MICROCHIP

AN1081

Interfacing a 4x4 Matrix Keypad with an 8-Bit GPIO Expander

Author: Mike Curran
Microchip Technology Inc.

INTRODUCTION

This application note discusses interfacing a 4x4 matrix keypad with MCP23X08 8-Bit GPIO Expander. This application note references the MCP23X08/17 GPIO Expander Keypad/LCD Demo Board (GPIODM-KPLCD).

GPIO Expanders provide easy I/O expansion using standard serial interfaces such as I^2C™ and SPI. They are especially useful in applications where pin count is limited on the microcontroller unit (MCU) or if remote inputs / outputs (I/O's) are needed. It is best to think of an 8-bit GPIO Expander like adding another 8-bit wide digital port to the MCU.

The MCP23X08 8-Bit GPIO Expanders family consists of two (2) devices which are similar, except for their serial interfaces:

- MCP23008 - I^2C
- MCP23S08 - SPI

This application note does not detail all of the features of the MCP23X08. Refer to the MCP23008/MCP23S08 Data Sheet, *"8-Bit I/O Expander with Serial Interface"* (DS21919) for more information.

FEATURE DESCRIPTIONS

The MCP23X08 devices have several features that make them ideal for controlling a 4x4 matrix keypad. These features have been broken down into two main groups:

1. The ports input and output characteristics.
2. The interrupt-on-change feature, which is an important aspect of the key scan method used.

Input and Output Characteristics

There are three (3) registers that control the port pins input and output characteristics that need to be manipulated:

- **GPIO Pull-Up Resistor Register (GPPU):** The GPPU register controls the individually selectable internal 100 kΩ pull-up resistors. When set, the pull-up resistor is enabled.
- **I/O Direction Register (IODIR):** The IODIR register controls the direction of the port pins. A high on the corresponding bit configures the pin as an input and a low configures it as an output.
- **General Purpose I/O Port Register (GPIO):** The GPIO register is the data port of the MCP23X08. Reading from this register reads the state of the port pins. Writing to the GPIO register will write that value to the output latches for each pin. Also, reading of GPIO unlocks the Interrupt Capture Register (INTCAP) and deactivates the interrupt pin (INT), which will be explained in detail later.

Interrupt-On-Change Feature

The MCP23X08 devices have one interrupt output pin (INT), which, if enabled, will activate when an input pin (GP7-GP0) changes state. This is known as Interrupt-on-change and can be configured to function with one of the following options:

1. <u>If the state of the pin changes from the current state.</u> This option is useful for monitoring a pin where anytime the state of the pin is changed, an action needs to be taken. For example, if the state of the pin switches from high to low, an interrupt will occur. Assuming the interrupt is serviced (cleared) while the pin is low, when the pin changes back to a high state another interrupt will occur. This is a useful option for many applications, but is not desired for this one.
2. <u>If the pin changes state as compared to a pre-configured default value in the Default Value Register (DEFVAL).</u> This option is used for the key scan which this application note discusses. An interrupt will occur if the pin is in an opposite state as compared to a pre-configured default input value. For example, if the default value (configured in DEFVAL) is set high, and assuming the pin's idle state is high, once the corresponding pin changes state from high to low an interrupt will occur. The interrupt condition will stay active as long as the input pin remains in the low state. Once the input pin returns to a high state and the INTCAP or GPIO register is read the interrupt will clear.

> **Note:** The interrupt condition will remain (i.e. the INT pin cannot be cleared) as long as there is a mismatch between the input and corresponding bit in DEFVAL.

Interrupt-On-Change Details

The interrupt pin (INT) can be configured as active low or active high with the IOCON.INTPOL bit. For this application, the default value of active low will be used.

To use the interrupt-on-change feature from a default value for the key scan, three (3) registers must be configured and one (1) register will need to be read. They include:

- **Interrupt-On-Change Enable Register (GPINTEN):** The GPINTEN register sets up each pin for interrupt-on-change. When set, the corresponding pin will be enabled for interrupt-on-change.
- **Interrupt-On-Change Control Register (INTCON):** The INTCON register controls how each pin is used for interrupt on change. If enabled, the pin will be setup for option two (2) (see **"Interrupt-On-Change Feature"**, item #2) and compared to the value in DEFVAL. If disabled, the pin will be compared to the previous pin value as in item #1 (see **"Interrupt-On-Change Feature**").
- **Default Value Register (DEFVAL):** The DEFVAL register sets the comparison value to determine when an interrupt has occurred. When a pin has a state opposite that of the corresponding DEFVAL bit, an interrupt will occur.
- **Interrupt Capture Register (INTCAP):** The INTCAP register is a "read-only" register. When an interrupt occurs, the INTCAP register captures the state of the port pins. INTCAP reflects the state of the port at the time of the interrupt and will remain unchanged until the interrupt is cleared by reading either INTCAP or GPIO registers.

