Microcontroller Concepts

A. What is a microcontroller?
B. Top-down view of microcontroller systems
C. Memory concepts
D. Microcontroller memory map
E. The Motorola microcontroller

What is a Microcontroller?

- The microcontroller is a programmable single-chip IC that controls the operation of a system
- CPU chips with reduced computing power and with built-in memory and interface circuits

Top-Down View of Microcontroller Systems

- The microcontroller system
- The microcontroller unit (MCU)
- The central processing unit (CPU)
The Microcontroller System

- Support components for the MCU
  - Buffers and converters
  - Bus – signals
    - data: instructions and values
    - address: location where data is stored
    - control: coordinate microcontroller operation with associated chips
  - Clock circuitry
  - Power circuit

- Buffers and converters
- Bus – signals
  - data: instructions and values
  - address: location where data is stored
  - control: coordinate microcontroller operation with associated chips
- Clock circuitry
- Power circuit

The Microcontroller Unit (MCU)

- Microcontroller unit has 3 basic parts connected by an internal bus
  - CPU
  - Memory
  - Registers
- I/O registers
  - data, control, status
- I/O port is a collection of I/O pins on the chip that represents a unit of data

Block diagram of a typical microcontroller shown in single-chip mode

The Microcontroller Unit (MCU) Example of an Input Operation

The Microcontroller Unit (MCU) Example of an Output Operation
The Microcontroller Unit (MCU)

- The I/O control and data registers control and monitor the microcontroller I/O process
- Microcontrollers have internal timers
- The microcontroller uses other external pins for power and control and, if necessary, data and address lines
- RESET is a control line that is used to bring the microcontroller to an initial state
- CLOCK is also a control line

**Example**

- Speed Signal
  - Speed Sensor
  - Counter
    - Set Counter to count on ↑ edges
  - CPU
    - CPU reads counter at every time interval
  - Timer
    - Program timer’s interrupt interval

**The Central Processing Unit (CPU)**

- CPU executes program instructions
- Program counter (PC) is a special register that points to the instructions
- Instruction decoder tells the ALU what to do with the data
- Control sequencer manages the transfer of instruction and data bytes along the internal data bus

**The Microcontroller Unit (MCU) Expanded Mode**

- Sometimes, a microcontroller requires more memory or I/O ports than are available in the chip mode
  - need external data and address line connections
  - some pins can be used either as I/O ports or as external data and address lines
  - set microcontroller’s mode of operation

Block diagram of a typical microcontroller shown in expanded multiplexed mode
Microcontroller Concepts

A. What is a microcontroller?
B. Top-down view of microcontroller systems
C. Memory concepts
D. Microcontroller memory map
E. The Motorola microcontroller

Memory Concepts

- Numbering conventions
  - binary: %01010101
  - hexadecimal: $2A
- Main types of memory:
  - ROM
  - RAM
  - EPROM
  - EEPROM

Microcontroller Memory Map

- A memory map is a diagram that shows the computer’s available addresses and how they are used
- The default condition is the condition of the chip, as supplied by the manufacturer
- Addressing range: 64KB for 2-byte address
Microcontroller Concepts

A. What is a microcontroller?
B. Top-down view of microcontroller systems
C. Memory concepts
D. Microcontroller memory map
E. The Motorola microcontroller

The Motorola Microcontroller

- Programmer’s model
- Pin descriptions
- Termination of unused pins
- Configuration and modes of operation
- On-chip memory
- Resets and Interrupts
68HC11 Family of MCUs

Pin Descriptions (V\text{DD}, V\text{SS}, MODB/V\text{STBY}, MODA/LIR\text{^}{})

- Power supply pins
  - V\text{DD} is the positive power input
  - V\text{SS} is the ground
- Mode select pins
  - Mode B/standby RAM supply (MODB/V\text{STBY})
    - functions as both a mode select input pin and a standby power-supply pin
  - Mode A/load instruction register (MODA/LIR\text{^}{})
    - used to select MCU operating mode while the MCU is in reset and it operates as a diagnostic output signal while the MCU is executing instructions

