SIEMENS

Microcontrollers

ApNote AP0824

Program Memory Expansion by using the Bankswitching Capability of the C5xx/80C5xx Family

This application note describes the proceeding for external program memory expansion from 64 KByte to 512 KByte by using the bankswitching capability of the C5xx/80C5xx Family

Author: Klaus Scheibert / HL MC AT Microcontroller Application Support

AP0824 ApNote - Revision History		
Actual Revision : Rel.01	Previous Revison: none (Original Version)	
Page of actual Rel.	Page of prev. Rel.	Subjects changes since last release)
		Original Version

1 Introduction

In some applications, using the 80C5xx/C5xx architecture, there is a need for more than 64 kByte external program memory space. This request becomes more and more important due to the increased usage of the high level language C51.

The following apnote proposes a solution with a standard device 74HC257 used as external glue logic to solve the 64 kByte boundary problem for the external program memory.

2 Memory Organization

The memory partitioning of the C5xx/80C5xx microcontrollers is typically a Harvard architecture where program and data areas are held in seperate memory areas. The program and data memory areas use the same physical address range from 0000H - FFFFH and a 8 bit instruction code/data format.

2.1 External Memory Organization in ´Normal Configuration´

When using external memory resources, the external program and data memories can be expanded up to 64 kByte each due to the external multiplexed memory bus interface of the C5xx/80C5xx architecture. Locations 0000H - 0100H in the program memory are used for the hard-wired interrupt vector addresses by the corresponding interrupt service routines. The external program memory is automatically selected if the pin EA# (**E**xternal **A**ccess) is held at logic low level. The external data memory is selected by MOVX instructions from the C5xx/80C5xx 8-bit instruction set. The following figure shows this ´normal configuration´ for external memory resources with pin EA# set to ´0´.

