# AN2475
# Application note

## STR9 In-Application Programming using UART

## Introduction

An important requirement for most Flash-based systems is the ability to update firmware while the system is installed in the end product. This is referred to as In-Application Programming (IAP).

STR9 MCUs have the capability of running user-specific firmware to perform In Application Programming of the MCU embedded Flash memory. This feature allows the use of any type of communication protocol for the reprogramming process (for example, CAN, UART, USB). In this application note, UART is used as an example.

This application note outlines general guidelines for creating an IAP application. The STR91x Eval Boards were used to validate the IAP driver

Software is provided and is available on *http://www.st.com/mcu*

# Contents

# 1 IAP overview

## 1.1 Principle

The IAP driver must be programmed by the user in the Flash bank 1 after being hardware remapped at address 0x00 (bank 0 is hardware remapped at address 0x80000).

Programming is performed via JTAG using a flasher toolset.

Using the UART, the IAP driver loads a binary file from HyperTerminal to the STR9 internal Flash bank 0, and then jumps to execute it.

## 1.2 IAP driver description

The IAP driver code size is 11.5 Kbytes

It contains a set of source files:

- *main.c*: where the UART initialization and PLL configuration are set. Then, a main menu is executed from the *common.c* file.
- *common.c*: contains display functions and the main menu routine. The main menu gives the choice between loading a new binary file and executing the binary file already loaded.
- *ymodem.c* and *download.c*: they allow all data to be received from the HyperTerminal (using the YMODEM protocol (see note 1 below) then loaded into STR9 internal RAM. In the event of a failure in data receival, the error message ‘Failed to receive the file’ is displayed in the HyperTerminal window. Otherwise, after successful reception, the data is then programmed into internal Flash at the appropriate address. A comparison between internal RAM and internal Flash contents is performed to check the data integrity. If there is a mismatch, the error message ‘Verification failed’ is displayed in the HyperTerminal window. Other error messages are also displayed when the image size is higher than the allowed space memory and when there is an abort by the user.

*Note: 1 Ymodem protocol sends data in 1024-byte blocks. Error detection is applied to data blocks transmitted to the STR91x internal RAM. This is done through a comparison between the transmitted and received data. Blocks received unsuccessfully are acknowledged with a NAK (Negative Acknowledgement). For more details about the Ymodem protocol, please refer to the existing literature.*

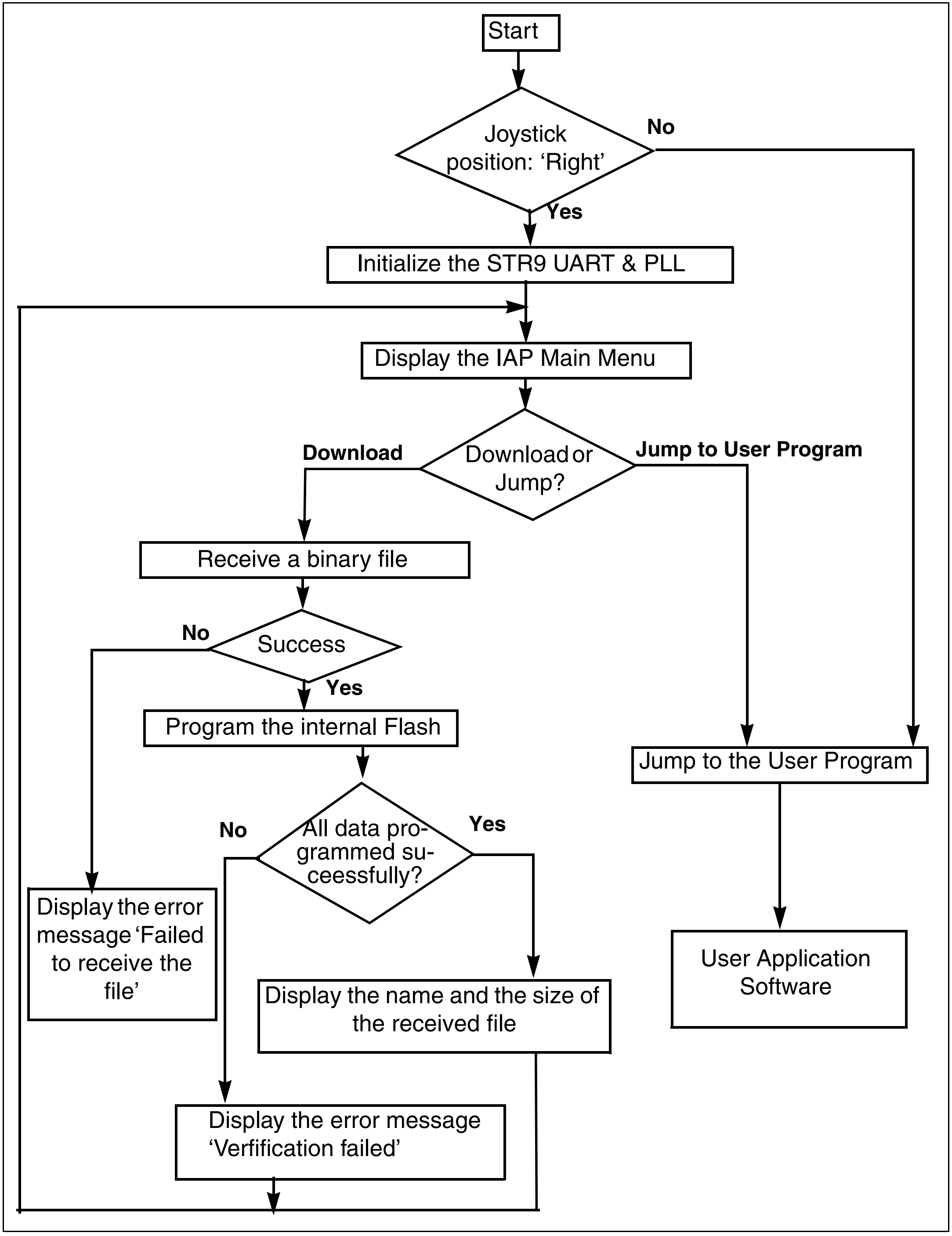
- Some STR91x Standard library source files and header files
  - *91x_fmi.c* and *91x_fmi.h*
  - *91x_gpio.c* and *91x_gpio.h*
  - *91x_scu.c* and *91x_scu.h*
  - *91x_uart.c* and *91x_uart.h*
  - *91x_lib.h*
  - *91x_map.h*
  - *91x_type.h*