Programmer's Model

- Power supply pins
  - V\text{DD} is the positive power input
  - V\text{SS} is the ground
- Mode select pins
  - Mode B/standby RAM supply (MODB/V\text{STBY})
    - functions as both a mode select input pin and a standby power-supply pin
  - Mode A/load instruction register (MODA/LIR\text{^}{})
    - used to select MCU operating mode while the MCU is in reset and it operates as a diagnostic output signal while the MCU is executing instructions
### Pin Description

**SMOD**: special mode control bit in the highest priority interrupt (HPRIO) register

**MDA**: mode A select control bit in HPRIO

**Bootstrap** mode is the special variation of the single chip mode

**Special Test** is the special variation of the expanded mode

### Pin Description

**EXTAL, XTAL, E**

- Crystal oscillator and clock pins
  - E clock is the bus frequency clock output
  - E clock is free running at one-fourth the crystal frequency

### Pin Description

**RESET^**

- **RESET^** is an active-low, bidirectional control signal
- **Input**: initialize the microcontroller to a known startup state
- **Output**: indicate that an internal failure has been detected

### Pin Description

**Interrupt pins (XIRQ^; IRQ^)**

- **XIRQ^**: provides a mean for requesting nonmaskable interrupts after reset
  - during reset, X bit in the condition code register (CCR) is set => all interrupts are masked
  - XIRQ^ is level sensitive and can be connected to a multiple-source wired-OR network
- **IRQ^**: provides means for requesting asynchronous interrupts to the controller
  - IRQ^ is program selectable having a choice of either level-sensitive or falling-edge-sensitive triggering
Pin Description
A/D Reference (VREFH; VREFL)
- The VREFH and VREFL pins provide the reference voltage for the A/D converter circuitry
  - These pins are normally connected to VDD and VSS through a low-pass filter network
  - There should be at least 2.5 V between VREFL and VREFH for full A/D accuracy

Pin Description
Port E pins (PE7-PE0)
- Port E input pins are used for general-purpose input and/or A/D analog inputs
  - the analog and digital functions of port E do not normally interfere with each other
  - digital input buffers are disabled at all times except for part of a cycle during an actual read of port E => reduced power-supply current drains

Pin Description
(Timer Port A Pins)
- Port A includes
  - 3 input-only pins: PA0/IC3, PA1/IC2, PA2/IC1
    - also serve as edge-sensitive timer input-capture pins
  - 4 output-only pins: PA3/OC5/OC1, PA4/OC4/OC1, PA5/OC3/OC1, PA6/OC2/OC1
    - also serve as main timer output-compare pins
    - these 4 pins can be controlled by output compare 1 (OC1) and/or another output compare
  - 1 pin that can be configured as input or output: PA7/PAI/OC1
    - general-purpose I/O; pulse-accumulator input; OC1 output pin

Pin Description
(Serial Port D Pins)
- Port D includes 6 general-purpose bidirectional I/O pins that can be individually configured
  - When SCI receiver is enabled => PD0/RxD becomes an input dedicated to RxD function
  - When SCI transmitter is enabled => PD1/TxD becomes an output dedicated to TxD function
  - When SPI system is enabled => PD2/MISO, PD3/MOSI, PD4/SCK, PD5/SS become dedicated SPI functions
  - The 6 port D pins can be configured for wired-OR operation
Pin Description

**Pin Description (Port B and C, STRA, and STRB)**

- The 18 pins are used for general purpose I/O while MCU is operating in single-chip mode.
- When extended mode is used, these 18 pins become a multiplexed address/data bus with AS (address select) and R/W* (read/write) control line.
- Single chip mode
  - Port B is an 8-bit output only port.
  - Port C is an 8-bit bidirectional I/O port.
  - Several automated handshake I/O functions are associated with ports B and C => strobe A (STRA) and strobe B (STRB) are used.

**Termination of Unused Pins**

- CMOS inverter
- Avoidance of pin damage
- Zap and latchup
- Internal circuitry. Digital input-only pin
- Internal circuitry. Analog input-only pin
- Internal circuitry. Digital I/O pin
- Internal circuitry. Input/Open-Drain-Output pin
- Internal circuitry. Digital output-only pin

