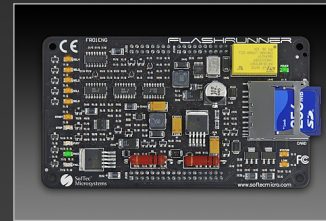


FlashRunner Series

Universal, Standalone In-System Programmers



General Presentation

Rev. 2.0 – May 15th, 2007 – DC01336

ISP Advantages

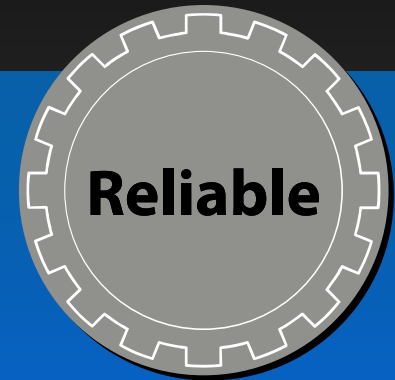
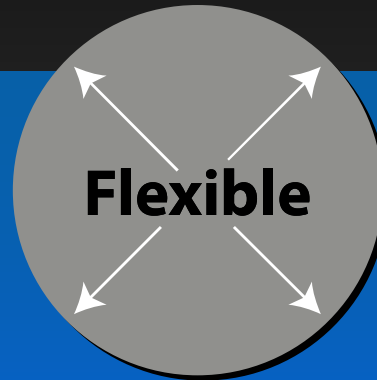
- **Higher Manufacturing Quality**—by elimination of device manual handling
- **Shorter Time-to-Market**—software development overlaps manufacturing startup
- **Lower Cost of Stock**—no need for inventory of different versions of product
- **Higher Integration**—by programming integration into the PCB manufacturing and test flow
- **Higher PCB Testability**—by programming a test firmware prior to real product firmware

Today's ISP Market Offer

- **Some programmers are not truly universal**
- **Some are slow**
- **Some don't allow for fault traceability**
- **Some don't easily integrate into ATE systems**

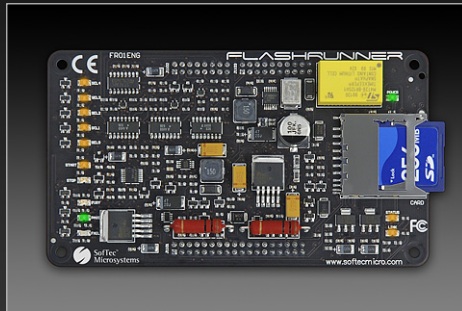
**No single In-System Programmer
fits all Production needs**

FlashRunner Overview



FlashRunner is the First **Complete Production In-System Programmer specifically designed for Manufacturing lines**

FlashRunner Models



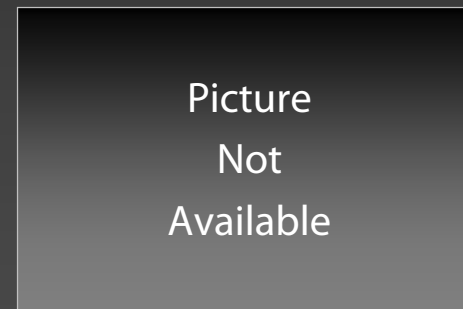
Piggyback Version
(code: FR01ENG)



Ethernet Version
(code: FR01LAN)



Rackmount Version
(code: FR01PRO)



8 Multiplexed Channels Version
(code: FR01M01)

Main Features

- **Universal In-System Programmer Engine**
Supports most popular microcontrollers and serial memories
- **Extremely Fast**
Optimized programming algorithms, as fast as target device's memory technology limit
- **Innovate Logging System**
For detailed error report and traceability
- **Easy ATE Integration**
Mechanical, hardware and software

Additional Features

- **Standalone or host-controlled operations**
- **Supports most ISP protocols (SPI, JTAG, SCI, BDM, I2C, MON08, and other serial interfaces)**
- **Host protocol connections: Ethernet, RS-232**
- **Gangable through Ethernet or control lines**
- **Standard memory card to store projects, binary images, log files**
- **Robust mechanical design**
- **Fully-protected: optoisolated I/O lines and anti-shock protection**
- **Fully upgradable: designed to support new devices and customer-specific programming algorithms**
- **Rapid start-up documentation**
- **Prompt and knowledgeable technical support**