4 pins connected to a joystick button are used to select between jumping to the user application and executing the IAP for reprogramming purposes:

- At reset, the joystick button is at position 'Right': The IAP main menu is displayed
- At reset, the joystick button is not at position 'Right': Jump to the user application

*Figure 1* shows the IAP flowchart.

**Figure 1. FlowChart of the IAP Driver**


Start
Joystick position: 'Right'
No
Yes
Initialize the STR9 UART & PLL
Display the IAP Main Menu
Download
Download or Jump?
Jump to User Program
Receive a binary file
No
Success
Yes
Program the internal Flash
Jump to the User Program
No
All data programmed successfully?
Yes
Display the error message 'Failed to receive the file'
Display the name and the size of the received file
User Application Software
Display the error message 'Verfification failed'

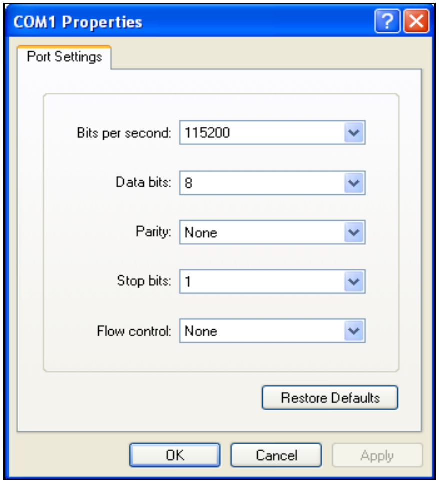
# 2 Running the IAP driver

On STR91x devices, the IAP driver is programmed in bank1 which is hardware remapped at address 0x00.

## 2.1 HyperTerminal configuration

The IAP requires a PC running HyperTerminal with the following settings:

Figure 2. COM Port Properties



Bits per second: 115200

Data bits: 8

Parity: None

Stop bits: 1

Flow control: None

*Note: A baud rate value of 115200 bps is used as an example.*

*Care should be taken when selecting the system clock frequency. Ensure that with the system clock frequency used in the application, a baud rate equal to 115200 bps can be generated in order to guarantee successful communication via UART.*