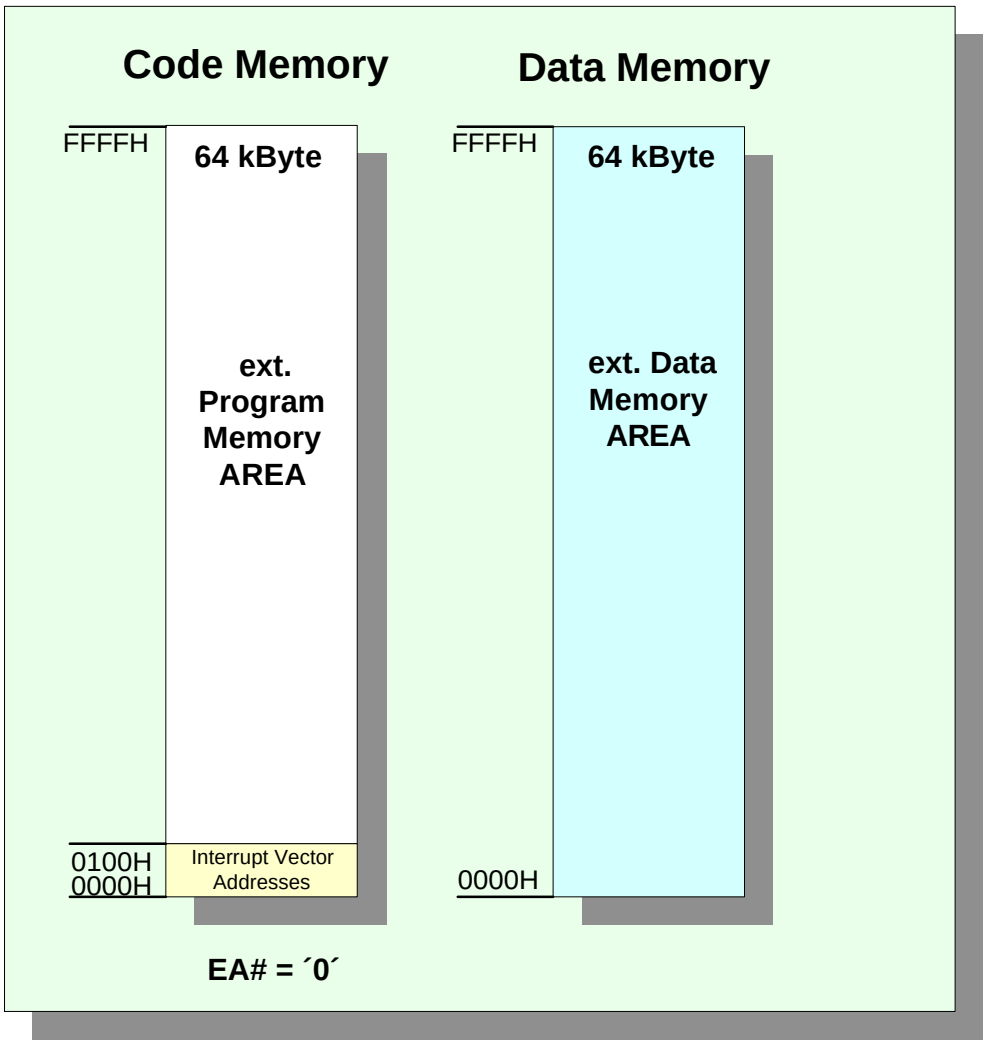

Figure 1:
External Memory Organization in ´Normal Configuration´

2.2 External Memory Organization in ´Expanded Configuration´

In this application example, the external program memory is expanded to 512 kByte in the ´Expanded Configuration´, the data memory remains in the ´Normal Configuration´. The program memory is split up into sixteen 32 kByte banks (named Bank 0-15); each addressed via address lines A0 - A14 in combination with bank select lines A15 - A18. The C5xx/80C5xx external bus interface supports A0 - A15 address lines at 8-bit ports P0 and P2. In the ´Expanded Configuration´only A0 - A14 address lines of the microcontrollers external bus interface are used directly to interface external program memory. Additionally, four normal standard I/O pins are used for interfacing the bank select lines A15 - A18 to the external program memory.

The P2.7/A15 address line of the microcontroller is used in connection with an external glue logic to select between a fixed bank 0 in the address range of 0000H - 7FFFH and any preselected upper bank 1-15 (preselected by the four standard I/O pins for program memory address lines A15 - A18). If address line P2.7/A15 of the microcontroller is not selected (logic ´0´ level), always the lowest 32 kByte bank 0 of external program memory is accessed via external glue logic. With activation of the microcontrollers address line P2.7/A15 (logic ´1´level), each of the remaining fifteen 32 kByte banks 1-15 can be accessed in connection with the four standard I/O pins. The external glue logic is used to synchronize the external program memory access timing between the directly connected microcontrollers address lines A0 - A14 and the four indirectly connected standard I/O pins for the bank selection of bank 1-15 (A15 - A18).

If interrupts are used, always the hard-wired interrupt vector addresses from 0000H - 0100H in the lowest 32 kByte bank 0 becomes active and therefore automatically deselects addressline P2.7/A15. It doesn´t exist any code size overlapping in the ´Expanded Configuration´ within the complete external program memory area.

The following figure gives an overview of the ´Expanded Configuration´.

