

MC68HC912/9S12 FLASH/EEPROM Programmer

User's Guide

TABLE OF CONTENTS

1. PREFACE	3
2. CHECKLIST AND REQUIREMENTS	3
3. INSTALLATION AND USE	4
3.1 PROGRAMMER CHECK AND CONNECTION TO PC.....	5
3.2 PROGRAMMER FIRMWARE UPDATE.....	5
4. INTERFACE TYPES	6
4.1 IN-CIRCUIT PROGRAMMING	6
4.2 ON-BOARD PROGRAMMING.....	6
5. WORKING WITH TARGET MCU	7
5.1 MC68 HC912 Devices.....	7
5.1.1 In-Circuit EEPROM/FLASH reading example	7
5.1.2 In-Circuit EEPROM/FLASH programming example.....	7
5.1.3 On-Board EEPROM/FLASH reading example	8
5.1.4 On-Board EEPROM/FLASH programming example	8
5.2 MC 9S12 Devices	9
5.2.1 Features of MC9S12xx128 Devices with 0L85D/1L85D masksets	9
6. FILE OPERATIONS	11
6.1 LOAD FILE INTO BUFFER	11
6.2 SAVE FILE FROM BUFFER.....	11
7. ERRORS AND TROUBLESHOOTING	12
8. WARRANTY STATEMENT.....	14
9. APPENDIX.....	15
Figure 2. In-Circuit programming schematic diagram for HC912/9S12 MCU.....	15
Figure 3. In-Circuit wiring diagram for MC68HC912B32, 80-PIN QFP Package	16
Figure 4. In-Circuit wiring diagram for MC68HC912D60/DG128, 112-PIN TQFP Package.....	17
Figure 5. In-Circuit wiring diagram for MC68HC912D60A/DG128A, 112-PIN LQFP Package..	18
Figure 6. In-Circuit wiring diagram for MC68HC(9)12D60, 80-PIN QFP Package	19
Figure 7. In-Circuit wiring diagram for MC68HC912D60A, 80-PIN QFP Package	20
Figure 8. In-Circuit wiring diagram for MC9S12Dx64/Dx128/Dx256, 80-PIN QFP Package	21
Figure 9. In-Circuit wiring diagram for MC9S12Dx64/Dx128/Dx256, 112-PIN LQFP Package..	22
Figure 10. In-Circuit wiring diagram for MC9S12H-Family 112-PIN LQFP Package	23
Figure 11. In-Circuit wiring diagram for MC9S12H-Family 144-PIN LQFP Package	24

1. PREFACE

This manual will guide you through the installation and operation of the ETL MC68HC912/9S12 Programmer, referenced hereafter as the **HC912-Programmer**.

The **HC912-Programmer** has been designed for Reading, Programming of FLASH, EEPROM contents of the next Motorola Microcontroller Unit (MCU):

- | | |
|-----------------------------------|--------------------------------|
| ✓ MC68HC912 DC128A (3K91D) | ✓ MC9S12 DG128B (0L85D) |
| ✓ MC68HC912 DC128 (0K50E) | ✓ MC9S12 DT128B (0L85D) |
| ✓ MC68HC912 DG128 (5H55W) | ✓ MC9S12 A128B (0L85D) |
| ✓ MC68HC912 DG128A (3K91D) | ✓ MC9S12 DB128B (0L85D) |
| ✓ MC68HC912 D60A (2K38K) | ✓ MC9S12 DT128B (1L85D) |
| ✓ MC68HC912 D60 (0K75F) | ✓ MC9S12 DG256C (2K79X) |
| ✓ MC68HC912 D60 (0K13J) | ✓ MC9S12 DT256C (2K79X) |
| ✓ MC68HC912 D60 (4F73K) | ✓ MC9S12 DP256C (2K79X) |
| ✓ MC68HC912 B32 (4J54E) | ✓ MC9S12 DP512 (1L00M) |
| ✓ MC68HC912 B32 (9H91F) | ✓ MC9S12 D64 (2L86D) |
| | ✓ MC9S12 DT128B (3L40K) |
| | ✓ MC9S12 H128 (1K78X) |
| | ✓ MC9S12 H256 (1K78X) |



Note: Most number of devices can be programmed in two operating modes In-Circuit and On-Board.



Note: On-Board programming must be used when device secured or BDM module disabled only. See Section 4.2 for details.



Note: Devices that not mentioned above in list can't be guaranteed of correct reading, programming by **HC912-Programmer**.

2. CHECKLIST AND REQUIREMENTS

The following describes what items are supplied with the **HC912-Programmer** and the system requirements if used by a PC.

- ✓ **HC912-Programmer** – *supplied*
- ✓ Two HC912 QFP112 Adaptors – *supplied*
- ✓ Two MC9S12 112QFP Adaptors– *supplied*
- ✓ Cable -A DB9 “straight-thru” cable - *supplied*
- ✓ **HC912-Programmer** PC software on CD-ROM – *Optional Extra*
- Desktop PC and a free Serial Communication Port (COM1...8)
- Memory - Minimum 32 Mbytes
- Display - Color SVGA display recommended
- Power supply 12 Volt/500 mA linear power supply source
- OS -MS-Windows (Win98, Win2000/XP/2003)

3. INSTALLATION AND USE

The HC912-Programmer includes two LED's and mode jumper (Figure 1). Color LED's indicates programmer state and external power supply voltage (Table 1, 2).

GREEN	HC912-Programmer is ready.
GREEN Flashing	HC912-Programmer is busy.
RED	Error occurred while operation.
DARK	Voltage applied to HC912-Programmer lower than 6 Volt.

Table 1. LED D1 color meaning

GREEN	Motorola device powered on.
DARK	Motorola device powered off.

Table 2. LED D4 color meaning

Jumper-1 (JP1) is intended for HC912-Programmer mode selection (see Table 3).

JP1 Shorted	Normal operation. Motorola device EEPROM/FLASH Reading/Programming.
JP1 Opened	Service mode. HC912-Programmer firmware update.

Table 3. HC912-Programmer operation modes



Note: When Service mode selected LED D1 will be **RED**.

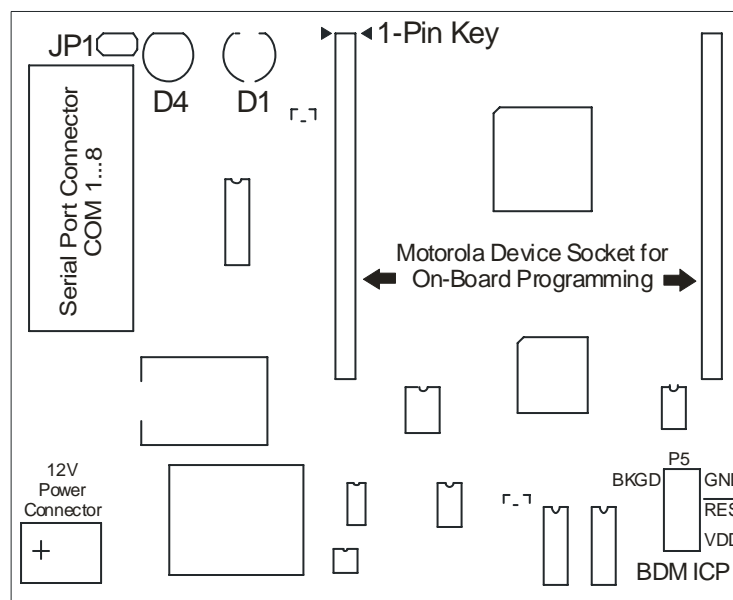


Figure 1. The HC912-Programmer board layout

3.1 PROGRAMMER CHECK AND CONNECTION TO PC

- Connect the power supply source to **HC912-Programmer** (an external 12 V DC power supply source is required).
- Attach a COM port cable to the 9-pin connector on the programmer and to a COM port on the PC.
- Insert jumper JP1 (see Figure 1).
- Remove any adaptor from Motorola device socket.
- Turn On power supply source and make sure that LED D1 appear green.
- Start **HC912-Programmer** software.
- After few seconds you should see on display message: “MC68HC912 Programmer Ver-X.X detected”.
- In case when you should see message: “MC68HC912 Programmer Ver-X.X not found”, change COM port number. To do that select “Tools” menu item and than select “Comm Port Options”. Select new COM port number. After pressing “OK” button the new setting will be applied and software reattempts connection with programmer.
- Now **HC912-Programmer** ready to operate.



Note: To speed up connection between **HC912-Programmer** and PC communication port baud rate must be set to maximum value.



Note: Update **HC912-Programmer** firmware if required (see Section 3.2).