### Pin Description

<table>
<thead>
<tr>
<th>Port</th>
<th>Bit</th>
<th>Single-Chip and Bootstrap Mode</th>
<th>Expanded-Multiplexed and Special Test Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>B 0</td>
<td>P81</td>
<td>Output</td>
<td>A8 Address Output</td>
</tr>
<tr>
<td>B 1</td>
<td>P82</td>
<td>Output</td>
<td>A9 Address Output</td>
</tr>
<tr>
<td>B 2</td>
<td>P83</td>
<td>Output</td>
<td>A10 Address Output</td>
</tr>
<tr>
<td>B 3</td>
<td>P84</td>
<td>Output</td>
<td>A11 Address Output</td>
</tr>
<tr>
<td>B 4</td>
<td>P85</td>
<td>Output</td>
<td>A12 Address Output</td>
</tr>
<tr>
<td>B 5</td>
<td>P86</td>
<td>Output</td>
<td>A13 Address Output</td>
</tr>
<tr>
<td>B 6</td>
<td>P87</td>
<td>Output</td>
<td>A14 Address Output</td>
</tr>
<tr>
<td>C 0</td>
<td>PC1</td>
<td>Input/Output</td>
<td>A20 Address Data Multiplexed</td>
</tr>
<tr>
<td>C 1</td>
<td>PC2</td>
<td>Input/Output</td>
<td>A21 Address Data Multiplexed</td>
</tr>
<tr>
<td>C 2</td>
<td>PC3</td>
<td>Input/Output</td>
<td>A22 Address Data Multiplexed</td>
</tr>
<tr>
<td>C 3</td>
<td>PC4</td>
<td>Input/Output</td>
<td>A23 Address Data Multiplexed</td>
</tr>
<tr>
<td>C 4</td>
<td>PC5</td>
<td>Input/Output</td>
<td>A24 Address Data Multiplexed</td>
</tr>
<tr>
<td>C 5</td>
<td>PC6</td>
<td>Input/Output</td>
<td>A25 Address Data Multiplexed</td>
</tr>
<tr>
<td>C 6</td>
<td>PC7</td>
<td>Input/Output</td>
<td>A26 Address Data Multiplexed</td>
</tr>
<tr>
<td>C 7</td>
<td>PC8</td>
<td>Input/Output</td>
<td>A27 Address Data Multiplexed</td>
</tr>
<tr>
<td>STRA</td>
<td>Output Stroke (High Imp)</td>
<td>A5 Address strobe (Out)</td>
<td>R/W* Ready/Write Select</td>
</tr>
</tbody>
</table>

### CMOS Inverter

- In CMOS devices, unused pins must be terminated to assure proper operation and reliability.
- The overall current drain of a CMOS device is directly proportional to the rate of switching.
  - If the input is not connected it can float to a midsupply level => P and N are on => added power supply current.
**Avoidance of Pin Damage**

- Some inputs on the MCU (RESET, EXTAL, MODA, MODB) cannot be left unterminated.
- VREFL and VREFH do not connect to the inputs of any CMOS gates within the MCU => no termination needed.
- Types of input pins:
  - A/D pins
  - input-only
  - input/output
- Unused inputs can be terminated with a pullup or pulldown resistor for each unused pin.
- If a pullup or pulldown resistor is used a signal can be easily connected to the input later.

**Zap and Latchup**

- Zap refers to damage caused by very high-voltage static-electricity exposure.
- Latchup refers to an usually catastrophic condition caused by turning on an unintentional, bipolar, silicon-controlled rectifier (SCR).
- SCR is formed by N and P regions in the layout of the IC which act as collector, base, and emitters for parasitic tr.
- Bulk resistance of silicon in the wells and substrate act as resistors in the SCR.
- Application of voltages above VDD or under VSS => can cause latchup.

**Protective Interface Circuits**

- In applications where MCU pins might be exposed to detrimental conditions, protective interfaces may be needed to protect MCU from damage.
- The goals of protective interfaces are:
  - prevent high currents from flowing
  - prevent illegal voltage levels at a pin
- Methods:
  - low-pass filters: protect high current and voltage
  - diode clamps: high voltage protection
Internal Circuitry. Digital Input-Only Pin

- Allowing a pin to float (or be driven) to a mid-supply level can result in both the N- and P-channel devices in the input buffer simultaneously being partially on, which causes excess current and noise on the VDD/VSS power supply.
- Port E inputs are exceptions because they are designed to be driven by analog levels.

