```

Store this in TREXPAND again. Let's try it. Enter $\mathrm{SIN}(X+Y)$ and press TREXPAND. After 76 seconds you get the result $\mathrm{COS}(X)\ \mathrm{SIN}(Y)+\mathrm{SIN}(X)\ \mathrm{COS}(Y)$. Press now TRLIN. After 17 seconds you get $\mathrm{SIN}(X+Y)$, the expression with which you started. Another example: Enter $\mathrm{COS}(X)^2$ and press TRLIN. In 14 seconds the HP48 returns $.5+.5\ \mathrm{COS}(2\ X)$. Now press TREXPAND. It takes 98 seconds for the HP48 to show $.5+.5\ \mathrm{COS}(X)^2-.5\ \mathrm{SIN}(X)^2$. As you can see, the execution time of TREXPAND is very long. compared to the execution time of TRLIN. This is because TRLIN only works with many subsequent MATCHes, while TREXPAND uses very often EXPAN to fully expand products of the intermediately produced exponentials. If somebody has an algorithm, that makes TREXPAND faster, then please post it to cure the slow TREXPAND.

Our command set is now pretty large and the interdependencies of the commands are more complex. (Picture on next page.)

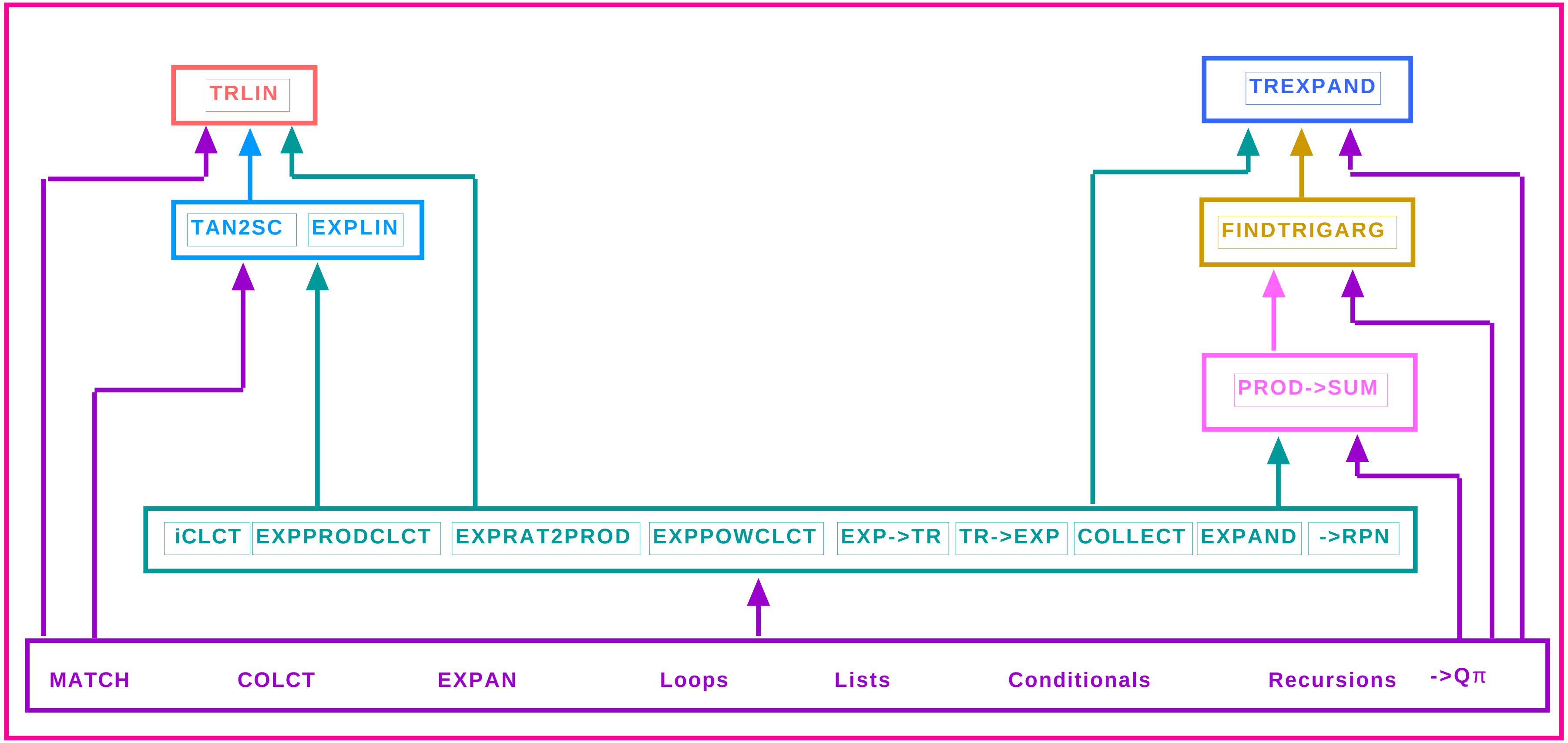
Let's now move on to the next thing to do, a program that converts SIN, COS and TAN functions to TAN functions of the half argument. This seems easy to do using MATCH, but there is something that we must take care of. If we use first match SIN and COS then the resulting TAN functions along with the TAN functions that were in our algebraic object right from the start, will be all matched again to TAN functions. If for example we start with $\mathrm{SIN}(X)+\mathrm{TAN}(X)$ and we match first $\mathrm{SIN}(X)$ to

$$\frac{2\ \mathrm{TAN}\left(\frac{X}{2}\right)}{\mathrm{TAN}\left(\frac{X}{2}\right)^2+1}$$

then the resulting expression will be: $\dfrac{2\ \mathrm{TAN}\left(\frac{X}{2}\right)}{\mathrm{TAN}\left(\frac{X}{2}\right)^2+1}+\mathrm{TAN}(X)$

If we match now $\mathrm{TAN}(X)$, then we will have:

$$\frac{-\dfrac{4\ \mathrm{TAN}\left(\frac{X}{4}\right)}{\mathrm{TAN}\left(\frac{X}{4}\right)^2-1}}{\left(\dfrac{2\ \mathrm{TAN}\left(\frac{X}{4}\right)}{\mathrm{TAN}\left(\frac{X}{4}\right)^2-1}\right)^2+1}-\frac{2\ \mathrm{TAN}\left(\frac{X}{2}\right)}{\mathrm{TAN}\left(\frac{X}{2}\right)^2-1}$$



