# USING THE ACOM2 ADAPTER WITH MOTOROLA MICROCONTROLLERS IN AUTOMOTIVE APPLICATIONS
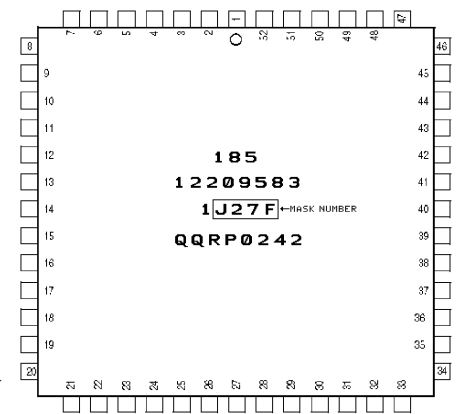
All automotive modules are based on some type of microprocessor or microcontroller. In many cases module specific data is contained in an 8 pin serial eeprom which is external to the microcontroller package. In cases where there is no external eeprom part, the module data will be stored in eeprom memory located internal to the microcontroller itself (not in an external 8 pin part). Motorola (now Freescale) created three families of micro-controllers, many with internal eeprom memory, which are commonly found in different body controller and vehicle modules. These include airbags, immobilizers, ECUs, digital clusters plus more. Unlike an external 8 pin eeprom, a microcontroller is a tiny computer on a chip. Accessing the eeprom memory in these parts re-quires establishing a communication link with the part itself. This is much like a modem link between two computers. The ACOM2 ayschronous communication adapter allows the Andromeda Research system to access and communicate with Motorola microcontrollers thereby allowing you to access the internal eeprom for reading and programming. This document describes the three common microcontroller families, how to identify the part with which you are working plus how to connect and establish communication with the part itself. This docu-ment also explains the technical specifics which you must understand in order to successfully establish communication with the part. This includes logic levels, voltages, and if necessary, how to identify a package pin which must be isolated (lifted). Also covered is the basic operation of the microcontroller itself and what is required for successful communication.

(**NOTE:** This document applies to both generations of the ACOM2 adapter. (REV B-Top Illustration | REV C/D-Bottom Illustration) If your ACOM2 is a REV B it must be upgraded to allow support of the 68HC08AZ family of parts. Contact Andromeda Research for upgrade information.)
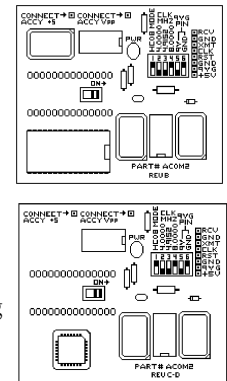
HOW TO IDENTIFY YOUR PART - The three Motorola microcontroller families

There are three families of Motorola microcontrollers which include an asychronous communication interface compatible with the ACOM2. These technology families are the 68HC11, 68HC08/908 and 68HC05. Note that these are families and that there are many parts in each family group. An actual part number is composed of the primary family followed by a suffix. For example, a common module part is the 68HC11E9. The E9 suffix defines a specific internal architecture (amount of ROM memory, I/O options, etc.) for this part. There are other members of this family including the 68HC11EA9, 68HC11E20, 68HC11K4 plus more. Suffixes also apply to the HC05 and HC08/908 families. Motorola microcontrollers are usually fabricated in two package types with four pin arrangements. The first package type is the PLCC (Plastic Leaded Chip Carrier). In automotive applica-tions there are three sizes of the PLCC package; 52 pin (13 pins on a side), 68 pin (17 pins on a side) and 84 pin (21 pins on a side). PLCC package pins are formed in a "J" shape where the lead is bent down the side and wraps under the package forming the "J" (J-lead). The second package type is the QFP (Quad Flat Pack). QFP parts are smaller than a PLCC and are commonly fabricated in a 64 pin package (16 pins on a side). Since the package is smaller the leads are also smaller. The leads exit the package body and form an "L" shape where each pin is bent down the package side and then bends out (away from the package body) forming the "L". An industry standard part number, such as 68HC11E9, is not always printed on the part package. If the part does not have an industry standard part number, in almost all cases it will have what is called a "mask number". A mask number references the photo mask used to fabricate the silicon chip inside the package. The mask number will cross reference to an industry standard part. The mask number is always four characters consisting of an alpha (letter), two numeric (numbers) and ending with an alpha. The upper illustra-tion on the right shows the mask number J27F. Use this number with the cross reference sheet (motomask.pdf) to determine the industry standard part 68HC08AS32. This is the part number you will choose when using the AR-32A software. Note that there is also a cross reference list which may be accessed using the system librarian (com-mand L). (Lower right illustration) To access this list use the PATH command to navigate to the EPROM directory then press TAB and hilite INFO_DOC. Press ALT-L to enter the librarian and hilite MOTOMASK. Press TAB then use the arrow keys to locate your mask number and standard part cross referemce. Use the part number at the ENTER DEVICE TYPE -> prompt or to capture (hilite and press Ins) the number from the MOTOROLA 68HC05/08/11 SERIES MICROCONTROLLER catagory screen.

HOW A MICROCONTROLLER WORKS

A microcontroller is a tiny computer on a chip. As such it includes all of the necessary components which are required to compose a computer; CPU, MEMORY and I/O (Input/Output). The microcontroller is sometimes referred to as the BRAIN of the module. In some respects this is true however the microcontroller is just a computer running a program. The program the computer (microcontroller) runs was written to cause the module to perform its specific function (airbag module controller, vehicle immobilizer, digital cluster, ECU, etc.). The program that the module runs, regardless of the module type, consists of a series of computer instructions which are stored in the memory of the microcontroller itself. This memory is called ROM (Read Only Memory). The part of the microcontroller which executes the instructions stored in ROM is the CPU (Central Processing Unit). The CPU executes the instructions in a specific order to perform its function. As the program runs it will send data out to the pins on the microcontroller package or read data in from pins. An example of an output would be a relay which disables the starter circuit for the engine. An example of an input would be a crash sensor or brake pedal switch. All module inputs and outputs are controlled (processed) by the program running on the microcontroller. If the microcontroller is not running its program, the module does not work.

DIFFERENT TYPES OF MEMORY

A microcontroller will always have two or three types of memory; ROM, RAM and EEPROM. All memory types retain information. It's how the information is retained which makes the memory types different. **ROM** stands for **R**ead **O**nly **M**emory. ROM is where the program instructions are stored. ROM (in most cases cannot be changed). It is call ROM because the CPU cannot write information into the memory. It can only read the instructions which compose its program. When the microcontroller chip is fabricated the preprogammed ROM is created at the same time. **RAM** stands for **R**andom **A**ccess **M**emory. RAM allows information to be both written and read. You can think of RAM as a fast scratch pad memory area which the program will use if it has to temporarily store information such as calculation results. Unlike ROM, RAM is not permanent. It will retain its contents until module power is removed or the program overwrites what is stored. Once module power is gone, RAM memory cells lose their data. **EEPROM** stands for **E**lectrically **E**rasable **P**rogrammable **R**ead **O**nly **M**emory. EEPROM are special memory cells which are fabricated using a technology whereby each bit of information can be permanently stored or changed. This is done using what is called an "insulated or floating gate". The floating gate is fabricated above the transistor memory cell. If a charge is placed on the floating gate (programmed) the charge will remain trapped even if power is removed from the module. This is because the gate floats and does not discharge. Placing the charge on the floating gate takes time and requires a high voltage which is why programming EEPROM appears to consume more time. However, once programmed, reading EEPROM is just as fast as ROM or RAM. Due to its unique ability to retain data after power is removed makes EEPROM a perfect solution for automotive applications. EEPROM is used anywhere vehicle specific information such as the VIN number, mileage, crash data or key sequences must be stored or changed.

