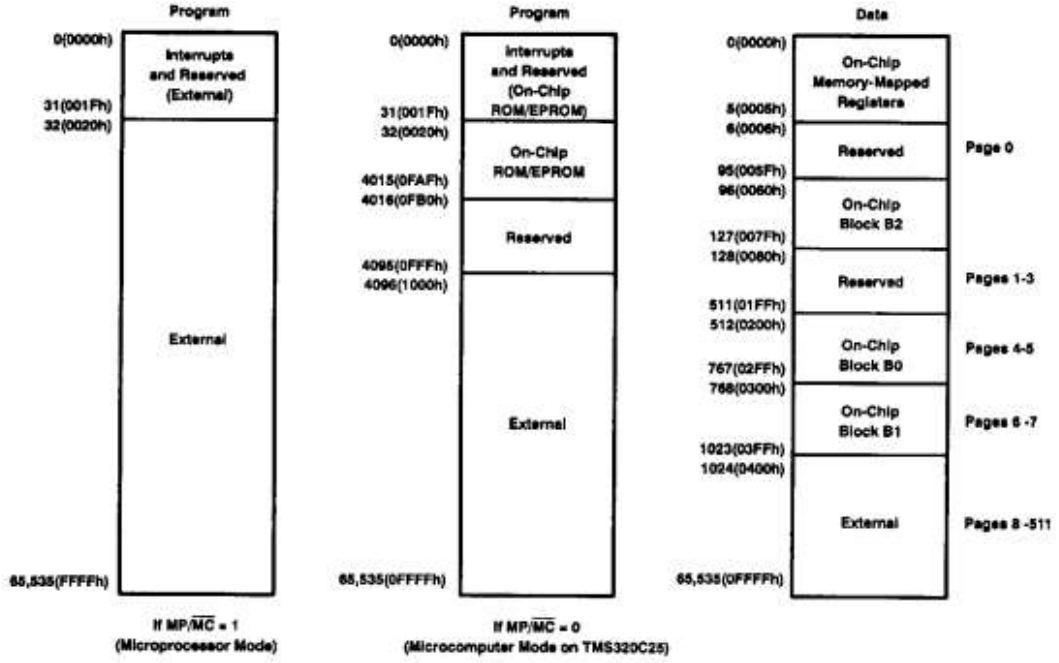
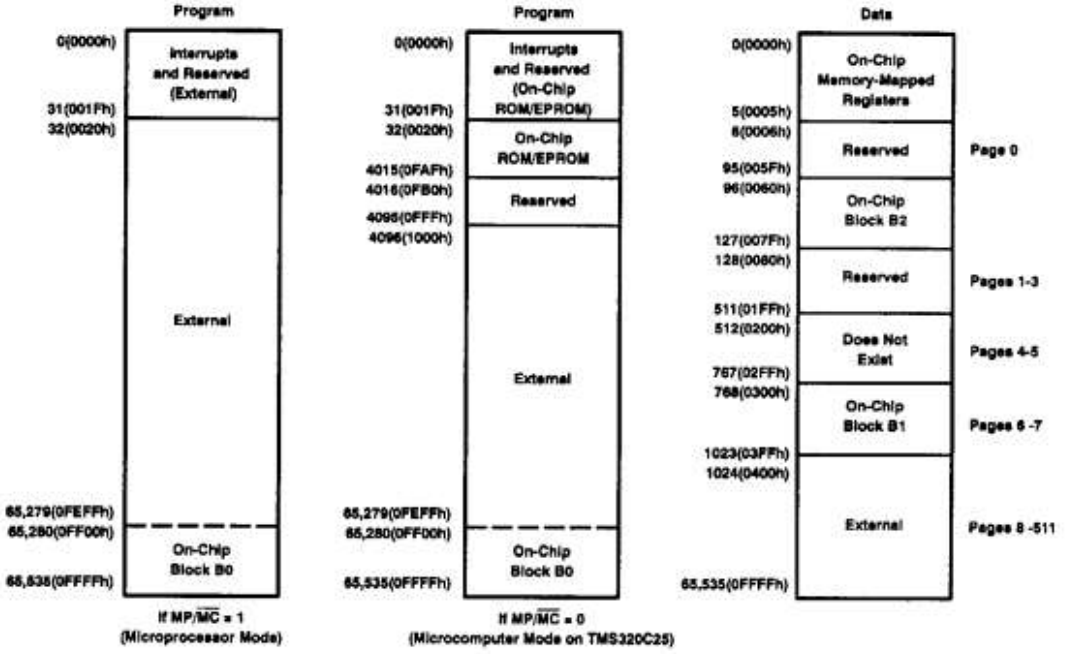