which means that we converted to TAN functions of the half *and* of the quarter argument. To avoid this we must first match TAN functions and then SIN and COS functions.

```
<<
   EXPAND COLLECT EXPAND COLLECT
   "TAN(X)->TAN(X/")
" 1 DISP
   { 'TAN(&X)' '2*TAN(&X/2)/(1-TAN(&X/2)^2)'}
   ↑MATCH DROP
   "SIN(X)->TAN(X/")
" 1 DISP
   { 'SIN(&X)' '2*TAN(&X/2)/(TAN(&X/2)^2+1)'}
   ↑MATCH DROP
   "COS(X)->TAN(X/")
" 1 DISP { 'COS(&X)'
     '(1-TAN(&X/2)^2)/(TAN(&X/2)^2+1)'}
   ↑MATCH DROP
   COLLECT
>>
```

Store that in HALFTAN. Try it with COS(X) TAN(X). The result is

$$\frac{2}{1+TAN(.5\ X)^2}\ TAN(.5\ X)$$

which comes in about 7 seconds.

We can also program TRIGSIN and TRIGCOS, for conversion of $COS(X)^2$ to $1-SIN(X)^2$ and of $SIN(X)^2$ to $1-COS(X)^2$.

Enter

```
<<
  { 'COS(&X)^2' '1-SIN(&X)^2 }
  ↑MATCH DROP COLLECT
>>
```

and Store it in TRIGSIN.

Enter

```
<<
  { 'SIN(&X)^2' '1-COS(&X)^2 }
  ↑MATCH DROP COLLECT
>>
```

and Store it in TRIGSIN.

Programming TRIGTAN for such conversions like for example $\frac{SIN(X)}{COS(X)}$ to TAN(X) is a bit more complicated. If would be really tough to search one algebraic expression for occurrences of $\frac{SIN(X)}{COS(X)}$, because such patterns could be "hidden for the eye of MATCH". Consider for example

$$\frac{\frac{SIN(X)+COS(X)}{2}+SIN(X)}{COS(X)}$$

Matching here $\frac{SIN(X)}{COS(X)}$ to TAN(X) would leave the expression unchanged. If we first expand, then we get:

$$\frac{\frac{SIN(X)}{2}}{COS(X)}+\frac{\frac{COS(X)}{2}}{COS(X)}+\frac{SIN(X)}{COS(X)}$$

If we collect this we get:

$$\frac{1.5}{COS(X)}\ SIN(X)+.5$$

We could of course sit down and experiment, in order to find *all* possible patterns which are equivalent to $\frac{SIN(X)}{COS(X)}$ but as Nick is a rather lazy person, he found an easier but also dirtier method. If there is a SIN function in the algebraic expression, then match SIN(X) to COS(X) TAN(X). If there is no SIN, then match COS(X) to $\frac{SIN(X)}{TAN(X)}$. We use → RPN to turn the algebraic in an RPN list, and check if the list contains SIN.