> **Note:** The application needs to take into account that the interrupt can only be cleared by reading of the INTCAP or GPIO register after the condition that caused the interrupt no longer exists.

KEY SCAN DESCRIPTION

The keypad is a 4x4 matrix of 16 keys and is connected to the MCP23X08 as shown in Figure 1. The rows of the keypad are connected to the upper nibble of the port, and the columns to the lower nibble. See Figure 2 for the top level flow diagram of the key scan.

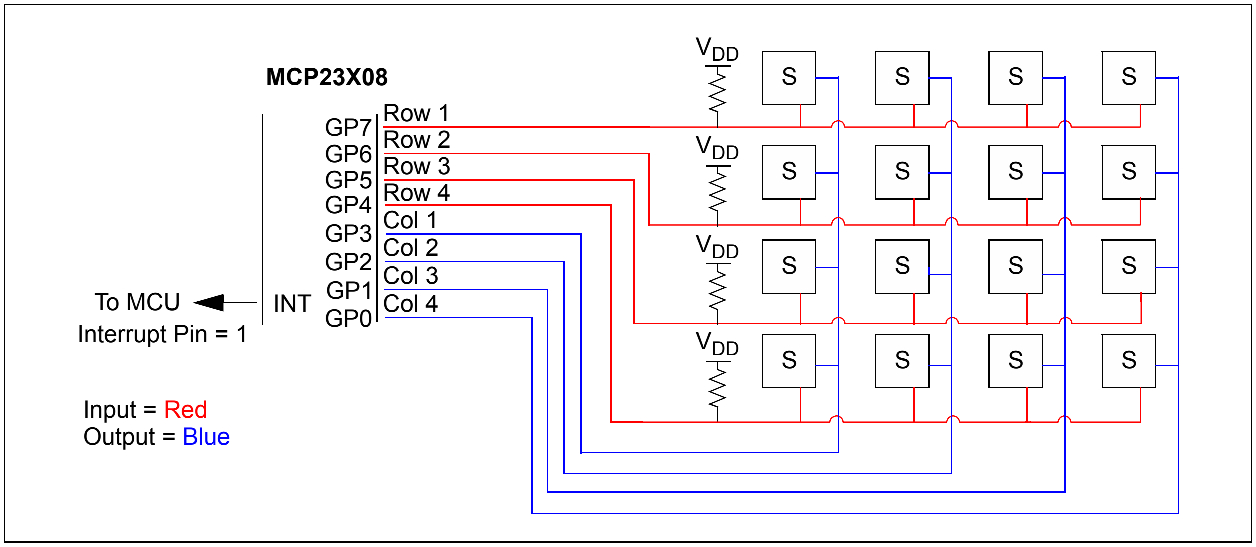

FIGURE 1: *GPIO to Keypad Interface.*

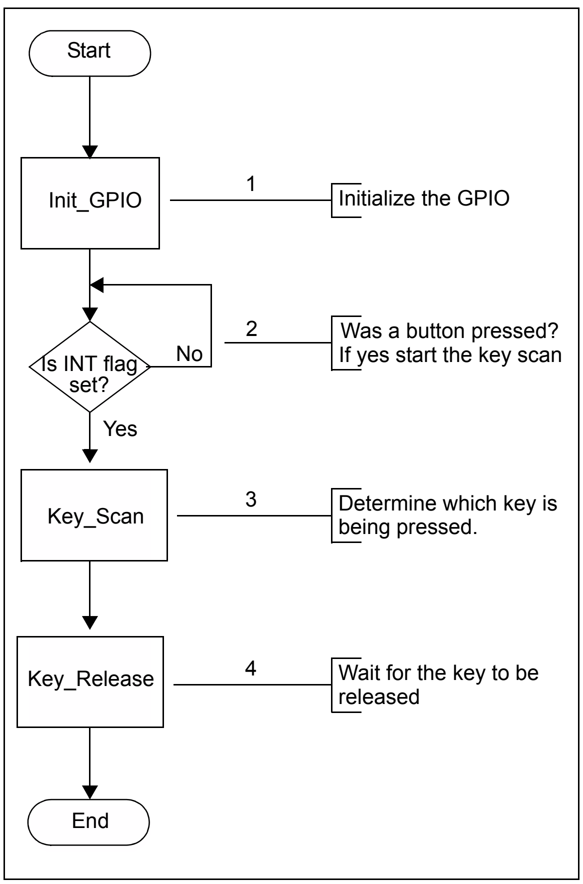

FIGURE 2: *Key Scan Flow Diagram.*

There are four things that must happen for the GPIO expander to read the correct key that was pressed:

1. The MCP23X08 must be initialized at startup of the MCU.
2. The MCU must determine if a button is being pressed. If yes, the MCU starts the key scan.
3. The MCU performs the key scan to determine which key is being pressed.
4. Finally, the MCU waits for the key to be released.

Refer to Figures 3 through 6 and Tables 1 through 4 for the register and pin values for the key scan routine.

TABLE 1: INITIAL SETUP

Register	Row 1	Row 2	Row 3	Row 4	Col 1	Col 2	Col 3	Col 4	
GPIO	1	1	1	1	0	0	0	0	0xF0
INTCAP	—	—	—	—	—	—	—	—	—
IODIR	1	1	1	1	0	0	0	0	0xF0
INTCON	1	1	1	1	0	0	0	0	0xF0
DEFVAL	1	1	1	1	0	0	0	0	0xF0
GPINTEN	1	1	1	1	0	0	0	0	0xF0
GPPU	0	0	0	0	1	1	1	1	0x0F
Pin Values	1	1	1	1	0	0	0	0	0xF0

Interrupt Pin Value	1

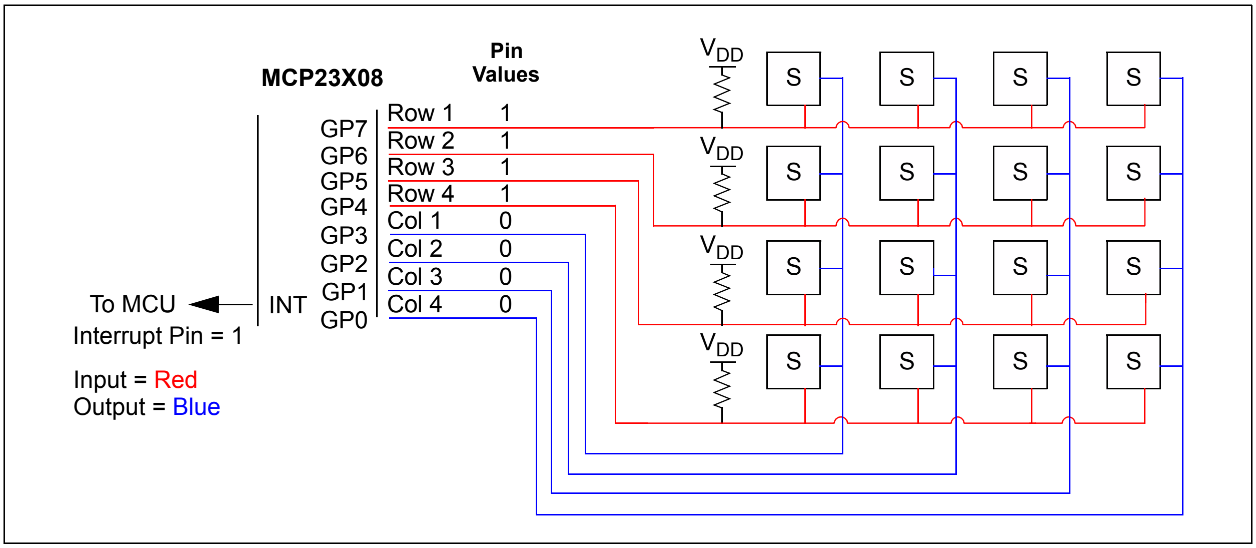

FIGURE 3: Initial Setup.

