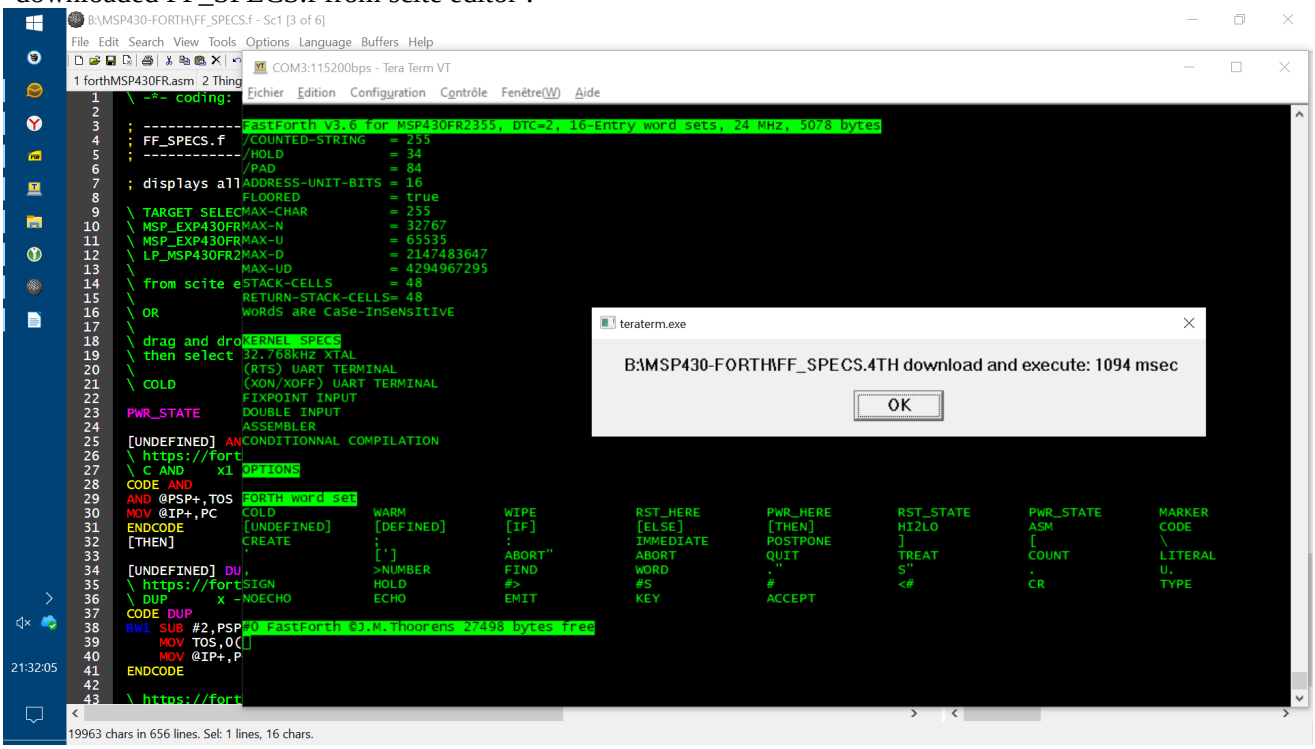
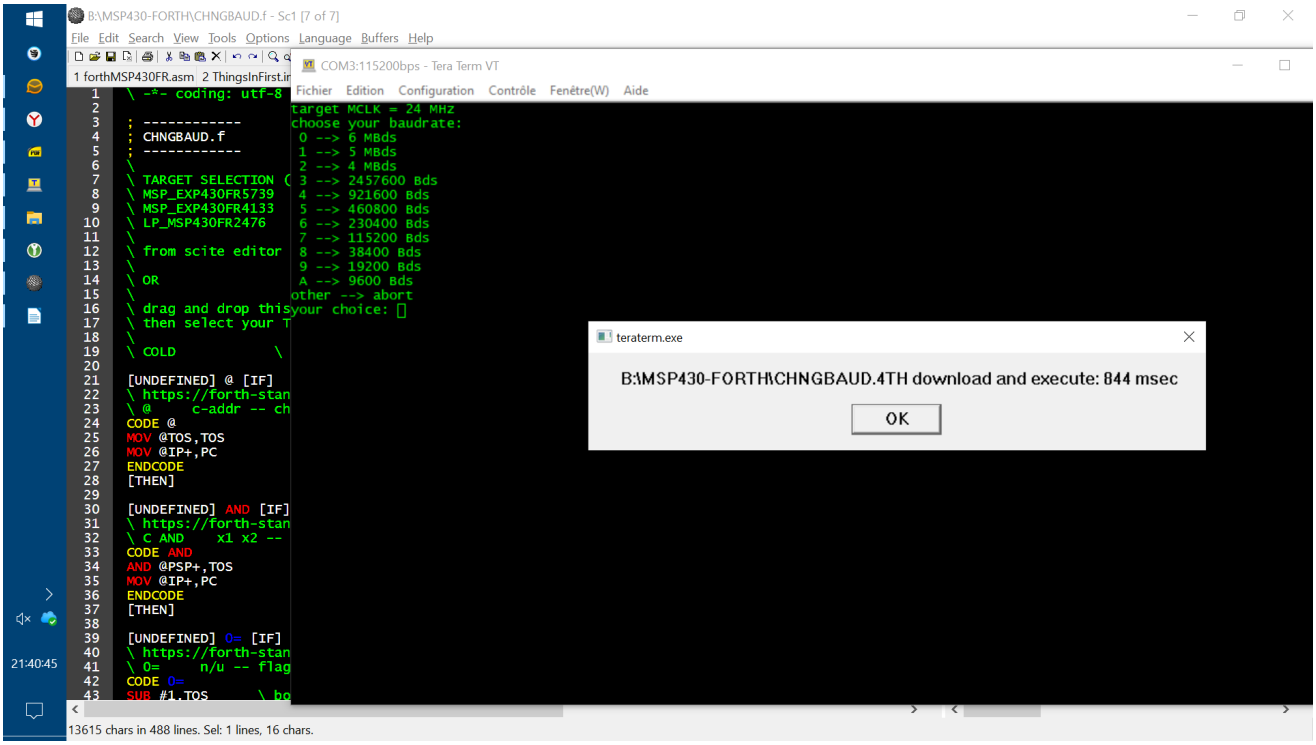
FastForth 3.7 out of the box