Pin current levels:
Max: 100mA
Specified: 25mA

Digital Circuitry. Analog Input-Only Pin

- The N-channel device [4] allows the analog input pins to be driven by intermediate levels.
- The device is only turned on for half an E-clock cycle during a digital read of port E.

Internal Circuitry. Digital I/O Pin

- The port C and port D I/O pins can be configured as open-drain-type outputs by disabling the gate signal to the P-channel device of the output buffer.

Pin current levels:
Max: 100mA
Specified: 25mA

Internal Circuitry. Input/Open-Drain-Output Pin

- Circuit applies for RESET^ and MODA/LIR^ pins.

[Diagrams and images of circuitry are shown in the images embedded in the text.]
**Internal Circuitry. Digital Output-Only Pin**

- Output-only pins react at illegal levels exactly like I/O pins.

**Internal Circuitry. MODB/V\_STBY Pin**

- This pin serves as a standby voltage and a mode select input.
- An MOS switch automatically connects the internal RAM power supply to the higher of V\_DD or V\_STBY.

**Internal Circuitry. IRQ^/V\_PPBULK Pin**

- The IRQ^ is used as a high voltage (20V) power source during factory testing.
**Typical Single-Chip-Mode System Connections**

![Diagram of typical single-chip-mode system connections]

**Configuration and Modes of Operation**
- Hardware mode selection
- EEPROM-based CONFIG register
- Protected control register bits
- Normal MCU operating modes
- Special MCU operating modes

**Hardware Mode Selection**

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Mode Description</th>
<th>Control bits in HPRIO (default at Reset)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MODA</td>
<td>SMOD</td>
<td>MODA</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

- There are two fundamental modes of operation: single and expanded
- Each mode has a normal variation and a special one
- After RESET\(^{\wedge}\) rises the mode select pins no longer influence the MCU operating mode

**Hardware Mode Selection**

Mode control bits in the HPRIO Register

- The captured MODA and MODB levels (during the reset state) determine the logic state of the special mode (SMOD) and the mode A select (MDA) control bits in the highest priority interrupt (HPRIO) register
- SMOD and MDA control the logic circuits involved in hardware mode selection
  - RBOOT: read bootstrap ROM
  - Writable only while SMOD equals one
EEPROM-Based CONFIG Register

- CONFIG register is used to enable or disable ROM, EEPROM, the computer operating (COP) watchdog system, and, optionally, the EEPROM security feature of the MCU
- Check the MCU version for correct information about this register

Protected Control Register Bits

- In some 68HC11 MCU versions several sensitive control registers and bits are protected against writes
- RAM and I/O mapping register (INIT)
- Protected control bits in the TMSK2 register
- Protected control bits in the OPTION register
- The protect mechanism include the ability to write these bits only within the first 64 bus cycles after any reset and/or the ability to write them only one time after each reset

Protected Control Register Bits RAM and I/O mapping register INIT

- The INIT register allows to reposition the internal 256-byte RAM and/or 64-byte register space to any 4K page boundary in the 64K-byte memory map
- Default location after reset are:
  - RAM from $0000-$00FF
  - Registers from $1000-$103F
- Reasons for reposition capability -

Protected Control Register Bits TMSK2 Register

- PR1-PR0 are time-protected timer prescale select register in the timer mask register 2 (TMSK2)
- PR1-PR0 select the prescale rate for the main 16-bit, free running timer system
Protected Control Register Bits

<table>
<thead>
<tr>
<th>Register</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADEN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSEL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IROE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BLV</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CR1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CR0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Option register

- Time protected control bits on the OPTION register

Normal MCU Operation Modes

- MODB = 1 during reset => normal modes
  - The reset vector is fetched from $FFFF,FFFF
  - The execution begins at the address indicated by this vector
- Normal single-chip mode operation
  - the internal 8K-byte program memory is enabled in this mode => reset vector is fetched from ROM
- Expanded mode
  - the internal 8K-byte ROM may or may not be enabled (ROMON bit in CONFIG tells this)

Special MCU Operating Modes

- The special mode variation are selected by having a logic zero on the MODB pin during reset
- In special mode variations, the reset and interrupt vectors are located at $BFC0-$BFFF