```
<<
  DUP ->RPN
  IF
    { SIN } HEAD POS
  THEN
    { ' SIN(&X)' 'COS(&X)*TAN(&X)' }
    ↑MATCH
```

```
   ELSE
      { ' COS(&X)' 'SIN(&X)/TAN(&X)' }
      ↑MATCH
   END
   DROP EXPAND COLLECT
>>
```

Store the program in TRIGTAN . To try it enter the expression

$$\frac{\frac{SIN(X)+COS(X)}{2}+SIN(X)}{COS(X)}$$

and press TRIGTAN.

The result is $.5+1.5\ TAN(X)$.

Enter also

$$\frac{SIN(X)}{COS(X)^{2}}$$

Press TRIGTAN once. You get $INV(COS(X))\ TAN(X)$. Now press TRIGTAN again. This time you get

$$\frac{TAN(X)^{2}}{SIN(X)}$$

because there was no SIN function in the expression. This is the result of the design of the function. Both results are equivalent, but you should always use TRIGTAN twice to check if one of the possible results fits your needs better.

Last thing we have to do is TAN2SC2.

```
<<
   { 'TAN(&X)' 'SIN(2*&X)/(1+COS(2*&X))' }
   ↑MATCH DROP COLLECT
>>
```

STtore this in TAN2SC2.

Now turn page to take a look at our command set in all its beauty.

We see that the HP48 is not so weak, regarding symbolic mathematics. Actually the set of UserRPL commands available to the "normal" user is mighty enough, to allow to program many manipulations of symbolics. It is the execution time of such programs that makes the HP48 looking less powerful, comparing it to the HP49G. The fact that we have such a language like UserRPL, that allows making such programs, but a processor that needs such a long time to execute these programs, should make some things clear. The theoretical concept of UserRPL is fantastic! It is nearly complete, at least complete enough to let almost anything be possible. If it only were put on faster hardware! It reminds me somehow of the theoretical predictions of physicists that were experimentally proven much later, when the needed hardware was available. Imagine what the HP48 could be used for, if it only could run such programs in shorter time.

