AN1363
Application Note

Workaround to ISP Mode Limitation on the ST7MDT1-DVP2 and ST7MDT2-DVP2

by Microcontroller Cores and Tools Division

1 Introduction

About this application note

This application note seeks to correct a problem that may occur when using the **In Situ Programming (ISP)** feature on **ST7MDT1-DVP2** and **ST7MDT2-DVP2** development kits.

Please note that this problem only concerns In Situ Programming, so only persons who use this feature, or plan to use this feature during the development of your application, need to perform the hardware and software modifications described in this application note.

What is the ISP limitation?

Certain application board configurations may prevent you from being able to program your target MCU using ISP mode.

What causes the limitation?

Inability to program using ISP may occur when there is high capacitance at the $\overline{\text{RESET}}$ pin of the target MCU, located on your application board. More specifically, high capacitance at the $\overline{\text{RESET}}$ pin leads to an uncertainty in the timing of signal transmission at the initiation of programming.

Normally, when there is no capacitance at the $\overline{\text{RESET}}$ pin, ISP mode is initiated by the following sequence, illustrated in *Figure 1*:

1. A chip reset must be performed (i.e. voltage of $\overline{\text{RESET}}$ pin falls to zero) then released, so that the voltage of the $\overline{\text{RESET}}$ pin returns to 5 V.
2. The moment that the reset is released, the target MCU starts counting a period of 256 CPU_CLK timer ticks.
3. Shortly after the $\overline{\text{RESET}}$ is released, and independent of the RESET pin's voltage, a specific 33-peak signal sequence is sent to the dedicated ISPSEL pin.

4 If the ISPSEL signal is received in its entirety during the 256 CPU_CLK timer ticks after the reset release, the MCU to be programmed will enter ISP mode, and programming may begin.

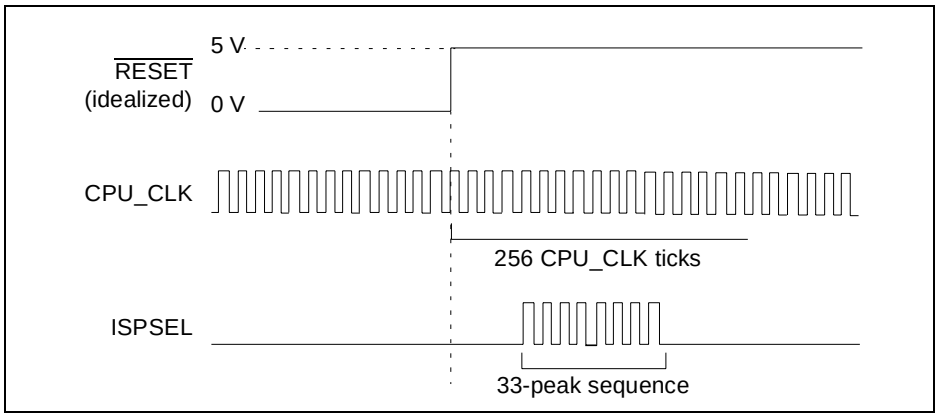


Figure 1: "Ideal" signal transmission at ISP mode initiation (RESET pin has no capacitance)

The problem arises when (for various worthy reasons) you have designed your application board with protections that result in a high capacitance at the target MCU's RESET pin. The result is that following a chip reset, the return of the pin to 5 V can be very gradual, rather than instantaneous.

As the voltage gradually rises at the RESET pin, at some unknown threshold voltage (probably in the vicinity of 3.5 V, but this is dependent on several factors), the target MCU begins to count out 256 CPU_CLK timer ticks, during which time the MCU is "listening" for the ISPSEL signal.

However, because the ISPSEL signal sequence is sent from the ISP driver independently of the RESET pin value, in a high capacitance scenario it becomes very likely that the 33-peak ISPSEL signal sequence will begin transmission before the MCU is ready to receive it (i.e. before the voltage on the RESET pin has reached its threshold value). *Figure 2* illustrates this sequence of events.

The result is that the target MCU doesn't receive the ISPSEL signal sequence in its entirety and therefore cannot be put into ISP mode to accept programming.