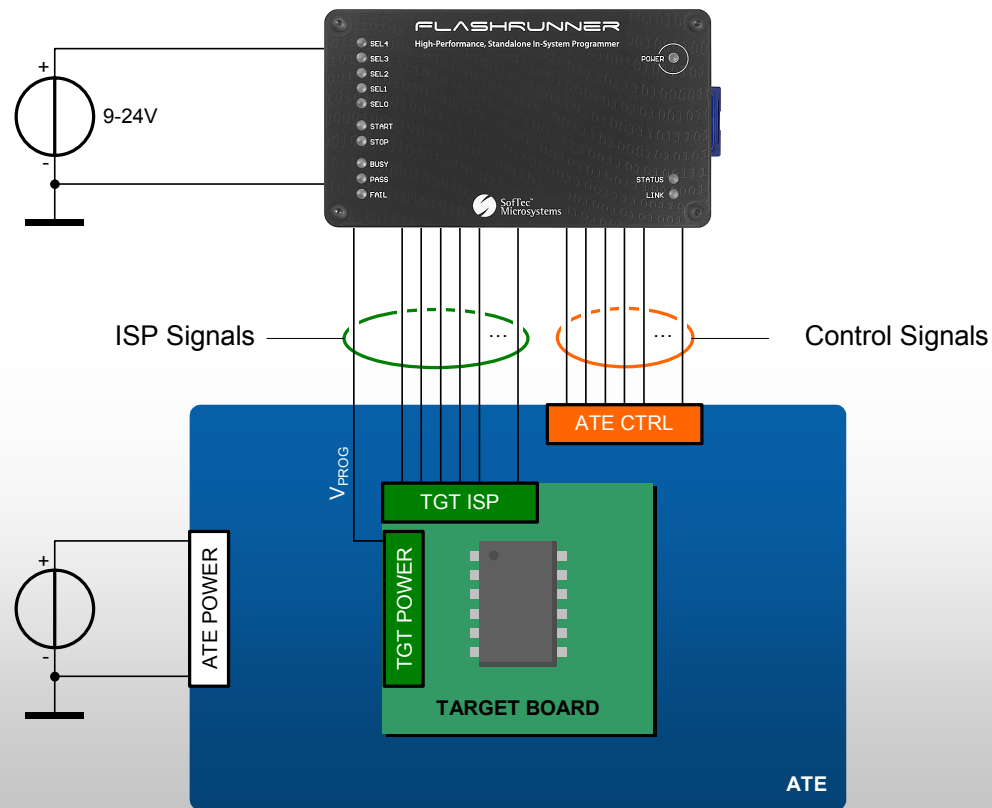
Device Support



Easy ATE Integration

- **Piggyback and Ethernet versions mount easily inside ATE and test fixtures**
- **Compatible with Agilent, Teradyne, Genrad, and any major in-circuit test system**
- **Fully autonomous—a signal from the test system starts the programmer automatically**
- **No external hardware components required**
- **Independent from ATE control system**
- **Standalone or host-controlled operation**
- **Fully upgradable**

Typical ATE Connections



Hardware Features

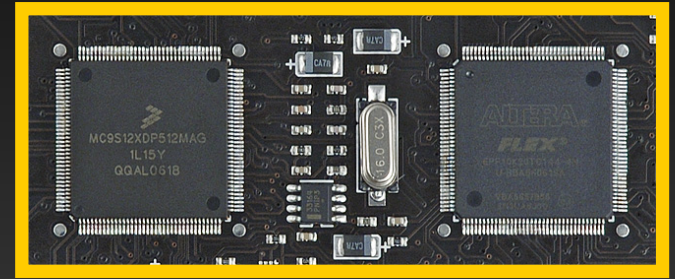
- **Wide range input power supply (9 to 24V)**
- **Expanded ISP connection lines:**
 - Two programmable output voltages
 - Five ISP digital I/O lines
 - Two digital I/O or analog output lines
 - One analog input line
 - One programmable clock output
- **Communication and parallel control lines:**
 - Optoisolated RS-232 or Ethernet
 - Optoisolated inputs for project selection
 - Two optoisolated command inputs (START, STOP)
 - Three optoisolated status outputs (FAIL, PASS, BUSY)
- **512 bytes on-board dynamic memory**
- **On-board timekeeper/calendar**
- **Fully protected against EOS and ESD**

