Forth for Education
- 4E4th and 4E4th IDE

Dirk Bruehl Dirk@4E4th.eu http://www.4E4th.com

First Steps with 4E4th

Installing 4E4th on the MSP430-LaunchPad
and on MicroBox using LP or the 6 wires.
Documentation slightly adapted for MicroBox

Here our twelve step program to get 4E4th onto a brand-new MSP430-LaunchPad with an MSP430G2553 chip:

Requirements:

A LaunchPad with **MSP430G2553** Microprocessor and **4E4th-IDE** on Windows XP.

Here it starts:

1. Start Windows XP PC and 4E4th-IDE for Windows XP..
2. Click on 4E4th-IDE Menu
   LaunchPad / Preparing LaunchPad Connection.
3. Unpack MSP430-LaunchPad: cut foil, take out circuit board and take out USB-Cable hidden under cardboard cover.
4. Check HW | UART position of RX/TX-Jumper,
   turn by 90° if necessary.
5. Connect USB-Cable out of MSP430-LaunchPad box
   to MSP430-LaunchPad PCB.
6. Connect USB-Cable to PC - green Power-LED lights,
   red and green LEDs flash.
7. MSP430 Application UART Hardware Wizard appears on screen:

   If not already set, click on "*Install Software automatically*",

   Click on button "*Next*>" - it will need some time until Software is found.

8. A message will come up telling the MSP430 Application UART has not passed Windows Logo testing to verify its compatibility with Windows XP.
   Click on button "*Continue Anyway*".
   Installation will need some seconds.
   Click on button "*Finish*".

9. Some messages are coming, telling that hardware is ready for use. Proceed with 4E4th-IDE.

10. Click on 4E4th-IDE Menu
    LaunchPad / FLASH the LaunchPad / 4e4th.a43

11. Popup *** All your Forth Words will be lost ***
    Proceed with FLASHING the 4e4th?

    Click on button "*Yes*".

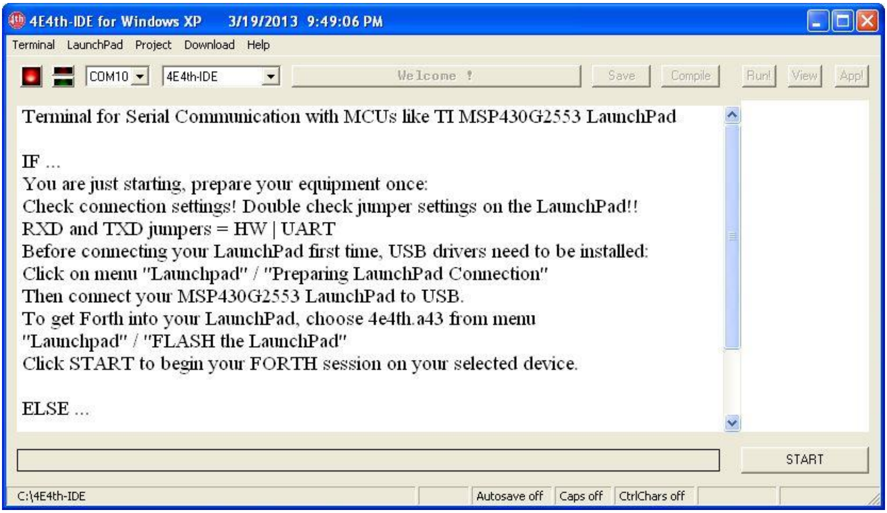
12. Flashing needs several seconds, when ready,
    message "*Flashing 4e4th done -*
    checking connections now" shows up.
    If everything is okay, the message
    "4e4thFLASH connections checked !
    Click 'Start' button to connect to target !"
    shows up.

Now 4E4th is installed on the LaunchPad!

The red and the green LED are lit as long as the LaunchPad is connected to the PC and no program switched off an LED.

For better explanation the same points again, but now with additional pictures:

1. Start the 4E4th-IDE:,

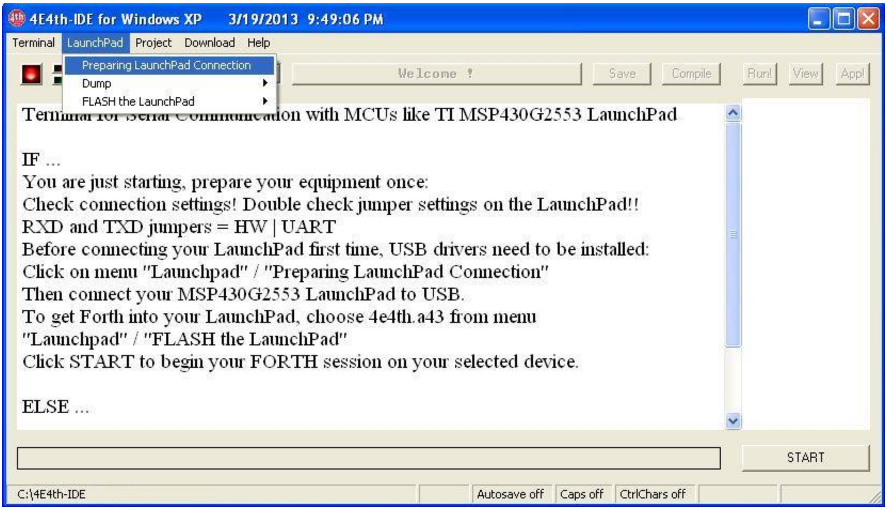


Terminal for Serial Communication with MCUs like TI MSP430G2553 LaunchPad

IF ...
You are just starting, prepare your equipment once:
Check connection settings! Double check jumper settings on the LaunchPad!!
RXD and TXD jumpers = HW | UART
Before connecting your LaunchPad first time, USB drivers need to be installed:
Click on menu "Launchpad" / "Preparing LaunchPad Connection"
Then connect your MSP430G2553 LaunchPad to USB.
To get Forth into your LaunchPad, choose 4e4th.a43 from menu
"Launchpad" / "FLASH the LaunchPad"
Click START to begin your FORTH session on your selected device.

ELSE ...

2. Click on Menu LaunchPad / Preparing LaunchPad Connection.

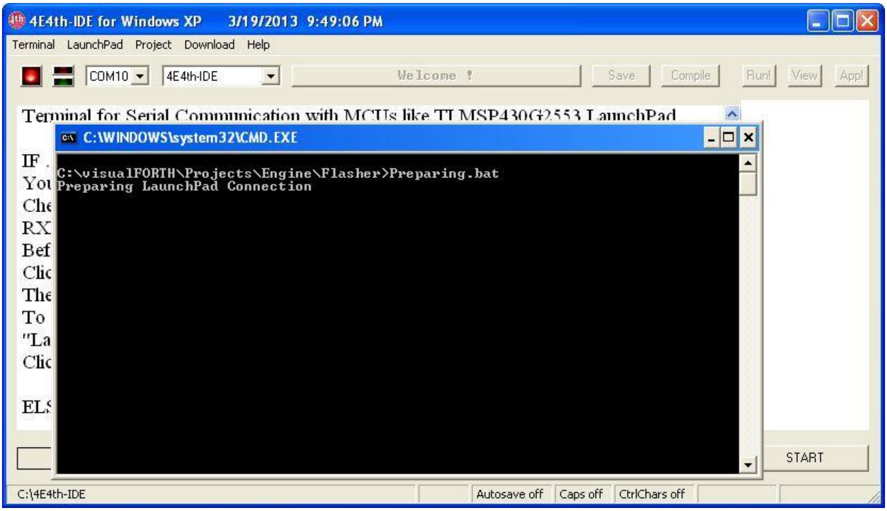


Terminal for Serial Communication with MCUs like TI MSP430G2553 LaunchPad

IF ...
You are just starting, prepare your equipment once:
Check connection settings! Double check jumper settings on the LaunchPad!!
RXD and TXD jumpers = HW | UART
Before connecting your LaunchPad first time, USB drivers need to be installed:
Click on menu "Launchpad" / "Preparing LaunchPad Connection"
Then connect your MSP430G2553 LaunchPad to USB.
To get Forth into your LaunchPad, choose 4e4th.a43 from menu
"Launchpad" / "FLASH the LaunchPad"
Click START to begin your FORTH session on your selected device.

ELSE ...

A batch-program does this fully automatic:

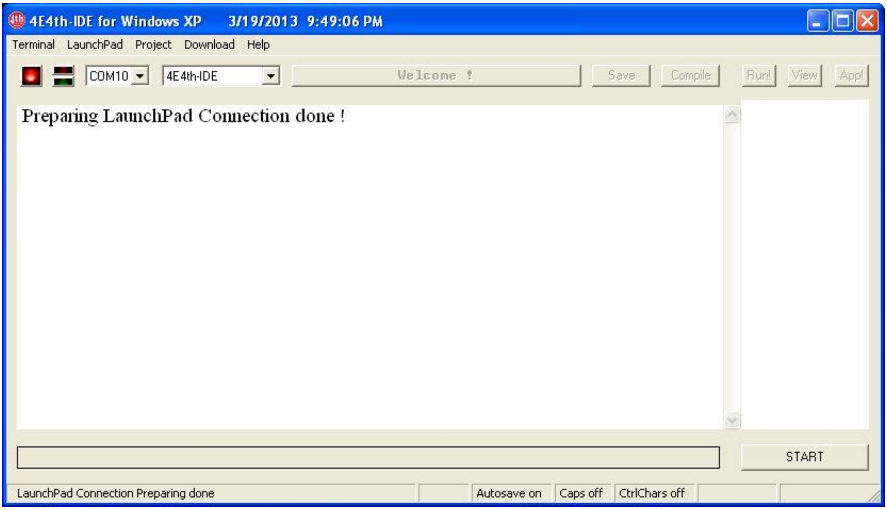


Terminal for Serial Communication with MCUs like TI MSP430G2553 LaunchPad

C:\WINDOWS\system32\CMD.EXE

```
C:\visualFORTH\Projects\Engine\Flasher>Preparing.bat
Preparing LaunchPad Connection
```

When the batch-program has finished, this message comes up:


4E4th-IDE for Windows XP 3/19/2013 9:49:06 PM
Terminal LaunchPad Project Download Help
COM10
4E4th-IDE
Welcome !
Save
Compile
Run!
View
Appl
START
Autosave on
Caps off
CtrlChars off

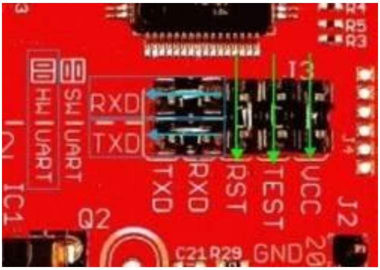
Preparing LaunchPad Connection done !

