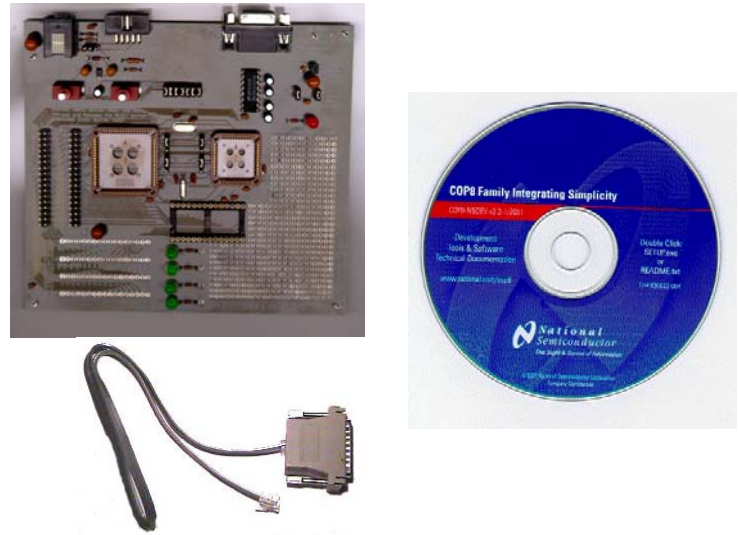


## INTRODUCTION

The COP8-SKFLASH-01 Starter Kit provides a low cost, easy-to-use environment for quickly developing and debugging COP8FLASH applications.

The development environment includes WCOP8 Windows IDE, with COP8 Assembler, Linker, Librarian, Simulator, COP8 ISP Utility, and NiceMon Debug Monitor Utility. Use the comprehensive online tutorial and sample source code to quickly develop and simulate your application, then download your code (with COP8Flash In-System-Programming) to your choice of COP8Flash devices on the 5v target board (\*see NOTES for <5v operation). Debug your application in a dual-clock, realtime environment with breadboard area, 4 LEDs, and probe pins.

Use the NiceMon Debug Monitor utility for simple (one breakpoint; single-step) debugging from your PC, or for full-featured, non-intrusive debugging you can easily connect one of our economical realtime COP8FLASH Emulators.



### COP8-SKFLASH-01 Starter Kit

Includes:

- COP8-SKFLASH-01, 5v Target Board with Dual Clock
- COP8CDR (68PLCC), and COP8AME (28DIP) Samples
- Target ISP cable for PC parallel port
- COP8-NSDEV Development CD
- SKFlash-01 QuickStart Manual

## SOFTWARE INSTALLATION

The enclosed CD contains the COP8 IDE tools, optional tools demos, datasheets, app notes, example source code, and other documentation (see NOTES section). All you need is a standard PC running win95/98/me/2k/nt/xp with minimum 32MB of disc space. Check National's COP8 web site for the latest software versions (\*See notes).

**1. WCOP8 IDE and Tools.** Includes Assembler, Linker, Librarian, Simulators, COP8-ISP software, NiceMon Debug Monitor, COP8 PM Programmer support, and COP8 Realtime Emulator support. Insert the CD, click on "Install Tools", and follow directions. All are

automatically installed and configured under WCOP8 IDE.

**2. C Compiler Demos, Processor Expert Demo, and Reference Designs.** Insert the CD, click on "Install Tools", and follow directions. Instead of the default install option, select "Custom" and select the demos and reference designs you want to install. All are automatically installed.

- a. Byte Craft C-Compiler demo.
- b. IAR Embedded Workbench™ with C Compiler demo
- c. A fully functional COP8 assembler-only version of IAR Embedded Workbench™.
- d. UNIS Processor Expert™ demo.
- e. COP8 Reference Design software/firmware.

**3. Software Source Code Modules, and Applications Source Code.** Insert the CD and browse to:

- a. /tools/skflash. Preconfigured demo code. (This code was designed for the SKFlash-00 board, but will work with the SKFlash-01 board).
- b. /software/an....sw. Source code for many of the app notes in /appnotes.
- c. /software/utility. Sample .asm and C source utilities.
- d. /software/\*.\*. Other sample source code and applications.