Software Features

- **Controllable by any host system through a terminal utility and simple ASCII commands**
- **Project handling**
(unlimited number of text-based projects)
- **Integrated, full logging system**
(thanks to the built-in real-time clock/calendar)
- **Erase, blank check, program, read, verify, oscillator trimming, etc.**
- **Serial numbering**
- **Data integrity guaranteed**
(every data transfer to/from the SD card is CRC tagged)
- **Ready-to-use device-specific programming algorithms**

In-Depth: Powerful Hardware

- **FlashRunner is based on a dual-core processing engine paired with a field programmable logic**
- **FlashRunner hardware and firmware has been designed to be open and easily reconfigurable**
- **FlashRunner is therefore fully upgradable and ready to support new devices and customer-specific programming algorithms**



In-Depth: Logging

- **FlashRunner's built-in logging system timestamps all operations performed, together with their result**
- **Production problems can thus be traced back to the exact moment they happened, and the cause of the problem can be easily investigated**
- **Logs are stored in the SD card**

```
01:000:06.07.06.15.21.00.00:START LOGGING
01:010:06.07.06.15.21.00.01:TCSETDEV FREESCALE MC68HC908EY16 HC08:>
01:011:06.07.06.15.21.02.15:TCSETPAR FOSC 16000000:>
01:012:06.07.06.15.21.02.20:TCSETPAR FDIV 4:>
01:013:06.07.06.15.21.02.25:TCSETPAR VDD 5000:>
01:014:06.07.06.15.21.02.30:TCSETPAR PWDOWN 10:>
01:017:06.07.06.15.21.02.35:TPSETSRC FILE TEST.FRB:>
01:020:06.07.06.15.21.02.40:TPSTART:>
01:023:06.07.06.15.21.02.41:TPCMD SETPSW CONST $FF $FF $FF $FF $FF $FF $FF $FF:>
01:026:06.07.06.15.21.02.42:TPCMD MASSERASE F:>
01:029:06.07.06.15.21.03.00:TPCMD BLANKCHECK F $C000 16384:>
01:032:06.07.06.15.21.05.32:TPCMD TRIM 4000000 $FF80:>
01:035:06.07.06.15.21.06.01:TPCMD PROGRAM F $C000 $C000 16384:>
01:038:06.07.06.15.21.08.54:TPCMD VERIFY F R $C000 $C000 16384:>
01:041:06.07.06.15.21.09.88:TPEND:>
01:000:06.07.06.15.21.09.89:END LOGGING:OK
```

Log Example

In-Depth: Easy Scripts

```
;  
; FLASHRUNNER SCRIPT EXAMPLE FOR FREESCALE MC68HC908EY16  
;  
; Use this example as a starting point for your specific programming needs  
;  
  
#LOG_ON 2 ; Turns off logging  
#HALT_ON FAIL ; Halt on errors  
  
TCSETDEV FREESCALE MC68HC908EY16 HC08 ; Sets device  
TCSETPAR FOSC 16000000 ; Oscillator frequency, Hz  
TCSETPAR FDIV 4 ; Oscillator divisor  
TCSETPAR VDD 5000 ; Target voltage, mV  
TCSETPAR PWDOWN 10 ; Power-down time, ms  
TCSETPAR PWUP 10 ; Power-up time, ms  
  
TPSETSRC FILE TEST.FRB ; Image file to be programmed  
  
TPSTART ; Starts programming block  
  
TPCMD SETPSW CONST $FF $FF $FF $FF $FF $FF $FF $FF ; Security bytes  
TPCMD MASSERASE F ; Mass erases Flash memory  
TPCMD BLANKCHECK F $C000 16384 ; Blank checks Flash memory  
TPCMD TRIM 4000000 $FF80 ; Trims internal oscillator  
TPCMD PROGRAM F $C000 $C000 16384 ; Programs Flash memory  
TPCMD VERIFY F R $C000 $C000 16384 ; Verifies Flash memory, read-out method  
  
TPEND ; Ends programming block
```

This simple script it's all that is needed to mass erase, blank check, trim, program and verify a Freescale HC08 device. The script is stored on the FlashRunner SD card and is launched via either control lines or serial command.