TABLE 2: BUTTON PRESSED

Register	Row 1	Row 2	Row 3	Row 4	Col 1	Col 2	Col 3	Col 4	
GPIO	1	1	1	**0**	0	0	0	0	0xE0
INTCAP	1	1	1	**0**	0	0	0	0	0xE0
IODIR	1	1	1	1	0	0	0	0	0xF0
INTCON	1	1	1	1	0	0	0	0	0xF0
DEFVAL	1	1	1	1	0	0	0	0	0xF0
GPINTEN	1	1	1	1	0	0	0	0	0xF0
GPPU	0	0	0	0	1	1	1	1	0x0F
Pin Values	1	1	1	**0**	0	0	0	0	0xE0

Interrupt Pin Value	**0**

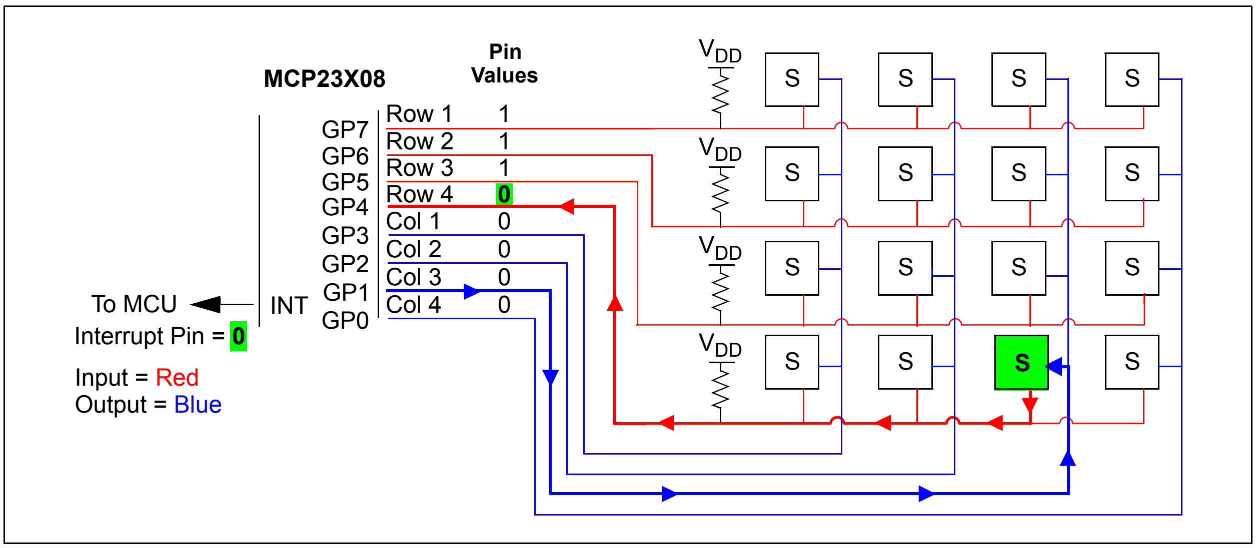

FIGURE 4: Button Pressed (Read Row).

TABLE 3: I/O FLIPPED TO READ COLUMN

Register	Row 1	Row 2	Row 3	Row 4	Col 1	Col 2	Col 3	Col 4	
GPIO	0	0	0	0	1	1	**0**	1	0x0D
INTCAP	1	1	1	0	0	0	0	0	0xE0
IODIR	0	0	0	0	1	1	1	1	0x0F
INTCON	0	0	0	0	1	1	1	1	0x0F
DEFVAL	0	0	0	0	1	1	1	1	0x0F
GPINTEN	0	0	0	0	1	1	1	1	0x0F
GPPU	0	0	0	0	1	1	1	1	0x0F
Pin Values	0	0	0	0	1	1	**0**	1	0x0D