**5. Technical Documentation and Literature.** Insert the CD, click on “Read docs”. Select the desired documentation and save to your disk as a .pdf file. Other documents of interest:

- a. COP8-SKFlash-01 PCB Schematics and Drawings. (/tools/skflash)
- b. COP8 Flash ISP User’s Guide, and other ISP App notes (/appnotes/an11...)
- c. NiceMon User’s Manual (/tools/flashmon)

**6. Getting the latest documentation and software.** Go to: [www.national.com/cop8](http://www.national.com/cop8)

## HARDWARE SETUP

Install only one COP8Flash device into an on-board socket. The 68pin PLCC COP8CDR9 devices are superset products for the COP8C and COP8S Flash families, and support all of the family features (except Brownout). The 28 DIP COP8AME9 is the same as the COP8ANE9 plus Brownout. For optional COP8T Flash Family support, refer to the COP8 web site for details (\*see NOTES).

The Clock Configuration jumpers J11, J12, J13, J14 should be in the default positions (32KHz, XTAL, 10MHz XTAL) for on-board crystal, dual-clock operation. (See the Target Board Details section for optional configurations).

Connect the included cable to a PC parallel port (make sure the LPT port is enabled, and configured for Standard (Unidirectional), or EPP mode). Connect the other end to the RJ11 ISP connector on the target board.

**Important:** Insure that J16 (ISP jumper) is shorting 2-3. (See ISP Section below).

Connect an optional COP8 Flash Emulator to the 14 pin header on the target board, otherwise the header pin jumpers should be installed.

Jumpers J10 and J15 can be left installed, unless the breadboard area will require separate Analog and Digital grounds.

The 5v target board requires any 5v-25v DC, regulated/unregulated power supply, or use a 9-volt battery. Connect the power supply connections to the GND and VCC 5v Regulator Inputs. Check that the red LED is on.

**NOTE:** Operating voltages below 5v are possible by replacing the 5v regulator, or simply bypassing it with a regulated supply.

**IMPORTANT:** Below 4.5v, to avoid violating the flash memory specs, the 10MHz Xtal must be replaced with a 3.33MHz Xtal.

Refer to your device datasheet for details.

## GETTING STARTED

First, set up your environment for code development. Refer to the IDE Help files for information and tutorials. National’s COP8 IDE supports a complete set of tools, including Simulator, ISP programming, NiceMon Debugger, and optional Byte Craft C-Compiler or Unis Processor Expert.

Next, test out your software/hardware installation by using the IDE and COP8-ISP software to download a test startup program to the target board. The program's Intel .hex file can be found in the default cop8isp directory on your PC: C:\cop8\cop8isp..\samples\src\kitcar\kitcar.hex (or download the file from the COP8 web site.)  
Continued...

On the target board, connect the J2/D0 test point to one of the 4 LEDs (TP6-9). Follow the instructions below for ISP, then start the program. The LED should flash.

You are now ready to develop, download, and debug code!

For more information on developing COP8 code, and COP8Flash architecture, refer to our "COP8FLASH Family User's Manual" and "COP8 Assembler/Linker/Librarian User's Manual" on the CD-ROM.

### Options for starting code development:

- 1) Start with COP8-SKFlash QuickStart Programs. Mix and match as needed.
- 2) Browse the CD for applications and source code that match your desired application. Use and modify as needed.
- 3) Browse the CD for source code example modules to build your application in pieces.
- 4) Start from scratch using the COP8 Assembler (or use an optional C-Compiler or Unis Processor Expert™).

## DOWNLOADING CODE via ISP

COP8-ISP Software, or NiceMon ISP Mode, is used to download your Intel .hex code files into the Flash device. It is recommended that you first read the COP8 Flash Application Note an1150 before downloading code.

- 1) Apply power to the target board.
- 2) Connect the ISP cable from your PC parallel port to P3 (RJ11).
- 3) Move J16 jumper to position 1-2.
- 4) Press and HOLD SW2 (2xVCC).
- 5) Press and Release SW1 (RESET)
- 6) Release SW2 (2xVCC)
- 7) Move J16 jumper back to position 2-3.  
**Do not push RESET again, or cycle power, or you must start over at step 3.**
- 8) Start the Flash ISP Software, and select Programmer/"COP8 Direct LPT".
- 9) Click on "Options Register" and set the Write Timing Register to 10MHz (\*see NOTES section for other frequencies).
- 10) Click on "Status" and confirm the device is detected.
- 11) Select File/Load and download your application code Intel .hex file.
- 12) **Manually configure the Options Register; Enable the Flex bit.**
- 13) If this is the first use of cop8isp, go to the Device/Auto-Program-Options menu and select: ERASE, PROGRAM, VERIFICATION, OPTIONS BYTE, RUN.
- 14) Pushing the F5 key now will automatically erase, program, and verify the device, then start your application.