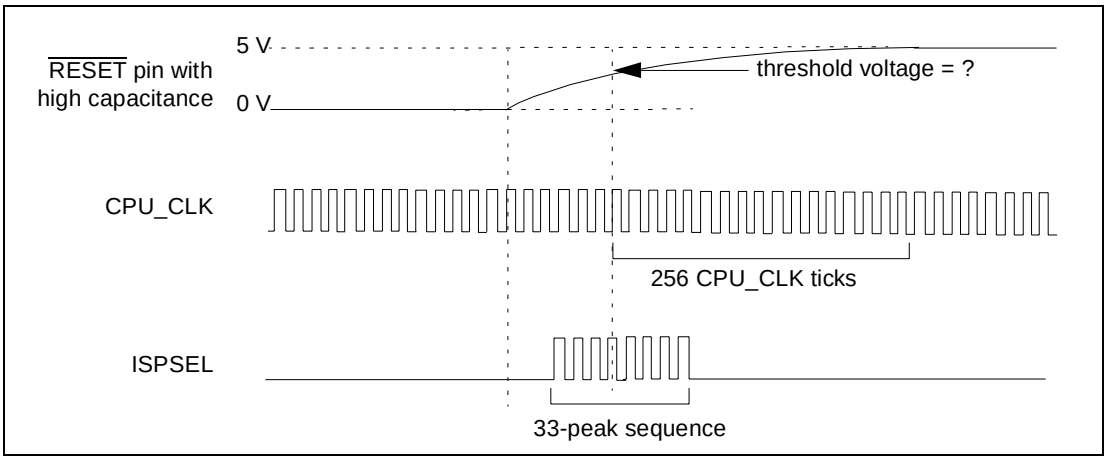


Figure 2: Signal transmission when RESET pin has high capacitance

2 Correcting the limitation

The first way to ensure that ISP mode functions correctly is to design your application board so that the $\overline{\text{RESET}}$ pin has low capacitance.

However, we recognize that this isn't always possible, and so have provided a workaround involving the following three steps:

1 Updating your version of WINEE (Windows Epromer) to version 4.0.

2 Modification to the DVP hardware—linking several pins of the ISP driver chip together by soldering two wires.

3 Reprogramming the ISP driver firmware.

The following sections describe each of these steps.

Downloading and running the patch

1 If you haven't already done so, download the **`st7dvp2_isp_patch.exe`** file from the downloads page of ST's Microcontroller website **(http://mcu.st.com)** and save it to a new folder on one of the host PC's local drives.

Note: The host PC should be the PC that you have been using with your DVP and must already have the Windows Epromer installed on it.

2 **`setup.exe`** is a self-extracting ZIP file. From Windows Explorer, double-click on it.

3 An install program will be launched to update your installed version of WINEE.

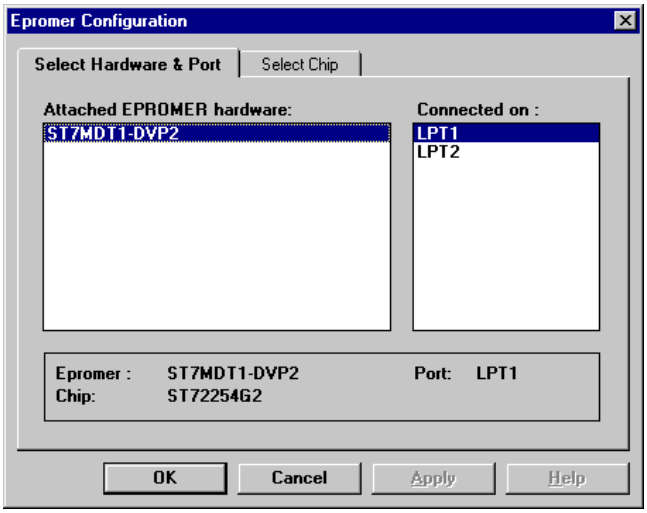
4 You will be asked to place a jumper across the ISP DRV PRG pins on your DVP board. The ISP DRV PRG pins are located on the DVP board close to the parallel port connector.