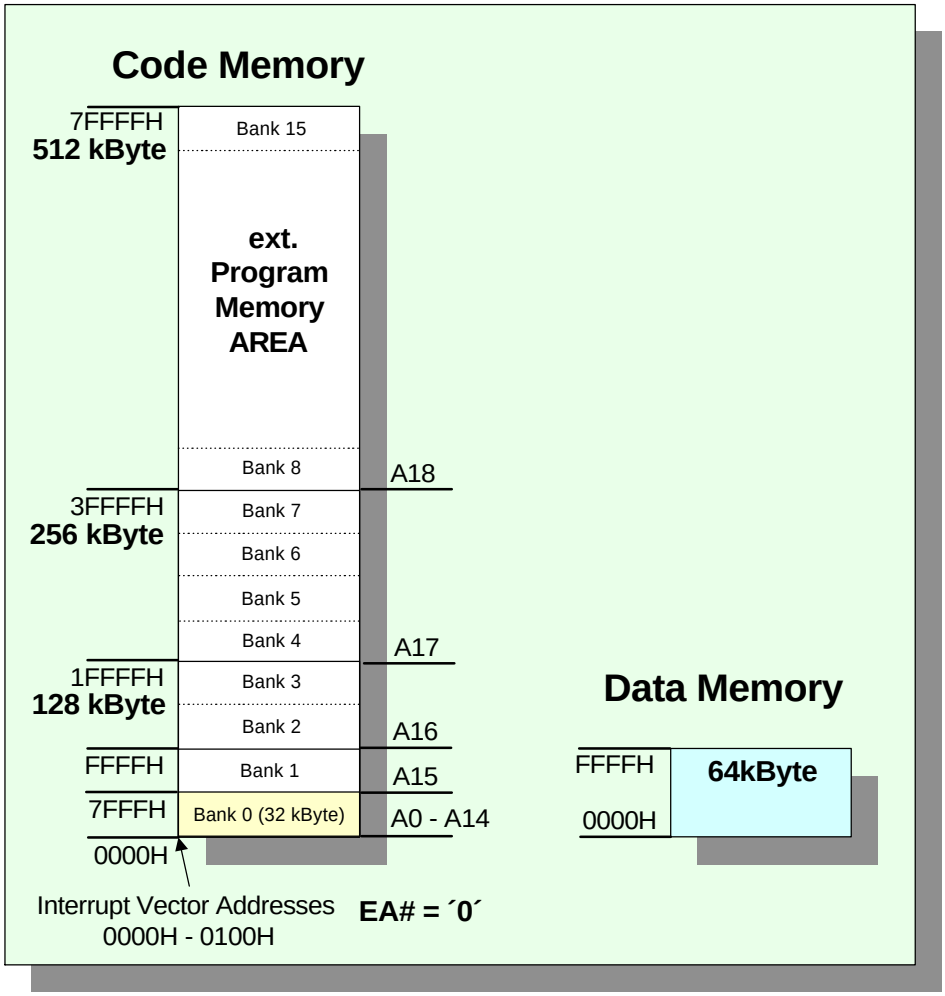

Figure 2:
External Memory Organization in ´Expanded Configuration´

3 Hardware Description

The following schematic shows an example of how to connect an external 512 kByte program memory and a 64 kByte data memory to the C5xx/80C5xx microcontroller family. For the program memory, a standard EPROM/EEPROM/FLASH can be used. A static RAM serves as external data memory. The x573 works as address latch for the 8 bit multiplexed port P0. The quad 2-line to 1-line data multiplexer 74HC257 (standard device) is used to select between the fixed bank 0 (0000H - 7FFFH) and the four standard I/O pins PX.0 - PX.3 for bank 0-15 selection. The microcontrollers P2.7/A15 address line is used for switching between bank 0 and bank 1-15 according to the AC characteristics for external program memory. Pin EA# is tied low, so all program memory accesses are performed from external program memory. Port 0 is the multiplexed address/data bus while port pins P2.0 - P2.6 emits the high address lines A8 - A14. P2.7/A15 is connected to the SELECT input line of the 74HC257 for selecting the upper fifteen 32 kByte memory bank 1-15 portions. Therefore, in this configuration port 0 and port 2 must not be used as general-purpose I/O ports.