3.2 PROGRAMMER FIRMWARE UPDATE

This section describes how to update firmware (ATMEGA and XILINX) of **HC912-Programmer**.

- Remove JP1 (see Figure 1).
- Turn On power supply; make sure that LED D1 appears red.
- Start **HC912-Programmer** software.
- Compare firmware version “MC68HC912 Programmer Ver-X.X detected” and version specified in “Help>About” menu item.
- If versions not coincide, follow next steps to update firmware.
- Select menu item “Tools>Firmware Update”.
- After message box “Firmware Update” appeared, press “OK” button.
- After firmware update completed, turn Off power supply, close **HC912-Programmer** software.
- Insert JP1.
- Turn On power supply. Make sure that LED1 appear green.
- Start **HC912-Programmer** software on PC. New firmware version “MC68HC912 Programmer Ver-X.X detected” will appear.

4. INTERFACE TYPES

This section describes two interfaces, In-Circuit Programming (ICP) and On-Board Programming (OBP) of HC912-Programmer.

4.1 IN-CIRCUIT PROGRAMMING

In-Circuit programming interface is basic for HC912-Programmer. With this interface HC912-Programmer automatically detects target MCU bus speed. Ceramic resonator connected to target MCU must be in range from 2 MHz to 16 MHz. If target MCU secured (9S12 devices) or BDM module disabled (912 devices) there is no way to establish connection between MCU and HC912-Programmer.



Note: ECLK Pin on target MCU must be connected to circuit via resistor 1Kohm or higher to avoid damaging of this pin.



Note: When using In-Circuit programming interface remove adaptor from Motorola device socket (see Figure 1).

4.2 ON-BOARD PROGRAMMING

On-board programming interface designed for establishing connection with target device when MCU secured (9S12 devices) or BDM module disabled (912 devices). This interface allows Read/Program EEPROM/FLASH without any restrictions. To work with this interface type MCU must be mounted to the corresponding QFP adaptor supplied with HC912-Programmer.



Note: When target adaptor with MCU mounted, check contacts careful to avoid short circuit. Otherwise target MCU can be damaged!



Note: When On-board programming interface used, remove cable from BDM ICP connector (see Figure 1).

5. WORKING WITH TARGET MCU

This section contains overall information about Motorola MCUs supported by HC912-Programmer. When HC912-Programmer successfully installed (see Section 3) target devices can be read, program and verify.

5.1 MC68 HC912 Devices

This section describes basic rules working with next devices:

- ✓ MC68HC912 **DC128A** (3K91D)
- ✓ MC68HC912 **DC128** (0K50E)
- ✓ MC68HC912 **DG128** (5H55W)
- ✓ MC68HC912 **DG128A** (3K91D)
- ✓ MC68HC912 **B32** (4J54E)
- ✓ MC68HC912 **D60A** (2K38K)

FLASH/EEPROM memory for these devices can be read, program in both OBP and ICP interfaces.



Note: On-Board programming interface not implemented for MC68HC912B32 device.

Extra care must be taken when working with EEPROM Shadow Word/Byte in ICP interface. If BDM Lockout bit (NOBDML) programmed to zero, further access to MCU will be blocked after next reset. The only one way to grant access to locked MCU is mount MCU on corresponding adaptor and use On-Board programming interface.

5.1.1 In-Circuit EEPROM/FLASH reading example

- Connect required pins to board with target MCU (see Appendix Figure 2).
- Remove any adaptor from Motorola device socket (see Figure 1).
- Apply power to HC912-Programmer. LED D1 became to green light (see Table 1).
- Select corresponding device in HC912-Programmer software (“Device” button).
- Select “Read Sequence” panel in HC912-Programmer software.
- Press “EEPROM/FLASH” button.
- Press “Start” button.
- LED D4 on HC912-Programmer became to green light. That means that +5 voltage regulator switched on and VCC applied to target MCU.
- Now HC912-Programmer automatically detects MCU bus speed.
- When LED D1 on HC912-Programmer became permanent green, reading completed. Target MCU powered off.
- After read sequence successfully completed it is necessarily to save memory dump to file (see Section 6.2).
- If some errors appeared during reading process refer to Section 7.

5.1.2 In-Circuit EEPROM/FLASH programming example

- Connect required pins to board with target MCU (see Appendix Figure 2).
- Remove any adaptor from Motorola device socket (see Figure 1).
- Apply power to HC912-Programmer. LED D1 became to green light (see Table 1).
- Select corresponding device in HC912-Programmer software (“Device” Button).

- Load EEPROM/FLASH data from file (see Section 6.1) or enter data to Hex Editor.
- Select “Program Sequence” panel in [HC912-Programmer](#) software.
- Press “EEPROM/FLASH” button.
- Press “Start” button.
- LED D4 on [HC912-Programmer](#) became to green light. That means that +5 voltage regulator switched on and VCC applied to target MCU.
- Now [HC912-Programmer](#) automatically detects MCU bus speed.
- When LED D1 on [HC912-Programmer](#) became permanent green, programming completed. Target MCU powered off.
- If some errors appeared during programming process refer to Section 7.



Note: Extra care must be taken when programming the EEPROM Shadow Word/Byte.



Note: Strongly recommended before EEPROM/FLASH programming, for a first time, read EEPROM/FLASH contents and save it to file.

5.1.3 On-Board EEPROM/FLASH reading example

- Mount target MCU on corresponding QFP adaptor.
- Insert adaptor to Motorola Device Socket on [HC912-Programmer](#) in according to 1 pin-key (see Figure 1).
- Remove cable from BDM ICP connector (see Figure 1).
- Apply power to [HC912-Programmer](#).
- Select device in [HC912-Programmer](#) software (“Device” Button).
- Select “Read Sequence” panel in [HC912-Programmer](#) software.
- Press “EEPROM/FLASH” button.
- Press “Start” button.
- LED D4 on [HC912-Programmer](#) became to green light. That means that +5 voltage regulator switched on and VCC applied to target MCU.
- When LED D1 on [HC912-Programmer](#) became permanent green light, reading completed.
- When read sequence successfully completed it is necessarily to save memory dump to file (see Section 6.2).
- If some errors appeared during reading process refer to Section 7.



Note: When target MCU mounted on adaptor check contacts careful to avoid short circuit. Otherwise target MCU can be damaged!



Note: During removing the adaptor with target MCU [HC912-Programmer](#) must be powered off to avoid damaging of programmer and target MCU!

5.1.4 On-Board EEPROM/FLASH programming example

- Mount target MCU on corresponding QFP adaptor.
- Insert adaptor to Motorola Device Socket on [HC912-Programmer](#) in according to 1 pin-key (see Figure 1).
- Remove cable from BDM ICP connector (see Figure 1).
- Apply power to [HC912-Programmer](#).
- Select device in [HC912-Programmer](#) software (“Device” Button).
- Load EEPROM/FLASH data from file (see Section 6.1) or enter data to Hex Editor.

- Select “Program Sequence” panel in **HC912-Programmer** software.
- Press “EEPROM/FLASH” button.
- Press “Start” button.
- LED D4 on **HC912-Programmer** became to green light. That means that +5 voltage regulator switched on and VCC applied to target MCU.
- When LED D1 on **HC912-Programmer** became permanent green light, programming completed.
- If some errors appeared during programming process refer to Section 7.



Note: When target MCU mounted on adaptor check contacts careful to avoid short circuit. Otherwise target MCU can be damaged!



Note: During removing the adaptor with target MCU **HC912-Programmer** must be powered off to avoid damaging of programmer and target MCU!



Note: Extra care must be taken when programming the EEPROM Shadow Word/Byte.



Note: Strongly recommended before EEPROM/FLASH programming, for a first time, read EEPROM/FLASH contents and save it to file.

5.2 MC 9S12 Devices

This section describes basic rules working with next devices:

- ✓ MC9S12 **D64** (2L86D)
- ✓ MC9S12 **DG128B** (0L85D)
- ✓ MC9S12 **DT128B** (0L85D)
- ✓ MC9S12 **A128B** (0L85D)
- ✓ MC9S12 **DB128B** (0L85D)
- ✓ MC9S12 **DT128B** (1L85D)
- ✓ MC9S12 **DG256C** (2K79X)
- ✓ MC9S12 **DT256C** (2K79X)
- ✓ MC9S12 **DP256C** (2K79X)
- ✓ MC9S12 **DP512** (1L00M)

FLASH/EEPROM memory for these devices can be read, program in both OBP and ICP interfaces.



Note: Use On-Board programming interface only if target MCU secured.