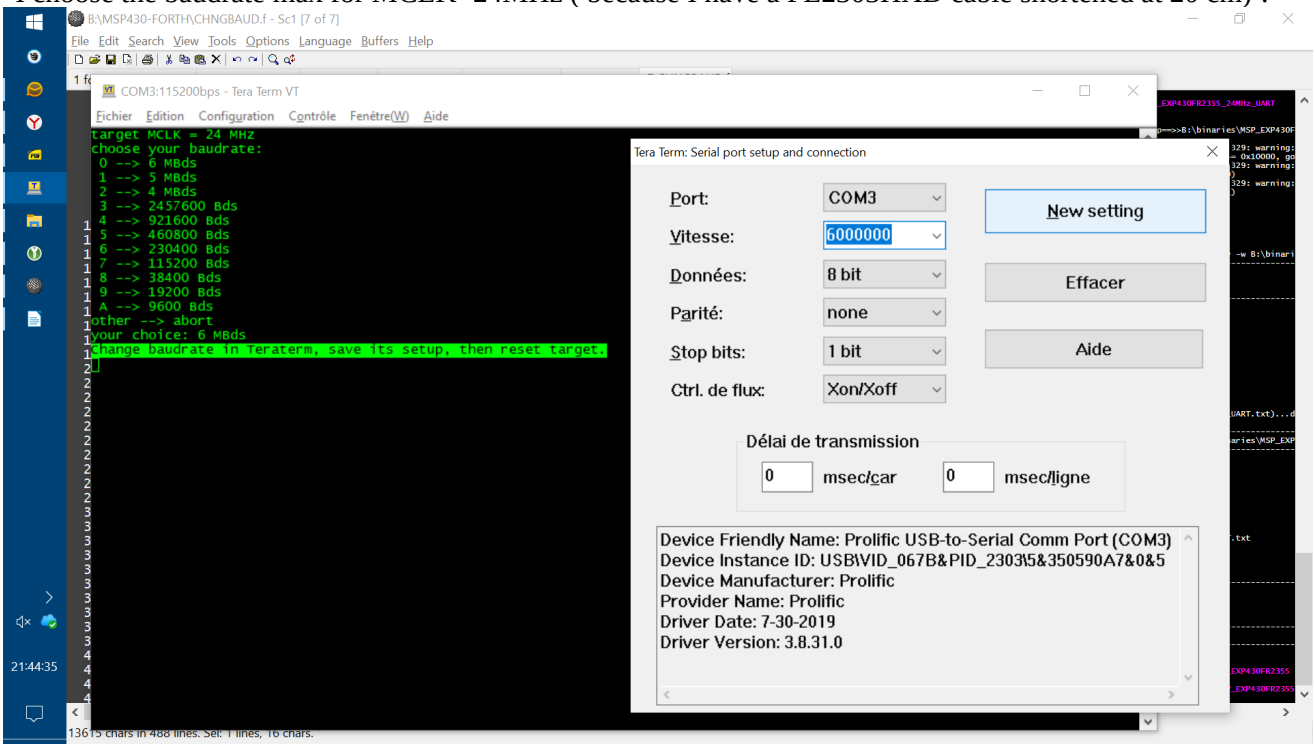
We have programmed MSP-EXP430FR2355 launchpad with MSP_EXP430FR2355_24MHz_UART.txt, then downloaded FF_SPECS.f from scite editor :