5 Once the jumper is placed

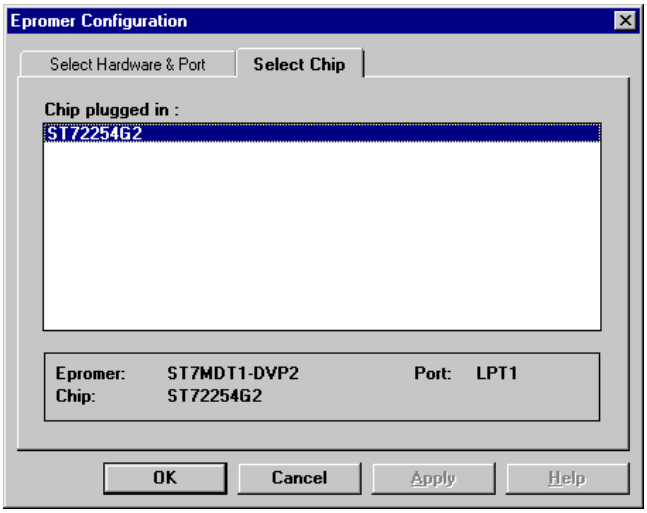
6 Connect your DVP via the parallel cable to one of your host PC's parallel ports.

7 From the Windows Epromer main menu, select ***Configure>Epromer***.

8 In the Configure window that appears, in the *Select Hardware & Port* tab, select **ST7MDT1-DVP2** and the parallel port on your host PC to which the DVP is connected.



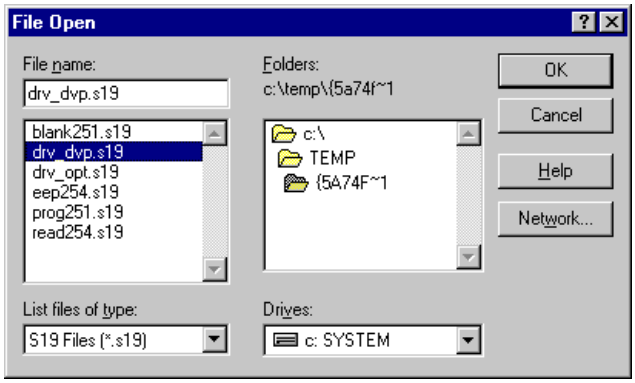
9 In the *Select Chip* tab, choose the **ST72254G2** chip.



10 Click ***OK***.

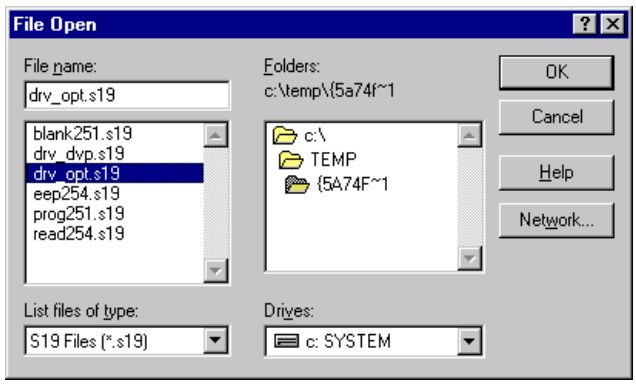
11 Click on the *EEPROM* tab of the Windows Epromer.

12 Select ***File>Open***. The folder containing the files to program is automatically opened. In the *List files of type* field, select **S19 Files (*.s19)** and select the **`drv_dvp.s19`** file. Click ***OK***.



13 Click on the *Options* tab of the Windows Epromer.

14 Select ***File>Open***. Select the **`drv_opt.s19`** file. Click ***OK***.



15 From the main menu, select ***Program>All*** to program the EEPROM and Options memory.

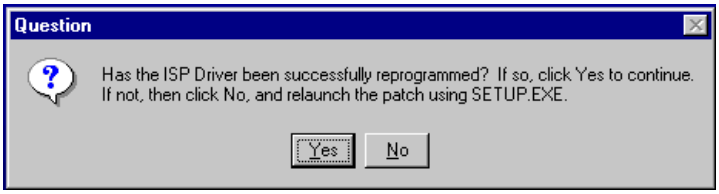
16 From the main menu, select ***Verify>All*** to check that the programing session was successfully completed (that the file contents match those programmed into the microcontroller memory).

17 Exit the Windows Epromer by selecting ***File>Exit***. A message will appear asking you whether the reprogramming has been successfully completed.

If you were able to correctly program and verify your DVP's firmware, click ***Yes*** to continue with the software patch to the Windows Epromer.

If you have **not** been able to successfully reprogram your DVP's ISP driver, it is very important to click ***NO*** — otherwise, the patch will continue by making changes to the Windows EPROMER program that will be incompatible with your DVP's firmware.

By clicking ***NO*** you will exit the installation program without making any changes to your Window EPROMER software, and you can try reprogramming again by relaunching the patch's **`setup.exe`**.