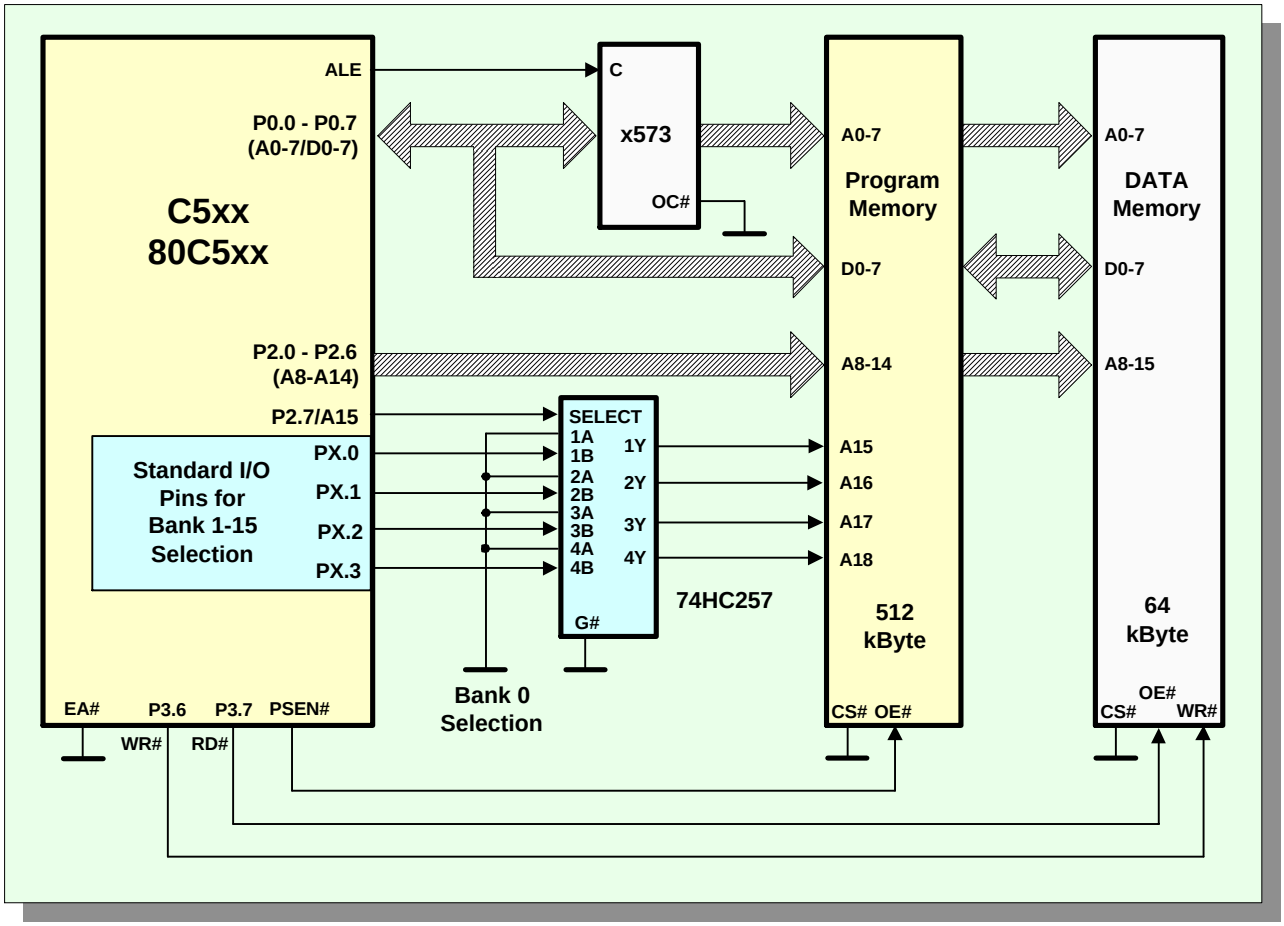

Figure 3:
Schematic for External Memory Bus Interface

The following table shows the logic table of the 74HC257.

Table 1:
74HC257 - Logic Table

G#	Select	xA	xB	xY
0	0	0	X	0
0	1	0	0	0
0	1	0	1	1

If address line P2.7/A15 shows a logic ´0´, the multiplexer 74HC257 always selects bank 0 (A15 - A18 = ´0´). Every software jump to a interrupt service routine within the hard-wired interrupt vector addresses at 0000H - 0100H uses this mechanism by an automatic deselection of P2.7/A15.
If addressline P2.7/A15 shows a logic ´1´, the multiplexer is switched transparent to the corresponding four standard I/O pins PX.0 - PX.3. In this way, every upper 32 kByte program memory bank 1-15 can be accessed.
The time delay $t_{257DELAY}$ between changing the SELECT input line of the 74HC257 and updating the corresponding Y output lines is approximately 10 ns.

$$\mathbf{t_{257DELAY} \approx 10\ ns}$$

Note: If there are used slower multiplexer types (e.g.not HC-Types), $t_{257DELAY}$ has to be adapted to the corresponding data sheet value.