Extra care must be taken when working with FLASH Option/Security byte in ICP interface. If MCU not erased (all EEPROM and FLASH bytes not equal to \$FF) and FLASH Security byte not equal to XXXXXX10B, further access to MCU will be blocked after next reset. To grant access to MCU device must be mounted on corresponding adaptor and On-Board programming interface must be used.

In all other respects EEPROM/FLASH programming technique similar to MC68HC912 devices (See sections 5.1.1-5.1.4)

5.2.1 Features of MC9S12xx128 Devices with 0L85D/1L85D masksets

This section describes specific behavior of MC9S12xx128 (0L85D/1L85D) devices when In-Circuit programming interface is used. In according to Motorola errata relative to this devices there

is some problem exists with running of Sector Erase and Program Commands when MCU secured. Only Bulk Erase of FLASH and EEPROM command possible. In many cases this problem can be solved by disabling of security via Backdoor Access Key. [HC912-Programmer](#) automatically read this key and passes the security sequence. But if Backdoor Access Key disabled there is no possibilities to disable security. To solve this problem the second method of security disabling must be used. After erasing FLASH and EEPROM, target MCU can be accessed in Special Single Chip mode ([HC912-Programmer](#) In-Circuit interface). Follow instructions below:

- Read full EEPROM and FLASH contents in On-Board programming interface.
- First of all, when link with MCU completed it is necessarily to read all FLASH and EEPROM contents and save it to File. For more information see sections 5.1.3 On-Board EEPROM/FLASH reading example. It is obligatory to save EEPROM and FLASH contents to file (see Section 6.2).
- Erase EEPROM and FLASH.
- Press Erase FLASH and EEPROM buttons on Program Sequence panel. Also, press Verify EEPROM and FLASH buttons to perform erase check.
- Press Start button.
- When LED D1 on [HC912-Programmer](#) became permanent green light, erasing completed.
- Disconnect power from [HC912-Programmer](#).
- Remove target MCU from programmer and solder it back to board.
- Prepare target MCU for In-Circuit Programming.
- Connect required pins to board with target MCU (see Appendix Figure 2).
- Remove any adaptor from Motorola device socket (see Figure 1).
- Apply power to [HC912-Programmer](#). LED D1 became to green light (see Table 1).
- Select corresponding device in [HC912-Programmer](#) software (“Device” Button).
- Load EEPROM/FLASH data from previous saved file.
- Change value of FLASH Security byte, \$7BF0F Address to \$FE in Hex Editor. That mean that device will be unsecured after programming. Also, some changes in EEPROM and FLASH areas are possible.
- If further access to MCU is not required, FLASH Security byte not need to be changed.
- Press “EEPROM/FLASH” button on Program Sequence Panel.
- Press “Start” button.
- LED D4 on [HC912-Programmer](#) became to green light. That means that +5 voltage regulator switched on and VCC applied to target MCU.
- When LED D1 on [HC912-Programmer](#) became permanent green, programming completed. Target MCU powered off.
- If some errors appeared during programming process refer to Section 7.



Note: There is no way to restore EEPROM and FLASH contents after erasing. That is strongly recommended read EEPROM and FLASH contents and save it to file before erasing.



Note: When target MCU mounted on adaptor check contacts careful to avoid short circuit. Otherwise target MCU can be damaged!



Note: During removing the adaptor with target MCU [HC912-Programmer](#) must be powered off to avoid damaging of programmer and target MCU!

6. FILE OPERATIONS

This section describes basic rules working with files. Memory dump from Hex Editor (Buffer) can be load/save from/to hard disk. Also short descriptions such as project name, MCU type and memory cell assignment can be done for future fast remind. **HC912-Programmer** accepts tree types of file formats:

- ✓ BIN (Binary format)
- ✓ Motorola S-Record (4 byte address)
- ✓ Motorola S-Record (6 byte address)
- ✓ EEF (Extended ETL Format)

6.1 LOAD FILE INTO BUFFER

- Select “File>Open” menu item.
- Press “Browse” button.
- In File Open Dialog window, select file witch need to be open.
- Press “Open” button.
- Than opposite “Auto Format Detected:” text, select correct file format. Note that software try automatically detects file format, but unknown records in file will fail this detection. “Load Entire file” check box must be checked if automatically loading procedures required.
- Than press “OK” button.

Sometimes load data from file to specific buffer allocations required. For Example if required load buffer from \$0400 address from binary file beginning from \$0000 address follow next steps:

- Select “File>Open” menu item.
- Press “Browse” button.
- In File Open Dialog window, select file which need to be open.
- Press “Open” button.
- Than, opposite “Auto Format Detected:” select Binary format.
- Uncheck “Load Entire File” check box.
- In field “Offset Value to Place Data to Buffer:” enter 0x0400.
- Than press “OK” button.
- Now data placed to Hex Editor Buffer from the beginning of 0x0400 address.

If more complicated operations with files required, for example load Hex Editor Buffer from many files “Lowest Address From File To Load”, “Highest Address From File To Load” and “Clear Buffer Before Loading File” options are available.

6.2 SAVE FILE FROM BUFFER

- Select “File>Save” menu item.
- Press “Browse” button.
- Select directory in which file will be saved.
- Type file name, for example “test1”
- Press “Save” button.
- Than select format in which file will be saved*.
- Press “OK” button.



Note: Use EEF Format for future “Load File Into Buffer” automatically processing. Also, only in EEF Format Project Description, Device Name and Memory Cells attributes can be saved.



Note: Use Motorola S-Record (6 byte address) Format to save all FLASH memory contents.

7. ERRORS AND TROUBLESHOOTING

This section describes most recently encountered problems, errors and fixing solutions.

- **Problem:** LED D1 (see Figure 1) Dark.
- **Causes:** This problem can accrue when external power supply connected to **HC912-Programmer** is damaged or connected in wrong polarity.
- **Solutions:** Check voltage on **HC912-Programmer** power clamps. It must be 12 V +/- 1V.

- **Problem:** LED D1 (see Figure 1) Red.
- **Causes:** These problem can appear in two cases: if **HC912-Programmer** is in the service mode (JP1 removed) and if some error accrue while operation of **HC912-Programmer**.
- **Solutions:** Insert jumper JP1 (see Figure 1), remove adaptors from Motorola device socket and cable from ICP connector, turn-off and than turn-on power supply connected to **HC912-Programmer**. If LED D1 still red contact ETL technical support.

- **Error Message:** MC68HC912 Programmer not found
- **Causes:** This message can appear when **HC912-Programmer** software couldn't establish connection with **HC912-Programmer** board.
- **Solutions:** Check connection of COM port cable from PC to **HC912-Programmer** board. Apply power from external power supply to **HC912-Programmer**. In menu item "Tools>Comm Port Options" select correct COM port number.

- **Error Message:** Communication Error
- **Causes:** This message can appear when Communication between **HC912-Programmer** and PC is broken.
- **Solutions:** Try to decrease communication baud rate in "Tools>Comm Port Options" dialog window. Also this message can appear when COM port cable has poor contact with DB-9 connectors.

- **Error Message:** RESET Line must be in 'high' state
- **Causes:** This message can appear when MCU Reset pin connected to ground through external components or no oscillation on EXTAL, XTAL pins.
- **Solutions:** Check MCU reset pin resistance relatively to VSS pin, and if it too low check passive and active components connected. Or check integrity of ceramic resonator and passive components connected to EXTAL, XTAL pins.

- **Error Message:** BKGD Line must be in 'high' state
- **Causes:** This message can appear when MCU BKGD pin connected to ground through external components.
- **Solutions:** Cut any component from pin.

- **Error Message:** Pin Tester Errors:
- **Causes:** This message can appear in On-Board programming interface when some pins has poor contact with QFP adaptor, or pins are damaged.
- **Solutions:** Clean pins from colophony. Check contacts between adaptor and target MCU pins. Check short circuits between pins to pins, pins to GND and pins to VCC.

- **Error Message:** BDM Speed Auto Detection failed

■ **Causes:** This message can appear in In-Circuit programming interface if target MCU not connected correctly to **HC912-Programmer** or MCU secured.

■ **Solutions:** Check contacts between **HC912-Programmer** and MCU pins. If this error not disappeared, place MCU on corresponding QFP adaptor and use On-Board programming interface.

- **Error Message:** BDM Activation Error

■ **Causes:** This message can appear when wrong Device selected.

■ **Solutions:** Select correct device type; take attention on MCU maskset.

- **Error Message:** No clock on ECLK pin

■ **Causes:** This message can appear in On-Board programming interface when wrong Device selected.

■ **Solutions:** Select correct device type; take attention on MCU maskset.

- **Error Message:** Parallel Boot Failed

■ **Causes:** This message can appear in On-Board programming interface when wrong Device selected.