LaunchPad Connection Preparing done

3. Unpack MSP430-LaunchPad: cut foil, take out circuit board and take out USB-Cable hidden under cardboard cover.

4. Check all of the jumpers: the HW UART position of the RX/TX-Jumpers might have to be changed, turned by 90° if necessary.

On delivery you will see that all jumpers from VCC- and TEST-Jumpers to RXD and TXD are all vertically placed on the LaunchPad.

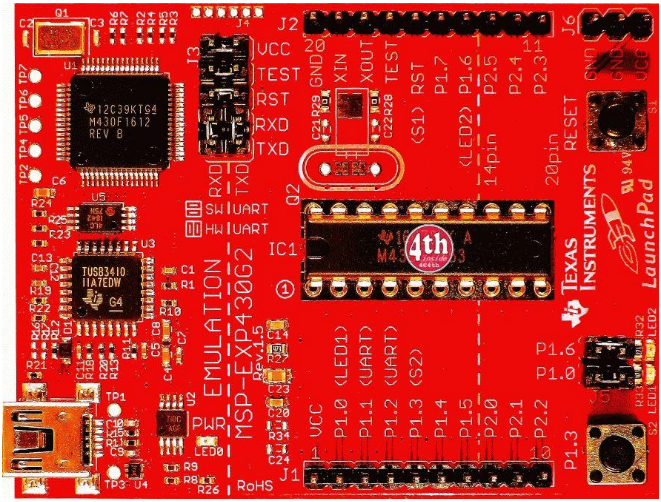


Please change the RXD- and TXD-Jumpers;

these 2 on the left have to be horizontal.

Here see the right jumper setting for the HW UART used by 4e4th:

In this picture on bottom left you see the USB connector:



There the USB-Cable has to be connected first.

5. Connect the USB-Cable out of the LaunchPad box to the MSP430-LaunchPad.
6. Connect USB-Cable to the PC - green Power-LED lights, red and green LEDs flash.
7. MSP430 Application UART Hardware Wizard appears on screen: If not already set, click on "*Install Software automatically*",

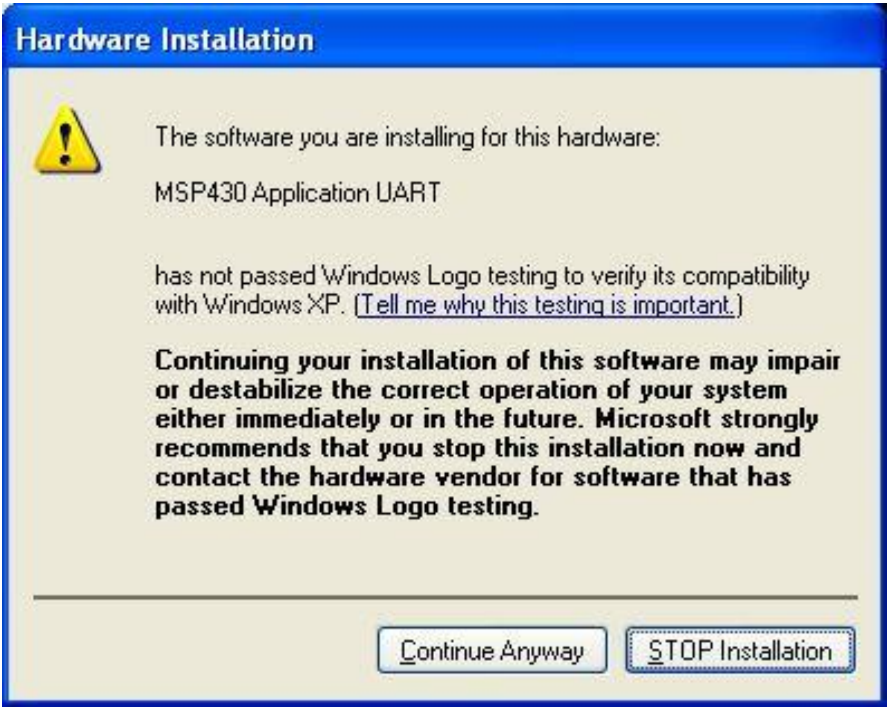
   Click on button "*Next>*"

- it will need some time till Software is found.

8. A message will probably come up telling that the MSP430 Application UART has not passed Windows Logo testing to verify its compatibility with Windows XP.

   Click on button "*Continue Anyway*". The Installation will need some seconds.

   "Found New Hardware Wizard" finally shows
   "Completing the Found New Hardware Wizard"
   - click on button "*Finish*".



Hardware Installation

The software you are installing for this hardware:

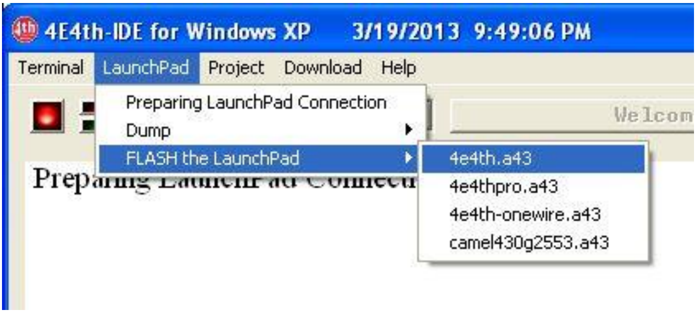
MSP430 Application UART

has not passed Windows Logo testing to verify its compatibility with Windows XP. (Tell me why this testing is important.)

Continuing your installation of this software may impair or destabilize the correct operation of your system either immediately or in the future. Microsoft strongly recommends that you stop this installation now and contact the hardware vendor for software that has passed Windows Logo testing.

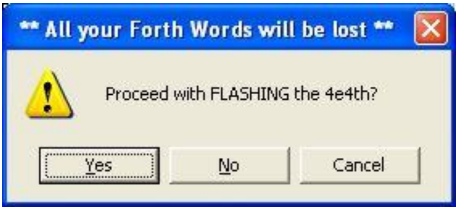
9. Some messages are coming, telling that hardware is ready for use. Proceed with 4E4th-IDE.

10. Click on Menu *LaunchPad / FLASH the LaunchPad / 4e4th.a43*



11. Popup ** *All your Forth Words will be lost*
** *Proceed with FLASHING the 4e4th?*

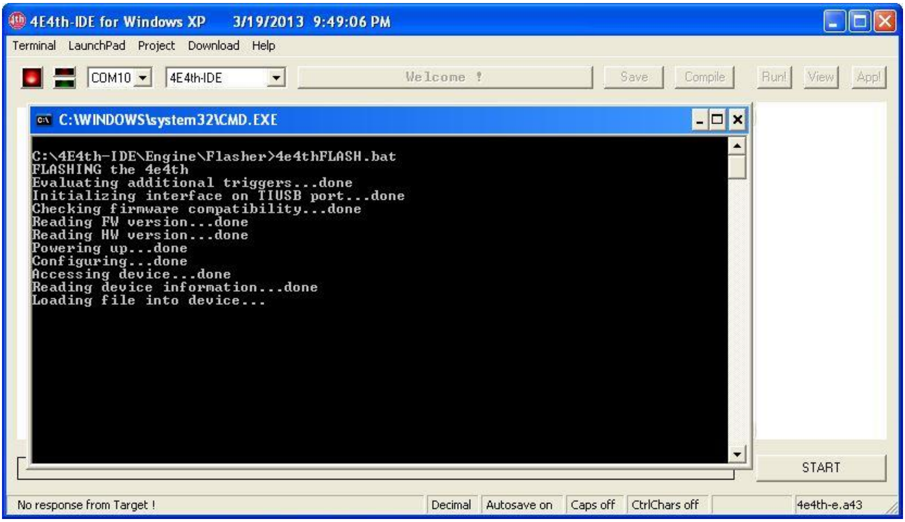
Click on button "*Yes*":



** All your Forth Words will be lost **

Proceed with FLASHING the 4e4th?

12. Flashing of the chip needs several seconds:



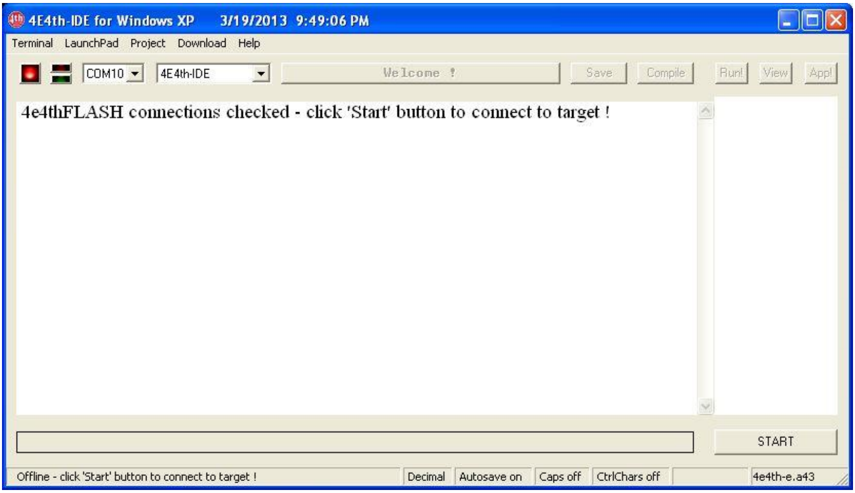
```
C:\4E4th-IDE\Engine\Flasher>4e4thFLASH.bat
FLASHING the 4e4th
Evaluating additional triggers...done
Initializing interface on TIUSB port...done
Checking firmware compatibility...done
Reading FW version...done
Reading HW version...done
Powering up...done
Configuring...done
Accessing device...done
Reading device information...done
Loading file into device...
```

No response from Target !

When ready, the message
"*Flashing 4e4th done - checking connections now*" shows.

If everything is okay, this message follows:

"*4e4thFLASH connections checked*
- click 'Start' button to connect to target !"



4e4thFLASH connections checked - click 'Start' button to connect to target !

Offline - click 'Start' button to connect to target !