(a) Memory Maps After a CNFD Instruction



(b) Memory Maps After a CNFP Instruction

Figure 1. Memory Maps

MEMORY MAP REGISTER DATA MEMORY PAGE 0

Register Name	Address Location	Definition
DRR(15–0)	0	Serial port data receive register
DXR(15–0)	1	Serial port data transmit register
TIM(15–0)	2	Timer register
PRD(15–0)	3	Period register
IMR (5–0)	4	Interrupt mask register
GREG(7–0)	5	Global memory allocation register

Interrupt Mask Register (IMR)

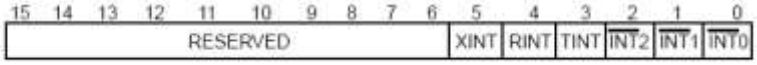


Table 3. TMS320C25 Instruction Set Summary

ACCUMULATOR MEMORY REFERENCE INSTRUCTIONS																	
MNEMONIC	DESCRIPTION	NO. WORDS	INSTRUCTION BIT CODE														
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ABS	Absolute value of accumulator	1	1	1	0	0	1	1	1	0	0	0	0	1	1	0	1	1
ADD	Add to accumulator with shift	1	0	0	0	0	←		S	→	M	←			D			→
ADDC	Add to accumulator with carry	1	0	1	0	0	0	0	1	1	M	←			D			→
ADDH	Add to high accumulator	1	0	1	0	0	1	0	0	0	M	←			D			→
ADDK	Add to accumulator short immediate	1	1	1	0	0	1	1	0	0	←				K			→
ADDS	Add to low accumulator with sign extension suppressed	1	0	1	0	0	1	0	0	1	M	←			D			→
ADDT	Add to accumulator with shift specified by T register	1	0	1	0	0	1	0	1	0	M	←			D			→
ADLK†	Add to accumulator long immediate with shift	2	1	1	0	1	←		S	→	0	0	0	0	0	0	1	0
AND	AND with accumulator	1	0	1	0	0	1	1	1	0	M	←			D			→
ANDK†	AND immediate with accumulator with shift	2	1	1	0	1	←		S	→	0	0	0	0	0	1	0	0
CMPL†	Complement accumulator	1	1	1	0	0	1	1	1	0	0	0	1	0	0	1	1	1
LAC	Load accumulator with shift	1	0	0	1	0	←		S	→	M	←			D			→
LACK	Load accumulator immediate short	1	1	1	0	0	1	0	1	0	←				K			→
LACT†	Load accumulator with shift specified by T register	1	0	1	0	0	0	0	1	0	M	←			D			→
LALK†	Load accumulator long immediate with shift	2	1	1	0	1	←		S	→	0	0	0	0	0	0	0	1
NEG†	Negate accumulator	1	1	1	0	0	1	1	1	0	0	0	1	0	0	0	1	1
NORM†	Normalize contents of accumulator	1	1	1	0	0	1	1	1	0	1	X	X	X	0	0	1	0
OR	OR with accumulator	1	0	1	0	0	1	1	0	1	M	←			D			→
ORK†	OR immediate with accumulator with shift	2	1	1	0	1	←		S	→	0	0	0	0	0	1	0	1
ROL	Rotate accumulator left	1	1	1	0	0	1	1	1	0	0	0	1	1	0	1	0	0
ROR	Rotate accumulator right	1	1	1	0	0	1	1	1	0	0	0	1	1	0	1	0	1
SACH	Store high accumulator with shift	1	0	1	1	0	1	←	X	→	M	←			D			→
SACL	Store low-order accumulator with shift	1	0	1	1	0	0	←	X	→	M	←			D			→
SBLK†	Subtract from accumulator long immediate with shift	2	1	1	0	1	←		S	→	0	0	0	0	0	0	1	1
SFL†	Shift accumulator left	1	1	1	0	0	1	1	1	0	0	0	0	1	1	0	0	0
SFR†	Shift accumulator right	1	1	1	0	0	1	1	1	0	0	0	0	1	1	0	0	1
SUB	Subtract from accumulator with shift	1	0	0	0	1	←		S	→	M	←			D			→
SUBB	Subtract from accumulator with borrow	1	0	1	0	0	1	1	1	1	M	←			D			→
SUBC	Conditional subtract	1	0	1	0	0	0	1	1	1	M	←			D			→
SUBH	Subtract from high accumulator	1	0	1	0	0	0	1	0	0	M	←			D			→
SUBK	Subtract from accumulator short immediate	1	1	1	0	0	1	1	0	1	←				K			→
SUBS	Subtract from low accumulator with sign extension suppressed	1	0	1	0	0	0	1	0	1	M	←			D			→