Interrupt Pin Value	**0**

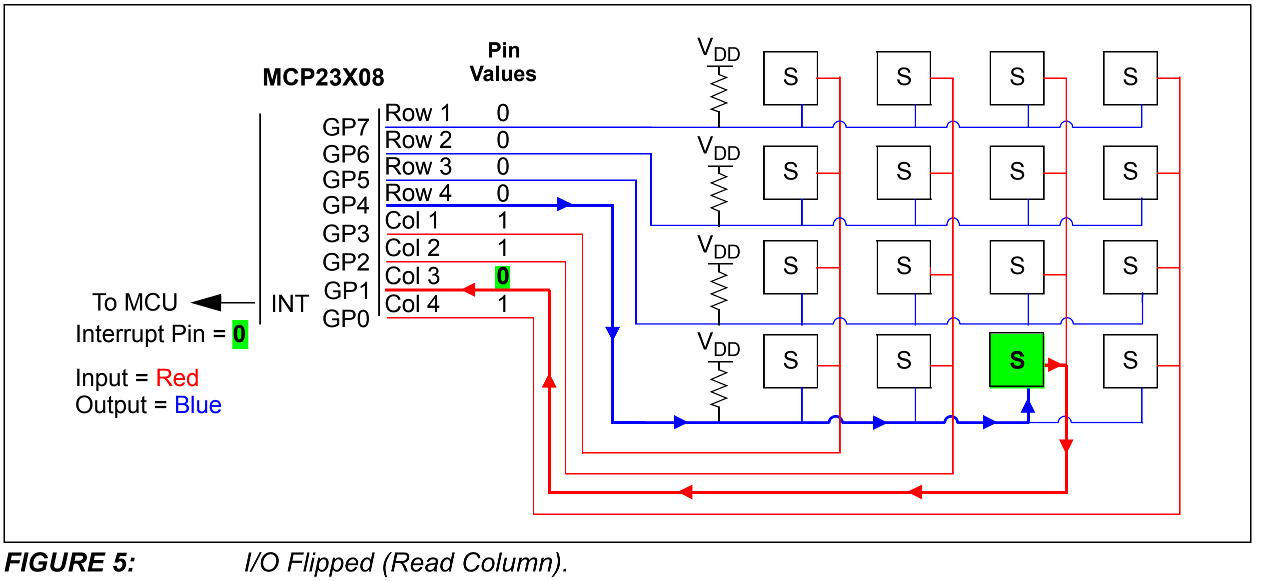

FIGURE 5: *I/O Flipped (Read Column).*

TABLE 4: WAITING FOR KEY RELEASE

Register	Row 1	Row 2	Row 3	Row 4	Col 1	Col 2	Col 3	Col 4	
GPIO	1	1	1	**0**	0	0	0	0	0xE0
INTCAP	1	1	1	0	0	0	0	0	0xE0
IODIR	1	1	1	1	0	0	0	0	0xF0
INTCON	1	1	1	1	0	0	0	0	0xF0
DEFVAL	1	1	1	1	0	0	0	0	0xF0
GPINTEN	1	1	1	1	0	0	0	0	0xF0
GPPU	0	0	0	0	1	1	1	1	0x0F
Pin Values	1	1	1	**0**	0	0	0	0	0xE0