ON-Chip Memory

- ROM
- RAM
- EEPROM
### On-Chip Memory

- **68HC11 includes**:  
  - on-chip random-access memory (RAM)  
  - read-only memory (ROM)  
  - electrically erasable programmable ROM (EEPROM) memories  
- RAM used for storage of variable and temporary information  
  - 192 bytes or more  
- ROM is used for storage of user program instructions and fixed data  
  - 4K or more  
- EEPROM is used for semipermanent information such as calibration tables, personality data, product history information, or program memory  
  - 512 bytes or more

### ROM

- **Primary use of ROM**:  
  - hold the user’s application program instructions  
  - These instructions are programmed into the MCU when it is manufactured  
  - they cannot be changed

### RAM

- **The on-chip RAM can be mapped to the beginning of any 4K block in the 64K-byte address space**  
- **By default**: $0000-$00FF  
  - the first 256 locations in memory are accessible using the direct addressing mode

### EEPROM

- **Information in EEPROM can be erased or reprogrammed under software control**  
  - no special power supplies are needed  
- The floating-gate transistor is the storage element in the EEPROM cell
EEPROM

- Reprogrammable nonvolatile memory has largely displaced PROMs.
- These memories, including EPROM, EEPROM, and Flash, use a second layer of polysilicon to form a floating gate between the primary gate and the channel.
  - High voltage on the upper gate \( \Rightarrow \) avalanche injection.

Resets and Interrupts

- Initial conditions established during reset
- Causes of reset
- Interrupt process
- Nonmaskable interrupts
- Maskable interrupts
- Interrupt request
- Interrupts from internal peripheral subsystems.

Initial Conditions Established During Reset

- Reset is used to force the MCU to assume a set of initial conditions and to begin executing instructions from a predetermined starting address.
- Internal registers and control bits are forced to an initial state.
- These initial states, in turn control on-chip peripheral systems to force them to known start-up-states.
Initial Conditions Established During Reset

- **CPU**: after reset, the CPU fetches the restart vector from locations $FFFF$, $FFFF$
- The stack pointer and other CPU registers are indeterminate
- X and I interrupt mask bits in CCR are set
- S bit in CCR is set to disable the STOP mode

- **Memory map**:
  - RAM and I/O mapping (INIT) register is initialized to $01 => RAM: $0000-$00FF
  - Control registers: $1000-$103F
  - ROM and EEPROM may or may not be present
  - CONFIG register

- **Parallel I/O (single)**
  - STAF, STAI and HNDS control bits in PIOC register are cleared
  - Port C is initialized as an input port (DDRC=$00)
  - Port B as a general purpose output port with all bits set at 0

- **Timer**
  - The timer system is initialized to a count of $0000
  - The prescaler bits are cleared
  - All output-compare registers are $FFFF
  - All input capture reg. are indeterminate
  - OC1F is cleared

- **COP watchdog**
  - computer operating properly (COP) watchdog system is enabled if NOCOP control bit in the CONFIG register is clear and disabled if NOCOP is set

- **Real-Time Interrupt**
  - Real-time interrupt flag is cleared
  - automatic hardware interrupts are masked
  - rate control bits are cleared

- **Pulse Accumulator**
  - pulse accumulator is disabled => pulse accumulator (PAI) pin defaults to being a general-purpose input pin

- **Serial communication interface (SCI)**
  - SCI baud rate is indeterminate
  - all transmit and receive interrupts are masked
  - both the transmitter and receiver are disabled
Initial Conditions Established During Reset

- Serial peripheral interface (SPI)
- SPI system is disabled
- the port pins associated with this function default to being general-purpose I/O lines
- Analog-to-digital converter
- A/D converter is indeterminate
- conversion complete flag is cleared
- A/D powerup (ADPU) bit is cleared => A/D system is disabled

Causes of Reset

- Poweron reset (POR)
  - POR circuit triggers and initiates a reset sequence
- COP watchdog timer reset
  - intended to detect software processing errors
- Clock monitor reset
  - if no MCU clock edges are detected within an RC time delay => the clock monitor can optionally generate a system reset: RESET^ is active – 4 Es
- External reset
  - reset can be forced by applying a low level to the RESET^ pin

Interrupt Process

- Interrupts can be enabled or disabled by mask bits (X and I) in the CCR and by the local enable mask bits in the on-chip peripheral control registers
- The instructions executed in response to an interrupt are called the interrupt service routine (ISR)
- ISR are called through the automatic hardware interrupt mechanism
- interrupt; end of current instruction; CPU registers => stack; the vector for highest priority => PC; execute ISR instructions; RTI concludes an interrupt