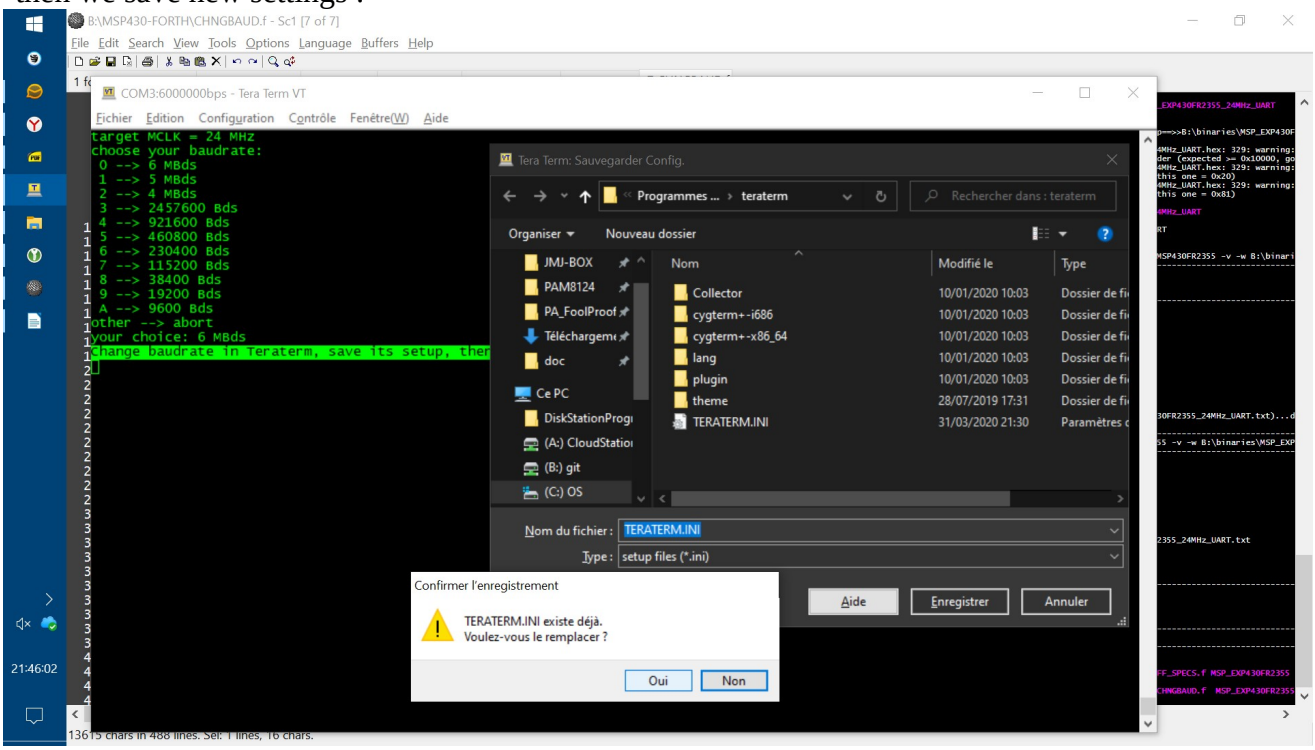
We try to speed up downloading, we download CHNGBAUD.f :



I choose the baudrate max for MCLK=24MHz (because I have a PL2303HXD cable shortened at 20 cm) :