4 Software Proceeding for ´Expanded Configuration´

The program memory software has to take care about the hardware situation in the ´Expanded Configuration´. The following list gives a rough overview about the recommended software items:

- The selection of a upper bank 1-15 (see table 2) always has to be executed in the lowest 32 kByte bank 0 (located at 0000H - 7FFFH) by setting up the corresponding standard I/O pins PX.0 - PX.3.

Table 2:
Bank Selection via Standard I/O Pins

PX.3	PX.2	PX.1	PX.0	selected bank [Dec]	physical address range [Hex]
0	0	0	0	0	**not allowed**
0	0	0	1	1	8000 - FFFF
0	0	1	0	2	10000 - 17FFF
0	0	1	1	3	18000 - 1FFFF
0	1	0	0	4	20000 - 27FFF
0	1	0	1	5	28000 - 2FFFF
0	1	1	0	6	30000 - 37FFF
0	1	1	1	7	38000 - 3FFFF
1	0	0	0	8	40000 - 47FFF
1	0	0	1	9	48000 - 4FFFF
1	0	1	0	10	50000 - 57FFF
1	0	1	1	11	58000 - 5FFFF
1	1	0	0	12	60000 - 67FFF
1	1	0	1	13	68000 - 6FFFF
1	1	1	0	14	70000 - 77FFF
1	1	1	1	15	78000 - 7FFFF

- The bank switching algorithm itself is automatically performed by a software jump to the microcontrollers address range 8000H - FFFFH. This software jump activates P2.7/A15 address line of the microcontroller and therefore accesses one of the fifteen upper 32 kByte memory banks 1-15 via the external glue logic.
- Every software jump below the microcontrollers address range 8000H - FFFFH accesses physically to the lowest 32 kByte memory bank 0 from 0000H - 7FFFH via deselection of P2.7/A15 - SELECT address line. Therefore, the combination PX.0 - PX.3 = ´0´ is not allowed. A software jump to the microcontrollers address range 8000H - FFFFH in this case would result in an access to the physical address range between 0000H - 7FFFH in bank 0.

- Direct bank switching within banks 1-15 is not allowed because the multiplexer 74HC257 in this case directly connects the standard I/O pins PX.0 - PX.3 with the program memory address lines A15 - A18. Every change of a PX.0 - PX.3 pin then causes a direct change of the corresponding A15 - A18 program memory input address line, which is not allowed according to the AC characteristics timings of the microcontroller.
- Interrupt service routines should be located in the lowest 32 kByte memory bank 0.
- The RETI (Return from Interrupt) instruction at the end of a interrupt service routine automatically branches back to the upper bank 1-15, where the last source code had been executed before.
- Software (sub-)routines shouldn´t be bigger than 32 kByte each.
- Software (sub-)routines should be completely located within one 32 kByte memory bank.
- Branches to fixed data tables stored in an upper program memory bank via MOVC access should be executed only within the same upper bank or from the lower bank 0.
- LCALL-routines have to regard the ´Expanded Configuration´. Therefore direct LCALL branches within two different upper banks are not allowed.

5 Timing Considerations concerning the External Program Memory Access

Due to the fixed AC characteristics for the external multiplexed bus interface of the C5xx/80C5xx family, a simple program memory expansion by using general purpose I/O pins for direct switching of a 32 kByte bank is not possible. The switching time of a general purpose I/O pin is not synchronized to the address output time for an external program memory access. Therefore the bus interface address output line P2.7/A15 is used to synchronize the preprogrammed I/O ports PX.0 - PX.3 for a upper 32 kByte memory bank 1-15 access. In the following, some considerations concerning the external program memory AC characteristic timings can be found.