## 2.2 Triggering execution of the IAP driver

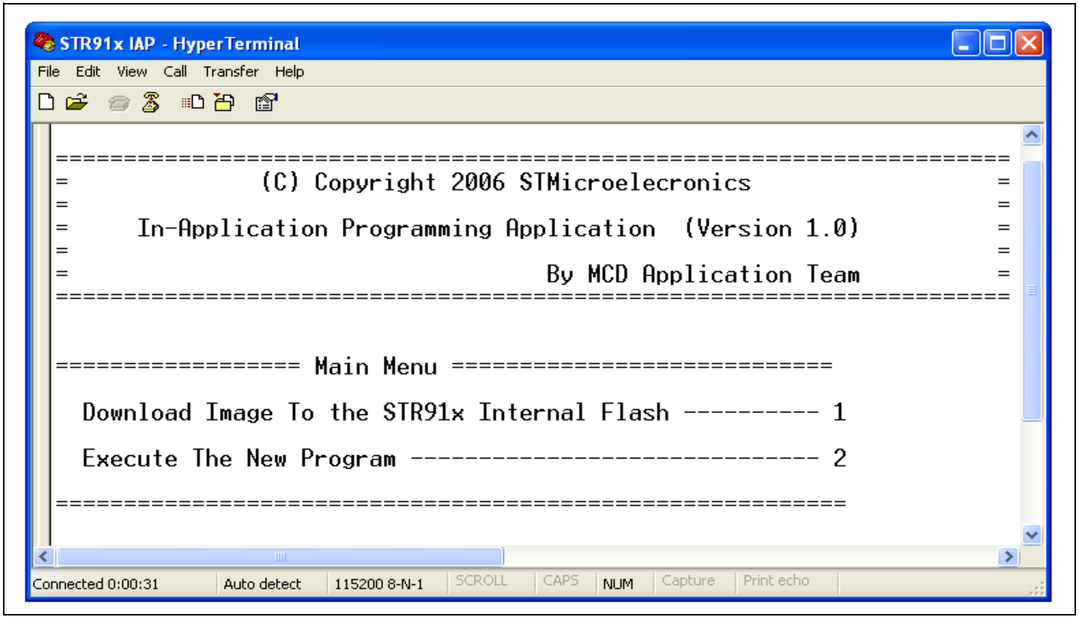
In this application note, a number of port 7 pins connected to the joystick button on the STR91x Eval board are used as an example. This configuration allows the execution of the IAP driver. By keeping the joystick at position ‘Right’ at reset, the user is able to run the IAP driver to reprogram the STR91x internal Flash.

It is not mandatory to use the joystick button; the user can simply enter signals on these pins with respect to their corresponding active levels. Refer to the table in *Section 4 on page 7* for the pins and their active levels.

# 3 IAP driver menu

The execution of the IAP firmware results in the following menu displayed in the HyperTerminal window.

Figure 3. IAP Driver menu



```
======================================================================
=              (C) Copyright 2006 STMicroelecronics                  =
=                                                                    =
=     In-Application Programming Application  (Version 1.0)          =
=                                                                    =
=                               By MCD Application Team              =
======================================================================


================== Main Menu ============================

  Download Image To the STR91x Internal Flash ---------- 1

  Execute The New Program ------------------------------ 2

==========================================================
```

## 3.1 Download image to internal Flash

To download a binary file via HyperTerminal to the STRxxx internal Flash, follow the procedure below:

- Press “**1**” on the keyboard to choose the menu “**Download Image To Internal Flash**”

Now, in the Transfer menu, select “**Send File**”:

- In the **filename** field, type the name and the path of the binary file to be sent.
- In the protocol list, choose the **Ymodem** protocol,
- Click the “**Send**” button.

Following these steps, the IAP driver loads the binary file into the STR91x internal Flash from bank0 sector0 base address and displays the binary file name and file size in the HyperTerminal window.

## 3.2 Execute the new program

After loading the new program from HyperTerminal by selecting the “**Download Image To Internal Flash**” menu, the code can be executed by selecting the “**Execute The New Program**” menu by pressing “**2**“ from the keyboard.

# 4 STR9 IAP implementation Summary

A summary of the STR91x IAP implementation is described in the below table:

| | **STR91x** |
|---|---|
| IAP driver location | Bank 1 |
| User application location | Bank 0 |
| Flash remapping | By default, bank0 is mapped at address 0x00. Given that the IAP driver must be located in bank1 and the user application is located  in bank0, a hardware remapping should be done using the CAPS tool (remap bank1 at address 0x00 and bank0 at address 0x80000).<br>In run time, a software remapping (bank0 at address 0x00 and bank1 at address 0x80000) will be performed just before jumping to the user application and executing it. |
| Push button<br>Active Level | P7 (pins 4, 5, 6, 7): Joystick<br>P7 = 0x60 (Right position) |
| Flash routines | In the IAP driver, the *91x_fmi.c* source file is copied into and executed from the internal RAM. |
| UART used | UART0 |

# 5 User program conditions

- On the STR91x, the IAP driver should be programmed at the bank1 base address after being hardware remapped at address 0x00 and the user application should be programmed from bank0 sector0 base address0. The bank 0 is hardware remapped at address 0x80000. If the user application uses interrupts, there is no problem because the STR91x IAP driver provides software remapping of bank0 at address 0x00 just before jumping to the user application and executing it.
- In the user application to be loaded using IAP, there is no need to remap bank0 and bank1 as it has already been done in the IAP software.

An example will be provided with this application note aiming to show the project settings of a user program which will be loaded in the STR91x internal Flash using IAP.