WHAT DOES A MICROCONTROLLER NEED TO WORK - THE TECHNICAL DETAILS

Since a microcontroller is a computer, there are certain external connections which must be supplied for the microcontroller to function. The first and most important is the power source. Microcontrollers operate from +5 volts DC. The power pins on the microcontroller package are Vdd and Vss. Vdd is the positive power supply (+5V). Vss is negative, 0 volts or ground (GND).

TECH-INFO: The Vdd and Vss designators originate from the transistor technology used to fabricate the microcontroller. The technology is called "High speed CMOS" which is from where the HC in the 68HC part number comes. CMOS (Complimentrary Metal Oxide Semiconductor) transistors have three connections. The "Source", the "Gate" and the "Drain". In order for a CMOS transistor circuit to function the Source connection must be at 0 volts while the Drain connection must be at 5 volts. In electronics terminology Vdd represents the voltage supply connection to the CMOS transistors Drain where Vss is the connection to the Source. The Gate element switchs the transistor on or off.

Another requirement for microcontroller operation is called the clock. The clock is the timing signal which causes the internal computer circuitry to sequence. The internal clock frequency of the microcontroller is established by an external component called a crystal. The crystal, which is an extremely small slice of quartz in a metal case, attaches to two external pins on the package called XTAL and EXTAL. The crystal, in conjunction with an internal oscillator on the microcontroller, will resonate at a specific and very accurate frequency. This frequency establishes the operating speed or clock rate of the microcontroller. You can see the operating frequency in Megahertz (MHZ) of a crystal by looking at the number stamped or printed on the case. It will be something such as 4.000, 8.000, etc. If you wish to prevent the microcontroller from running you can short or place a jumper across the crystal itself or between the two pins (XTAL, EXTAL) on the package. Shorting the crystal does not damage the crystal or microcontroller. **NOTE:** The crystal is always physically close to the microcontroller package.

Probe connections for 68HC11E9 and 68HC11E20

The illustration on the right shows the pin connections for a 68HC11E9 and 68HC11E20 microcontroller. Note the Vdd and Vss connections. Also note the XTAL and EXTAL connections for the crystal. The ACOM2 adapter provides its own clock generator which attaches to the EXTAL pin on the part. This will override the crystal and forces the microcontroller to operate at the specific frequency required for communication. Also note the RESET pin which resets or restarts the internal ROM program to its beginning. (The bar over the signal name indicates that a logic 0 or connection to ground will cause the reset to occur.)
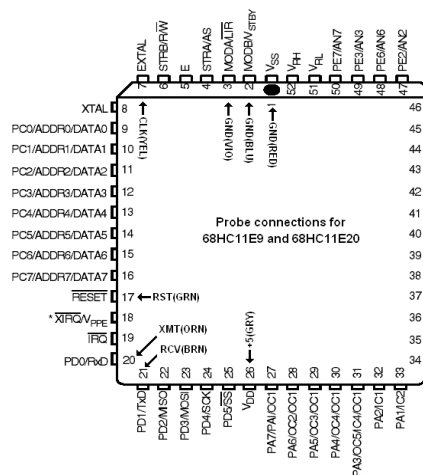
WHAT HAPPENS IN THE MODULE WHEN THE VEHICLE IGNITION IS SWITCH ON

With the iginition off, most modules and vehicle circuitry are unpowered as any excessive current draw will drain the battery. One exception is the security system. When the vehicle ignition is switched to ON the various modules receive power and come to life. It is extremely important that each module perform an orderly power up procedure. This procedure is called Power On Reset or POR. The POR circuit insures that the microcontroller begins properly executing its program. This is much like pushing the RESET button on a personal computer. Everything restarts and the microcontroller starts program execution at the first instruction. If all vehicle modules are working the engine will start and the dash lights go out (airbag, ECU, seatbelt, etc.) indicating that all electronic system are functioning. This happens because all of the modules communicate with each other over the cars network. Each module must confirm correct operation or a warning is indicated to the driver in the form of a dash light or display.

COMMUNICATING WITH A MODULE WHEN IT IS REMOVED FROM THE VEHICLE

When you remove a module from the vehicle it obviously cannot operate in its normal vehicle mode however you are interested in reading, copying, editing and programming the internal eeprom of the microcontroller. For this to be possible the module microcontroller must allow stand-alone communication after reset. It works like this; when the module microcontroller exits the reset state (after POR) even before it begins executing its module program, it will test (what it tests will be covered later) to confirm that it will enter what is called "single chip mode". Single chip mode is the standard mode for normal vehicle operation. If following the test after reset, the microcontroller senses that it is not to enter "single chip mode" it will enter an alternate mode, which, if possible, will allow external communication with the microcontroller. The alternate modes used by the ACOM2 for communication are called "BOOTSTRAP" and "MONITOR" depending on the microcontroller family.

LOGIC LEVELS AND PIN VOLTAGES - HOW THE MICROCONTROLLER PERFORMS THE MODE TEST

A logic level is simply a voltage which represents a logic 1 or a logic 0. These voltages are the same as the power supply of the microcontroller where a logic 0 is 0 volts or ground (GND) and a logic 1 is between 3 and 5 volts. The microcontroller is programmed, when it exits reset, to test the logic levels of specific pins on the part package. For the 68HC11E9 part (illustration) the pins are called MODA and MODB (MODE A and MODE B). These are pins 2 and 3 on the 52 pin PLCC package. Both of these pins must be at logic 0 (GND) for the part to enter bootstrap mode after reset. Notice that both pins are marked GND with the corresponding probe colors as (VIOLET and BLUE). When the probes are attached the MODA and MODB pins are forced to GND or logic 0.

BOOTSTRAP MODE - The term "bootstrap" originates from the early days of computers where a small program was loaded into a computers memory (usually through front panel switches) and then run. The small program would then cause a larger program to load (bootstrap), either from a hard drive or paper tape. The larger program would be the operating system or application. Bootstrap mode on a microcontroller works in a similar manner. A very small "bootstrap" program already exists in the ROM of the microcontroller. If after reset the microcontroller senses that it is to enter "bootstrap" mode, it begins executing the "bootstrap" program. The "bootstrap" program waits for a small program to be uploaded into the microcontrollers RAM. Once the program is uploaded the microcontroller exits bootstrap mode and begins executing the uploaded program. The uploaded program for the AR-32A system is a special communication program. This program allows the microcontroller to perform reading and programming operations based on commands sent and received by the system software. Bootstrap mode is used for the 68HC11 and 68HC05 families of microcontrollers. The illustration on the right shows the ACOM2 probe connections which are required to force a 68HC11E9 or 68HC11E20 into "bootstrap" mode.