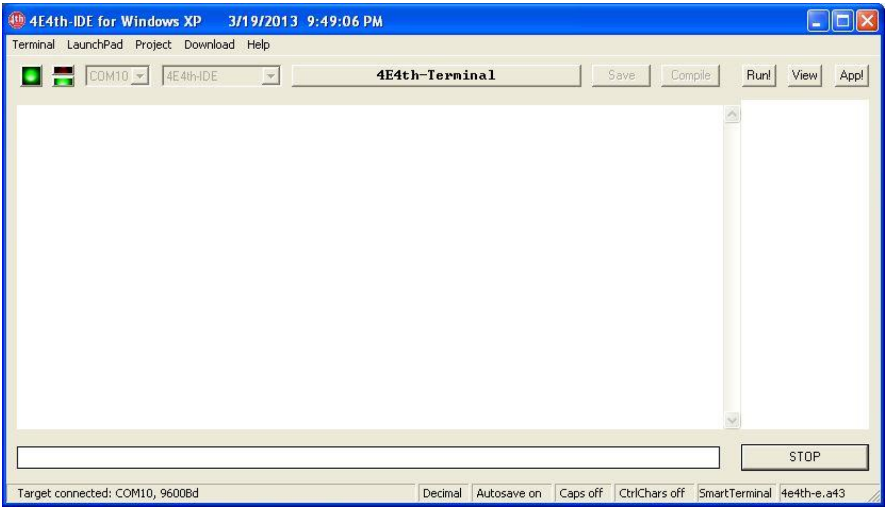
Now 4E4th software is installed on the LaunchPad!

The red and the green LED are lit as long as the LaunchPad is connected to the PC and no program switched off an LED.

When these 12 installation points have been successful, 4E4th and the 4E4th-IDE are ready to use.

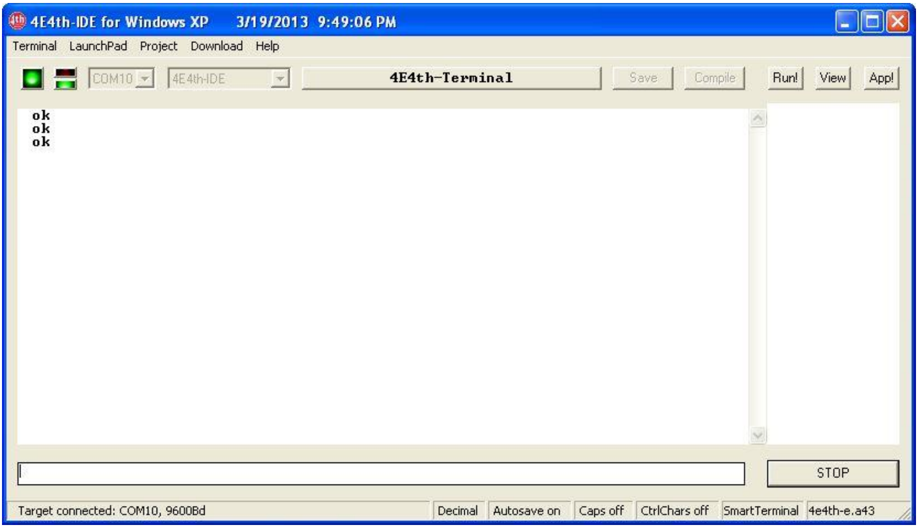
A click on "*START*", and the 4E4th LaunchPad is connected:

Both LEDS on the IDE change to green.



The cursor is blinking at the entry line.

Everytime when actuating the PC Enter key,
4E4th reports with "`ok`" and a new line:



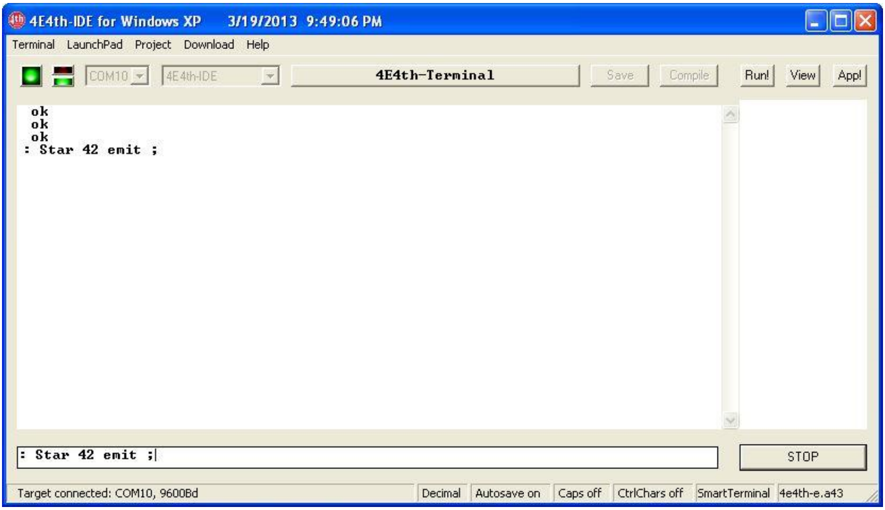
```
ok
ok
ok
```

Now our adventure "Programming" is ready to start!

Our first example is from Leo Brodie's book "Starting Forth":

```
: Star 42 emit ;
```

Type it into the IDE entry line at the bottom and hit the Enter key.



```
 ok
 ok
 ok
: Star 42 emit ;
```

```
: Star 42 emit ;
```

Target connected: COM10, 9600Bd

Explanation:

The colon marks the start of a new Word-Definition.

"Star" is this new Word, the name of this new Word.

All 4E4th texts are coded in ASCII-CODE, 42 is the ASCII-Code for the Star: *

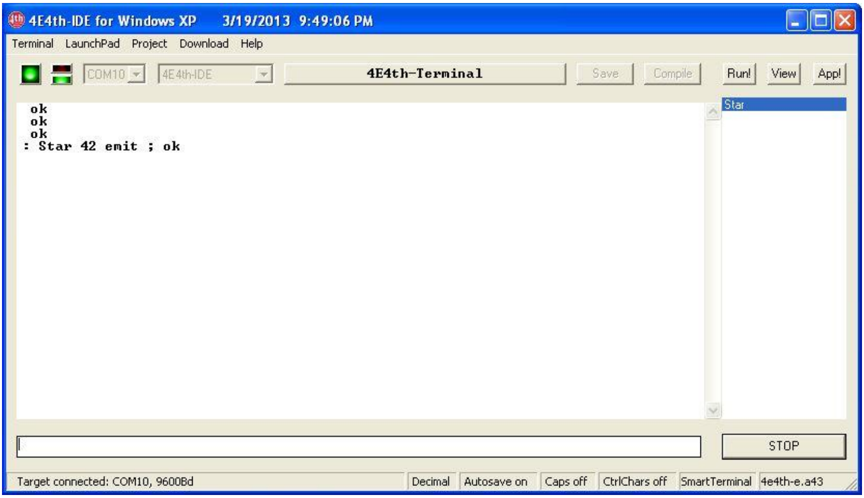
"*emit*" means send to the terminal.

The semicolon marks the end of this new Word-Definition.

This is really easy to comprehend:

With this line "`: Star 42 emit ;`"
we define a new word, which sends a Star * to our Terminal – it's so simple!

We type this line into the 4E4th-IDE entry line and actuate the PC Enter key:

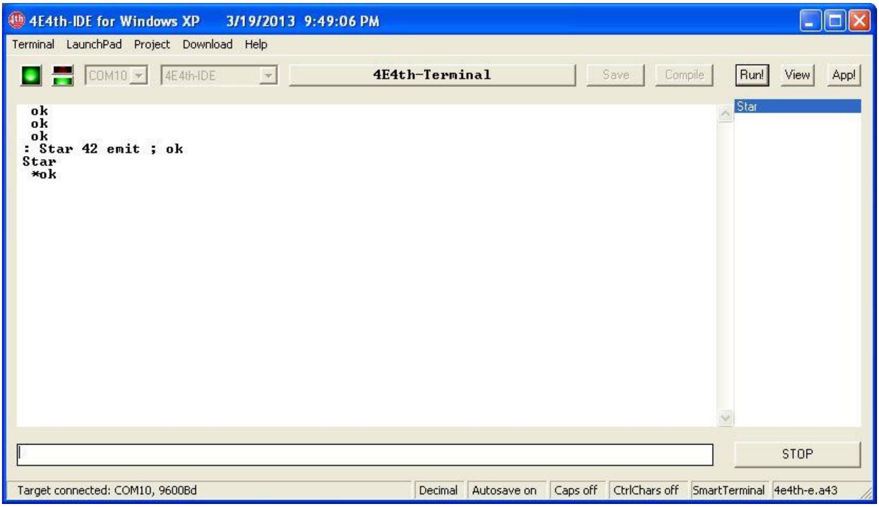


```
 ok
 ok
 ok
: Star 42 emit ; ok
```

Target connected: COM10, 9600Bd

4E4th answers with `ok`, the entry line is empty again, and in the white field, on the right side, the Word "Star" is shown in white on blue. In this box the new Words defined are shown.

A click on "*Run!*" bottom on the right, and the Word `Star` appears in the next Terminal line, followed by "* `ok`" in the second next Terminal line:



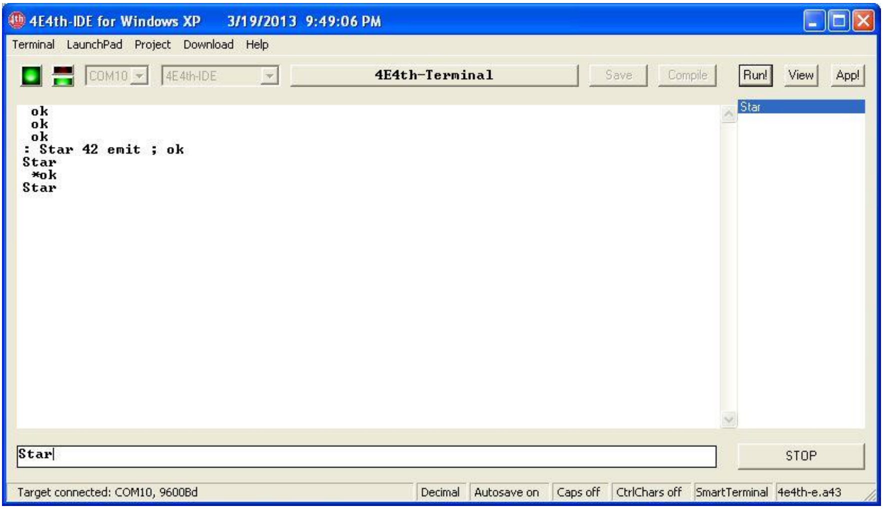
```
 ok
 ok
 ok
: Star 42 emit ; ok
Star
 *ok
```

Target connected: COM10, 9600Bd

Now we have written and tested our first “Program”! Cool!

This first program is named *“Star”,* the Word, which gave its name to this Program. Therefore from now on we call it “Word” instead of Program.