# 6 IAP restrictions

The maximum size of the image to be loaded in the STR91x Internal Flash is limited by the size of bank0, i.e. 512 Kbytes.

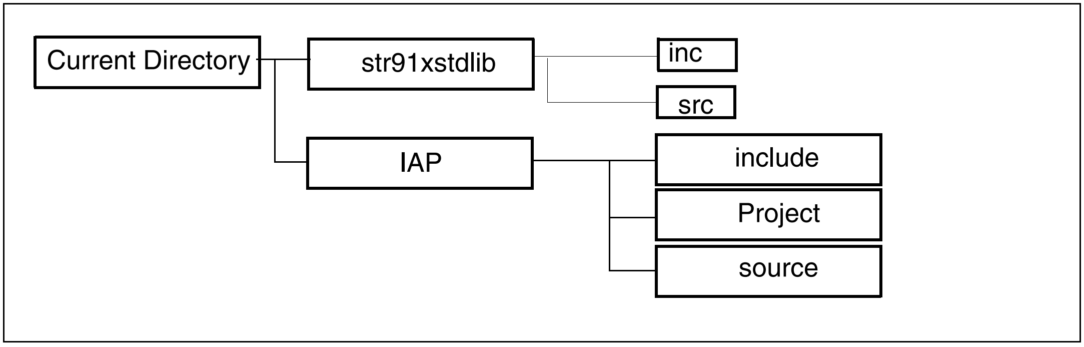
# 7 How to use the IAP driver

The IAP package consists of an archived file containing 2 subfolders:

- **IAP**: This contains the main application and the IAP source and hearder files.
- **str91xstdlib**: This contains the STR91x standard library files used by the IAP driver.

The file structure is described in the following figure:

**Figure 4. IAP driver directory structure**



To use efficiently the IAP driver, please follow the different steps below:

- Remap the bank0 at address 0x80000 and the bank1 at address 0x00 using CAPS tool.
- Download the IAP driver into the STR91x internal Flash via JTAG using a Flasher toolset.
- Open a HyperTerminal window with the settings already defined in the *Section 2.1*
- To run the IAP driver, keep the joystick button at position ‘Right’ at reset. The IAP main menu will be displayed on the HyperTerminal window.
- To download an application, press 1 and use the YMODEM protocol as described in the *Section 3.1*.

Two examples are provided with the application note:

– *Toggling_Led.bin*: This aims to toggle LEDs connected to P9.0, P9.1, P9.2 and P9.3 pins in an infinite loop.

– *STR910-Eval_Demo.bin*: This is the STR91x software demo running on the STR91x Eval board.

# 8 Revision history

Table 1. Revision history

| Date | Revision | Description of changes |
|---|---|---|
| 12-Jan-2007 | 1 | Initial release |

**Please Read Carefully:**

Information in this document is provided solely in connection with ST products. STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, modifications or improvements, to this document, and the products and services described herein at any time, without notice.

All ST products are sold pursuant to ST’s terms and conditions of sale.

Purchasers are solely responsible for the choice, selection and use of the ST products and services described herein, and ST assumes no liability whatsoever relating to the choice, selection or use of the ST products and services described herein.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any part of this document refers to any third party products or services it shall not be deemed a license grant by ST for the use of such third party products or services, or any intellectual property contained therein or considered as a warranty covering the use in any manner whatsoever of such third party products or services or any intellectual property contained therein.

**UNLESS OTHERWISE SET FORTH IN ST’S TERMS AND CONDITIONS OF SALE ST DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY WITH RESPECT TO THE USE AND/OR SALE OF ST PRODUCTS INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.**

**UNLESS EXPRESSLY APPROVED IN WRITING BY AN AUTHORIZED ST REPRESENTATIVE, ST PRODUCTS ARE NOT RECOMMENDED, AUTHORIZED OR WARRANTED FOR USE IN MILITARY, AIR CRAFT, SPACE, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS, NOR IN PRODUCTS OR SYSTEMS WHERE FAILURE OR MALFUNCTION MAY RESULT IN PERSONAL INJURY, DEATH, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE. ST PRODUCTS WHICH ARE NOT SPECIFIED AS "AUTOMOTIVE GRADE" MAY ONLY BE USED IN AUTOMOTIVE APPLICATIONS AT USER’S OWN RISK.**

Resale of ST products with provisions different from the statements and/or technical features set forth in this document shall immediately void any warranty granted by ST for the ST product or service described herein and shall not create or extend in any manner whatsoever, any liability of ST.

ST and the ST logo are trademarks or registered trademarks of ST in various countries.

Information in this document supersedes and replaces all information previously supplied.

The ST logo is a registered trademark of STMicroelectronics. All other names are the property of their respective owners.

STMicroelectronics group of companies

Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia - Malta - Morocco - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

**www.st.com**