18 Once the installation program has finished, it is **VERY IMPORTANT** to remember to remove the jumper from the ISP DRV PRG pins on your DVP board!

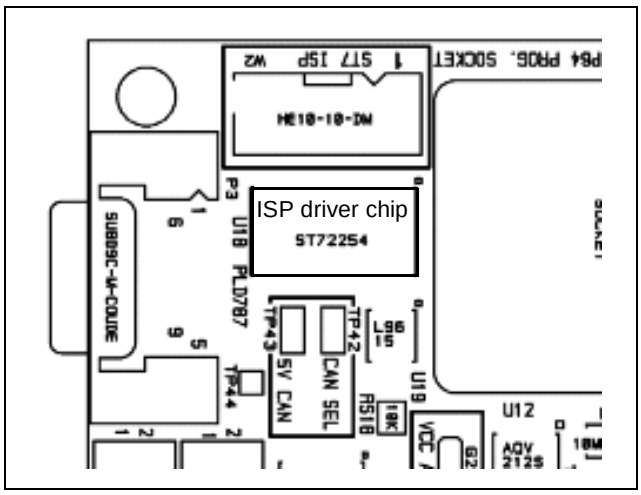
The ISP driver firmware is now reprogrammed.

You must still perform the hardware modifications described in the next section in order for your DVP's ISP mode to work correctly.

Modifying the DVP hardware

1 Locate the ISP driver chip (ST72254), in the corner of the DVP board, close to the ISP connector. *Figure 3* shows the location of the driver chip (labelled U18) for the ST7MDT2-DVP2. The driver chip is in a similar location (labelled U17) on the ST7MDT1-DVP2.

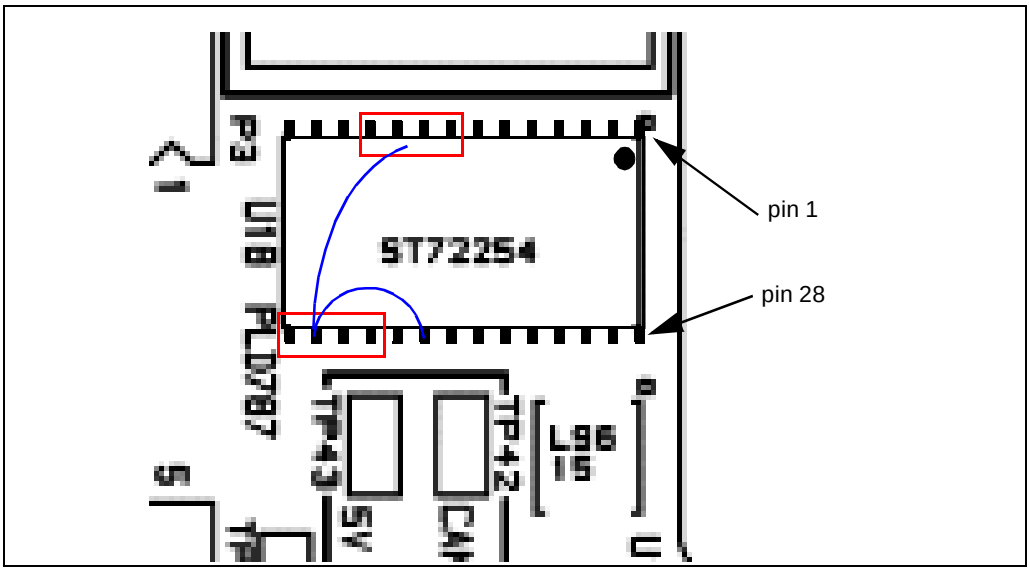
Figure 3: Location of ISP driver chip on the ST7MDT2-DVP2



2 Once you have located the ST72254 ISP driver chip, find pin 1 of the chip which is indicated by an indentation in the top of the chip's package.

3 Solder the following pins together:

- Pins 8, 9, 10, 11
- Pins 15, 16, 17, 18



4 Solder one end of an insulated wire from Pins 8, 9,10 and 11 (which are already soldered together). Solder the other end of the same insulated wire to Pins 15, 16, 17 and 18 (also soldered together).

5 Take a second insulated wire, and solder the first end to Pins 15, 16, 17 and 18 (soldered together) and the other end to pin 20.

Your hardware and firmware modifications are now complete, and your ISP mode should now work properly.