■ **Solutions:** Select correct device type; take attention on MCU maskset.

- **Error Message:** Monitor Loader time-out Error

■ **Causes:** This message can appear when wrong Device selected.

■ **Solutions:** Select correct device type; take attention on MCU maskset.

- **Error Message:** Monitor Check-sum Error

■ **Causes:** This message can appear after monitor firmware loaded into target MCU with errors.

■ **Solutions:** Try to reconnect to target MCU. If this error not disappeared, possible target MCU has damaged RAM.

- **Error Message:** Monitor Heap Check-sum Error

■ **Causes:** This message can appear after monitor firmware transferred data into target MCU RAM with errors.

■ **Solutions:** Try to reconnect to target MCU. If this error not disappeared, possible target MCU has damaged RAM.

- **Error Message:** BDM active status or monitor firmware time-out Error

■ **Causes:** This message can appear if target MCU “hangs up”.

■ **Solutions:** Try to reconnect to target MCU. If this error not disappeared, check contacts between **HC912-Programmer** and MCU pins, power supply voltage on VCC pins.

- **Error Message:** Parallel Bus active status or monitor firmware time-out Error

■ **Causes:** This message can appear if target MCU “hangs up”.

■ **Solutions:** Try to reconnect to target MCU. If this error not disappeared, check contacts between **HC912-Programmer** and MCU pins, power supply voltage on VCC pins.

- **Error Message:** Device still locked
- **Causes:** This message can appear in On-Board programming interface when target MCU secured.
- **Solutions:** Check accuracy of device type selected; take attention on MCU maskset. Try to reconnect to target MCU.

- **Error Message:** Device can't be unsecured
- **Causes:** This message can appear in In-Circuit programming interface when MCU EEPROM/FLASH erased (all data equal to \$FF) and accordingly device still secured.
- **Solutions:** Check voltage on VCC pins. Try to reconnect to target MCU.

- **Error Message:** Voltage on VFP pin is below normal programming voltage level
- **Causes:** This message can appear when no or low voltage applied to VFP pin.
- **Solutions:** Apply specified by Motorola voltage to VFP pin.

- **Error Message:** On-Board Programming Interface not supported for the Device
- **Causes:** This message can appear when wrong Device selected.
- **Solutions:** Select correct device type; take attention on MCU maskset.

- **Error Message:** External power supply must be disconnected from BDM ICP connector
- **Causes:** This message can appear in On-Board programming interface when external voltage applied to BDM ICP connector (see Figure 1).
- **Solutions:** Disconnect cable from BDM ICP connector.

- **Error Message:** Vreg Overcurrent Protection
- **Causes:** This message can appear when current consumption from built-in power supply grater than 500 mA.
- **Solutions:** Check target MCU power pins on short circuit.

- **Error Message:** Vreg Output Voltage out of range
CYPRESS IIC BUS Busy
CYPRESS IIC Protocol Failed
CYPRESS Frequency out of range
Wrong Subroutine call
- **Causes:** These messages can appear when fatal problems with **HC912-Programmer** hardware accrued.
- **Solutions:** Contact ETL technical support.



Note: **HC912-Programmer** has Log Window which can be stored to file. To perform this operation right clicks on Log Window. Than click on “Save to hc912prog.log file“ menu item. Now this file can be found in the same directory as **HC912-Programmer** software. Log File can be send by E-mail to techsupport@etlweb.net for non described problem solution.

8. WARRANTY STATEMENT

ETL warrants that Product delivered shall conform to applicable. Report any defects for a 45 days period, from the applicable data on invoice.