Probe connections for 68HC11E9 and 68HC11E20

The required connections are:

POWER (PIN#1[Vss], PIN#26[Vdd]) | PURPOSE: Provides power (+5 and GND) to the part.

CLK (PIN#7[EXTAL]) | PURPOSE: Provides the clock signal to the part which establishes the operating frequency.

RST (PIN#17 [RESET]) | PURPOSE: Resets the microcontroller to its initial condition to insure proper startup.

XMT (PIN#20 [PD0/RxD] | PURPOSE: This pin receives (RxD) the transmitted data from the ACOM2.

RCV (PIN#21 [PD1/TxD] | PURPOSE: This pin transmits (TxD) data from the microcontroller to the ACOM2.

MODE PINS (PIN#2 [MODB/Vstby], PIN#3[MODA/LIR]) | PURPOSE: The mode pins determine the operating mode of the microcontroller following reset. For the 68HC11E9 to enter "bootstrap" mode both mode pins must be at 0V or GND (logic 0).

MONITOR MODE - A "MONITOR" is a small program which is used to exercise and test various parts or areas of a computer. The monitor is designed to accept simple commands from a computer terminal or other communication device. Functions include reading and writing the computers memory plus other rudamentary functions such as uploading and running a test program. The 68HC08/908 family of microcontrollers include a monitor programmed into the ROM of the part. When the microcontroller exits reset it will test to determine if it should enter MONITOR mode. Once MONITOR mode has been entered, communication with the internal memory of the microcontroller is possible using the ACOM2.

To enter MONITOR mode the HC08/908 microcontrollers require logic levels on specific pins but also require a higher voltage (9V) on one specific pin. This is the IRQ pin. The IRQ (Interrupt Request) pin must be forced to a voltage level above the standard operating voltage (5V) of the microcontroller. The illustration shows the 68HC08AS32. Note there are nine connections identified on the package:

**POWER (PIN#6[Vss], PIN#25[Vdd]) | PURPOSE:** Provides power (+5 and GND) to the part.

CLK (PIN#3[OSC1]) | PURPOSE: Provides the clock signal to the part which establishes the operating frequency.

RST (PIN#10 [RESET]) | PURPOSE: Resets the microcontroller to its initial condition to insure proper startup.

RCV (PIN#27 [PTA0]) | PURPOSE: The microcontroller uses this pin to both send (transmit) data to and receive data from the ACOM2.

MONITOR MODE ENTRY PINS (PIN#9 [IRQ-9V]), (PIN#5[PTC3-GND]), (PIN#7 [PTC1-GND]), (PIN#4 [PTC0-5V]) | PURPOSE: The monitor entry mode pins determine the operating mode of the microcontroller following reset. Each of the four pins (#9, #5, #7, #4) must be at the proper voltage or logic level for the microcontroller to enter monitor mode.
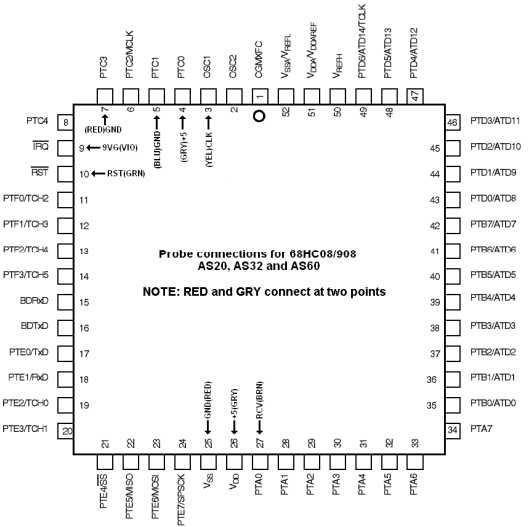
INSTALLING THE ACOM2 INTO THE AR-32A PROGRAMMING UNIT

Before proceeding with adapter installation apply power to the AR-32A programming unit and start the control program. From the ENTER DEVICE TYPE prompt enter the number of the part with which you are working and press ENTER. Before installing the ACOM2 adapter, be sure that the programming unit DIP SWITCH is set correctly (3 and 5 ON, all others OFF) as the switch cannot be accessed after the adapter is installed. The ACOM2 adapter has a 28 pin base and is therefore installed in the AR-32A

```
┌─────────────────────────────┬─────────────────────────────────┐
│     COMMAND/DATA ENTRY       │        ACTIVITY/STATUS           │
├─────────────────────────────┼─────────────────────────────────┤
│   PROGRAMMING UNIT SWITCH    │  REQUIRES ASYNC COMMUNICATION ADAPTER │
│     ├── DIP SWITCH ──┤       │       (PART# ACOM1 OR ACOM2)     │
│     1 2 3 4 - 5 6 7 8        │  USE COMMAND "Z" TO UPLOAD 68HCXX COM │
│     ↓↓↑↓  ↑↓↓↓               │  PROGRAM BEFORE ACCESSING THE DEVICE │
│  - PRESS ANY KEY TO CONTINUE -│  SET ACOM2 DIP SWITCH -> ↓↓↑↓↑↓ │
└─────────────────────────────┴─────────────────────────────────┘
```

programming unit left justified. To install the adapter, lift the ZIF socket release handle on the 32 pin socket to about 45 degrees. This will release the socket mechanism. With the adapter fully left insert the pins into the socket. Release the handle to lock the adapter in place. The ACOM2 adapter requires power from the programming unit ACCESSORY CONNECTOR. Be sure the adapter power switch is OFF (left position). Attach the +5V (ORANGE) wire to the +5 pin on the accessory connector. Attach the Vpp (YELLOW) wire to the Vpp pin on the accessory connector. Set the switches on the adapter according to the right diagram shown on the COMMAND LIST screen. The adapter is now ready for use. **NOTE:** If you find it more convenient you may attach the ORANGE and YELLOW wires to the ACCESSORY CONNECTOR before installing the adapter into the 32 pin socket.

ATTACHING THE ACOM2 ADAPTER TO THE TARGET PROCESSOR

The ACOM2 adapter attaches to the target processor using a probe set. There are two probe sets available; the standard surface mount probe set (#SMP8) and the optional precision probe set (#PPS8). Both probe sets terminate with the standard 10 pin black plug.. To attach the black plug to the ACOM2 in-line connector the PBCP1 standard terminator board is required. The terminator board attaches to the ACOM2 with the black plug facing left (toward the ON/OFF switch). Both probe sets consist of eight individual colored probes. The probe colors are brown, red, orange, yellow, green, blue, violet and gray. Each color represents a number where brown through gray is 1 through 8. The probe colors are shown on the connection diagrams located at the end of this document. If more than one probe color is required to satisfy all of the pin connections required for communication, use the probe extensions supplied with the ACOM2 to expand a connection point. To complete the ACOM2 connections to the target processor attach each probe to the corresponding pin. **NOTE:** If there is any humidity sealant or coating on the chip thoroughly clean the pins to allow a positive electrical contact. The probe jaws must fully close which pulls the probe tip against the pin side.