† These instructions are not included in the TMS320C1x instruction set.

Table 2. Instruction Symbols

SYMBOL	DEFINITION
B	4-bit field specifying a bit code
CM	2-bit field specifying compare mode
D	Data memory address field
FO	Format status bit
M	Addressing mode bit
K	Immediate operand field
PA	Port address (PA0–PA15 are predefined assembler symbols equal to o–15, respectively.)
PM	2-bit field specifying P register output shift code
AR	3-bit operand field specifying auxiliary register
S	4-bit left-shift code
X	3-bit accumulator left-shift field

Table 3. TMS320C25 Instruction Set Summary (continued)

ACCUMULATOR MEMORY REFERENCE INSTRUCTIONS																		
MNEMONIC	DESCRIPTION	NO. WORDS	INSTRUCTION BIT CODE															
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SUBT†	Subtract from accumulator with shift specified by T register	1	0	1	0	0	0	1	1	0	M	←			D			→
XOR	Exclusive-OR with accumulator	1	0	1	0	0	1	1	0	0	M	←			D			→
XORK†	Exclusive-OR immediate with accumulator with shift	2	1	1	0	1	←		S	→	0	0	0	0	0	1	1	0
ZAC	Zero accumulator	1	1	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0
ZALH	Zero low accumulator and load high accumulator	1	0	1	0	0	0	0	0	0	M	←			D			→
ZALR	Zero low accumulator and load high accumulator with rounding	1	0	1	1	1	1	0	1	1	M	←			D			→
ZALS	Zero accumulator and load low accumulator with sign extension suppressed	1	0	1	0	0	0	0	0	1	M	←			D			→

AUXILIARY REGISTERS AND DATA PAGE POINTER INSTRUCTIONS																		
MNEMONIC	DESCRIPTION	NO. WORDS	INSTRUCTION BIT CODE															
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADRK	Add to auxiliary register short immediate	1	0	1	1	1	1	1	1	0	←				K			→
CMPR†	Compare auxiliary register with auxiliary register AR0	1	1	1	0	0	1	1	1	0	0	1	0	1	0	0	←	CM →
LAR	Load auxiliary register	1	0	0	1	1	0	←	R	→	M	←			D			→
LARK	Load auxiliary register short immediate	1	1	1	0	0	0	←	R	→	←				K			→
LARP	Load auxiliary register pointer	1	0	1	0	1	0	1	0	1	1	0	0	0	1	←	R	→
LDP	Load data memory page pointer	1	0	1	0	1	0	0	1	0	M	←			D			→
LDPK	Load data memory page pointer immediate	1	1	1	0	0	1	0	0	←					DP			→
LRLK†	Load auxiliary register long immediate	2	1	1	0	1	0	←	R	→	0	0	0	0	0	0	0	0
MAR	Modify auxiliary register	1	0	1	0	1	0	1	0	1	M	←			D			→
SAR	Store auxiliary register	1	0	1	1	1	0	←	R	→	M	←			D			→
SBRK	Subtract from auxiliary register short immediate	1	0	1	1	1	1	1	1	1	←				K			→

† These instructions are not included in the TMS320C1x instruction set.