9. APPENDIX

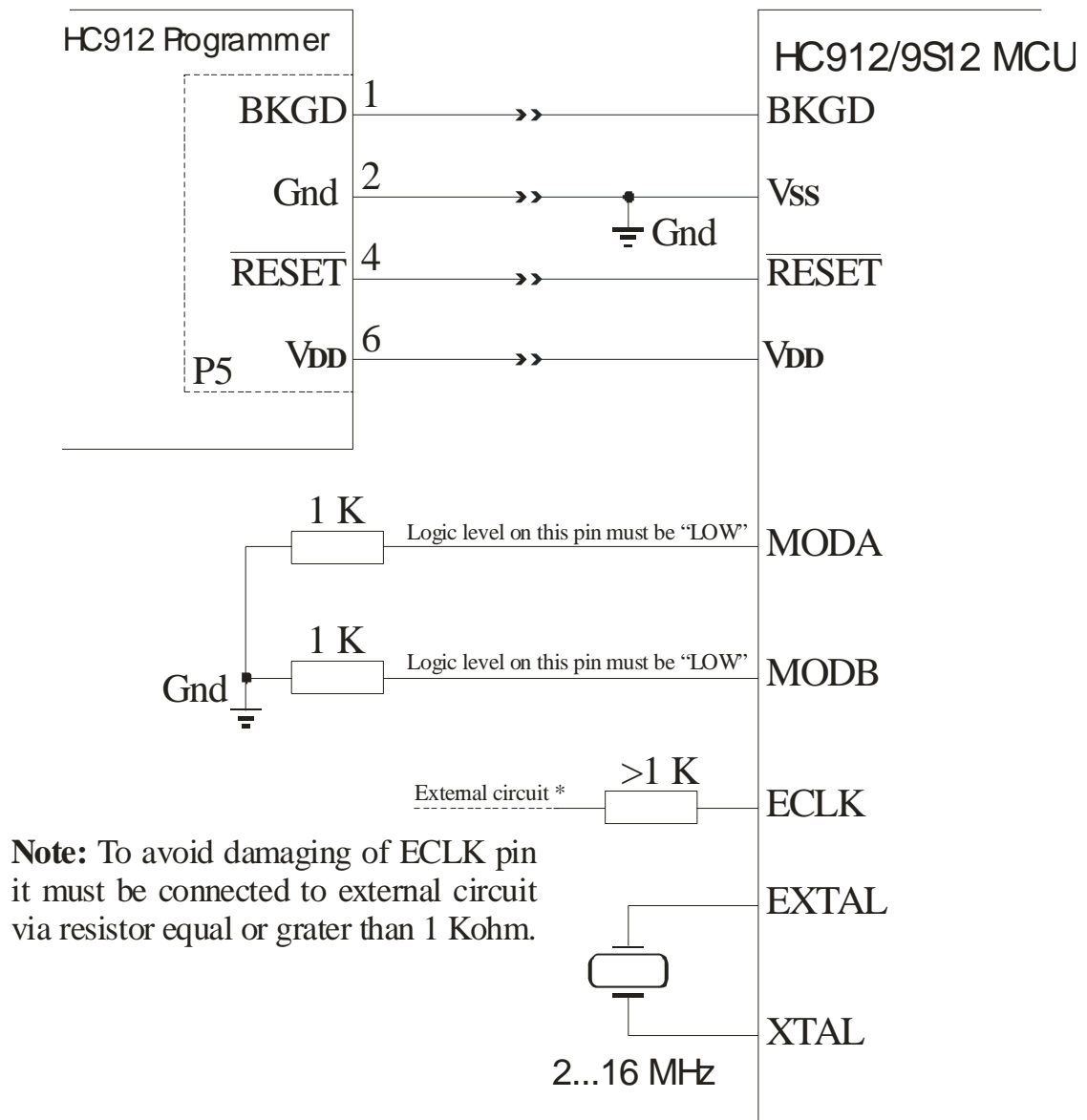


Figure 2. In-Circuit programming schematic diagram for HC912/9S12 MCU

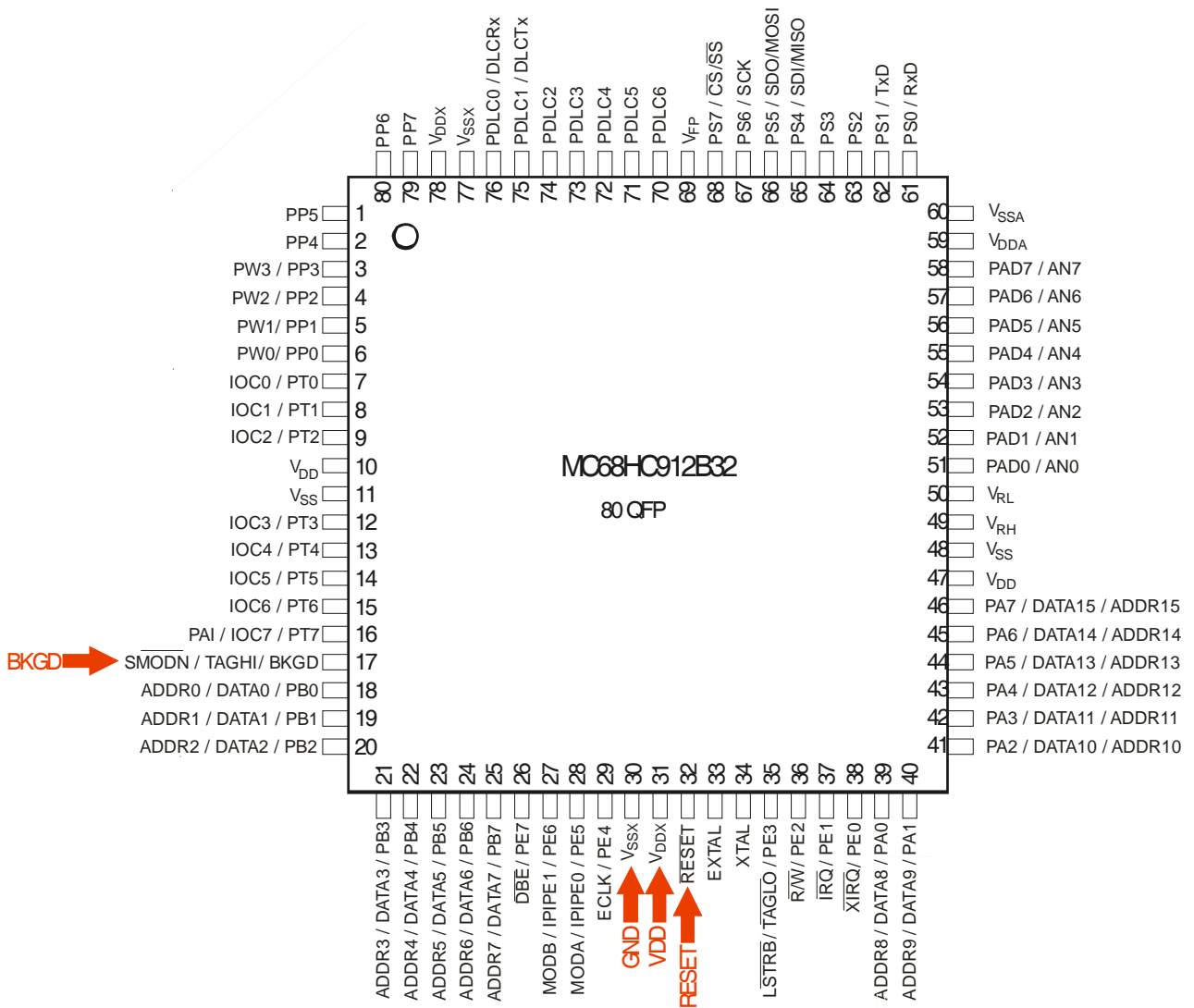


Figure 3. In-Circuit wiring diagram for MC68HC912B32, 80-PIN QFP Package

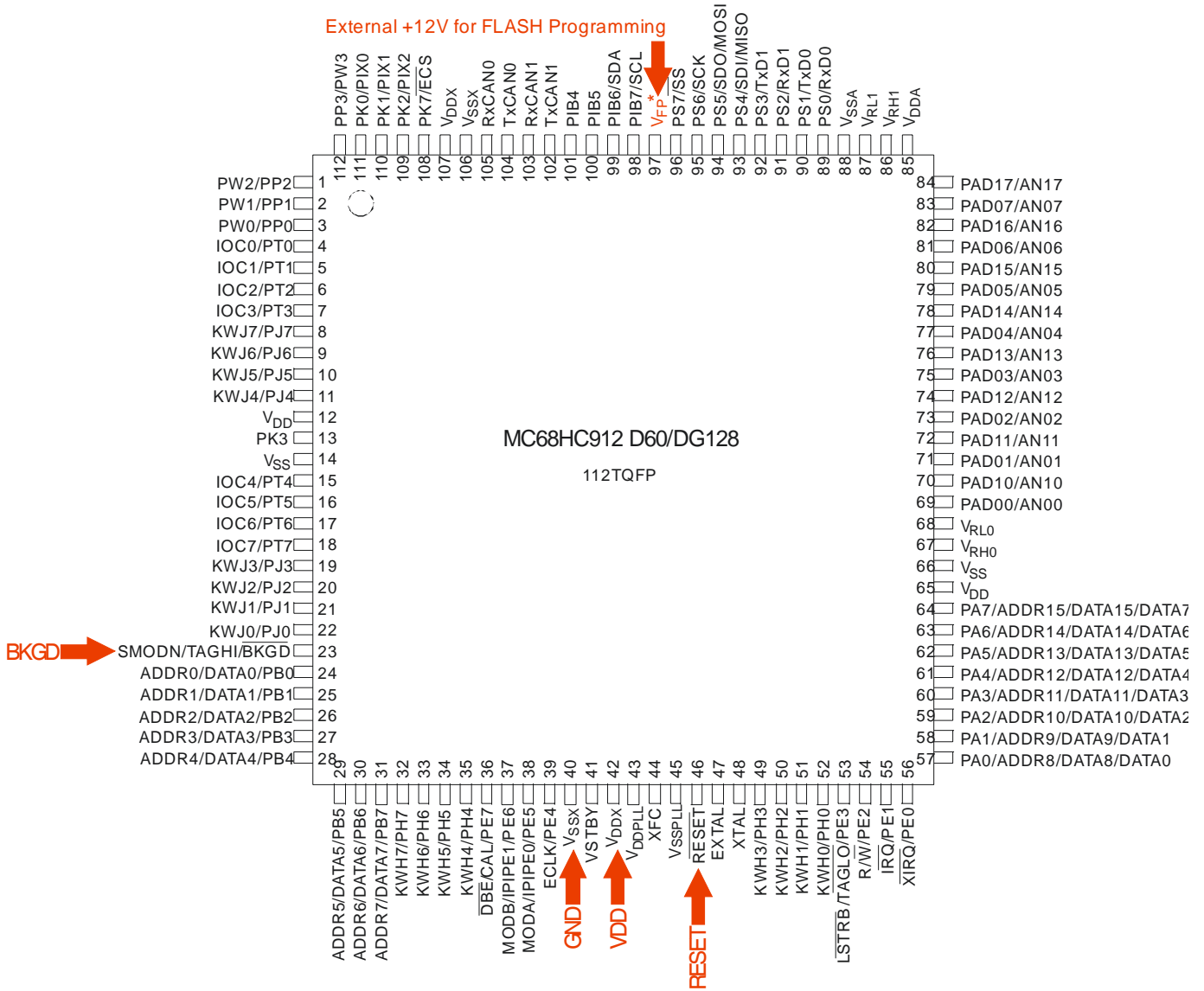


Figure 4. In-Circuit wiring diagram for MC68HC912D60/DG128, 112-PIN TQFP Package

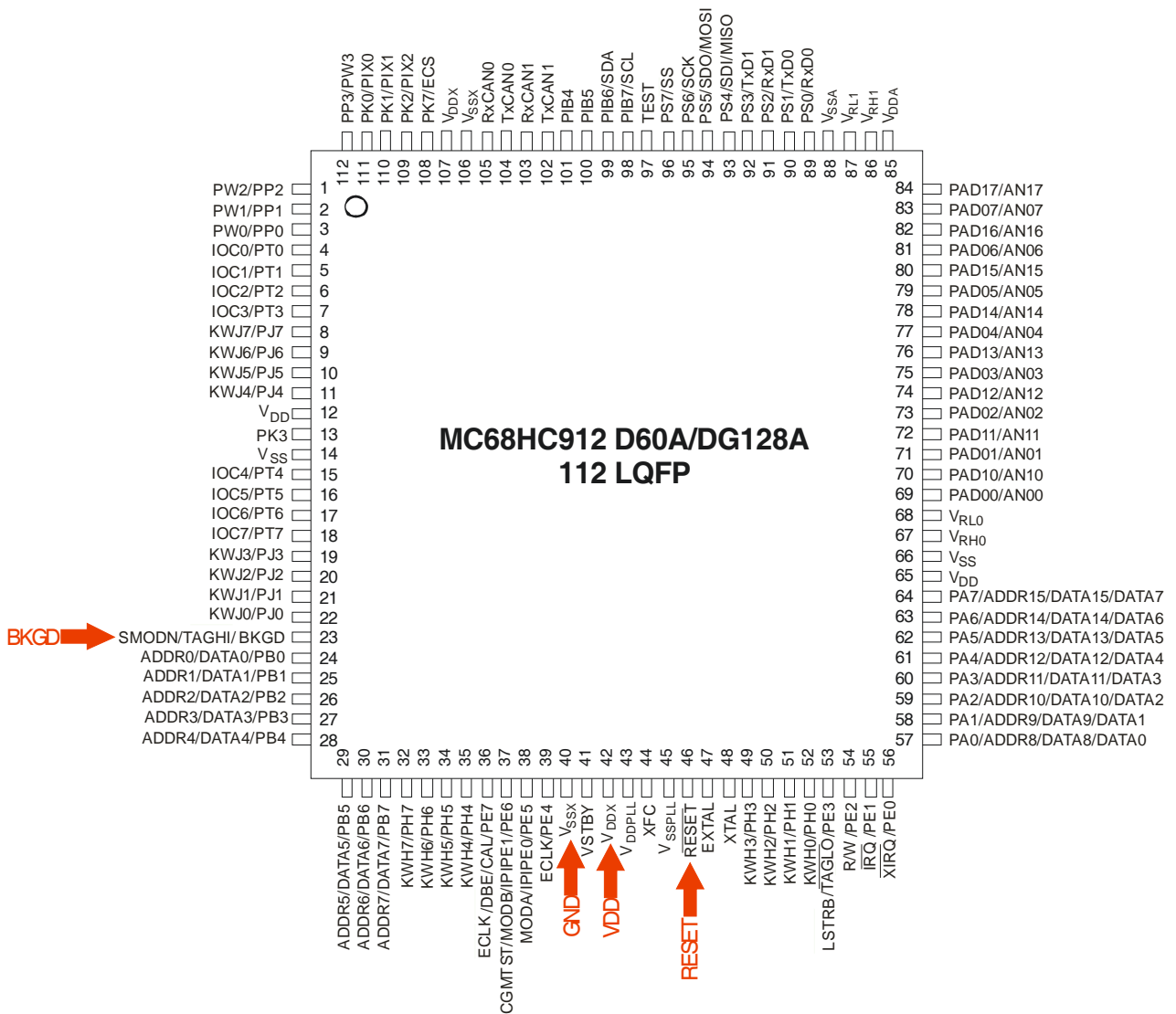


Figure 5. In-Circuit wiring diagram for MC68HC912D60A/DG128A, 112-PIN LQFP Package