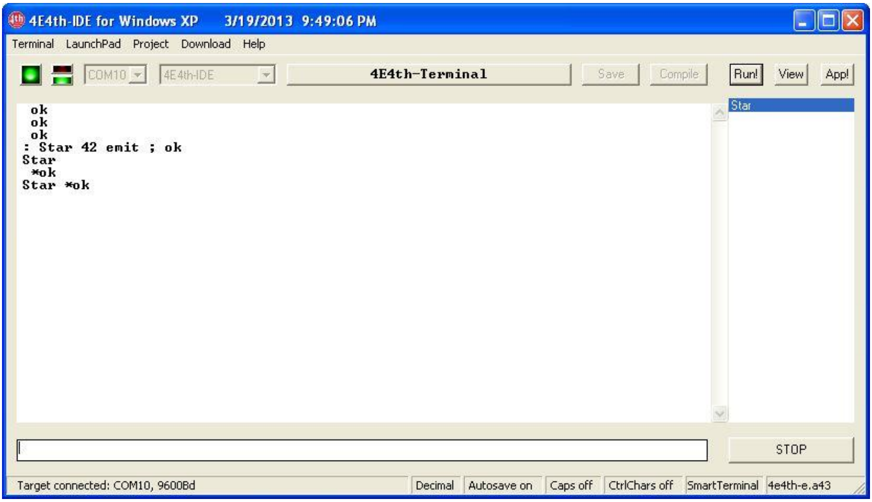
Another kind of testing:
We type the Word “Star” into our entry line



```
 ok
 ok
 ok
: Star 42 emit ; ok
Star
  *ok
Star
```

```
Star
```

and actuate the PC Enter key:

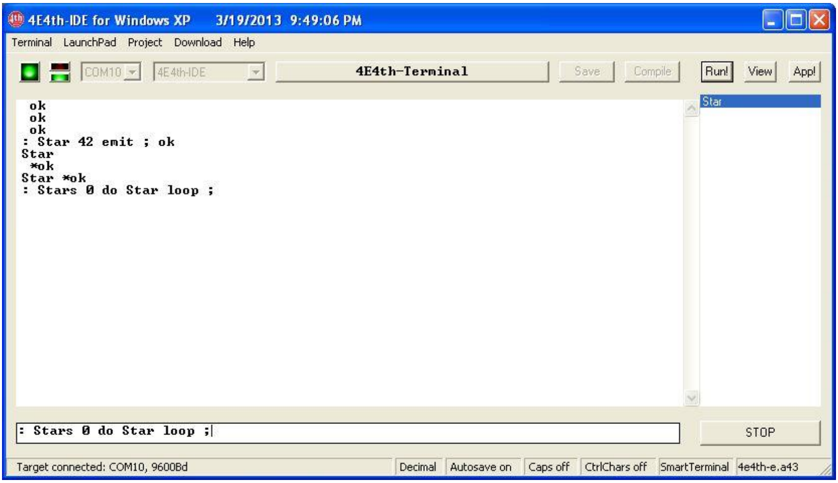


```
 ok
 ok
 ok
: Star 42 emit ; ok
Star
  *ok
Star *ok
```

This time we see the Word and the Star in one line, followed by "*ok".
Now we are sure that Programming works and we are able to proceed:

```
: Stars 0 do Star loop ;
```

Hint: We can use Copy/Paste to copy this line into our entry line!

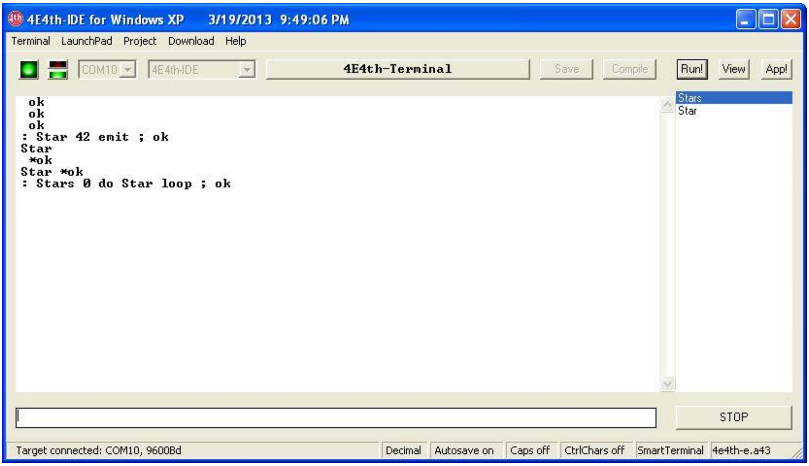

4E4th-IDE for Windows XP 3/19/2013 9:49:06 PM
Terminal LaunchPad Project Download Help
COM10
4E4th-IDE
4E4th-Terminal
Save
Compile
Run!
View
Appl
Star
STOP

```
 ok
 ok
 ok
: Star 42 emit ; ok
Star
 *ok
Star *ok
: Stars 0 do Star loop ;
```

```
: Stars 0 do Star loop ;
```

Target connected: COM10, 9600Bd | Decimal | Autosave on | Caps off | CtrlChars off | SmartTerminal | 4e4th-e.a43

The whole line will now be sent to our microprocessor.


4E4th-IDE for Windows XP 3/19/2013 9:49:06 PM
Terminal LaunchPad Project Download Help
COM10
4E4th-IDE
4E4th-Terminal
Save
Compile
Run!
View
Appl
Stars
Star
STOP

```
 ok
 ok
 ok
: Star 42 emit ; ok
Star
 *ok
Star *ok
: Stars 0 do Star loop ; ok
```

Target connected: COM10, 9600Bd | Decimal | Autosave on | Caps off | CtrlChars off | SmartTerminal | 4e4th-e.a43

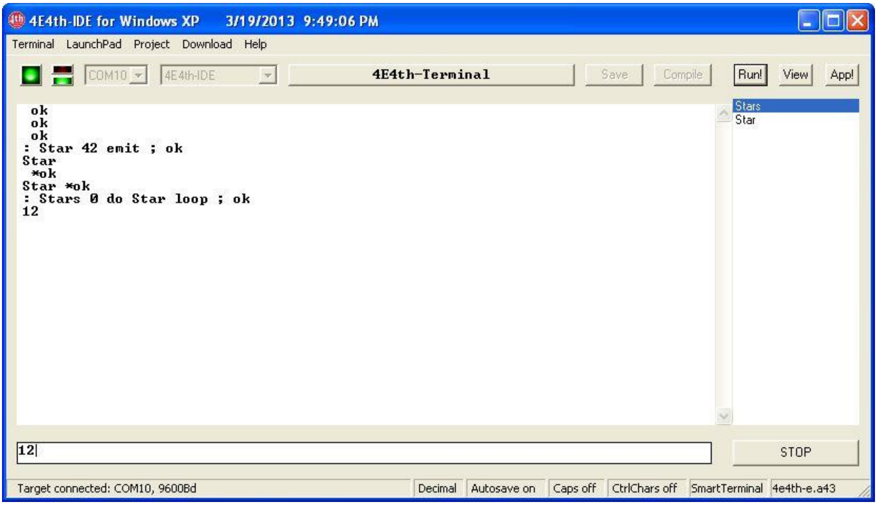
We now have a new Word "*Stars*" on top of our Word List, and here how it works:

This Program starts with zero, that's what the "*0*" is telling.

Then it does (*do*) something: sending a "*Star*", and this function will be repeated in a "*loop*".

4E4th has to be informed how often to repeat, that means it needs the number of Stars to send, before the Word "*Stars*" will be started.

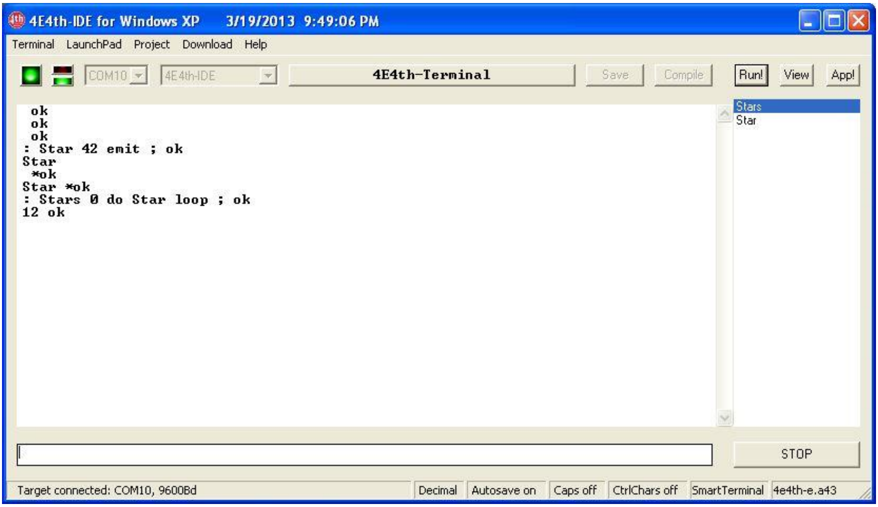
To accomplish this, type a number into our entry line, let's say 12:



```
 ok
 ok
 ok
: Star 42 emit ; ok
Star
 *ok
Star *ok
: Stars 0 do Star loop ; ok
12
```

```
12
```

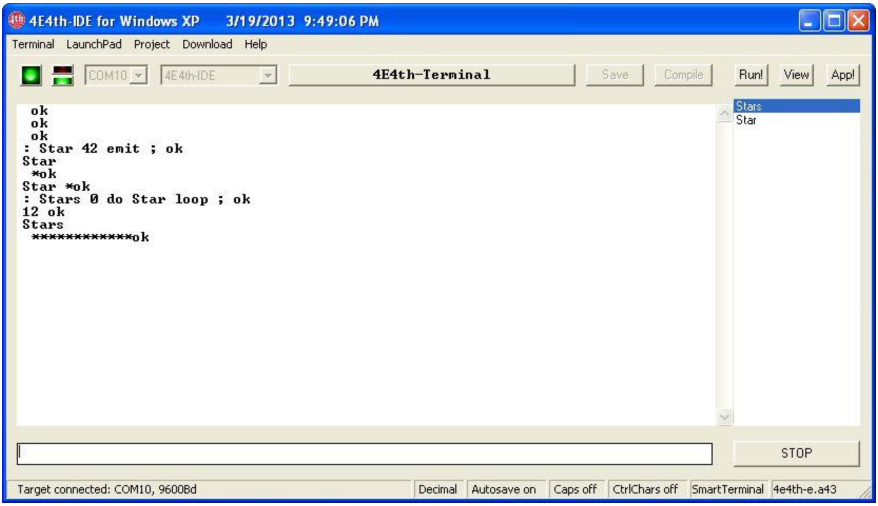
As usual we actuate the PC Enter key - 4E4th answers with "`ok`":



```
 ok
 ok
 ok
: Star 42 emit ; ok
Star
 *ok
Star *ok
: Stars 0 do Star loop ; ok
12 ok
```