APPLYING POWER TO THE PROCESSOR

Once all probes are attached to the target processor slide the ACOM2 power switch to the ON position. Be sure the ACOM2 power LED lights. **IMPORTANT:** If the power LED does not light immediately turn the ACOM2 power switch OFF. This is because there is a short between the +5 supply and GND. You must determine which probe is connected to a pin which causes the overload. The problem will be with one of the probes associated with "bootstrap" or "monitor" mode entry. Disconnect one probe at a time and then turn the ACOM2 adapter on. If the LED lights the last probe disconnected is causing the problem. To address the problem the pin must be isolated from the module circuitry. This is done by lifting the pin or cutting the connecting track. (See HOW TO ISOLATE A PIN). If the ACOM2 power LED lights proceed to COMMUNICATING.

COMMUNICATING WITH A 68HC05 OR 68HC11 PROCESSOR BASED MODULE

With the ACOM2 powered on and the probes properly attached to the target chip, attempt to establish communication with the processor. To establish communication press Z (DEVICE OPTIONS)(TOP Illustration). Then Press 1 (UPLOAD 68HCXX COMMUNICATION PROGRAM). The system will begin the upload and display BOOTSTRAP UPLOAD IN PROGRESS. (Second illustration). Once the buffer upload is complete the system will display * BUFFER UPLOAD COMPLETE *. (Third illustration). If the upload is successful and the communication program is running on the target processor the system will display (COMMUNICATION VERIFIED) below BUFFER UPLOAD COMPLETE.

IMPORTANT: If you do not see the (COMMUNICATION VERIFIED) message the communication program was not successfully uploaded and is not running. Do not proceed. You may try the upload again by pressing Z then 1. If you still do not see (COMMUNICATION VERIFIED) you must determine the reason. (See COMMUNICATION FAILURE). **IMPORTANT NOTE:** If the communication program upload is not successful the communication program will remain in the system buffer. If you select command 5 you will see data in the buffer. This is the communication program, not data from the microcontroller. The system clears the communication program from the buffer once communication has been verified.

If you see (COMMUNICATION VERIFIED) you may now proceed to perform any operation you desire on the EEPROM area of the microcontroller using any of the system commands.

EXAMPLE: READING THE EEPROM DATA INTO THE BUFFER

To read the microcontroller data into the system buffer press 5. This will invoke the buffer editor. You will see a screen of FF. Press 3 then Y. The system will display READING DEVICE... as the eeprom data is read into the buffer. Once all data is read it will appear on the display.

```
════ ANDROMEDA RESEARCH LABS EPROM+ PROGRAMMING SYSTEM - V6.0 ════
┌────────────────────────────┬───────────────────────┬───────────┐
│ DEVICE = 68HC11KA4 (PRESS S KEY) │ DEVICE SIZE = 0 - 27F │ Vpp = 5.00 │
├────────────────────────────┴───────────────────────┴───────────┤
│                        COMMAND LIST                             │
├─────────────────────────────────┬───────────────────────────────┤
│ 0 - EXIT PROGRAM/SYSTEM OPTIONS  │ A - SAVE BUFFER TO DISK FILE   │
│ 1 - PROGRAM DEVICE FROM DISK FILE│ B - SAVE DEVICE TO DISK FILE   │
│ 2 - PROGRAM DEVICE FROM BUFFER   │ D - DIRECTORY OF FILES         │
│ 3 - READ DEVICE INTO BUFFER      │ F - DISPLAY/SELECT PROG ALGORITHM│
│ 4 - READ DISK FILE INTO BUFFER   │ K - DEVICE/FILE CHECKSUM       │
│ 5 - BUFFER EDITOR                │ L - SYSTEM LIBRARIAN           │
│ 6 - VERIFY DEVICE IS ERASED      │ P - SET DRIVE/PATH             │
│ 7 - COPY DEVICE                  │ S - DIP SWITCH SETTINGS        │
│ 8 - COMPARE DEVICE WITH BUFFER   │ V - MANUALLY SET PROGRAMMING VOLTAGE│
│ 9 - CHANGE DEVICE TYPE           │ Z - DEVICE OPTIONS             │
├─────────────────────────────────┼───────────────────────────────┤
│      COMMAND/DATA ENTRY          │      ACTIVITY/STATUS           │
├─────────────────────────────────┼───────────────────────────────┤
│ UPLOAD (BOOTSTRAP) PROGRAM TO DEVICE │ NOTE: BEFORE UPLOADING PROGRAM │
│ 1 - UPLOAD 68HCXX COMMUNICATION PROGRAM │ VERIFY THAT THE TARGET PROCESSOR IS │
│ 2 - UPLOAD 256 BYTES FROM ADDRESS 0 NOW │ PROPERLY CONNECTED TO THE ACOM │
│ 3 - SPECIFY ENDING ADDRESS FOR UPLOAD │ ADAPTER. CONFIRM THAT BOTH THE ACOM │
│ SELECT OPERATION (1/2/3/(0)PT) -> │ AND TARGET ARE POWERED. │
└─────────────────────────────────┴───────────────────────────────┘
```

```
┌─────────────────────────────────┬───────────────────────────────┐
│      COMMAND/DATA ENTRY          │      ACTIVITY/STATUS           │
├─────────────────────────────────┼───────────────────────────────┤
│                                  │                               │
│                                  │   BOOTSTRAP UPLOAD IN PROGRESS...│
│                                  │                               │
└─────────────────────────────────┴───────────────────────────────┘
```

```
┌─────────────────────────────────┬───────────────────────────────┐
│      COMMAND/DATA ENTRY          │      ACTIVITY/STATUS           │
├─────────────────────────────────┼───────────────────────────────┤
│                                  │                               │
│                                  │   * BUFFER UPLOAD COMPLETE *  │
│                                  │                               │
└─────────────────────────────────┴───────────────────────────────┘
```

```
┌─────────────────────────────────┬───────────────────────────────┐
│      COMMAND/DATA ENTRY          │      ACTIVITY/STATUS           │
├─────────────────────────────────┼───────────────────────────────┤
│                                  │                               │
│                                  │   * BUFFER UPLOAD COMPLETE *  │
│                                  │     (COMMUNICATION VERIFIED)  │
└─────────────────────────────────┴───────────────────────────────┘
```

COMMUNICATION FAILURE

If the system fails to display COMMUNICATION VERIFIED you must determine the reason for the failure. There are several possibilities for the problem. The most common is that the pin which transmits data (TXD) from the microcontroller to the ACOM2 does not have sufficient drive capability. The ACOM2 is usually capable of driving the pins on the target processor however there is no way to guarantee that the target processor can provide sufficient drive back to the ACOM2. When you press Z then 1 the system issues a RESET command to the target. Once the target (68HC11 family only) receives the RESET it will transmit a byte of 00 back to the ACOM2. This confirms that the RESET was received by the processor. If the ACOM2 does not receive the 00 byte you will see the message WARNING! CONNECTION NOT CONFIRMED FROM TARGET PROCESSOR AFTER RESET | PROGRAM UPLOAD IN PROGRESS.....