Causes of Reset

<table>
<thead>
<tr>
<th>Cause of Reset</th>
<th>Normal Mode Vector</th>
<th>Special Test or Bootstrap Vector</th>
</tr>
</thead>
<tbody>
<tr>
<td>POR or RESET Pin</td>
<td>$FFE,FFFF</td>
<td>$FFE,FFFF</td>
</tr>
<tr>
<td>Clock Monitor Fail</td>
<td>$FFC,FFFF</td>
<td>$FFC,FFFF</td>
</tr>
<tr>
<td>COP Watchdog Time-Out</td>
<td>$FFA,FFFF</td>
<td>$FFA,FFFF</td>
</tr>
</tbody>
</table>
Interrupt Process

- Interrupts obey a fixed hardware-priority circuit to resolve simultaneous requests.
- The first 6 interrupt sources are not masked by the I bit in the CCR:
  - reset; clock monitor fail; COP fail; illegal opcode; and XIRQ^.
- The highest I-bit related priority input is assigned under software control (of the HPRIO register) to be connected to any one of the remaining I-bit related interrupt sources.

Interrupt Process

- These 4 bits allow any one maskable interrupt to be elevated to the highest priority position.

<table>
<thead>
<tr>
<th>PSEL3</th>
<th>PSEL2</th>
<th>PSEL1</th>
<th>PSEL0</th>
<th>Interrupt Source/Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Timer Overflow</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>Timer Accumulator Overflow</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>Timer Accumulator Input Edge</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>SPI Transfer Complete</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>SDI Server</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>Reserved (Settable by IRQ)</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>IRQ External Pin (Pulse 10)</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>Real-Time Interrupt</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Timer Input Capture 1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>Timer Input Capture 2</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>Timer Input Capture 3</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>Timer Output Capture 1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>Timer Output Capture 2</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>Timer Output Capture 3</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>Timer Output Capture 4</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>Timer Output Capture 5</td>
</tr>
</tbody>
</table>

Nonmaskable Interrupt Request (XIRQ^)

- The most common use for such an interrupt is for very serious system problems such as program runaway or power failure.
- Importance of X bit:
  - X is set after reset => XIRQ^ is inhibited
  - after software has established initial conditions the X bit may be cleared with a TAP instruction => XIRQ enabled
  - When XIRQ^ occurs => CCR value is stacked with X bit cleared
  - the X bit is automatically set to inhibit nested interrupts
  - when RTI is exec. => X bit is restored to clear

Nonmaskable Interrupt Request (XIRQ^)

- The most common use for such an interrupt is for very serious system problems such as program runaway or power failure.
- Importance of X bit:
  - X is set after reset => XIRQ^ is inhibited
  - after software has established initial conditions the X bit may be cleared with a TAP instruction => XIRQ enabled
  - When XIRQ^ occurs => CCR value is stacked with X bit cleared
  - the X bit is automatically set to inhibit nested interrupts
  - when RTI is exec. => X bit is restored to clear
Maskable Interrupts

- After the CCR value is stacked the I bit in the CCR is set to inhibit further interrupts => there is no need for a SEI at the beginning of ISR
- The last instruction in the ISR is the RTI instruction
  - restore the registers and PC to the preinterrupt values
  - clear the I bit during normal execution of RTI => no need for CLI before RTI

Interrupts from Internal Peripheral Subsystems

- All on-chip interrupt sources have software accessible control bits
- Each source has
  - a flag bit, which indicates service is required, and
  - an interrupt enable bit which enables the flag to generate hardware interrupt request
- The global interrupt mask I can be used to inhibit all maskable interrupts
- The interrupt status flags must be cleared after service

Interrupt Request

- Only one IRQ pin but:
  - The pulse accumulator, IC3-IC1, and STRA input pins can be used as edge-sensitive interrupt inputs
  - The default configuration for IRQ is low-level-sensitive wired-OR-network
  - many interrupt sources can be accommodated to IRQ
- IRQE bit in OPTION register is used to select IRQ to be level sensitive or low-going edge sensitive
- The interrupt sources within MCU all operate as a wired-OR level-sensitive network
- In an edge-sensitive network MCU is responsible for latching a request