T REGISTER, P REGISTER, AND MULTIPLY INSTRUCTIONS

MNEMONIC	DESCRIPTION	NO. WORDS	INSTRUCTION BIT CODE															
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
APAC	Add P register to accumulator	1	1	1	0	0	1	1	1	0	0	0	0	1	0	1	0	1
LPH†	Load high P register	1	0	1	0	1	0	0	1	1	M	◄			D			►
LT	Load T register	1	0	0	1	1	1	1	0	0	M	◄			D			►
LTA	Load T register and accumulate previous product	1	0	0	1	1	1	1	0	1	M	◄			D			►
LTD	Load T register, accumulate previous product, and move data	1	0	0	1	1	1	1	1	1	M	◄			D			►
LTP†	Load T register and store P register in accumulator	1	0	0	1	1	1	1	1	0	M	◄			D			►
LTS†	Load T register and subtract previous product	1	0	1	0	1	1	0	1	1	M	◄			D			►
MAC†	Multiply and accumulate	2	0	1	0	1	1	1	0	1	M	◄			D			►
MACD†	Multiply and accumulate with data move	2	0	1	0	1	1	1	0	0	M	◄			D			►
MPY	Multiply (with T register, store product in P register)	1	0	0	1	1	1	0	0	0	M	◄			D			►
MPYA	Multiply and accumulate previous product	1	0	0	1	1	1	0	1	0	M	◄			D			►
MPYK	Multiply immediate	1	1	0	1	◄					K							►
MPYS	Multiply and subtract previous product	1	0	0	1	1	1	0	1	1	M	◄			D			►
MPYU	Multiply unsigned	1	1	1	0	0	1	1	1	1	M	◄			D			►
PAC	Load accumulator with P register	1	1	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0
SPAC	Subtract P register from accumulator	1	1	1	0	0	1	1	1	0	0	0	0	1	0	1	1	0
SPH	Store high P register	1	0	1	1	1	1	1	0	1	M	◄			D			►
SPL	Store low P register	1	0	1	1	1	1	1	0	0	M	◄			D			►
SPM†	Set P register output shift mode	1	1	1	0	0	1	1	1	0	0	0	0	0	1	0	◄	PM►
SQRA†	Square and accumulate	1	0	0	1	1	1	0	0	1	M	◄			D			►
SQRS†	Square and subtract previous product	1	0	1	0	1	1	0	1	0	M	◄			D			►

† These instructions are not included in the TMS320C1x instruction set

BRANCH/CALL INSTRUCTIONS

MNEMONIC	DESCRIPTION	NO. WORDS	INSTRUCTION BIT CODE															
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
B	Branch unconditionally	2	1	1	1	1	1	1	1	1	1	←			D			→
BACC†	Branch to address specified by accumulator	1	1	1	0	0	1	1	1	0	0	0	1	0	0	1	0	1
BANZ	Branch on auxiliary register not zero	2	1	1	1	1	1	0	1	1	1	←			D			→
BBNZ†	Branch if TC bit ≠ 0	2	1	1	1	1	1	0	0	1	1	←			D			→
BBZ†	Branch if TC bit = 0	2	1	1	1	1	1	0	0	0	1	←			D			→
BC	Branch on carry	2	0	1	0	1	1	1	1	0	1	←			D			→
BGEZ	Branch if accumulator ≥ 0	2	1	1	1	1	0	1	0	0	1	←			D			→
BGZ	Branch if accumulator > 0	2	1	1	1	1	0	0	0	1	1	←			D			→
BIOZ	Branch on I/O status = 0	2	1	1	1	1	1	0	1	0	1	←			D			→
BLEZ	Branch if accumulator ≤ 0	2	1	1	1	1	0	0	1	0	1	←			D			→
BLZ	Branch if accumulator < 0	2	1	1	1	1	0	0	1	1	1	←			D			→
BNC	Branch on no carry	2	0	1	0	1	1	1	1	1	1	←			D			→
BNV†	Branch if no overflow	2	1	1	1	1	0	1	1	1	1	←			D			→
BNZ	Branch if accumulator ≠ 0	2	1	1	1	1	0	1	0	1	1	←			D			→
BV	Branch on overflow	2	1	1	1	1	0	0	0	0	1	←			D			→
BZ	Branch if accumulator = 0	2	1	1	1	1	0	1	1	0	1	←			D			→
CALA	Call subroutine indirect	1	1	1	0	0	1	1	1	0	0	0	1	0	0	1	0	0
CALL	Call subroutine	2	1	1	1	1	1	1	1	0	1	←			D			→
RET	Return from subroutine	1	1	1	0	0	1	1	1	0	0	0	1	0	0	1	1	0