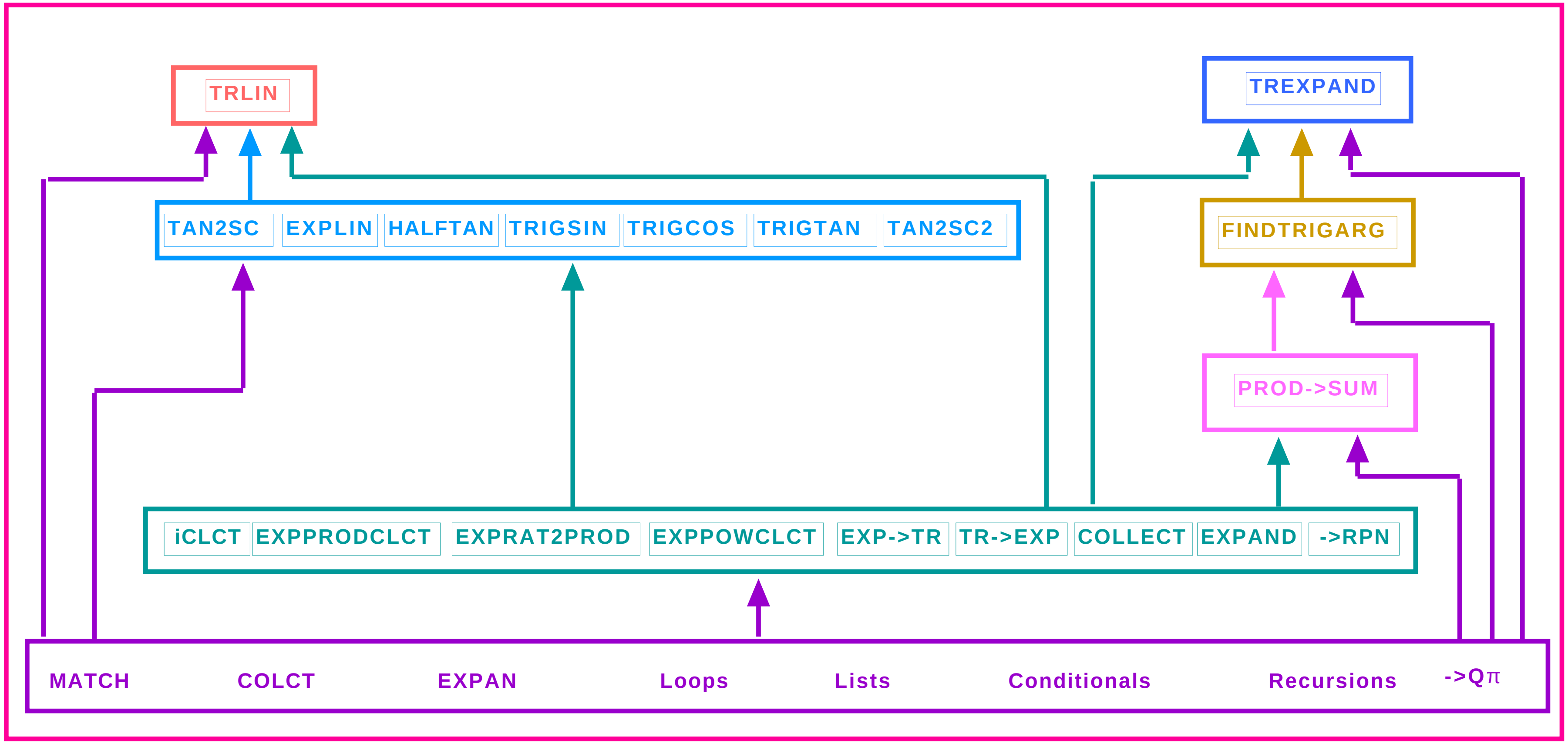
The whole set of commands occupies about 4.8 KBytes which shows, that UserRPL is also compact.

Now let's try our commands in some examples that we already have done with the HP49G and its built in CAS.

1) Show that:

$$SIN(X)^{4}-COS(X)^{4}=SIN(X)^{2}-COS(X)^{2}$$

Enter the left hand side of the equation. We have powers of trigonometric functions, so let's use TRLIN. We get:



$$-\text{COS}(2\ X)$$

in 108 seconds. Now we have a trigonometric function with a multiple of X as argument, so let's try TREXPAND. In 91 seconds the HP48 returns:

$$-\text{COS}(X)^2+\text{SIN}(X)^2$$

2) Show that:

$$\text{SIN}(X)^4-\text{COS}(X)^4=1-2\ \ \text{COS}(X)^2=2\ \ \text{SIN}(X)^2-1$$

Use example 1 to turn the expression $\text{SIN}(X)^4-\text{COS}(X)^4$ to $-\text{COS}(X)^2+\text{SIN}(X)^2$. Now press TRIGCOS. In about 2 seconds you get:

$1-2\ COS(X)^2$

Now press TRIGSIN. In about 1 second you get:

$1-2\ \left(1-SIN(X)^2\right)$

Press EXPAND and the COLLECT to simplify this to:

$-1+2\ SIN(X)^2$

---

3) Show that:

$$\left(SIN(X)+COS(X)\right)^2+\left(SIN(X)-COS(X)\right)^2=2$$

Enter the left hand side of the equation on stack level 1 and press TRLIN. In about 31 seconds you get a nice round 2.

---

4) Simplify the expression:

$$SIN\left(X+\frac{\pi}{2}\right)+SIN\left(X-\frac{\pi}{2}\right)$$

Enter expression and press TREXPAND. In about 230 seconds you get:

$2\ COS(.5\ \pi)\ SIN(X)$

Now, it would be nice if the built-in commands could simplify $COS(.5\ \pi)$ to 0, but they don't. Even if you use ->Qπ, you get

$2\ COS\left(\frac{1}{2}\ \pi\right)\ SIN(X)$, which can't be simplified to 0.

---

4) Simplify the expression:

$SIN(X-Y)\ COS(X+Y)+COS(X-Y)\ SIN(X+Y)$

Enter expression and press TRLIN. In about 28 seconds you get:

$SIN(2\ X)$

---

5) Turn $SIN(X+Y+Z)$ to a sum of products of trigonometric functions.