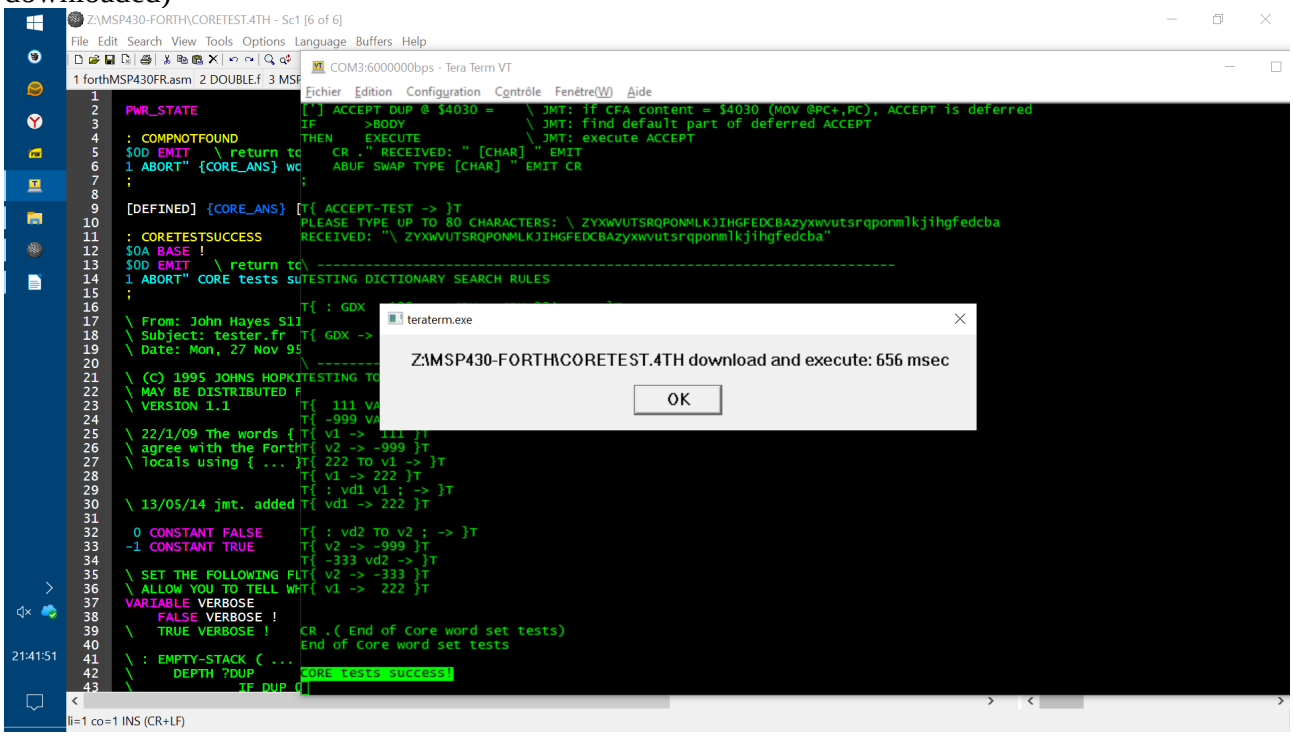


then we save new settings :



