

Peculiarities in Programming of the Flash Memory of the PIC18FXX2 microcontroller

Nina Bencheva

***Peculiarities in Programming of PIC18FXX2 Flash memory:** The Read and Write operations of the Flash program memory of the microcontroller PIC18FXX2 has considerable differences in comparison to those of PIC16F87X family. In the paper a comparative analysis of Flash memory organization is made. Some peculiarities of the write operations to PIC18FXX2 Flash memory are presented.*

Keywords: microcontroller, memory, Flash, write, read, holding registers

INTRODUCTION

The Read and Write operations of the Flash program memory of the microcontroller PIC18FXX2 has considerable differences in comparison to those of PIC16F87X family. The Flash memory organization of the PIC16F87X is 8Kx14bit words, as with PIC18FXX2 is 2Mx16 bit words. Different hardware and software tools are used to access the memory of the two families. In the paper a comparative analysis of memory organization is presented which may need to be observed when migrating an application from a PIC16F87X device to a PIC18FXX2 device. Some peculiarities of the write and erase operations to PIC18FXX2 Flash memory are presented.

1. COMPARATIVE ANALYSIS OF FLASH PROGRAM MEMORY ORGANIZATION

The program memory of PIC16F87X allows word reads and writes. A byte or word write automatically erases the location and writes the new data (erase before write). When interfacing to the program memory block, the EEDATH:EEDATA registers form a 2 byte word which holds the 14-bit data for read/write, and the EEADRH:EEADR registers form a 2 byte word which holds the 13-bit address of the location being accessed. A given sequence for loading EECON2 register must be followed to initiate the write cycle. PIC16F87X can have up to 8K words of Flash program memory with an address range from 0000h to 3FFFh.

Byte table access to Flash of PIC18FXX2 is provided, which ensures fast transfer of operands from Flash to RAM. Write can be performed maximum in blocks of 8 bytes. A new operation for erasing of blocks of 64 bytes is implemented. For write and read operations other data and address registers are used. Only the sequence for loading EECON2 register is the same.

Access to the program memory is performed in two cases – when the program is executed, i.e. read of instructions which are two bytes long, and when reading preliminary written data, which are one byte long. The access is organized in a different way and the program memory is addressed by different tools. The program counter addresses the memory in order to fetch the instructions. The least significant bit of the program counter is fixed in “0” which provides the correct addressing of the instructions. They are always stored starting from even address. A read of data from PIC18FXX2 program memory is executed one byte at a time. This is so called table read.

A write in the Flash memory in the process of execution of an application, so called table write, has two stages – short write and long write. At the stage of the short write the preparation for the real, physical (long) write is implemented. The write to program memory is executed in blocks of 8 bytes at a time.

Program memory is erased in blocks of 64 bytes at a time. The additional FREE bit, implemented in EECON1 register, when set, allows a program memory erase operation that is initiated on the next write command. When FREE is clear, only writes are enabled.

1.1 Table read/write (for PIC18FXX2)

In order to read and write program memory, there are two operations that allow the

processor to move bytes between the program memory space and the data RAM – Table Read (TBLRD) and Table Write (TBLWT).

The program memory space is 16-bits wide, while the data RAM space is 8-bits wide. Table Reads and Table Writes move data between these two memory spaces through an 8-bit register TABLAT. The bytes within the program memory are addressed by Table Pointer (TBLPTR). The TBLPTR is comprised of three SFR registers: Table Pointer Upper Byte, Table Pointer High Byte and Table Pointer Low Byte (TBLPTRU:TBLPTRH:TBLPTRL). These three registers form a 22-bit wide pointer. The low order 21 bits allow the device to address up to 2 Mbytes of program memory space. The 22-nd bit allows access to the Device ID, the User ID and the Configuration bits. Fig. 1 shows the operation of a Table Read.

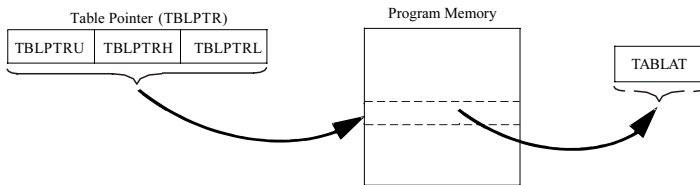


Fig. 1. Table read operation

Table Write operations (Fig. 2) store data from the data memory space into holding registers in program memory. A table block containing data is not required to be word aligned. If a Table Write is being used to write an executable code into the program memory, program instructions will need to be word aligned.

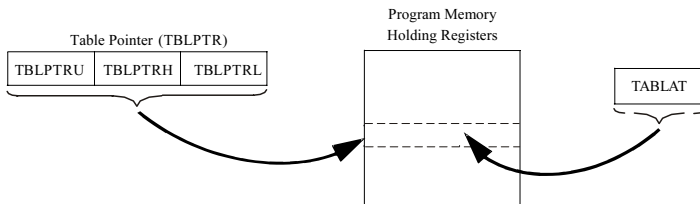


Fig. 2. Table write operation

The TBLRD and TBLWT instructions can update the TBLPTR in one of four ways based on the table operation. These operations are shown in Table 1. They only affect the low order 21 bits.