The condition for the external program memory parameter t_{ACC} (address to output delay) is:

$$\mathbf{t_{ACC} \leq t_{AVIV}}$$

where the t_{AVIV} (address to valid instr. in) parameter can be found in the AC characteristics of the corresponding C5xx/80C5xx data sheets.

For program memory selection, the time $t_{257DELAY}$ (see chapter 3) has to be regarded. The new ´address to valid instr. in´ calculation for the 74HC257 configuration is:

$$\mathbf{t_{AVIV257} = t_{AVIV} - t_{257DELAY} = t_{AVIV} - 10\ ns}$$

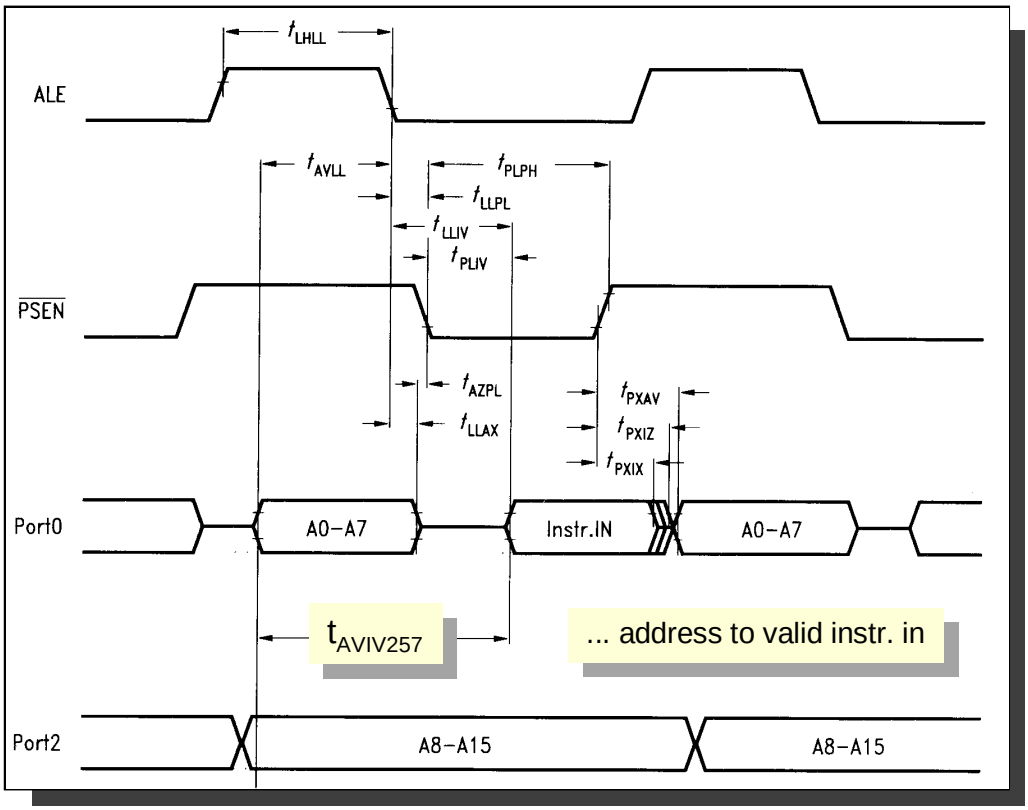

Figure 4:
Program Memory Read Cycle

In the following table, some data sheet values for the program memory parameter $t_{AVIV257}$ (calculated with the formula $t_{AVIV257} = t_{AVIV} - t_{257DELAY} = t_{AVIV} - 10$ ns) can be found.

Table 3:
´Address to Valid Instruction In´ $t_{AVIV257}$ in ´Expanded Configuration´

f_{OSC} [MHz]	$t_{AVIV257}$ [ns]
40	55
24	138
18	170
12	292
6	708

6 Conclusion

The proposed solution shows an efficent way to overcome the 64 kByte boundary problem of the C5xx/80C5xx architecture. With low hardware efforts and only small software restrictions, the bankswitching ability for opening the code size limitation has become feasible. This application example can be easily adapted to code size ranges with more than 512 kByte by using more then four standard I/O pins for the program memory expansion.