Enter $SIN(X+Y+Z)$ and press TREXPAND. The HP48 needs 224 seconds to return:

$COS(X)\ COS(Y)\ SIN(Z)+COS(X)\ SIN(Y)\ COS(Z)+$
$SIN(X)\ COS(Y)\ COS(Z)-SIN(X)\ SIN(Y)\ SIN(Z)$

---

6) Show that:

$$SIN(X+Y)\ SIN(X-Y)=SIN(X)^2-SIN(Y)^2=COS(Y)^2-COS(X)^2$$

Enter $SIN(X+Y)\ SIN(X-Y)$, press TRLIN and you get

$-(.5\ COS(2\ X))+.5\ COS(2\ Y)$

in 26 seconds. Now press TREXPAND. In 230 seconds the HP48 returns:

$$-(.5\ \mathrm{COS(X)}^2)+.5\ \mathrm{SIN(X)}^2+.5\ \mathrm{COS(Y)}^2-.5\ \mathrm{SIN(Y)}^2$$

Press TRIGCOS. In 6 seconds you get:

$$.5\ (1-\mathrm{COS(X)}^2)-.5\ (1-\mathrm{COS(Y)}^2)-.5\ \mathrm{COS(X)}^2+.5\ \mathrm{COS(Y)}^2$$

Press EXPAND and then COLLECT to get:

$$-\mathrm{COS(X)}^2+\mathrm{COS(Y)}^2$$

Press TRIGSIN now and you get:

$$\mathrm{SIN(X)}^2-\mathrm{SIN(Y)}^2$$

This example shows that adding EXPAND COLLECT instead of only COLLECT at the end of TRIGCOS and TRIGSIN seems to be a good idea.

---

7) Show that:

$$\frac{2\ \mathrm{SIN(X+Y)}}{\mathrm{COS(X+Y)+COS(X-Y)}}=\mathrm{TAN(X)+TAN(Y)}$$

Enter the numerator $2\ \mathrm{SIN(X+Y)}$ of left hand side of the equation. Press TREXPAND to expand the trigonometric function of $X+Y$ to trigonometric functions of X and Y. The HP48 needs 71 seconds to return:

$$2\ \mathrm{COS(X)}\ \mathrm{SIN(Y)}+2\ \mathrm{SIN(X)}\ \mathrm{COS(Y)}$$

Now enter the denominator $\mathrm{COS(X+Y)+COS(X-Y)}$ of left hand side of the equation and press TREXPAND again, to get

$$2\ \mathrm{COS(X)}\ \mathrm{COS(Y)}$$

in 118 seconds. Press ÷, EXPAND and then COLLECT. Now you have:

$$\mathrm{INV(COS(X))}\ \mathrm{SIN(X)}+\mathrm{INV(COS(Y))}\ \mathrm{SIN(Y)}$$

Press now TRIGTAN to get $\mathrm{TAN(X)+TAN(Y)}$.

---

8) Show that:

$$\mathrm{SIN(X)}=2\ \mathrm{SIN}\left(\frac{X}{2}\right)\ \mathrm{COS}\left(\frac{X}{2}\right)$$

Enter $\mathrm{SIN(X)}$. Since we want to transform this to trigonometric functions of $\frac{X}{2}$, we use HALFTAN first. The result is:

$$\frac{2}{1+\mathrm{TAN}(.5\ X)^2}\ \mathrm{TAN}(.5\ X)$$

Let's convert all TAN functions to SIN and COS. We use TAN2SC for this. The result is:

$$\frac{2}{1+\left(\frac{\mathrm{SIN}(.5\ X)}{\mathrm{COS}(.5\ X)}\right)^2}\ \frac{\mathrm{SIN}(.5\ X)}{\mathrm{COS}(.5\ X)}$$

Let's EXPAND and COLLECT this. We get:

$$\frac{\mathrm{INV}\left(.5+.5\ \ \mathrm{COS}(.5\ \ X)^{-2}\ \ \mathrm{SIN}(.5\ \ X)^{2}\right)}{\mathrm{COS}(.5\ \ X)}\ \mathrm{SIN}(.5\ \ X)$$

Now we use TRIGCOS to convert $\mathrm{SIN}(.5\ \ X)^2$ to $1-\mathrm{COS}(.5\ \ X)^2$.
The expression now is:

$$\frac{\mathrm{INV}\left(.5+.5\ \left(1-\mathrm{COS}(.5\ \ X)^{2}\right)\ \mathrm{COS}(.5\ \ X)^{-2}\right)}{\mathrm{COS}(.5\ \ X)}\ \mathrm{SIN}(.5\ \ X)$$

EXPAND and COLLECT this again to get the result:

$$2\ \ \mathrm{COS}(.5\ \ X)\ \ \mathrm{SIN}(.5\ \ X)$$

Finishing this last part, I must say again that the programs here are way far from being perfect. I only wanted to wake your appetite for using your HP48. Change the programs, add new ones, think of better algorithms, do what you think is best for your needs. And post your ideas so that we can join you in the continuing trigonometry marathon.

RPL-Greetings,
Nick.