Table 1

Instruction	Operation on Table Pointer
TBLRD* TBLWT*	TBLPTR is not modified
TBLRD*+ TBLWT*+	TBLPTR is incremented after the read/write
TBLRD*- TBLWT*-	TBLPTR is decremented after the read/write
TBLRD+* TBLWT+*	TBLPTR is incremented before the read/write

1.2 Read/write FLASH program memory (for PIC18FXX2)

Executing TBLRD places the byte pointed by TBLPTR into TABLAT. In addition, TBLPTR can be modified automatically for the next Table Read operation. Fig. 3 shows the interface between the internal program memory and the TABLAT.

Table Writes are used internally to load the holding registers needed to program the Flash memory. There are 8 holding registers used by the Table Writes for programming. Since the TABLAT is only a single byte, the TBLWT instruction has to be executed 8 times for each programming operation. All of the Table Write operations will essentially be short writes, because only the holding registers are written. At the end of updating 8 registers, the EECON1 register must be written to, to start the programming operation with a long write. The long write is necessary for programming the Flash. Fig. 4 shows table writes to the Flash memory. As in the case of PIC16F87X a given sequence for loading EECON2 register must be followed to initiate the write cycle.

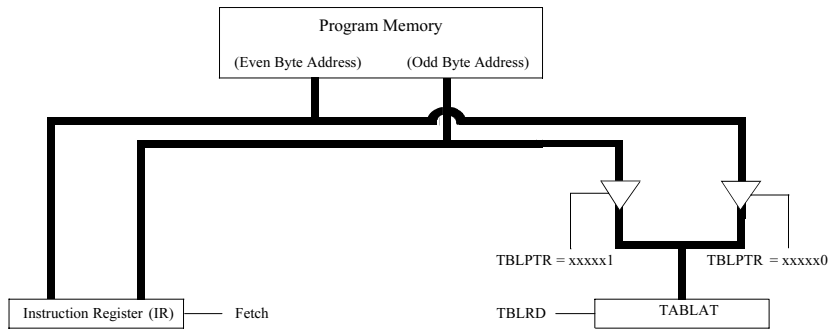


Fig. 3. Read from Flash program memory

Although there are 8 TABLAT registers for the short write, they are not accessible individually. They are internal and are used solely for the purpose of the short write using the TBLWT instruction [1].

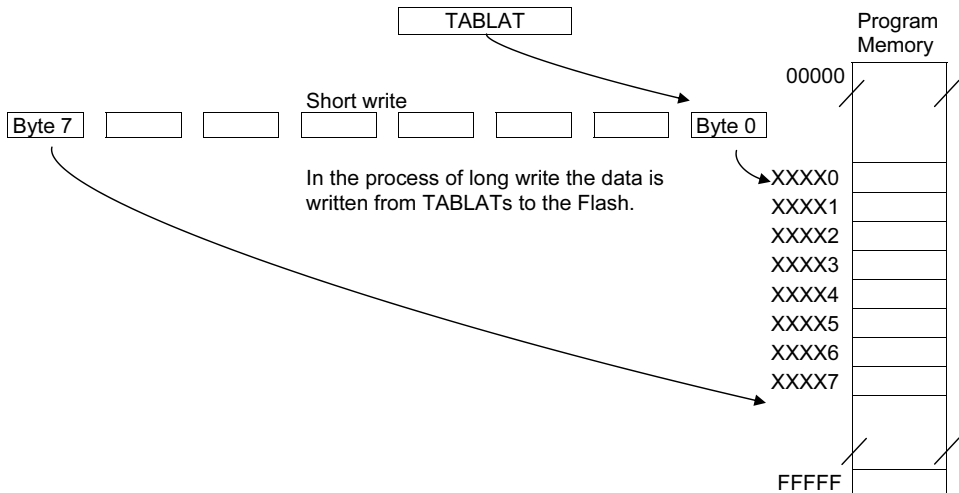
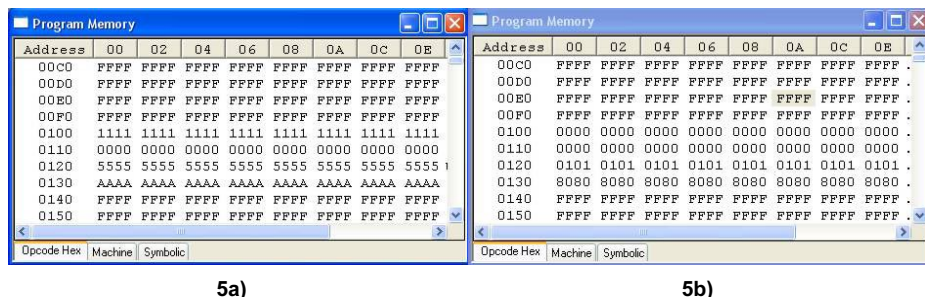