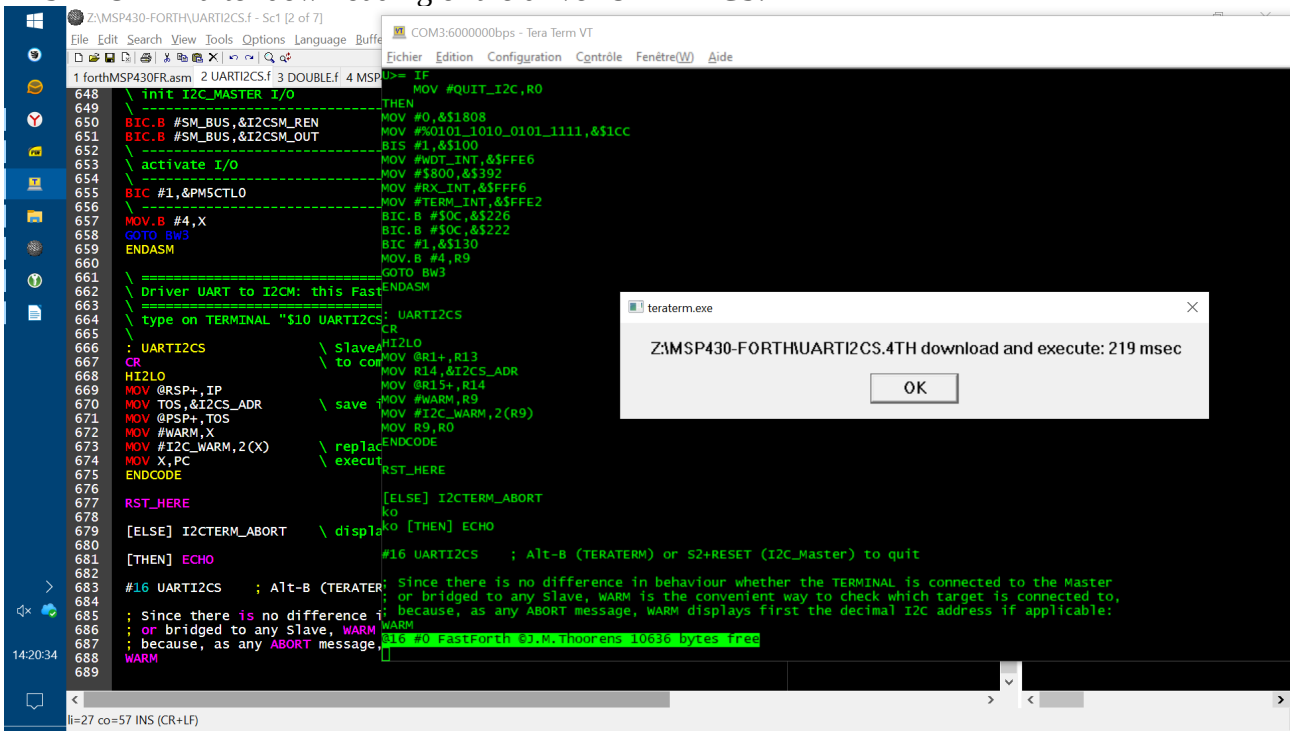
and we reset the launchpad before the next.

FAST FORTH @ 24MHz downloading CORETEST.4TH, (after CORE_ANS.f has been downloaded)



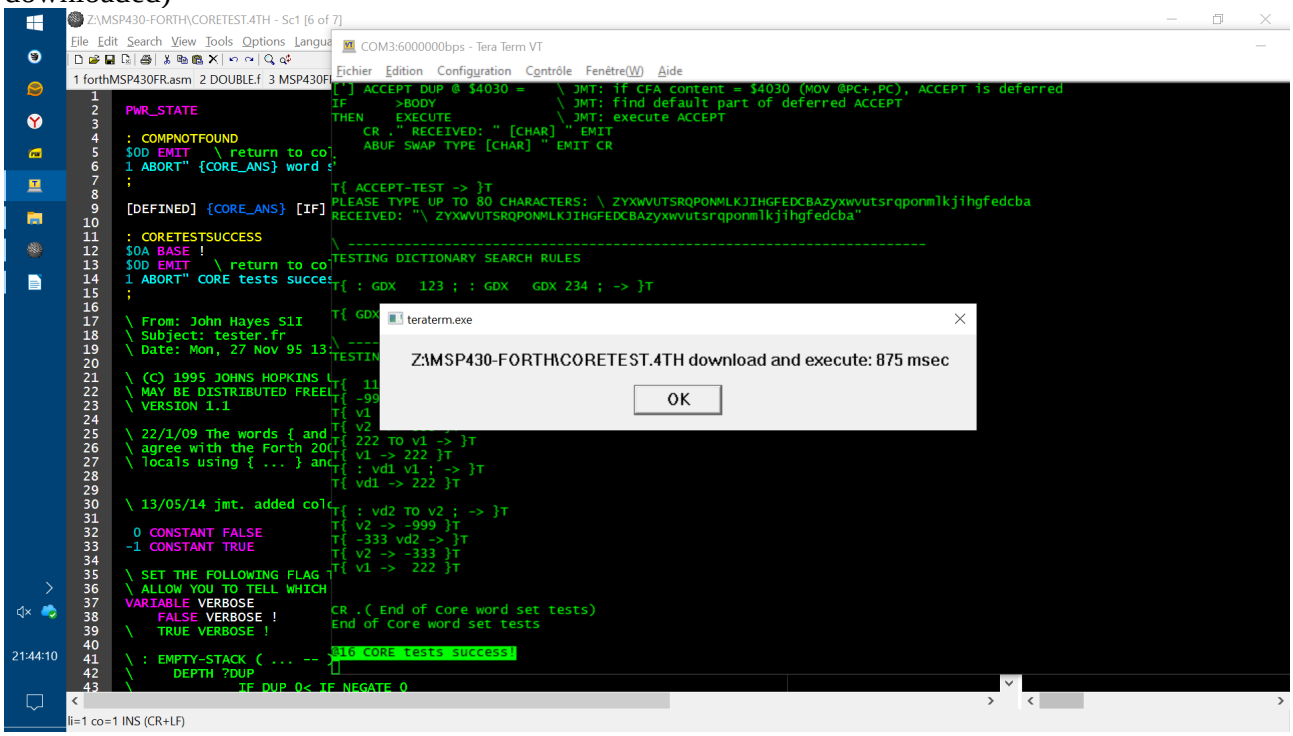
Notice the simplest of IDE's : scite editor, FastForth with its terminal. And the baudrate of COM3...