The system will continue with the upload however you will very likely not see COMMUNICATION VERIFIED. To address the problem you must isolate the TXD pin from the underlying module circuitry.

NOTE: The 68HC05 family of microcontrollers does not issue the 00 byte following a RESET so you will not see the warning. If you do not see COMMUNICATION VERIFIED isolate the TXD pin and retry.

If after you lift the processor TXD pin you do not see COMMUNICATION VERIFIED lift the following pins in this order:

IRQ (68HC05 family only) - The voltage at this pin must be 8 volts or above. Before you lift the pin you can measure the voltage using a standard multimeter. Set the meter for DC VOLTS. Place the negative (black) probe on Vss (GND) and the positive (RED) probe on the IRQ pin. If the voltage is below 8 lift the pin.

NOTE: You can also measure the voltage at the ACOM2 connector with the probes attached between 9VG and GND.
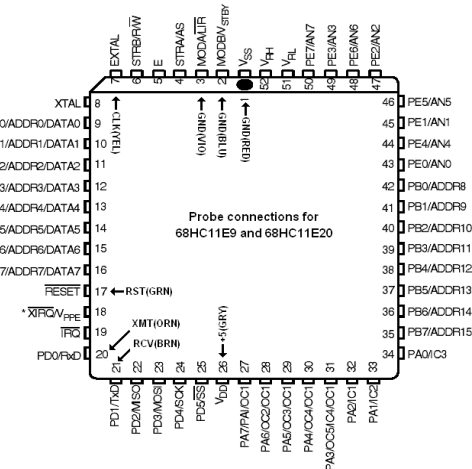
RXD - Receive data input (Transmit data from the ACOM2 to the target processor)

RESET - Reset signal to the target processor

IMPORTANT NOTE: The communication program which is running on the target processor is lost when the ACOM2 power is switched off. If for any reason the ACOM2 power is turned off you must again establish communication with the target processor by pressing Z then 1.

```
════ ANDROMEDA RESEARCH LABS EPROM+ PROGRAMMING SYSTEM - V6.0 ════
┌────────────────────────────┬───────────────────────┬───────────┐
│ DEVICE = 68HC11KA4 (PRESS S KEY) │ DEVICE SIZE = 0 - 27F │ Vpp = 5.00 │
├────────────────────────────┴───────────────────────┴───────────┤
│                        COMMAND LIST                             │
├─────────────────────────────────┬───────────────────────────────┤
│ 0 - EXIT PROGRAM/SYSTEM OPTIONS  │ A - SAVE BUFFER TO DISK FILE   │
│ 1 - PROGRAM DEVICE FROM DISK FILE│ B - SAVE DEVICE TO DISK FILE   │
│ 2 - PROGRAM DEVICE FROM BUFFER   │ D - DIRECTORY OF FILES         │
│ 3 - READ DEVICE INTO BUFFER      │ F - DISPLAY/SELECT PROG ALGORITHM│
│ 4 - READ DISK FILE INTO BUFFER   │ K - DEVICE/FILE CHECKSUM       │
│ 5 - BUFFER EDITOR                │ L - SYSTEM LIBRARIAN           │
│ 6 - VERIFY DEVICE IS ERASED      │ P - SET DRIVE/PATH             │
│ 7 - COPY DEVICE                  │ S - DIP SWITCH SETTINGS        │
│ 8 - COMPARE DEVICE WITH BUFFER   │ V - MANUALLY SET PROGRAMMING VOLTAGE│
│ 9 - CHANGE DEVICE TYPE           │ Z - DEVICE OPTIONS             │
├─────────────────────────────────┼───────────────────────────────┤
│      COMMAND/DATA ENTRY          │      ACTIVITY/STATUS           │
├─────────────────────────────────┼───────────────────────────────┤
│                                  │ WARNING! CONNECTION NOT CONFIRMED│
│                                  │ FROM TARGET PROCESSOR AFTER RESET│
│                                  │ PROGRAM UPLOAD IN PROGRESS......│
└─────────────────────────────────┴───────────────────────────────┘
```

Probe connections for
68HC11E9 and 68HC11E20

COMMUNICATING WITH A 68HC08/908 PROCESSOR BASED MODULE

With the ACOM2 powered on and the probes properly attached to the target chip, attempt to establish communication with the processor. To establish communication press Z (DEVICE OPTIONS)(TOP Illustration). Then Press 1 (COMMUNICATE (EE-PROM ONLY). The system will issue a reset to the target processor and wait for a response. If the processor responds, the system will then attempt to establish communication. You will see ATTEMPTING COMMUNICATION... WAIT! (Second illustration) While this message is displayed the system will attempt to write and read two bytes into the RAM of the target processor. If this operation is successful the system will display * COMMUNICATION CONFIRMED * (Third illustration) as both writing and reading the processor RAM was successful. Once successful communication with the part has been confirmed the system will upload a program into the processor RAM which is used to program the target EEPROM memory. You will see UPLOADING EEPROM ACCESS PROGRAM...WAIT! (Fourth illustration) The system displays a counter to indicate the upload progress. Once the eeprom access program is uploaded the system will display * BUFFER UPLOAD COMPLETE * (Fifth illustration). This message is briefly displayed after which the system will return to the SELECT COMMAND prompt. At this point you may now proceed to perform any operation you desire on the EEPROM area of the microcontroller using any of the system commands.

COMMUNICATION FAILURE

If communication with the target processor is not successful the system will display a message indicating where the communication process failed. Following the processor reset command sent by the ACOM2, the processor expects to receive a string of 8 security bytes. The security bytes must match a corresponding string stored in the processor itself. As each security byte is transmitted the same byte will be echoed back to the system via the ACOM2. If the security bytes are not echoed the processor fails the first and most basic test to confirm that the target is active. The system will respond with the message WARNING! PROCESSOR NOT RESPONDING | SWITCH ACOM2 POWER OFF & ON THEN RE-TRY. If you see this message turn the ACOM2 power switch off then on. After cycling the power switch press 1. If the message persists you must determine which pin or pins are causing the problem and then lift or isolate them from the underlying module circuitry.

The first step is to confirm that the IRQ pin is indeed at 8 volts or above. Before you lift the pin you can measure the voltage using a standard multimeter. Set the meter for DC VOLTS. Place the negative (black) probe on Vss (GND) and the positive (RED) probe on the IRQ pin. If the voltage is below 8 lift the pin. **NOTE:** You can also measure the voltage at the ACOM2 connector with the probes attached between 9VG and GND. The second pin to lift is the pin which the ACOM2 uses to transmit and receive data with the target processor. In the illustration this is PIN#26 (RCV) however it may be a different pin depending on the processor or package. The last pin to lift would be reset (RST). If the target processor is free from interference or loading from the underlying module circuitry it will respond.

NOTE: In rare cases the processor may have been fabricated to prevent access to the internal RAM or monitor mode itself. If either of these conditions exist the system will display one of the following messages:
COMMUNICATION FAILED!
REASON: The target entered monitor mode but would not write bytes into the processor RAM. If this message appears the internal eeprom cannot be accessed.
MONITOR ENTRY FAILED!
REASON: After the 8 byte security string was sent to the target the target is expected to send a special character called a "break" back to the ACOM2. If this character is not received the processor has not entered monitor mode and therefore the internal eeprom cannot be accessed.