Target connected: COM10, 9600Bd | Decimal | Autosave on | Caps off | CtrlChars off | SmartTerminal | 4e4th-e.a43

With a click on "*Run!*" the Word "*Stars*" will be transferred to the Microprocessor, shown in the Terminal window, and the Microprocessor answers with 12 Stars, followed by "`ok`":

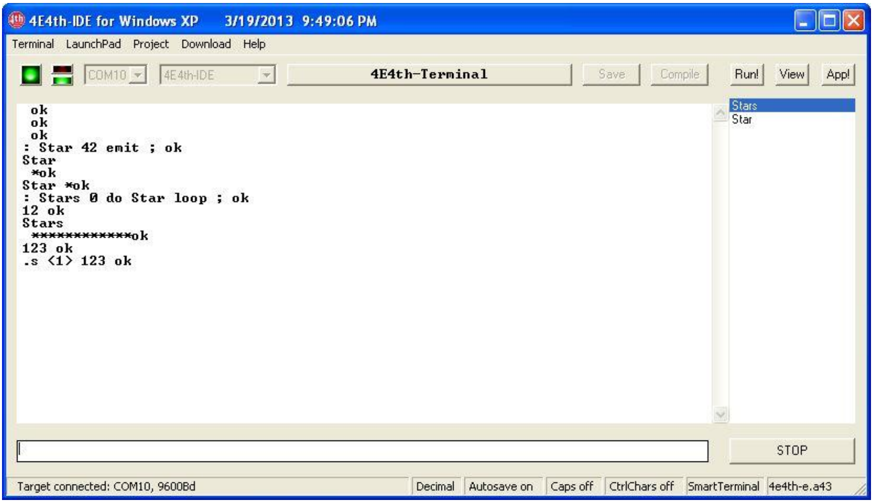


```
 ok
 ok
 ok
: Star 42 emit ; ok
Star
 *ok
Star *ok
: Stars 0 do Star loop ; ok
12 ok
Stars
 ************ok
```

Target connected: COM10, 9600Bd | Decimal | Autosave on | Caps off | CtrlChars off | SmartTerminal | 4e4th-e.a43

By the way, when we like to know, if 4E4th took our number, then we can ask by typing "`.s`":

So, type a number, actuate the PC Enter key, type "`.s`", and actuate the PC Enter key again:



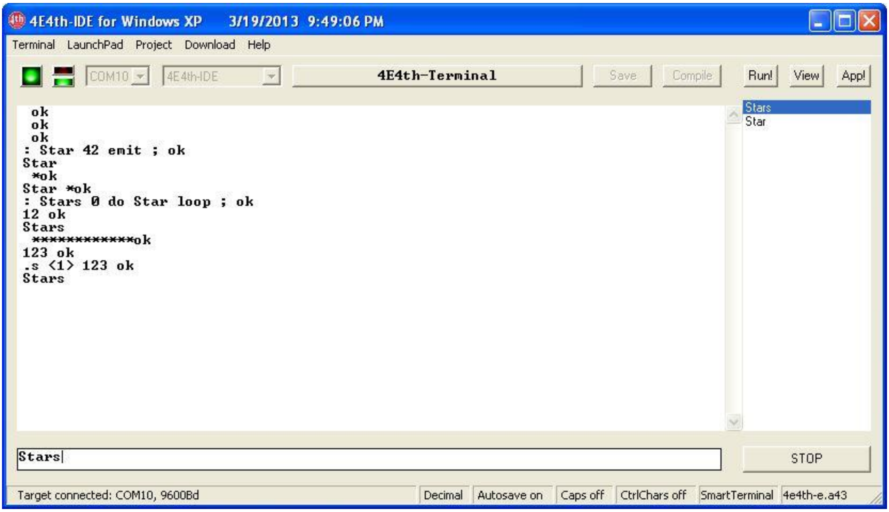
```
 ok
 ok
 ok
: Star 42 emit ; ok
Star
 *ok
Star *ok
: Stars 0 do Star loop ; ok
12 ok
Stars
 ************ok
123 ok
.s <1> 123 ok
```

4E4th answers after "`.s`" with

`<1> 123 ok`

`<1>` means, that 4E4th got *one* number. This number was 123.

Then we type in "Stars":



```
 ok
 ok
 ok
: Star 42 emit ; ok
Star
 *ok
Star *ok
: Stars 0 do Star loop ; ok
12 ok
Stars
 ************ok
123 ok
.s <1> 123 ok
Stars
```

```
Stars
```

Actuating the PC Enter key, 4E4th answers, following the Word "Stars" with 123 Stars and "`ok`":

```
Stars
**************************************************
**************************************************
***********************ok
```

These first examples clearly show, that the PC Enter key has to be actuated to tell 4E4th to evaluate what we typed into our entry line.

Each character, which we typed into the 4E4th-IDE entry line, will be sent to our 4E4th LaunchPad Microprocessor immediately, and 4E4th echoes each character back to our IDE – printable characters from 4E4th will then be shown in our Terminal window.

With actuating the PC Enter key, the input of a line is closed, and 4E4th will react on what is written in this line. Later we will report how that works.

I am sure now that I don't have to mention the PC Enter key any more.

To actuate the PC Enter key at the end of a line should be natural now, of course.

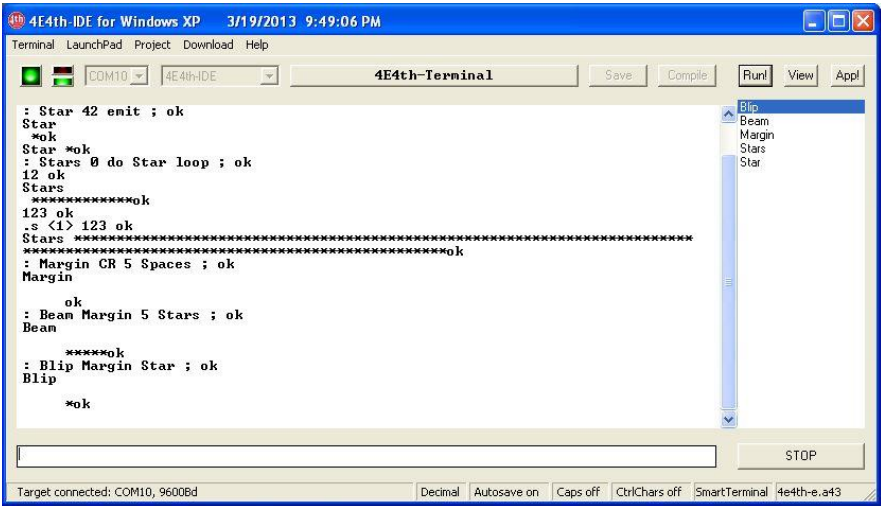
Let's go forward with our Program – here are the next lines:

```
: Margin CR 5 Spaces ;

: Beam Margin 5 Stars ;

: Blip Margin Star ;

: F Beam Blip Beam Blip Blip Blip ;
```

Of course it makes sense to test each Word after each Word-Definition.

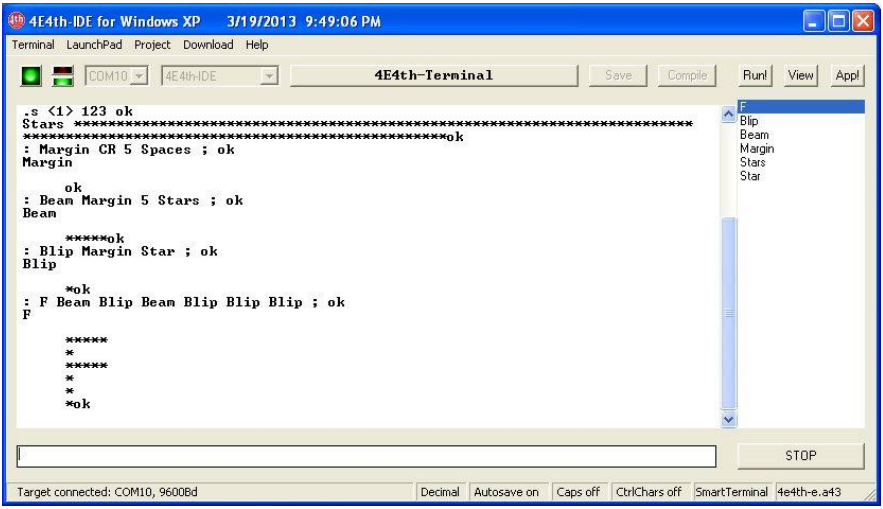
These Words don't expect a number.
Click on "*Run!*" after each line to look what this Word does.

This is really simple and fast, using Copy/Paste for each line, followed by actuating the Enter key and test with a simple click on "*Run!*":



```
: Star 42 emit ; ok
Star
 *ok
Star *ok
: Stars 0 do Star loop ; ok
12 ok
Stars
 ************ok
123 ok
.s <1> 123 ok
Stars *************************************************************************
**************************************************ok
: Margin CR 5 Spaces ; ok
Margin

     ok
: Beam Margin 5 Stars ; ok
Beam

     *****ok
: Blip Margin Star ; ok
Blip

     *ok
```

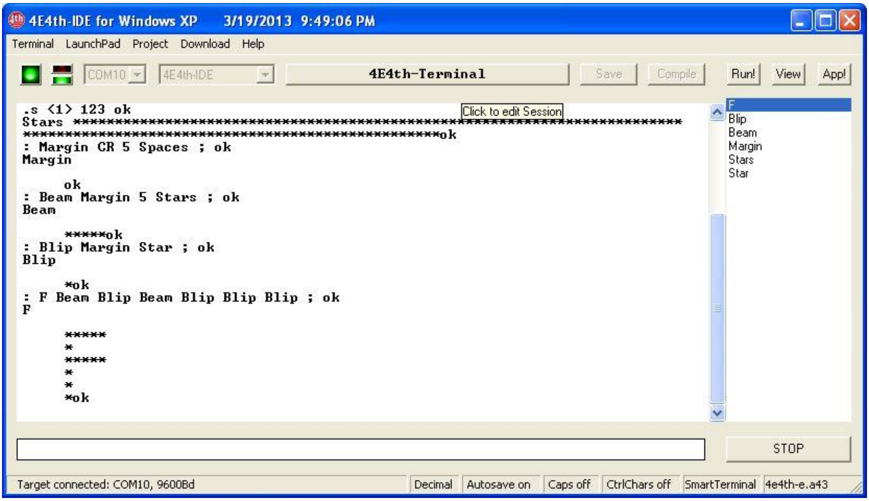
You see, this is a really simple programming example, to learn the first steps with 4E4th. The last line paints an F, using stars:



```
.s <1> 123 ok
Stars *************************************************************************
**************************************************ok
: Margin CR 5 Spaces ; ok
Margin

     ok
: Beam Margin 5 Stars ; ok
Beam

     *****ok
: Blip Margin Star ; ok
Blip

     *ok
: F Beam Blip Beam Blip Blip Blip ; ok
F

     *****
     *
     *****
     *
     *
     *ok
```