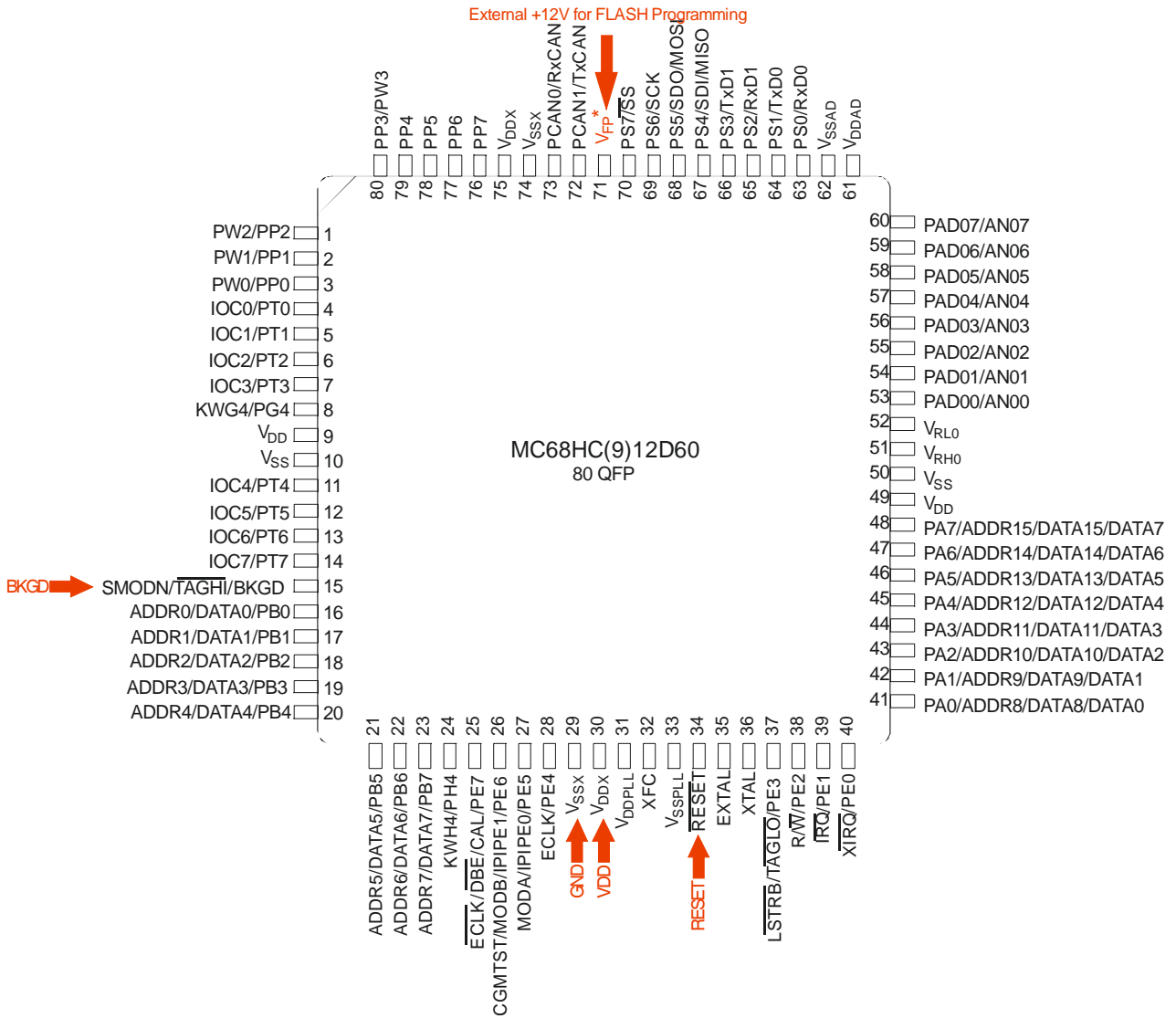


Figure 6. In-Circuit wiring diagram for MC68HC(9)12D60, 80-PIN QFP Package

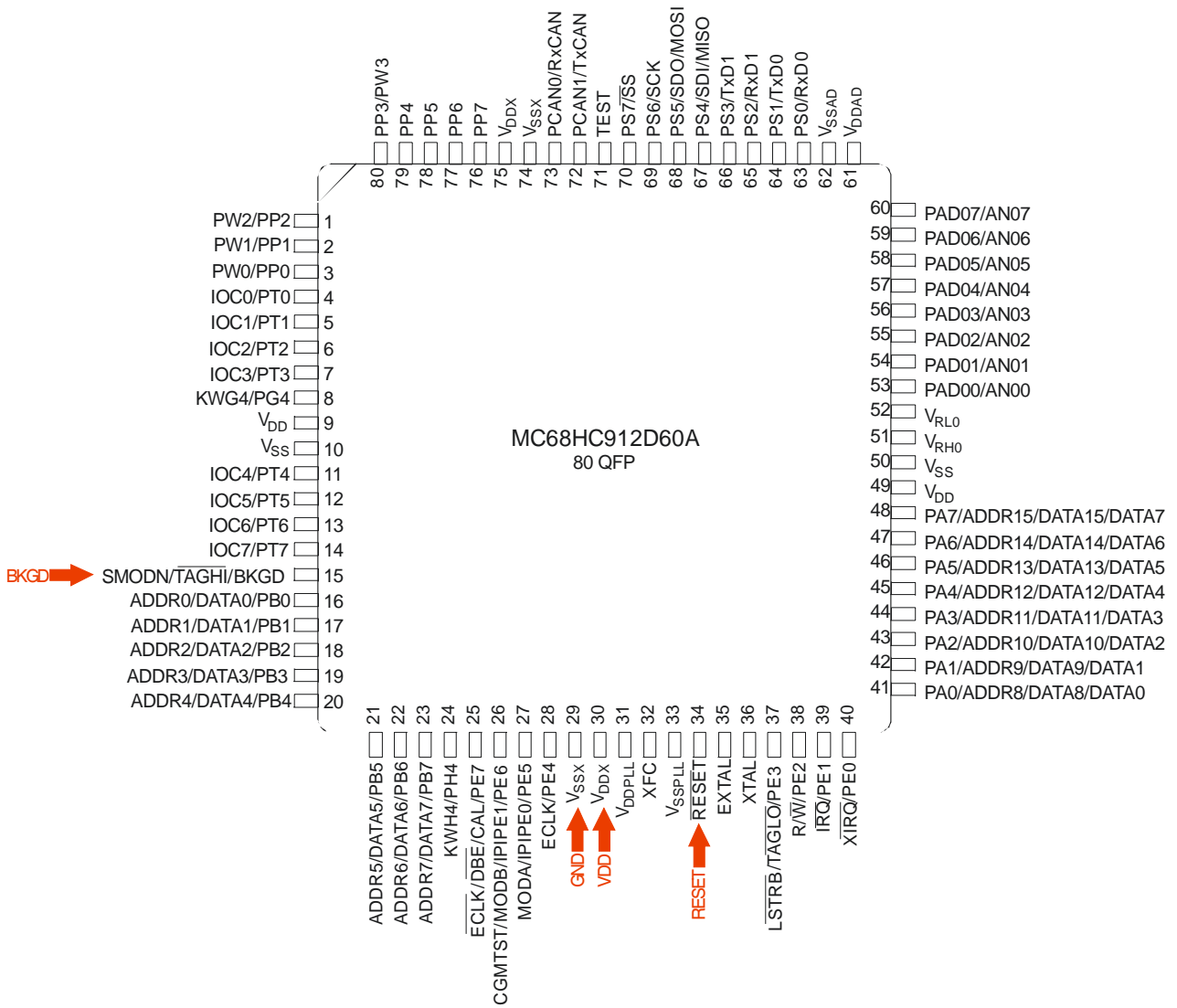


Figure 7. In-Circuit wiring diagram for MC68HC912D60A, 80-PIN QFP Package