HOW TO ISOLATE A PIN

There are two options which may be used to isolate a pin from the underlying circuitry. The first is to cut the track which connects tto the pin. If you choose this option be sure that the track does not continue under the processor package to additional connection points. The second option is to physically heat the pin until the solder melts and then lever the pin up therby breaking the connection to the underlying circuitry.

HOW TO CUT A TRACK

Locate an X-ACTO knife with a pointed blade. Locate a spot on the track and scrape away the solder mask (about 1/8th inch) until the underlying copper is revealed. The copper must be shiney. Carefully cut through the copper track. Be certain that the knife blade has passed completely through the track into the underlying glass-epoxy board material. It may take several strokes. Becareful to not cut any adjacent tracks. After the track is cut carefully use the tip of the knife blade to slightly lever up one side of the track. Don't peel the track up. Move it just enough to insure that the two ends are not touching. This insures that the track is indeed cut. You may also perform two cuts to create a gap but this makes track repair more difficult. If you want to insure that the track is cut you may use an ohmmeter (set to X1). Place a probe on each side of the cut and be sure you do not have zero ohms indicated. Some deflection is possible but zero ohms means the track is not cut.

HOW TO REPAIR THE CUT TRACK

To repair the cut track simply press the elevated side of the track down until both sides are flush. Take a small low wattage soldering iron such as a Weller Princess Model #WM120 (12Watt) with a sharp tip and apply a small amount of solder on top of the cut. Be sure the solder flows over both ends of the cut forming a bridge. The track is repaired.

HOW TO LIFT A PIN (PLCC PACKAGE)

The primary point of focus to successfully lift a pin is to be sure the solder which bonds the pin to the circuit board has indeed melted. To accomplish this a soldering iron with a 700 or 800 degree pointed tip works best. Once you have located the pin you need to first gently scrape away any sealant or coating. Once the pin is clean, apply a very small amount of solder to the pin side. Be carefull not to bridge adjacent pins. The purpose of the solder is to allow maximum heat transfer down the pin and into the solder attaching the pin to the board. Once the solder is applied remove the iron and insert a dental pick behind the pin. You may choose which pick end is best however remember that the pick will serve as the lever to pop the pin loose from the underside of the package. Remember that the pin does not just attach to the board with solder but also at the end forms a hook which is inserted in a small indent in the bottom of the processor package. This is why the term "J" lead is used as the bottom of the pin forms the lower part of the letter "J". Not only do you have to melt the solder but you also have to dislodge the part of the pin which is hooked into the indentation on the bottom of the package. To lift the pin first position the dental pick behind the pin. With the pick in place touch the soldering iron tip to the solder on the pin. With the soldering iron tip in place allow a few moments for the pin to heat and the solder to melt then gently lever the dental pick to force the pin from beneath the package body. Lift the pin until the connection clears the pad on the circuit board. Do not over extend the pin angle as this will stress the lead and possibly break the pin.

HOW TO REATTACH THE PLCC PIN TO THE CIRCUIT BOARD PAD

With the pin lifted and bent out at an angle from the package body, the bottom of the "J" will be visible. Using fine point needle nose pliers uncurl the "J" until the length of the pin is straight. On the very end of the pin use the needle nose pliers and bend a small section of the pin in the opposite direction from how the "J" was formed. You are creating a small flat portion on the end of the pin which will be soldered to the circuit board pad when the pin is pressed back down. The bottom of the pin will appear as an "L" where the bottom of the "L" will face outward from the package. Once you have formed the "L" at the bottom of the pin press the pin down until the bottom of the "L" contacts the circuit board pad. Continue pressing until the "L" is flush with the pad. Once the pin is in place use the soldering iron with a small amount of solder to connect the pin to the pad. The pin is now reattached to the circuit board.

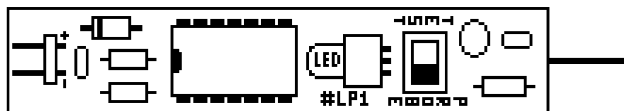## HOW TO LIFT A PIN (QFP PACKAGE)

Lifting a pin on a QFP (Quad Flat Pack) package is simply a matter of removing as much solder as possible from the pin/pad connection and then levering the pin up after applying heat with the soldering iron. The first step, after you have identified the pin to be lifted, is to use solderwik (.080" or smaller) with a soldering iron to wick as much solder away from the pin/pad junction as possible. If you do not wick the solder away from the pad the solder will bridge the space between the pin and the pad as the pin is lifted. After the solder is wicked from the pin/pad junction carefully place your dental pick behind the pin and touch the soldering iron to the pin/pad junction until the remaining solder melts. Very carefully lever the pin up (try not to cause the pin to twist) until the pin is free from the pad. Once the pin is free insert a small sliver of paper or other insulsting material under the pin. This will prevent the pin from accidentally contacting the pad while the probe is attached.

HOW TO REATTACH THE QFP PIN TO THE CIRCUIT BOARD PAD

To reattach the pin to the circuit board pad first remove the insulating material and press the pin down so that it contacts the pad. Using the soldering iron touch the top of the pin while applying a small amount of solder to the pin/pad junction. Be careful not to apply too much solder which can bridge to adjacent pads. Once the solder flows into the junction remove the iron. The pin is reattached.

OPTIONAL TOOL - THE LP-1 LOGIC/TEST PROBE

The LP-1 logic/test probe is an optional tool specifically designed to perform two functions. The first is to stimulate a pin on the microcontroller package to determine if the pin must be isolated for successful communication to occur. The second function is to indicate the logic level (0 or 1) of a specific pin. A problem you may encounter is that after you attach the probes to the necessary pins on the microcontroller and you attempt to establish communication, the system indicates COMMUNICATION FAILED or you do not see COMMUNICATION VERIFIED. You do not know which pin or pins are causing the failure. The LP-1 includes similar drive circuitry to that of the AR-32A programming unit. You use the LP-1 before you attach all of the probes to the microcontroller to test each pin for drive compatibility. It works like this; first you attach the LP-1 power leads to the microcontroller power (Vdd) and ground (Vss) probes. Each probe has an extra connection pin for the LP-1. Once the LP-1 is connected to the power probes you then attach *only* the power probes to the microcontroller. No other probes are attached. Both the LP-1 and microcontroller will be powered. You place the switch on the LP-1 to TEST and turn on the ACOM2 power switch. The LED on the LP-1 will begin to alternate between RED and GREEN. You now take the LP-1 and touch the probe tip to each pin on the microcontroller where a probe would be attached. If you touch a pin and the LED locks on RED or GREEN then that pin cannot be driven by the ACOM2 and must be isolated. This pretest saves significant time by preventing the unnecessary unsoldering of pins or cutting of tracks.