Fig. 4. Table writes to the Flash program memory

2. PECULIARITIES OF THE WRITE IN THE FLASH MEMORY OF THE PIC18FXX2 MICROCONTROLLER

2.1. Write in a memory area not erased

All the bits are set in the erased memory. If the memory is not erased, write can be performed but only where the corresponding bits are "1". On the bits in "0" a write cannot be done. The result of the execution of a program for a write of 64 bytes from a begin address 0x100 in a Flash program memory not erased is shown in Fig. 5b. The data in the memory before the execution of the write program are displayed in Fig. 5a. The addresses, the preliminary data in the memory, the write data and the final data after the execution of the write program in a Flash memory not erased are presented in the Table 2.



5a) 5b)
Fig.5. Screen shot before and after a write in a program memory not erased

Table 2.

Begin address, h	End address, h	Preliminary data, h	Write data, h	Final data, h
0100	010F	11	00	00
0110	011F	00	FF	00
0120	012F	55	01	01
0130	013F	AA	80	80

2.2. Write not in the boundaries of an 8 byte block

It is recommended that the write has to start from a begin address in which the least address lines A2, A1, A0 are "0", i.e. the begin addresses are divisible by 8.

If the begin address is not divisible by 8 and the number of the bytes, written in the holding registers, is less than the subtraction value of the end address of the block and the begin address, the data are written in the same block. If the begin address is not divisible by 8 and the number of the bytes, written in the holding registers, is greater than the subtraction value of the end address of the block and the begin address, the data are written in the next 8 byte block. In this case the first byte is written into the next address divisible by 8 and the remaining bytes – at the bottom of the block. Table 3 demonstrates this peculiarity of the write.

Table 3

Begin address Flash, h	Number of bytes	Long write addresses, h	Number of bytes	Long write addresses, h
101	7	101 ÷ 107	7	101 ÷ 107
102	6	102 ÷ 107	7	108, 10A ÷ 10F
103	5	103 ÷ 107	6	108, 10B ÷ 10F
104	4	104 ÷ 107	5	108, 10C ÷ 10F
105	3	105 ÷ 107	4	108, 10D ÷ 10F
106	2	106 ÷ 107	3	108, 10E ÷ 10F
107	1	107	2	108, 10F

With short write it is possible to write separate byte/bytes in the holding registers modifying the TBLPTR register. After the long write the holding registers are erased. If all 8 holding registers are not loaded, the long write burns only the loaded bytes. Between the sequential short writes, a read can be performed without any influence on the long write. Data are read from the Flash memory, not from the holding registers. In TABLAT the data of the last operation read/write stay.

2.3. Erase not in the boundaries of a 64 byte block

It is recommended that the write has to start from a begin address in which the least address lines A5, A4, A3, A2, A1, A0 are "0", i.e. the begin addresses are divisible by 64.

Fixed 64 byte blocks are erased, i.e. the memory is divided into sequential pages with begin and end addresses: 00 to 3F; 40 – 7F; 80 – BF etc. If the address is not divisible by 64, the block comprising the address is erased. For example, if the address 50 is written in TBLPTR, the block with addresses 40 to 7f will be erased. This peculiarity in the erase organization of PIC18FXX2 Flash program memory requires careful address space allocation – the area of the program code and the areas where the data will be written by the program execution.

CONCLUSIONS

The access to the PIC18FXX2 Flash program memory is organized in different way and it is addressed by different tools. Byte table access to Flash is provided, which ensures fast transfer of operands from Flash to RAM. A write in the Flash memory in the process of execution of an application has two stages – short write and long write. Write can be performed maximum in blocks of 8 bytes. A new operation for erasing of blocks of 64 bytes is implemented.

The possibility for write in a memory not erased can cause write/read of false data. If due to a logical error (undetected by the programmer) the erasing program is not working properly and a write of data (for example data obtained from measuring) in the same area of the Flash memory is executed, the measurement data will be distorted.

The PIC18FXX2 memory features are significantly improved compared to those of PIC16F87X. This enlarges the fields of PIC18FXX2 application.

REFERENCES

- [1] Mazidi A. M., R. D. McKinlay, D. Causey, PIC Microcontroller and Embedded Systems. Using Assembly and C for PIC18, Pearson Prentice Hall, 2008.
- [2] Microchip Technology Inc., *PICmicro™ Mid-Range MCU Family Reference Manual*. DS30292A, 1998.
- [3] Microchip Technology Inc., *PIC18FXX2 Data Sheet*, DS39564A, 2001.
- [4] Peatman J. B., *Embedded Design with the PIC18F452 Microcontroller*, Pearson Education Inc, ISBN 0-13-046213-6, 2003.

Author:

Nina Bencheva, PhD, Telecommunications Dept., Rousse University, tel.: +359 82 888677, e-mail: nbencheva@ecs.ru.acad.bg

The paper has been reviewed.