Interrupt Pin Value	**0**

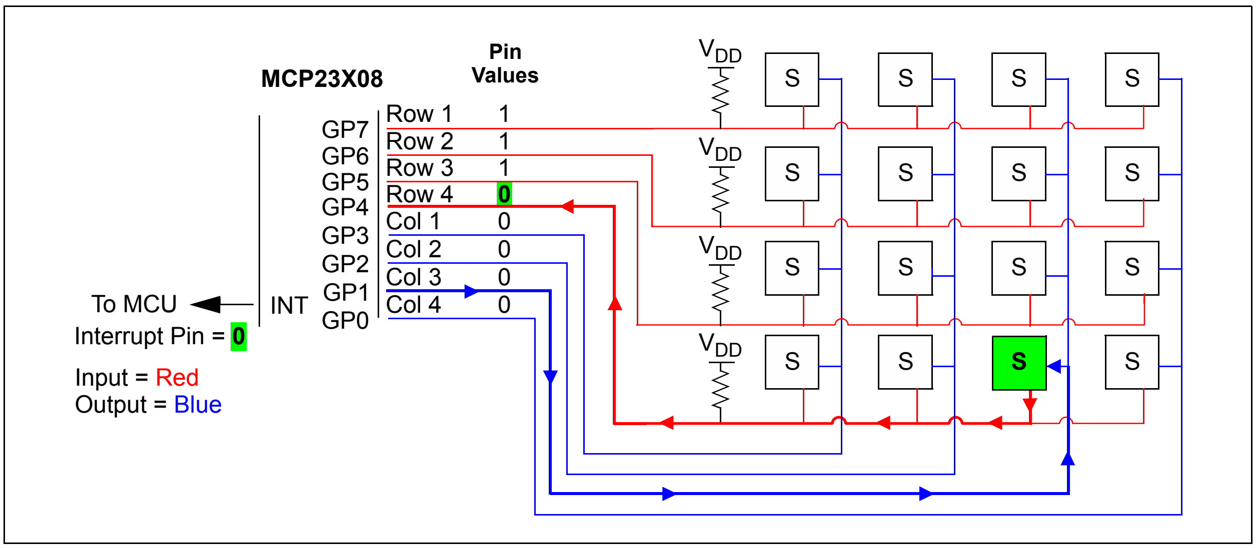

FIGURE 6: *Waiting for Key Release.*

INITIALIZATION OF MCP23X08

Refer to Figure 3 and Table 1 for initial setup and configuration of the registers and the initial state of the port pins.

The MCP23X08 must be preconditioned for the key scan at startup of the MCU. Six registers need to be configured during initialization.

Two of the registers need to be configured only during the initialization and will not require further manipulation:

1. **GPIO Pull-Up Resistor Register (GPPU):** The GPPU register controls the 100 kΩ internal pull-up resistors. They are disabled for the rows and enabled for the columns. External resistors are used for the rows to give the designer control over how fast the line will “pull up” when the button is released. This effectively controls the minimum time required between the release of a button and pressing of another.
2. **General Purpose I/O Port Register (GPIO):** The rows and columns are set to a low state through the GPIO register. This is done so that when configured as outputs via the IODIR register the pins will be outputting a low.

The other four registers will have to be initially set up, and will also be manipulated during the key scan, those registers include:

3. **I/O Direction Register (IODIR):** Initially the row pins are set as inputs and the column pins as outputs via the IODIR register. This is done so that the row pins on the MCP23X08 are now inputs with a high state because of the 2.2 kΩ external pull-up resistors, and the columns are now outputting a low.
   Pressing a button will close a row/column circuit causing an interrupt to occur. (See Figure 4)
4. **Interrupt Control Register (INTCON):** The INTCON register which controls the interrupt-on-change option for each pin, is set for the rows and cleared for the columns. This will set the device to interrupt if a value opposite that of DEFVAL is detected on a row pin.
5. **Default Compare Register for Interrupt-on-change (DEFVAL):** The default value for interrupt on change is then set in DEFVAL with the rows set and the columns cleared. With DEFVAL configured this way, if a row pin changes state from high to low an interrupt will occur.
6. **Interrupt-on-Change Control Register (GPINTEN):** The last step is to enable the rows for interrupt-on-change in the GPINTEN register. **Important:** the GPINTEN register must be set up last so that setting of the other registers does not inadvertently cause an interrupt.

INITIATE THE KEY SCAN

There are six things that must happen to initiate the key scan:

1. Initially the row pins are configured as inputs through the IODIR register during initialization.
2. The row pins are high because of the 2.2 kΩ external pull-up resistors shown in Figure 1.
3. A single key is pressed by the user.
4. Once a key is pressed, a row/column circuit will be closed and the row (input) will pull low due to the column (output) being driven low. (See Figure 4).
5. The row pin in which the key was pressed is now low, which is opposite of what was set up in the DEFVAL register during initialization.
6. An interrupt will now occur signaling the MCU that a key has been pressed.

DETERMINE WHICH KEY IS PRESSED

When a key depression has been detected and an interrupt generated, the MCU will need to determine which of the keys is being pressed.

See Figure 7 for flow diagram of the key scan and Figures 3 through 6 and Tables 1 through 4 for both pin and register values through the key scan.