**\* Important Note:** If your application uses the G4,G5,G6 pins, do not select RUN in the Auto Program menu. Instead, remove the ISP cable and start the application by pushing the on-board Reset button.

### NOTES:

- \* The Flex bit MUST be Enabled and Programmed for your application to run.
- \* This ISP procedure temporarily overrides the Flex bit, and forces execution from the COP8's ISP boot ROM. This is the ONLY time that SW2 (G6 High Voltage) is used.
- \* Downloading a .hex file does not automatically configure the Options Register. It must be manually configured.
- \* Using Device/Program/Flash (instead of F5) does not program the Options Register. You must also use Device/Program/Options.
- \* Using Device/RUN (F9) will automatically set the Flex bit, program the Options Register, and start the application. (This can be used instead of using Device/Program/Options.)
- \* The device MUST always be Erased before Programming. Using F5 makes sure this is done only if Erase is selected in Auto-Program-Options.

### If You Have Problems with ISP:

- 1) Check the position of J16.
- 2) Did you set the Clock Freq correctly?
- 3) Check the cable connections.
- 4) Check your parallel port settings.
- 5) Did you set the FLEX bit properly?
- 6) Did you erase the part first?
- 7) If all else fails, try replacing the COP8 device.

## TARGET DEBUGGING

### Notes:

- 1) If your application uses the G4,G5,G6 pins, the ISP cable should be disconnected from the target board. (voltages from the PC's parallel port can interfere with your application.) Start the application by pushing the on-board Reset button.
- 2) If you are not using pins G4,G5,G6, you can leave the cable connected.
- 3) If you are using NiceMon for debugging, the cable must be left connected.

The SKFLASH-01 target board comes configured for 5v, dual-clock, realtime operation. Jumpers are provided for optional external clock configurations, with an onboard Reset pushbutton (SW1) for easy Reset/Restart.

Test points for each of the device's 68 pins are brought out for easy probing and connection to your customized circuits in the breadboard area. The test point headers can also be used for custom daughter boards, or connection to external applications.

The four LEDs can be connected to any output pin test points for visual debugging.

An RS232 connector is provided for external communications, and the RJ11 connector can be used as a Microwire/PLUS interface.

When you are ready to download a new version of your code, you must re-connect the ISP cable,

and re-initialize the device for ISP execution from the boot ROM (instructions are on the PCB). This is the ONLY time SW2 (2xVCC) is used.

### NiceMon Debug Monitor

The NiceMon User Control Interface (UCI) runs on your PC, and provides COP8Flash device program download, debug control, and internal flash/register visibility. The NiceMon Debug Monitor is a separate piece of firmware code that is automatically downloaded into flash memory with your application code, and interfaces with the UCI through the Microwire/PLUS interface (the RJ11 ISP connector).

Your application code executes normally, in realtime, without intervention from the monitor. Control is transferred to the UCI when a break point is reached or the UCI requests control, such as Halt or Single-Step.

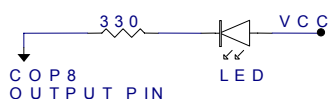
The Debug Monitor and UCI includes the following features:

- Read/Write/Erase Byte, Page, or All
- Reset/Stop/Run/Single-Step.
- Specific-address break point.
- Variables Watch Area with Modify
- PC and Register Status Bar with Modify
- Source/Program/Symbols files display
- Console Area with a Scheme Command Interpreter , and message logging.

## DEBUGGING TIPS

If you don't have a scope handy for probing output levels, one or more of the on-board

### LED Indicators for Visual Probing



LEDs can be attached to the device output probe pins for easy visual observation of output signals.

With some practice, you can quickly determine what is happening on each pin.

- LED Bright = Vol
- LED Dim = Fast Switching
- LED Off = Voh
- LED Flashing = Slow Switching