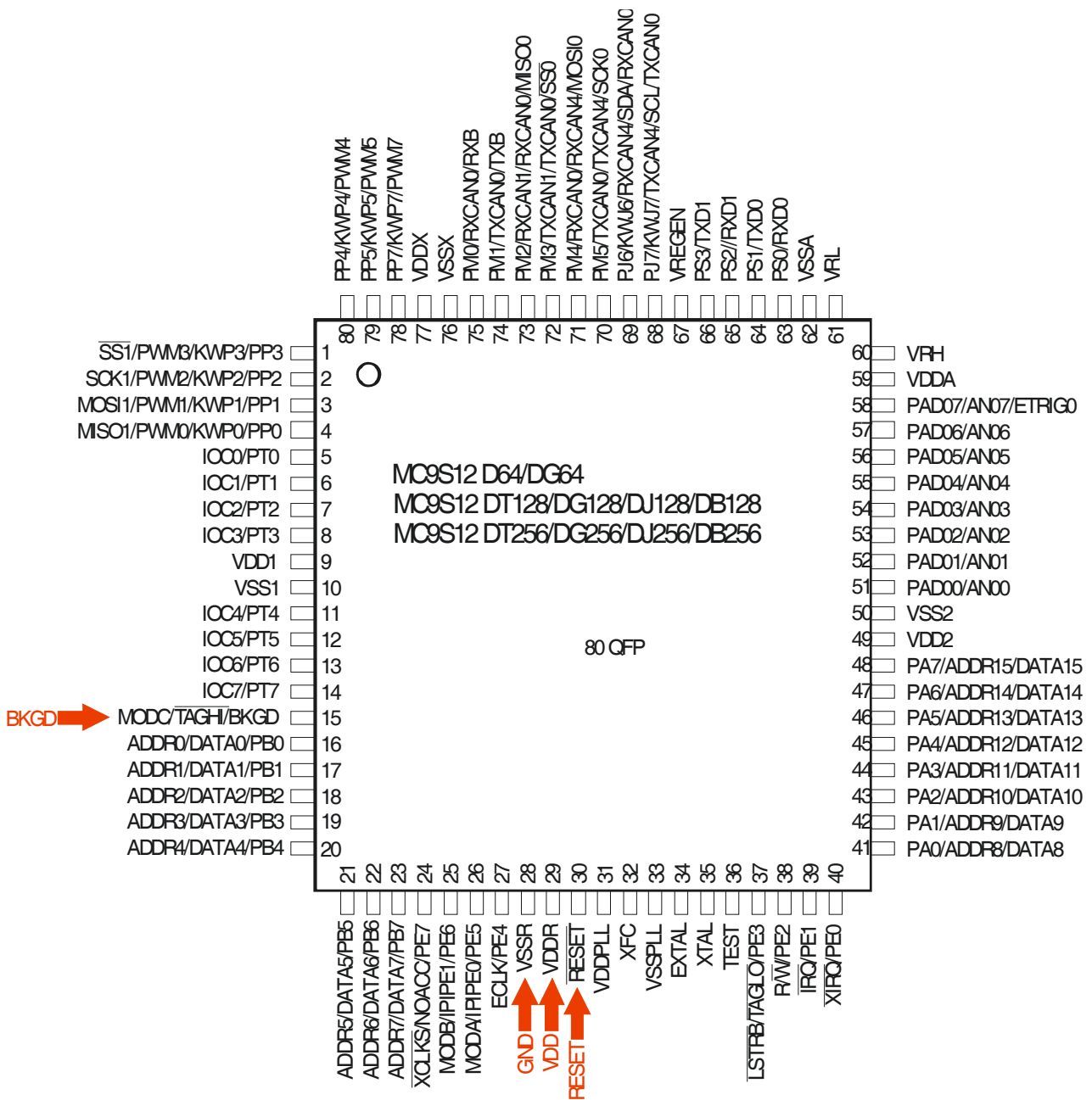


Figure 8. In-Circuit wiring diagram for MC9S12Dx64/Dx128/Dx256, 80-PIN QFP Package

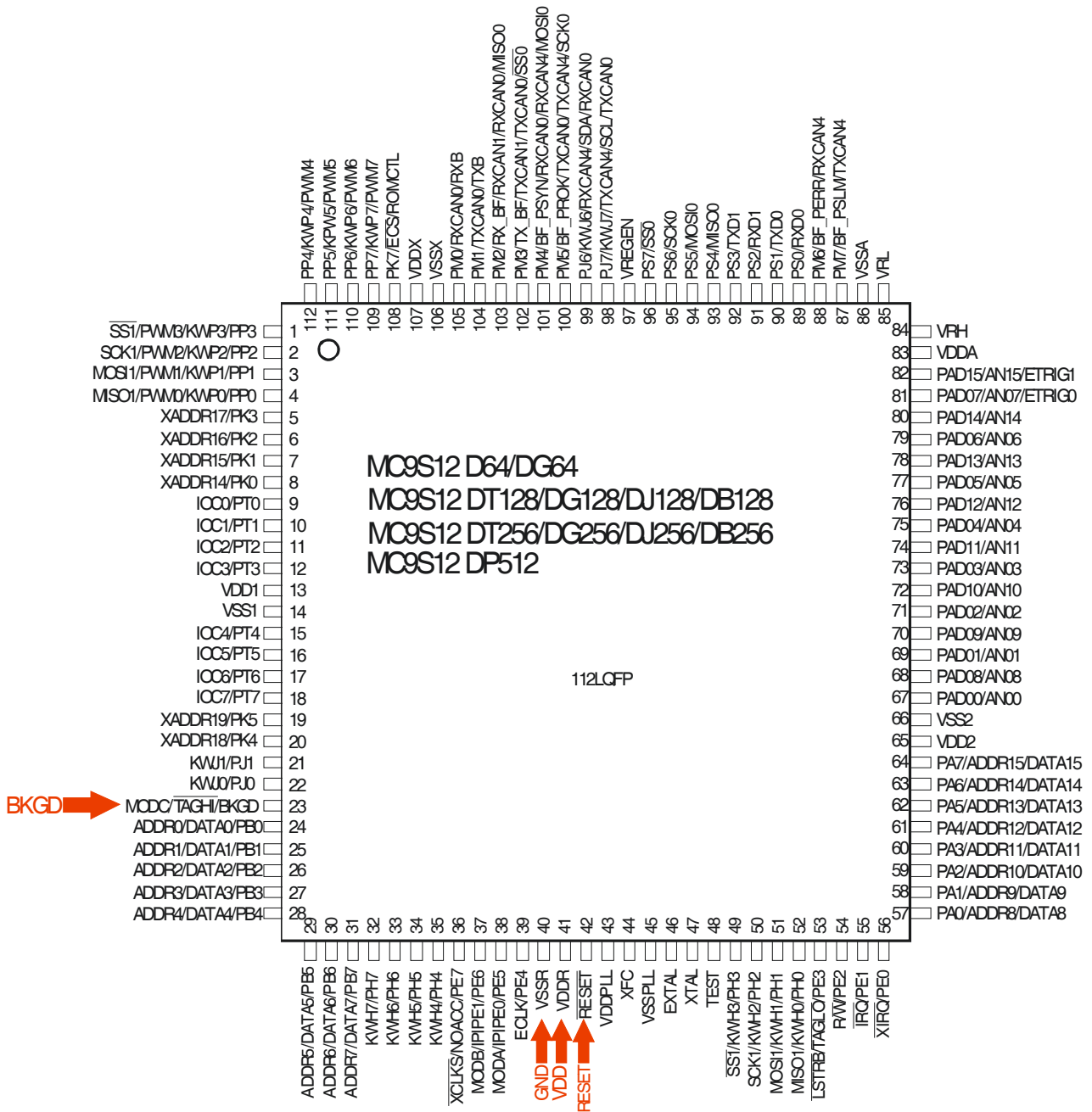


Figure 9. In-Circuit wiring diagram for MC9S12Dx64/Dx128/Dx256, 112-PIN LQFP Package