As another launchpad with I2C_FastForth is wired via an I2C BUS onto this launchpad, we can connect it :
FAST FORTH after downloading of the driver UARTI2CS.f



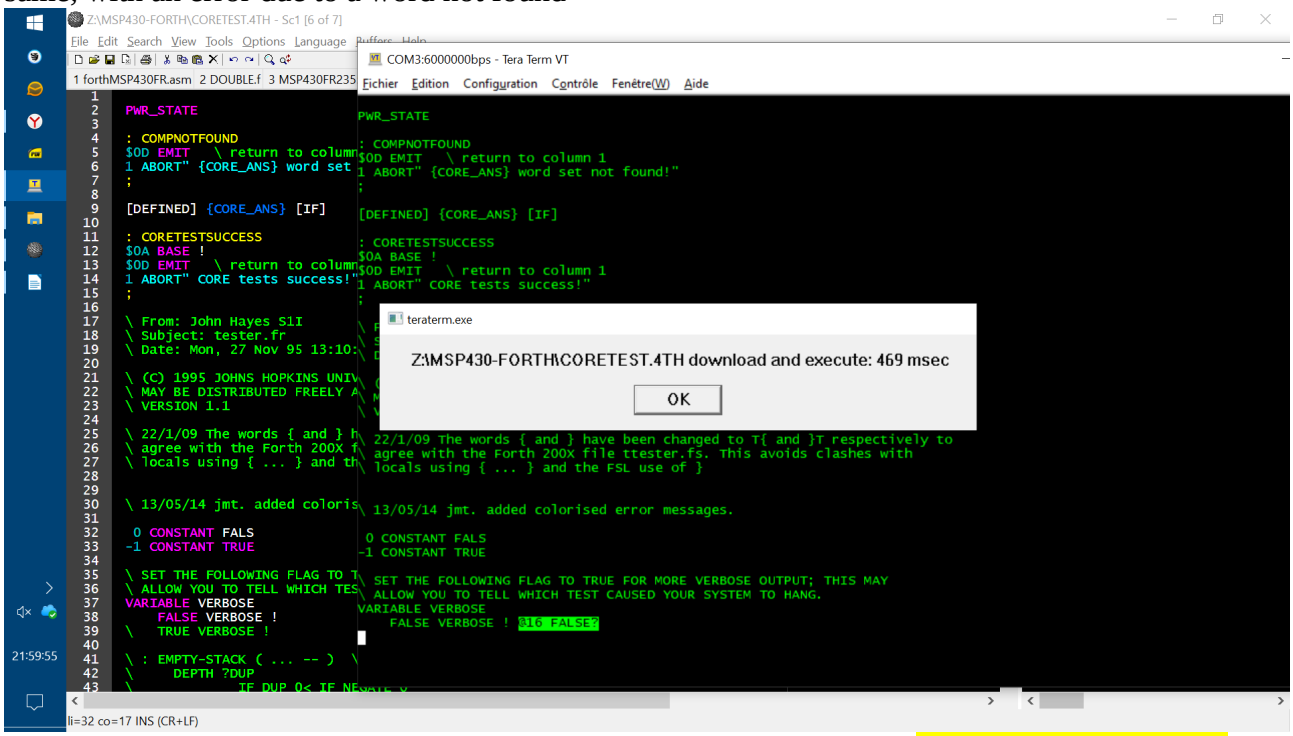
We see the work of preprocessor GEMA that replaces symbolic names by assembler's ones.
We also see the decimal address of the connected I2C_FastForth at the beginning of the message in reverse video.