TWO PINS WHERE IT MAY NOT WORK

The LP-1 will successfully test pins on the microcontroller which must be driven to valid logic levels. There are two pins where this may not apply. The first is the IRQ pin. As stated previously in this document, this pin must be forced to 8 volts or above for successful communication to occur. You can test the pin with the LP-1 but it does not guarantee that with the ACOM2 attached, the pin will indeed rise to the required voltage. If communication cannot be established, test the pin voltage with a multimeter. The second pin where the LP-1 indication cannot provide a guarantee is the TXD pin (68HC11/68HC05) or the shared TXD/RXD pin (68HC08/908) on the microcontroller. This is the pin which sends data back to the ACOM2 therefore this pin is not driven by the ACOM2. The ACOM2 receives data from this pin. If the drive capability of the microcontroller is not sufficient for the ACOM2 to sense the incoming data (valid logic levels), you must isolate the pin to allow valid data reception. You may use the LP-1 indication as a pretest on the pin however this does not guarantee that the ACOM2 will receive the incoming data. If communication cannot be established, isolate the pin.

YOU MAY NOT NEED TO ATTACH A PROBE

The LP-1 also has a mode (PROBE) which allows you to determine the logic level (0 or 1) on a selected pin. Since the microcontroller must have predetermined logic levels applied to specific pins in order to allow communication, it is possible that the required logic level is already applied by the existing circuitry on the module itself. If this is the case you do not need to attach a probe to a pin which is already at the correct logic level. To test the logic level on a pin set the switch on the LP-1 to PROBE and turn on the ACOM2. With the ACOM2 power switch set to ON the LED will glow YELLOW (open circuit-no logic level). When you touch a pin which is at a logic 1 the LED will glow RED; a logic 0 will glow GREEN. Touch each pin on the microcontroller which is required to enter "bootstrap" or "monitor" mode. If the microcontroller pin is already at the correct level you will see it on the the LP-1 LED. If the LED color matches the required logic level you do not need to connect a probe to the pin.

HOW TO TEST THE ACOM2

The ACOM2 provides the circuitry required to communicate with Motorola microcontrollers which support an asynchronous interface. The ACOM2 performs this function by sending and receiving data to and from the microcontroller itself. If there is any question regarding the proper functioning of the ACOM2 adapter you may test the adapter for correct operation. The test is called a "loopback" test where the transmit (data out) signal is connected (looped back) to the receive (data in) input. During the test the ACOM2 exercises the internal circuitry by sending data to itself. To perform the "loopback" test enter the buffer editor by pressing 5 then press the "^" (SHIFT 6) key. You will see the illustration on the right. Set ACOM2 HC08 MODE (SW1) to ON then turn on the ACOM2. Press "T" to execute the test. If the ACOM2 is working properly you will see *** LOOPBACK TEST PASSED ***. (Second illustration) If the ACOM2 is not working properly you will see *** LOOPBACK TEST FAILED ***. (Third illustration)

```
══════════ BUFFER EDITOR ══════════
│ EDITOR MODE = BYTE │ BUFFER END = FFFFFF │ DEVICE SIZE = 0 - 1FF │

  0  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF    ................
 10  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF    ................
 20  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF    ................
 30  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF    ................
 40  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF    ................
 50  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF    ................
 60  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF    ................
 70  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF    ................
 80  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF    ................
 90  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF    ................
 A0  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF    ................
 B0  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF    ................
 C0  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF    ................
 D0  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF    ................
 E0  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF    ................
 F0  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF    ................

ACOM ADAPTER LOOPBACK TEST
ACOM1:SHORT XMT&RCV:ACOM2:HC08 MODE ON (SW1↑)
PRESS "T" TO TEST : "E" TO EXIT (T/E) ->
```

```
ACOM ADAPTER LOOPBACK TEST
ACOM1:SHORT XMT&RCV:ACOM2:HC08 MODE ON (SW1↑)  *** LOOPBACK TEST PASSED ***
PRESS "T" TO TEST : "E" TO EXIT (T/E) ->t
```

```
ACOM ADAPTER LOOPBACK TEST
ACOM1:SHORT XMT&RCV:ACOM2:HC08 MODE ON (SW1↑)  *** LOOPBACK TEST FAILED ***
PRESS "T" TO TEST : "E" TO EXIT (T/E) ->t
```

ACOM2 ADVANCED INFORMATION - If you are interested in the advanced features of the system software when used with the ACOM2 you may download from the Andromeda Research website (www.arlabs.com) the document ACOM2_ADV.PDF.