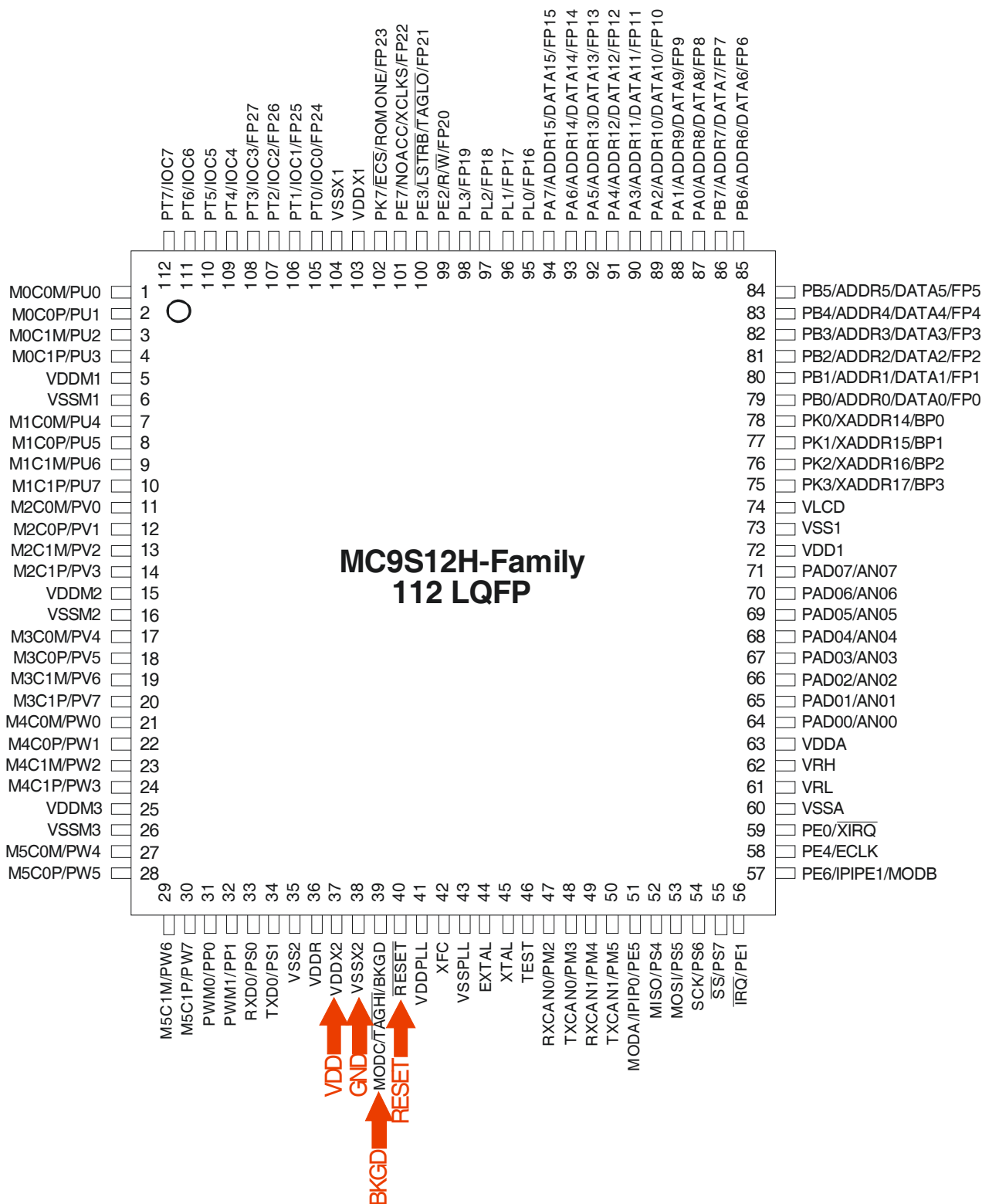


Figure 10. In-Circuit wiring diagram for MC9S12H-Family 112-PIN LQFP Package

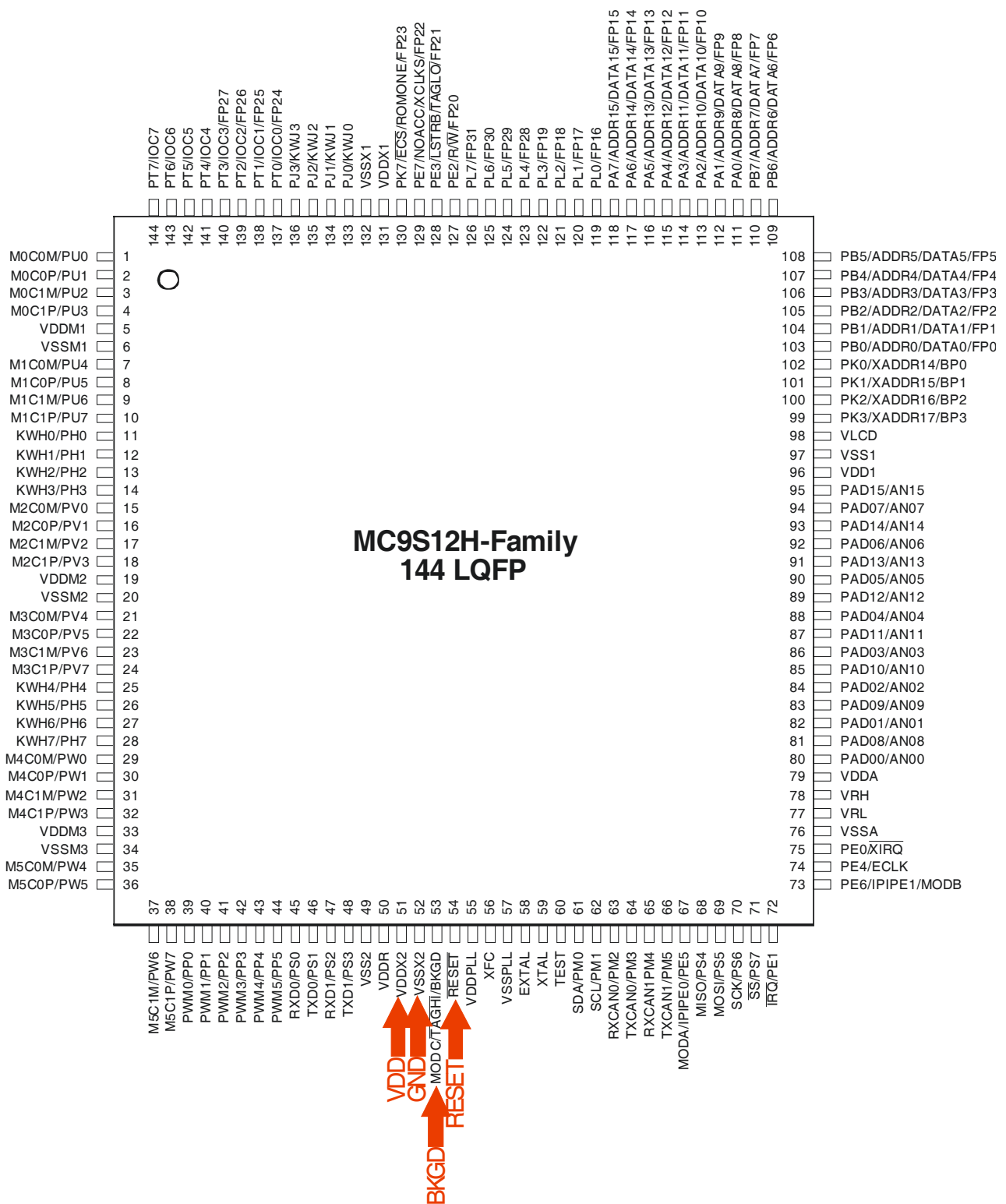


Figure 11. In-Circuit wiring diagram for MC9S12H-Family 144-PIN LQFP Package