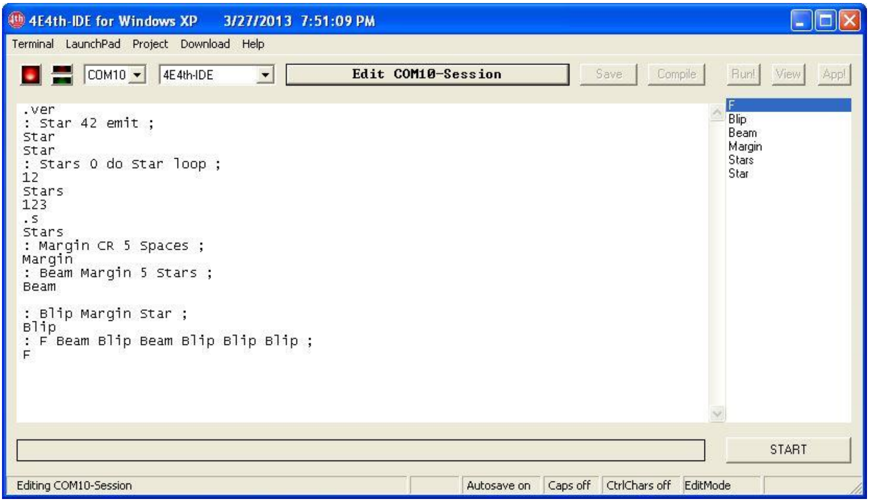
It bothers me that the "ok" is touching the last star.
To make the "F" look better, we should change it!

Nothing simpler than that:



```
.s <1> 123 ok
Stars *************************************************************************
**************************************************ok
: Margin CR 5 Spaces ; ok
Margin

     ok
: Beam Margin 5 Stars ; ok
Beam

     *****ok
: Blip Margin Star ; ok
Blip

     *ok
: F Beam Blip Beam Blip Blip Blip ; ok
F

     *****
     *
     *****
     *
     *
     *ok
```

A click on the wide "4E4th-Terminal" bar above the Terminal window switches to the Editor Mode:
LEDs have changed to red, as this is a local activity.



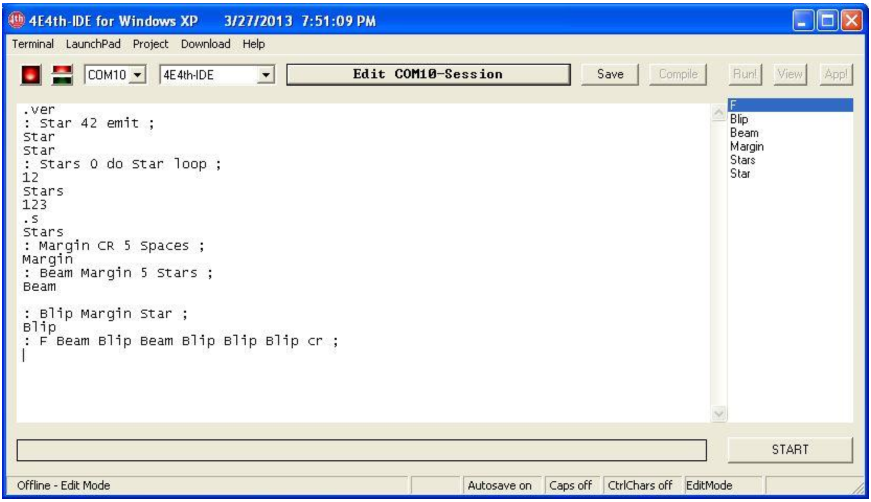
```
.ver
: Star 42 emit ;
Star
Star
: Stars 0 do Star loop ;
12
Stars
123
.s
Stars
: Margin CR 5 Spaces ;
Margin
: Beam Margin 5 Stars ;
Beam

: Blip Margin Star ;
Blip
: F Beam Blip Beam Blip Blip Blip ;
F
```

Now the caption of this bar has changed to "*Edit COMxx-Session*", and our text, which we typed is shown in the Terminal window, written in another font, and we are able to do changes.

That's what we do now. The second line from bottom is:

```
: F Beam Blip Beam Blip Blip Blip ;
```

We simply put a "`cr and a space`" before the semicolon:



```
.ver
: Star 42 emit ;
Star
Star
: Stars 0 do Star loop ;
12
Stars
123
.s
Stars
: Margin CR 5 Spaces ;
Margin
: Beam Margin 5 Stars ;
Beam

: Blip Margin Star ;
Blip
: F Beam Blip Beam Blip Blip Blip cr ;
```

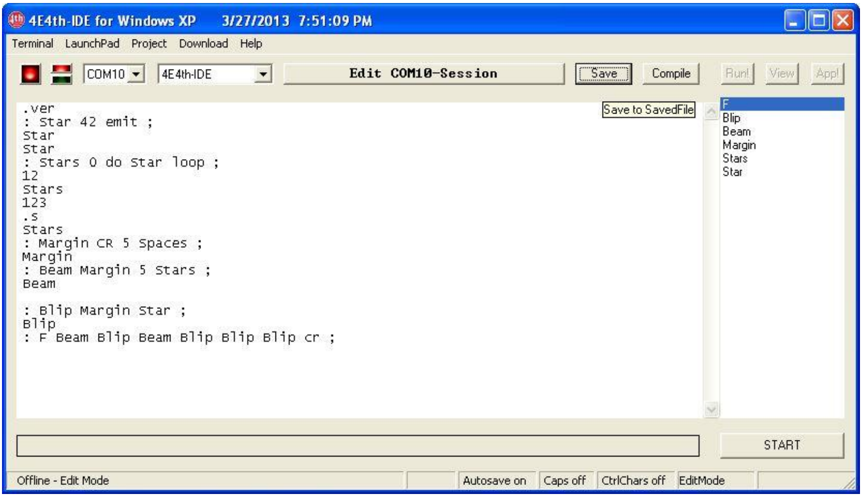
"`cr`" is a shortcut for "Carriage Return", inherited from the old times of teletypes. With the "`cr`" inserted, this line looks like on the picture above.

To be on the safe side, we have removed the single character "`F`" in the last line.

Now a clue to be remembered: between two Words always has to be a space - Colon and Semicolon are counted as Words, too.

This is the only way to allow 4E4th to be able to read each Word and recognize it.

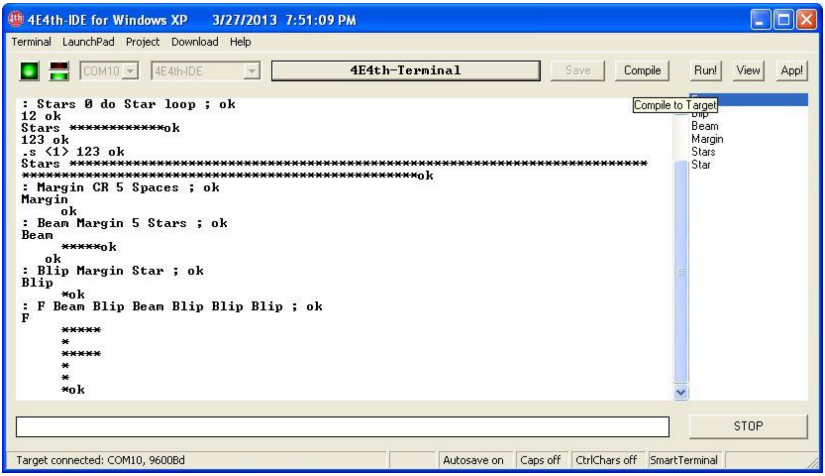
Next we click on button "*Save*" (right of the "*Editor*" bar), and then button "*Compile*" <u>two times</u>



```
.ver
: Star 42 emit ;
Star
Star
: Stars 0 do Star loop ;
12
Stars
123
.s
Stars
: Margin CR 5 Spaces ;
Margin
: Beam Margin 5 Stars ;
Beam

: Blip Margin Star ;
Blip
: F Beam Blip Beam Blip Blip Blip cr ;
```

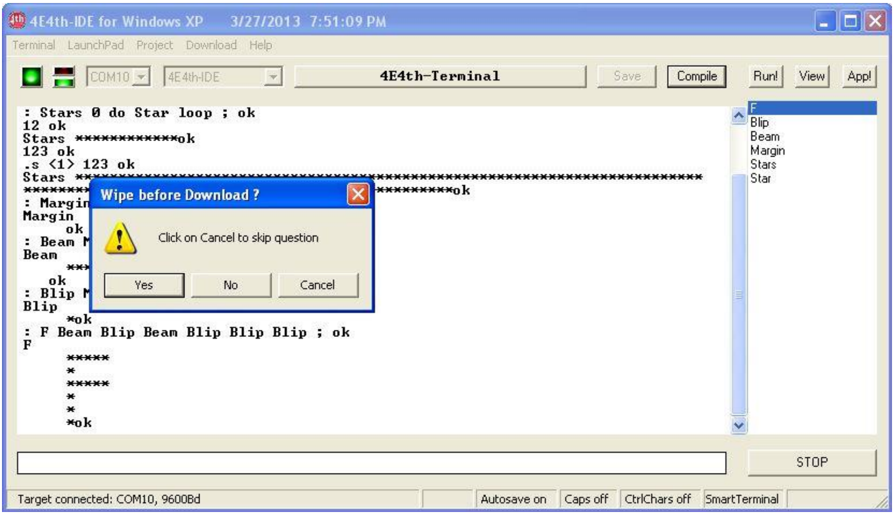
The first click on "*Compile*" switches 4E4th-IDE back to Terminal-Modus:

The LEDS change back to green



```
: Stars 0 do Star loop ; ok
12 ok
Stars ************ok
123 ok
.s <1> 123 ok
Stars *************************************************************************
**************************************************ok
: Margin CR 5 Spaces ; ok
Margin
     ok
: Beam Margin 5 Stars ; ok
Beam
     *****ok
   ok
: Blip Margin Star ; ok
Blip
     *ok
: F Beam Blip Beam Blip Blip Blip ; ok
F
     *****
     *
     *****
     *
     *
     *ok
```

The second click is followed by a Popup asking "*Wipe before Download?*":



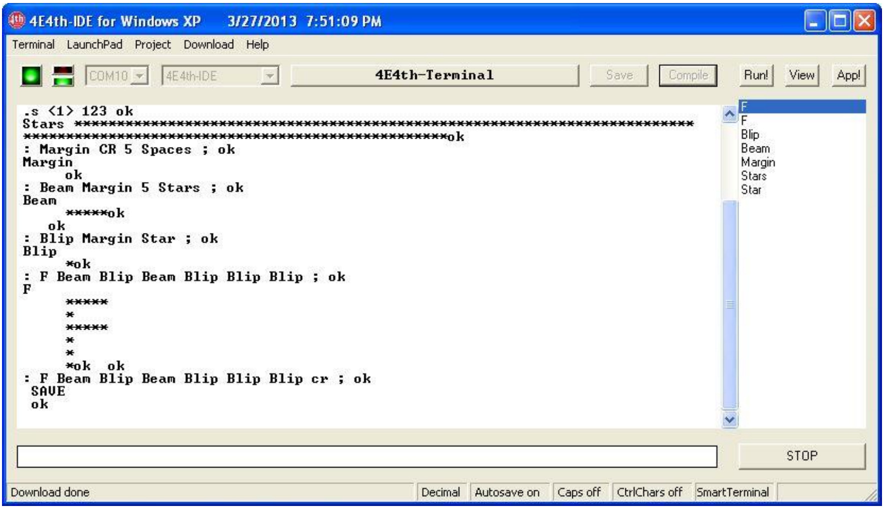
```
: Stars 0 do Star loop ; ok
12 ok
Stars ************ok
123 ok
.s <1> 123 ok
Stars ************************************************************************
********                                              **********ok
: Margin
Margin
     ok
: Beam M
Beam
     ***
   ok
: Blip M
Blip
     *ok
: F Beam Blip Beam Blip Blip Blip ; ok
F
     *****
     *
     *****
     *
     *
     *ok
```

Wipe before Download ?

Click on Cancel to skip question

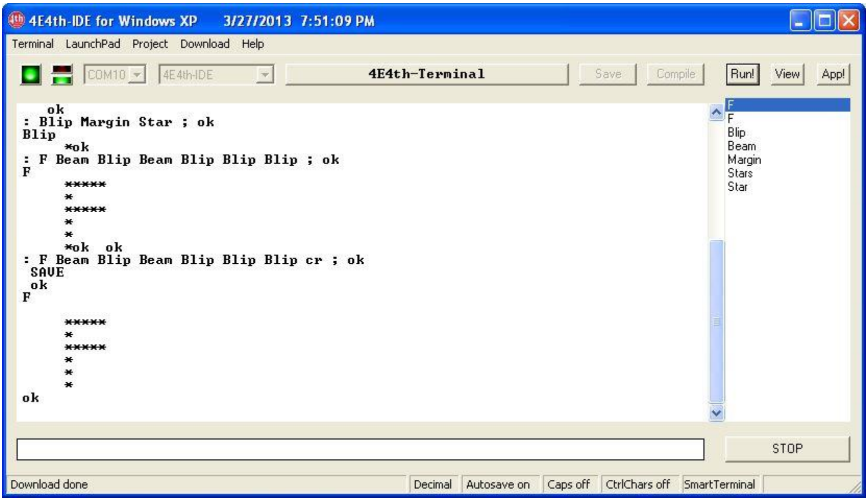
We don't like to compile everything again, so we click on the middle button "*No*".

Doing this, only the line which we changed will be sent to 4E4th:



```
.s <1> 123 ok
Stars ************************************************************************
***************************************************ok
: Margin CR 5 Spaces ; ok
Margin
     ok
: Beam Margin 5 Stars ; ok
Beam
     *****ok
   ok
: Blip Margin Star ; ok
Blip
     *ok
: F Beam Blip Beam Blip Blip Blip ; ok
F
     *****
     *
     *****
     *
     *
     *ok  ok
: F Beam Blip Beam Blip Blip Blip cr ; ok
 SAVE
 ok
```

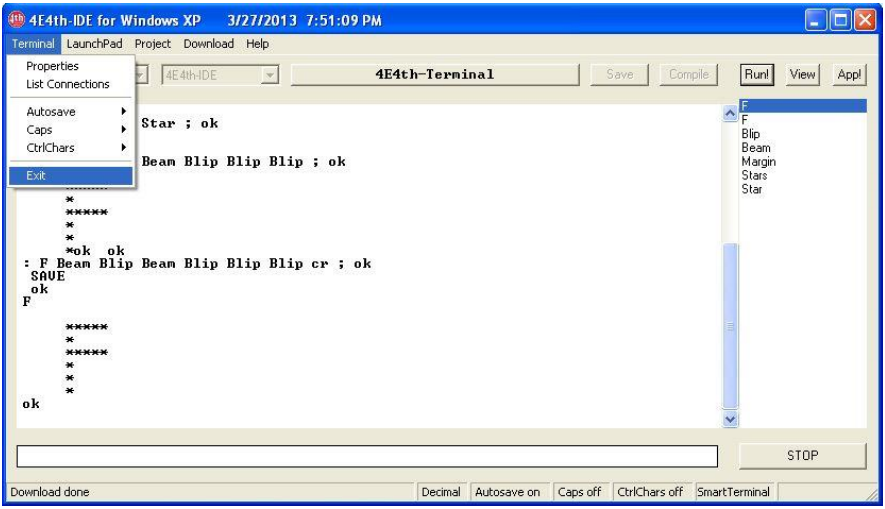
A click on "*Run!*" shows, our operation was successful:



```
   ok
: Blip Margin Star ; ok
Blip
     *ok
: F Beam Blip Beam Blip Blip Blip ; ok
F
     *****
     *
     *****
     *
     *
     *ok  ok
: F Beam Blip Beam Blip Blip Blip cr ; ok
 SAVE
 ok
F

     *****
     *
     *****
     *
     *
     *
ok
```

Now a final Test!

A click on *Terminal* (top left), and the *Terminal-Menu* shows up:

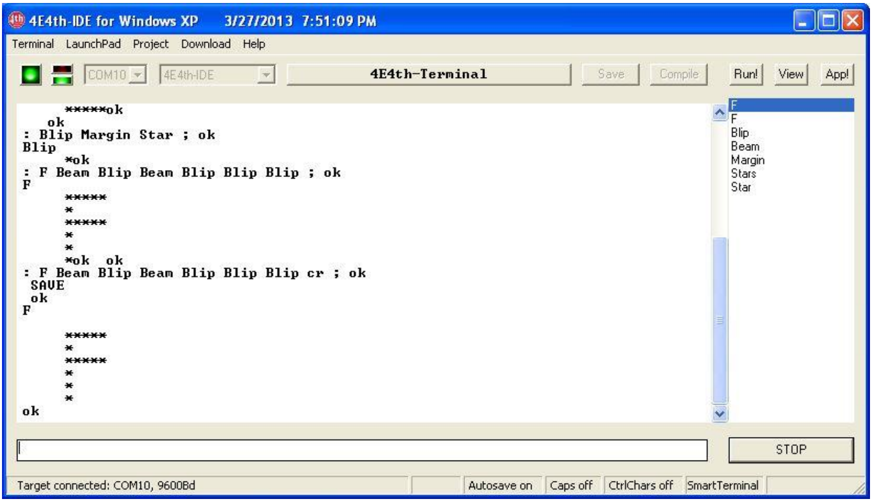


```
                Star ; ok

                Beam Blip Blip Blip ; ok

     *
     *****
     *
     *
     *ok  ok
: F Beam Blip Beam Blip Blip Blip cr ; ok
 SAVE
 ok
F

     *****
     *
     *****
     *
     *
     *
ok
```

A click on *Exit* ends our 4E4th-IDE-Session ("session" is the name for sitting at our computer for a while).

Then we pull the USB-plug of the Launchpad-cable from our PC, and the LaunchPad LEDs are off. Of course, without power nothing works.

To see what happens, we connect our 4E4th-LaunchPad, which we used for our first programming steps with our PC again;
after waiting for ten seconds or more to give our PC the chance to check that our USB connection was interrupted first and then connected again - all three LEDs on the board are lit again now.



```
     *****ok
   ok
: Blip Margin Star ; ok
Blip
     *ok
: F Beam Blip Beam Blip Blip Blip ; ok
F
     *****
     *
     *****
     *
     *
     *ok  ok
: F Beam Blip Beam Blip Blip Blip cr ; ok
 SAVE
 ok
F

     *****
     *
     *****
     *
     *
     *
ok
```

Then we start our *4E4th-IDE.exe* – a click on the button "*START*" bottom right, and our work appears in our Terminal window again (see picture above).

A click on "*Run!*" shows that our work wasn't been in vain.
It still works!

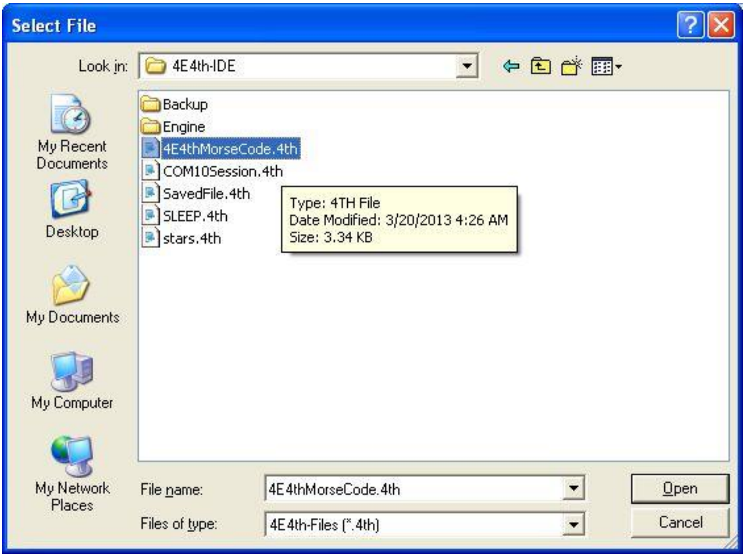
Now our Programming Adventure is really starting to go!

To have some fun, we prepared a Morse program.