ACOM2 PROBE CONNECTIONS FOR MOTOROLA MICROCONTROLLERS

The following illustrations indicate which connections are necessary to allow the ACOM2 adapter to successfully communicate with the listed family member. Both the ACOM2 signal name and the probe color (#SMP8 or #PPS8) are shown. Packages were selected to illustrate the most common devices, however other parts are supported. In order to connect a microcontroller not shown, locate the part data sheet and reference the identical or corresponding pins. Attach the required ACOM2 signal to these pins and proceed according the the instructions in the applicable section of this document (68HC11, 68HC05 or 68HC08).
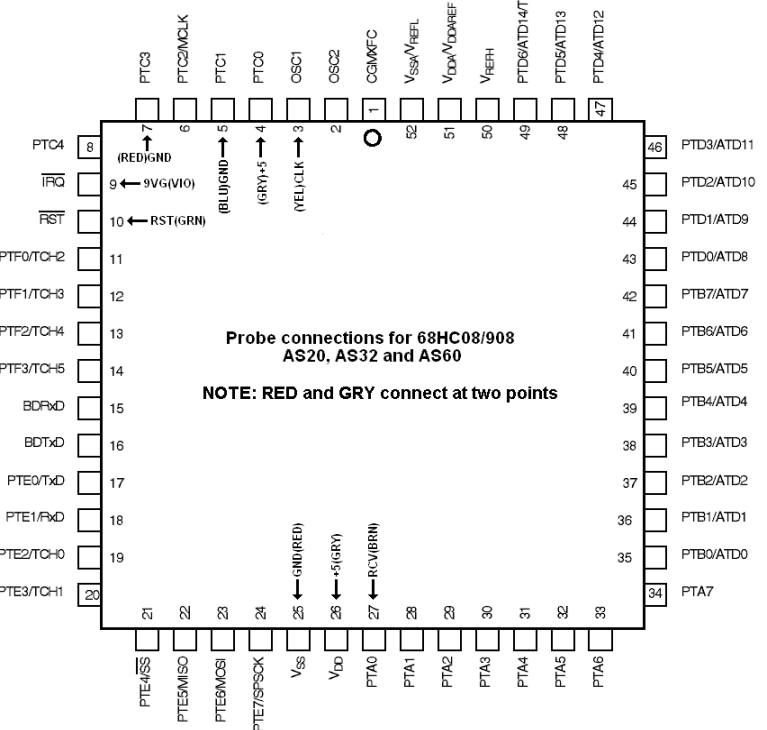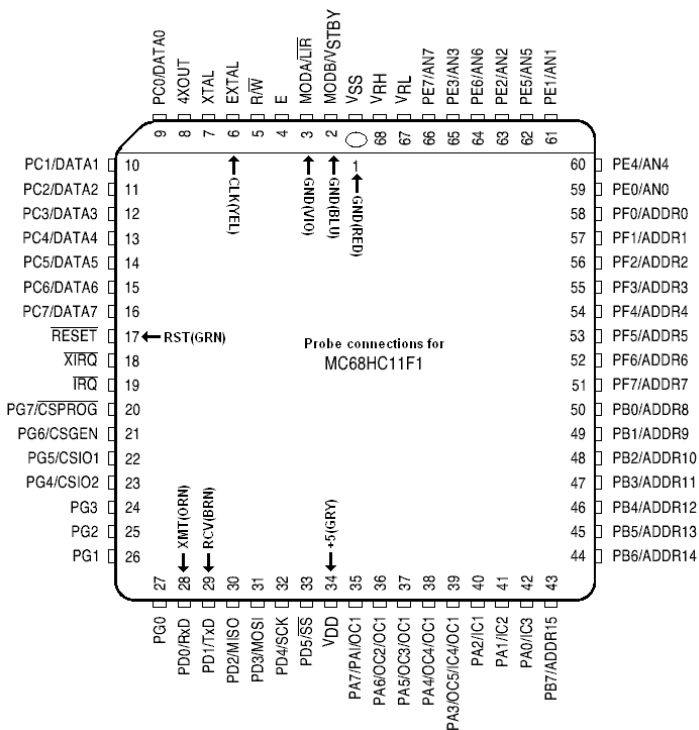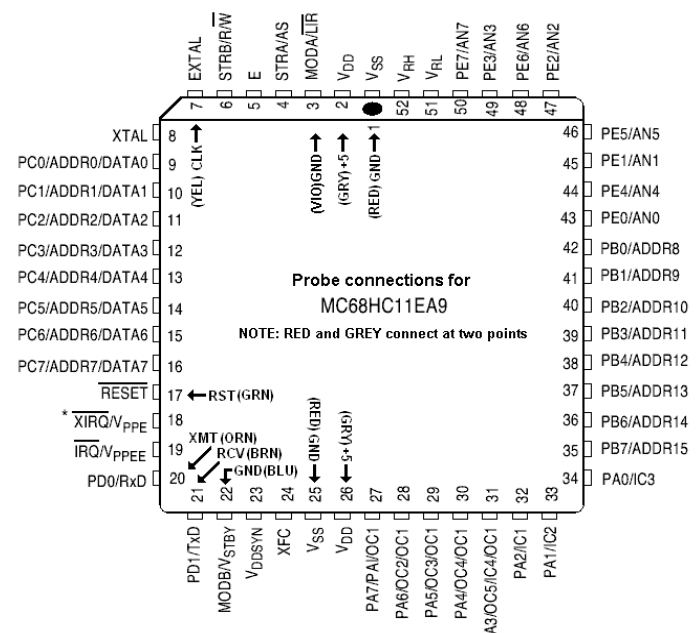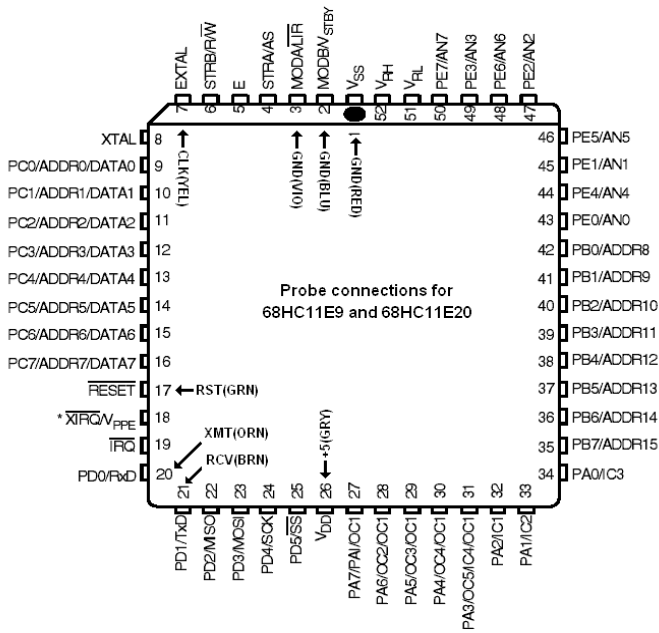
IMPORTANT NOTE 1: The ACOM2 adapter is designed with high current drivers which in most cases allow the attached microcontroller to be read and programmed in-circuit. If, however, you are not successful, you must determine which microcontroller pin is unable to be driven by the ACOM2 and disconnect (isolate) that pin from the circuit or module assembly. This can be done by carefully heating the pin with a low wattage soldering iron and using a small, sharp instrument to carefully lift the pin once the solder has melted. Once the pin has been lifted (isolated from the underlying board) connect the appropriate probe.

IMPORTANT NOTE 2 (HC11 family): To establish communication with the an HC11 family member both MODE (MODA and MODB) pins must be at a logic 0 (ground) state. This is accomplished using the BLUE and VIOLET probes. If all probes are connected and the ACOM2 power switch is set to ON the LED on the adapter should light. If the LED does not light then one of the MODE pins must be lifted as it is connected directly to the +5 volt module supply (GRY probe). This shorts out the power to the module which causes the programmer to current limit and the LED not to light. Turn the adapter switch OFF and remove one MOD probe (BLU or VIO) at a time until the switch can be set to ON with the LED lit. The probe color which causes the LED to remain OFF when connected is the pin which must be lifted. NOTE: You may also use the Andromeda Research Logic/Test Probe (#LP-1) to test a pin.

OTHER PINS WHICH MAY CAUSE PROBLEMS: HC11 family (RST, RCV, XMT), HC05 family (9VG, RST, RCV, XMT), HC08/908 family (9VG, RST, RCV) The HC05, 08/908 parts require that the IRQ pin voltage be at +9 volts. This is sourced from the 9VG (VIO) probe. If communication cannot be established measure the voltage between this pin and GND to confirm the correct value (9V). If the value is below 8 volts the pin must be lifted.

IMPORTANT NOTE 3: You should never lift a power supply pin (Vdd or Vss) on any microcontroller. These pins must remain connected as they supply current to other pins and use the module power supply by-pass capacitors for power stability. Also you only need to connect +5 and GND to one set of Vdd and Vss pins.

IMPORTANT NOTE 4: If multiple connections of a single voltage probe are required, attach an extension probe to the primary signal probe for the connection(s).

Probe connections for 68HC11E9 and 68HC11E20

Probe connections for MC68HC11EA9

NOTE: RED and GREY connect at two points

Probe connections for MC68HC11F1

Probe connections for 68HC08/908 AS20, AS32 and AS60

NOTE: RED and GRY connect at two points

Probe connections for 68HC05B6 and B16

NOTE: GRAY connected at three points

MC68HC11KA4
MC68HC11KA2

Probe connections for 68HC11K4

68HC05X16 and X32

Probe connections for
68HC08/908AS20, AS32 and AS60
68HC08/908AZ32, AZ48 and AZ60

64 PIN QFP PACKAGE

NOTE: FOR 68HC08/908AS PARTS PIN 63 = GND(RED)
FOR 68HC08/908AZ PARTS PIN 63 = +5(GRY)

Probe connections for the
68HC11L6