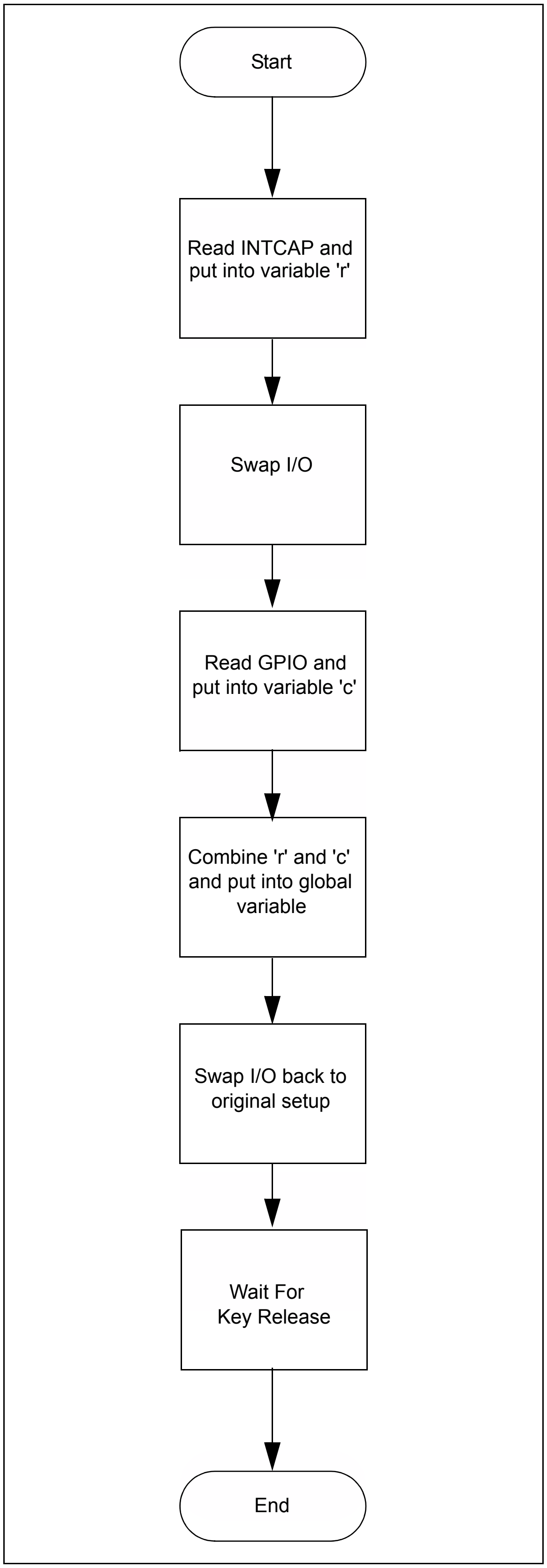

FIGURE 7: *Key Press Flow Diagram.*

KEY SCAN

Because the keys are in a matrix, the MCU must determine which row and column matches the key press. The MCU will then know which key was pressed.

There are six steps to servicing a key press:

1. **Determine Rows:**

   Refer to Figure 4 and Table 2 for the pin and register values for this step. To determine which row the key being pressed is in, the INTCAP register which contains the port value at the time of the interrupt must first be read and put into variable ‘r’. The upper nibble contains the state of the rows with 3 bits high and one low. The highs exist from the 2.2 kΩ external pull-up resistors, and the low comes from the low output of the column pin through the key being pressed.

2. **Reconfigure registers to read columns:**

   Refer to Figure 5 and Table 3. To obtain the column value, four of the registers that were initially setup will now have their values switched:

   - **IODIR:** The IODIR register has its value switched so that the rows are now outputs and the columns inputs.
   - **INTCON/DEFVAL/GPINTEN:** These registers which control the interrupt-on-change feature also all have their values switched. The interrupt-on-change feature is not used for reading of the columns. But if left in its original setup, when the IODIR register has it’s value changed from 0xF0 to 0x0F and the GPIO register is then read to obtain the column value (explained in next section) the interrupt will be cleared. Once the IODIR register is flipped back to its original setup an interrupt will occur once more. To avoid this, the interrupt-on-change register values are also switched.

3. **Determine Columns:**

   Refer to Figure 5 and Table 3 for the pin and register values. Since the key is still being pressed, reading of the GPIO register will indicate the column of the key press. Once the GPIO register is read, it is put into the variable ‘c’. The 8-bit value in ‘c’ now reflects the state of the port pins on the MCP23X08. The lower nibble contains the state of the column which has three bits high (internal pull-up resistors) and one bit low. The column with the key being pressed is low because of the now low output of the row pin through the key being pressed.

4. **Calculate which key was pressed:**

   Next, the separate row and column values need to be combined into one variable. Variable ‘r’ and ‘c’ are ORed together so that the upper nibble represents the row and the lower nibble the column of an 8-bit value. The resulting value represents the key that was pressed.

5. **Swap I/O Back to Original Setup:**

   Refer to Figure 6 and Table 4. The last step before checking if the key has been released, is to reconfigure the MCP23X08 registers back to their original setup during initialization. They just need to be set appropriately and in the same order as during the initialization. Take care when setting the GPINTEN register back to its original setting that it be set last or an unnecessary interrupt will occur.