---

## **TARGET BOARD DETAILS** (see schematics and PCB silkscreen for details)

1. **Power** - 5-25v DC external power source attached to TP1 and TP5. <5v sources can be directly attached to the output of the 5v regulator (bypassing the regulator), or a lower voltage regulator can be installed. The red LED indicates power to the PCB.
2. **Breadboard Area** - The breadboard area has multi-layer ground and power planes for noise-free digital and analog circuit operation. Installing J15 and J10 provides a common power/ground . Removing one (or both) of J15 and J10 allows separate Analog and Digital power and grounds. DGND and AGND test points are located at the corners of the PCB.
3. **Clock Sources** - J11-J14 provide flexible clock configurations, with a combination of on-board and external clock sources. The jumpers come installed for on-board 10MHz and 32KHz, dual-clock operation. Other crystal frequencies can be used, or external clocking can be customized.
4. **Application Reset Button** - A simple reset to the device (SW1). **If the Flex bit is Enabled, your application code will run. If the Flex bit is not enabled, the device will start up in the ISP mode.** (SW1 is also used with SW2 to force a programmed device into the ISP mode.)
5. **ISP Reset Button** - When J16 is in position 1-2, pushing SW2 connects a 2xVcc voltage to the device G6 pin. **This button is used ONLY for forcing a programmed device into the ISP mode.** When J16 is in position 2-3, this button is isolated from the G6 pin, and is not connected to anything.
6. **Device Sockets** - The three device sockets are connected in parallel, sharing the clock circuits, Reset, ISP connections, I/O, and test points. Only one device should be inserted at any time.
7. **Test Points** - Two sets of 64 test point headers are provided, with common connections. One set has dual headers installed for easy attachment of external circuits or daughter boards. The other set allows custom connection to the breadboard area.
8. **4 LEDs** - The four LEDs are connected to DVCC only. TP6-TP9 can be connected to any device outputs, or your own circuits, for visual debugging.
9. **RS232 Port** - A DB9 serial connector (P1) and on-board circuitry (U5) provide a turn-key RS232 interface. UART Software drivers are available on the CD.
10. **RJ11 Connector** - This connector (P3) is tied to the G4, G5, G6 device pins, and supports the Microwire/PLUS™ interface for ISP, or any other external application.
11. **KANDA ISP Connector** - This 2x5 connector (P2) is for the Kanda COP8-ISP programming tool. This tool supports both Parallel and Serial PC connections, a wide variety of clock frequencies, and eliminates the need for the manual ISP reset operation.
12. **COP8 Flash Emulator Header** - This 2x7 header (J1) allows connection of a realtime, full-featured emulator for full, interactive debugging. **When not used, the jumpers must be installed.**

---

## **NOTES**

### 1) <5v Operation with COP8S and COP8C Flash devices.

Non-Brownout (COP8SDx, COP8CDx) and Low-Brownout (COP8SBx, COP8CBx) Flash devices operate down to 2.7v, but not at full speed. If the board is set up for less than 4.5v operation, the 10MHz crystal must be replaced with a 3.33MHz crystal. This meets the datasheet specs, and insures the correct flash timing parameters. (The COP8 can actually run faster at low voltages, and will erase and program the flash, but the effects on the flash memory over time are unknown.) **Hi-Brownout versions (COP8SCx, COP8CCx), and COP8A flash devices are 5v only devices.**

### 2) Low voltage RS232 operation.

At low voltages, the RS232 interface may require its own external power source for correct operation.  
Continued....

### 3) 2.25v-2.75v Operation with COP8T Flash devices.

The COP8T flash family devices are designed to operate at full speed from 2.25v-2.75v max. It is not necessary to change the 10MHz crystal, but the voltage regulator must be changed (or bypass the regulator). The COP8T flash devices are not available in PLCC or DIP, so a package adapter must be used. Package adapter information, and an application note on this subject, are available from National's web site.

4) **The following software programs are automatically installed from the enclosed CD.** Updates can be downloaded from National's web site at [www.national.com/cop8](http://www.national.com/cop8).

1. WCOP8 IDE from K&K Development ApS
2. COP8 Assembler from National Semiconductor Corp.
3. COP8 Simulator software for Windows 32 bit.
4. COP8 Emulator/Debugger software for Windows 32 bit.
5. COP8 PM software for Windows 32 bit.
6. COP8FLASH NiceMon for Windows 32 bit from Ammunerve
7. COP8FLASH ISP for Windows 32 bit from Kanda Systems Ltd.
8. COP8 Simulator for Windows 32 bit from Unis s.r.o.

5) **The following optional tools can be installed using the Custom option:**

1. COP8C C-Compiler (DEMO) from Byte Craft Limited
2. EWCOP8 Embedded Workbench (DEMO) from IAR Systems
3. EWCOP8A Embedded Workbench COP8 Assembler-only version.
4. COP8 Processor Expert with Simulator (DEMO) for Windows 32 bit from Unis s.r.o.
5. COP8FLASH FL1 Reference software from A Brand New World(ICU)
6. COP8-REF-AM Reference software from A Brand New World(ICU)
7. COP8-REF-AJ Reference software from A Brand New World(ICU)
8. COP8-REF-SGR Reference software from A Brand New World(ICU)
9. COP8-REF-ACC Reference software from A Brand New World(ICU)