We connect a little 8Ohm speaker to MSP430-LaunchPad pins P1.4 and P1.5, and there we go:

This time we choose a ready made Program, and we send it with "*Download*" from the PC to our 4E4th-LaunchPad.

With a click on menu “*Download / Selected File”* the well-known Windows-File-Selection appears:



Look in: 4E4th-IDE

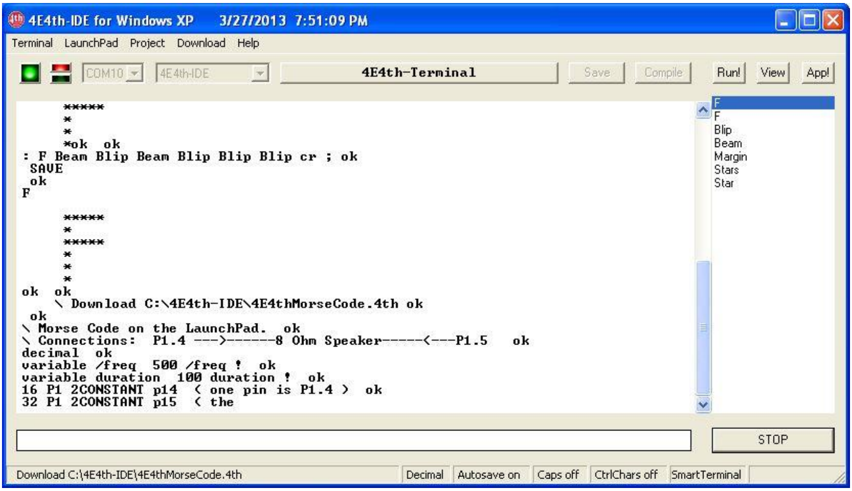
4E4thMorseCode.4th
COM10Session.4th
SavedFile.4th
SLEEP.4th
stars.4th

Type: 4TH File
Date Modified: 3/20/2013 4:26 AM
Size: 3.34 KB

File name: 4E4thMorseCode.4th

Files of type: 4E4th-Files (*.4th)

A click on 4E4thMorseCode.4th
and then on "*Open*",
and the Download starts:

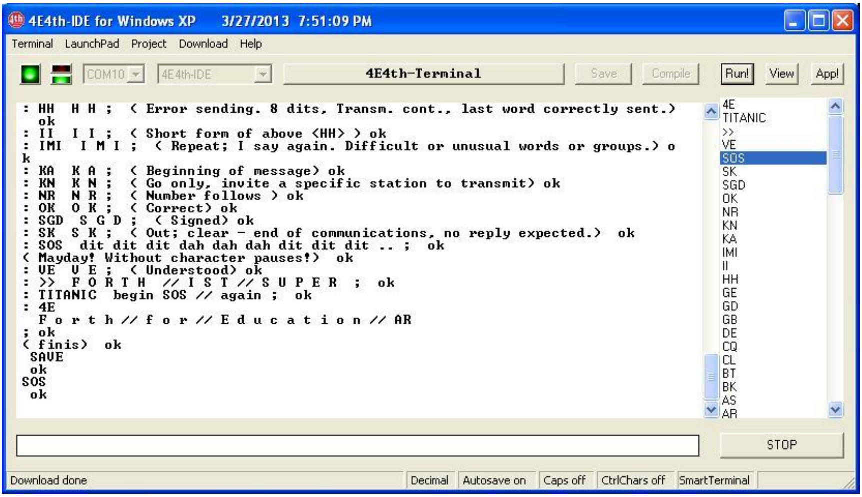


```
     *****
     *
     *
     *ok  ok
: F Beam Blip Beam Blip Blip Blip cr ; ok
 SAVE
 ok
F

     *****
     *
     *****
     *
     *
     *
ok  ok
    \ Download C:\4E4th-IDE\4E4thMorseCode.4th ok
 ok
\ Morse Code on the LaunchPad.  ok
\ Connections:  P1.4 --->------8 Ohm Speaker-----<---P1.5   ok
decimal  ok
variable /freq  500 /freq !  ok
variable duration  100 duration !  ok
16 P1 2CONSTANT p14  ( one pin is P1.4 )  ok
32 P1 2CONSTANT p15  ( the
```

Download C:\4E4th-IDE\4E4thMorseCode.4th | Decimal | Autosave on | Caps off | CtrlChars off | SmartTerminal

The Download needs some time – it needs a bit of patience.

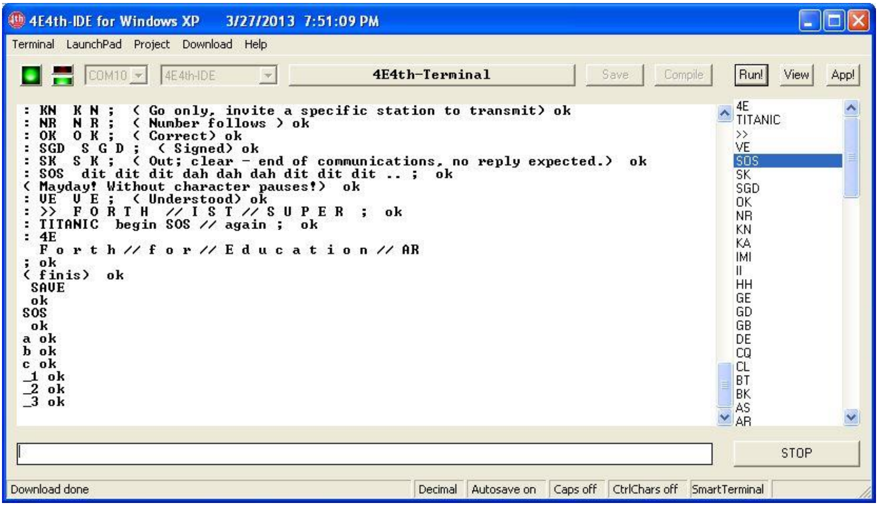
The Download has to be ready for testing – soon it is done:



```
: HH  H H ;  ( Error sending. 8 dits, Transm. cont., last word correctly sent.)
  ok
: II  I I ;  ( Short form of above <HH> ) ok
: IMI  I M I ;  ( Repeat; I say again. Difficult or unusual words or groups.) o
k
: KA  K A ;  ( Beginning of message) ok
: KN  K N ;  ( Go only, invite a specific station to transmit) ok
: NR  N R ;  ( Number follows ) ok
: OK  O K ;  ( Correct) ok
: SGD  S G D ;  ( Signed) ok
: SK  S K ;  ( Out; clear - end of communications, no reply expected.)  ok
: SOS  dit dit dit dah dah dah dit dit dit .. ;  ok
( Mayday! Without character pauses!)  ok
: VE  V E ;  ( Understood) ok
: >>  F O R T H  // I S T // S U P E R  ;  ok
: TITANIC  begin SOS // again ;  ok
: 4E
  F o r t h // f o r // E d u c a t i o n // AR
; ok
( finis)  ok
 SAVE
 ok
SOS
 ok
```

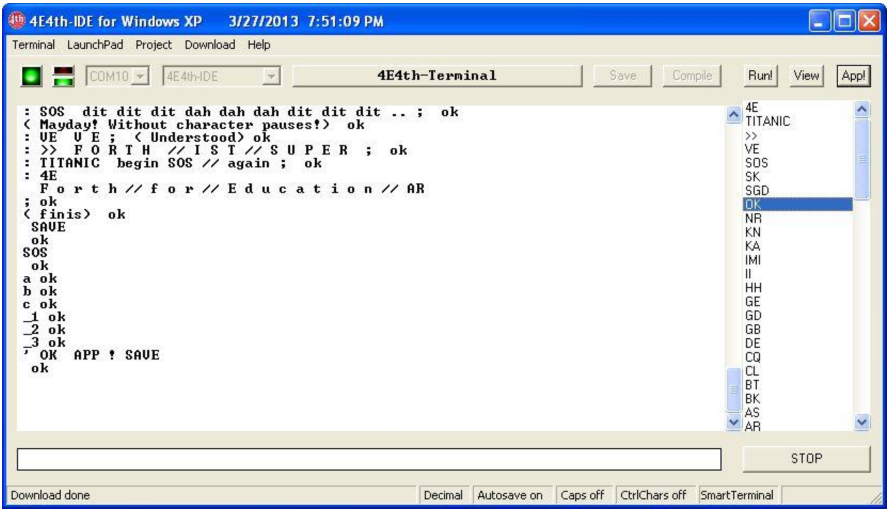
A click on "SOS" and "*Run!*", and we hear an SOS – as long as the speaker is connected in the right manner – and the red LED is blinking SOS.

We can type in single characters into the Terminal entry line and hear the sound pattern. It works with single digits too, but these need an underscore to be recognized. Of course we can click on characters, digits, and combinations to make them sound as Morse code.



```
: KN  K N ;  ( Go only, invite a specific station to transmit) ok
: NR  N R ;  ( Number follows ) ok
: OK  O K ;  ( Correct) ok
: SGD  S G D ;  ( Signed) ok
: SK  S K ;  ( Out; clear - end of communications, no reply expected.)  ok
: SOS  dit dit dit dah dah dah dit dit dit .. ;  ok
( Mayday! Without character pauses!)  ok
: VE  V E ;  ( Understood) ok
: >>  F O R T H  // I S T // S U P E R  ;  ok
: TITANIC  begin SOS // again ;  ok
: 4E
  F o r t h // f o r // E d u c a t i o n // AR
; ok
( finis)  ok
 SAVE
 ok
SOS
 ok
a ok
b ok
c ok
_1 ok
_2 ok
_3 ok
```

For completion we click on "OK" at the Wordlist (right panel), then a click on "*App!*", and from now on we will be greeted with an "OK" Morsecode each time when we hit the LaunchPad RESET button or when we connect our LaunchPad to power again:



```
: SOS  dit dit dit dah dah dah dit dit dit .. ;  ok
( Mayday! Without character pauses!)  ok
: VE  V E ;  ( Understood) ok
: >>  F O R T H  // I S T // S U P E R  ;  ok
: TITANIC  begin SOS // again ;  ok
: 4E
  F o r t h // f o r // E d u c a t i o n // AR
; ok
( finis)  ok
 SAVE
 ok
SOS
 ok
a ok
b ok
c ok
_1 ok
_2 ok
_3 ok
' OK  APP ! SAVE
 ok
```

Now we have a complete Program which starts by itself!

Congratulations! You have done it – even if it needed some patience!

Salisbury, 11/24/2012 - updated 3/27/2013 Dirk Bruehl.
Exeter,UK 19 March 2016 – changed to A5 size and MicroBox added by Juergen Pintaske
Preliminary Version – change without Notice