I2C_Slave FAST FORTH @24MHz downloading CORETEST.4TH, (after CORE_ANS.f has been downloaded)



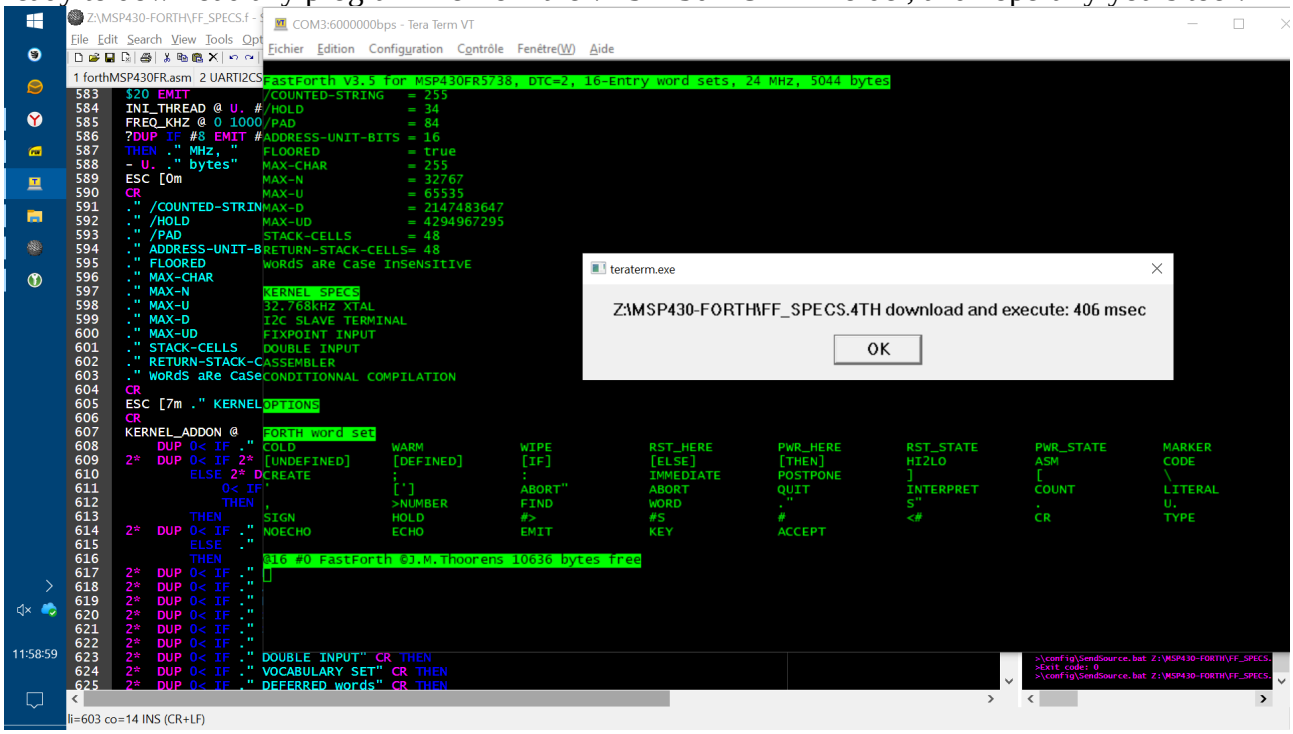
No difference in behavior of FastForth with UART TERMINAL or with I2C TERMINAL, apart from the download time…

same, with an error due to a word not found



Once the error is fixed in the source file, FastForth is ready to reload it without any other action, due to the smart error process that automatically shortens the main program to its state defined by PWR_STATE.

After downloading FF_SPECS.f, we see the FastForth specifications in its minimum configuration, ready to download any program file from the \MSP430-FORTH folder, and hopefully yours too !



By leveraging the best of FRAM technology and thanks to its amazing innovations, FastForth is impressive in efficiency during the most expensive development phase of a program with its very many round trips between real-time testing and code fixes.

And, by modifying the I2C addresses in the UARTI2CS.f file we can work with many I2C_FastForth targets...