6. **Wait for key release:**

   Refer to Figure 8 for the flowchart of checking for key release. Reading of either the GPIO or INTCAP register will clear the interrupt. However, if the interrupt condition still exists (i.e. the button is still being pressed), the interrupt will not clear. This can be used to determine when the key has been released. Because even a very fast key press lasts more than 20 ms, there is no need to constantly be checking for the key to be released. Instead, the MCU might wait 5 ms, then read INTCAP or GPIO to try to clear the interrupt. If the key has been released, the interrupt clears and the MCU goes and performs other tasks. If the key is still being pressed, the MCU waits another 5 ms and tries to clear the interrupt again. This will continue until the interrupt is cleared.

SUMMARY

GPIO Expanders provide easy I/O expansion using standard serial protocols such as I^2C and SPI. They are especially useful for applications that require remote I/O or if pin count is limited on the MCU.

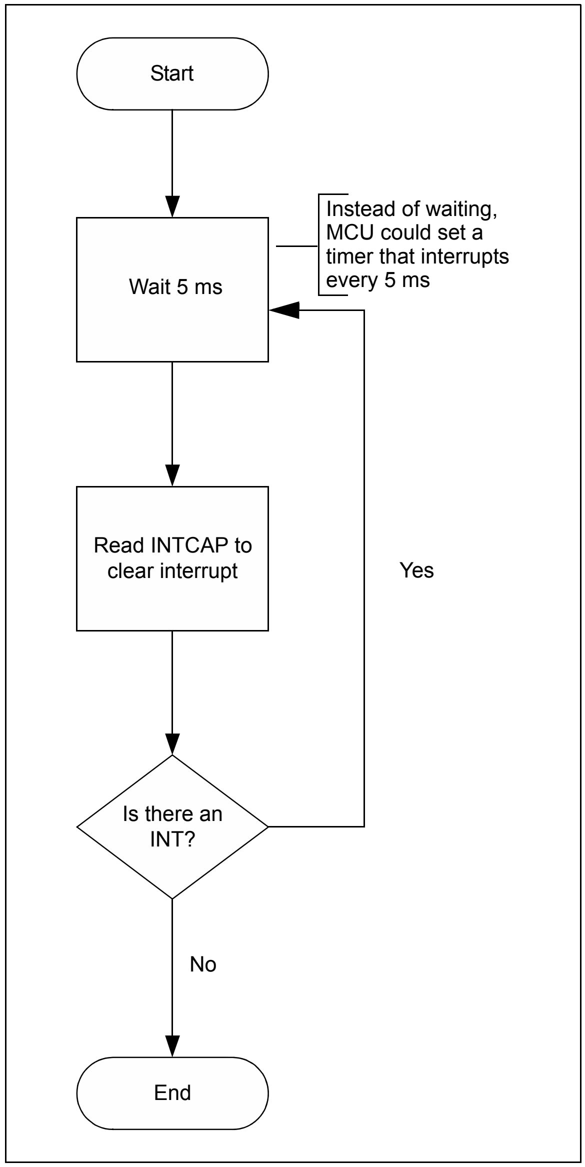

FIGURE 8: *Key Release Flow Diagram.*

WORLDWIDE SALES AND SERVICE

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://support.microchip.com
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

Kokomo
Kokomo, IN
Tel: 765-864-8360
Fax: 765-864-8387

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

Toronto
Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Habour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Fuzhou
Tel: 86-591-8750-3506
Fax: 86-591-8750-3521

China - Hong Kong SAR
Tel: 852-2401-1200
Fax: 852-2401-3431

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

China - Shunde
Tel: 86-757-2839-5507
Fax: 86-757-2839-5571

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7250
Fax: 86-29-8833-7256

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-4182-8400
Fax: 91-80-4182-8422

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

Japan - Yokohama
Tel: 81-45-471- 6166
Fax: 81-45-471-6122

Korea - Gumi
Tel: 82-54-473-4301
Fax: 82-54-473-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Penang
Tel: 60-4-646-8870
Fax: 60-4-646-5086

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-572-9526
Fax: 886-3-572-6459

Taiwan - Kaohsiung
Tel: 886-7-536-4818
Fax: 886-7-536-4803

Taiwan - Taipei
Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

12/08/06