I/O AND DATA MEMORY OPERATIONS

MNEMONIC	DESCRIPTION	NO. WORDS	INSTRUCTION BIT CODE															
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BLKD†	Block move from data memory to data memory	2	1	1	1	0	1	1	0	1	M	←			D			→
BLKP†	Block move from program memory to data memory	2	1	1	1	1	1	1	0	0	M	←			D			→
DMOV	Data move in data memory	1	0	1	0	1	0	1	1	0	M	←			D			→
FORT†	Format serial port registers	1	1	1	0	0	1	1	1	0	0	0	0	0	1	1	1	FO
IN	Input data from port	1	1	0	0	0	←		PA	→	M	←			D			→
OUT	Output data to port	1	1	1	1	0	←		PA	→	M	←			D			→
RFSM	Reset serial port frame synchronization mode	1	1	1	0	0	1	1	1	0	0	0	1	1	0	1	1	0
RTXM†	Reset serial port transmit mode	1	1	1	0	0	1	1	1	0	0	0	1	0	0	0	0	0
RXF†	Reset external flag	1	1	1	0	0	1	1	1	0	0	0	0	0	1	1	0	0
SFSM	Set serial port frame synchronization mode	1	1	1	0	0	1	1	1	0	0	0	1	1	0	1	1	1
STXM†	Set serial port transmit mode	1	1	1	0	0	1	1	1	0	0	0	1	0	0	0	0	1
SXF†	Set external flag	1	1	1	0	0	1	1	1	0	0	0	0	0	1	1	0	1
TBLR	Table read	1	0	1	0	1	1	0	0	0	M	←			D			→
TBLW	Table write	1	0	1	0	1	1	0	0	1	M	←			D			→

CONTROL INSTRUCTIONS																		
MNEMONIC	DESCRIPTION	NO. WORDS	INSTRUCTION BIT CODE															
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BIT†	Test bit	1	1	0	0	1	← B →				M	← D →						
BITT†	Test bit specified by T register	1	0	1	0	1	0	1	1	1	M	← D →						
CNFD†	Configure block as data memory	1	1	1	0	0	1	1	1	0	0	0	0	0	0	1	0	0
CNFP†	Configure block as program memory	1	1	1	0	0	1	1	1	0	0	0	0	0	0	1	0	1
DINT	Disable interrupt	1	1	1	0	0	1	1	1	0	0	0	0	0	0	0	0	1
EINT	Enable interrupt	1	1	1	0	0	1	1	1	0	0	0	0	0	0	0	0	0
IDLE†	Idle until interrupt	1	1	1	0	0	1	1	1	0	0	0	0	1	1	1	1	1
LST	Load status register STO	1	0	1	0	1	0	0	0	0	M	← D →						
LST1†	Load status register ST1	1	0	1	0	1	0	0	0	1	M	← D →						
NOP	No operation	1	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0
POP	Pop top of stack to low accumulator	1	1	1	0	0	1	1	1	0	0	0	0	1	1	1	0	1
POPD†	Pop top of stack to data memory	1	0	1	1	1	1	0	1	0	M	← D →						
PSHD†	Push data memory value onto stack	1	0	1	0	1	0	1	0	0	M	← D →						
PUSH	Push low accumulator onto stack	1	1	1	0	0	1	1	1	0	0	0	0	1	1	1	0	0
RC	Reset carry bit	1	1	1	0	0	1	1	1	0	0	0	1	1	0	0	0	0
RHM	Reset hold mode	1	1	1	0	0	1	1	1	0	0	0	1	1	1	0	0	0
ROVM	Reset overflow mode	1	1	1	0	0	1	1	1	0	0	0	0	0	0	0	1	0
RPT†	Repeat instruction as specified by data memory value	1	0	1	0	0	1	0	1	1	M	← D →						
RPTK†	Repeat instruction as specified by immediate value	1	1	1	0	0	1	0	1	1	← K →							
RSXM†	Reset sign-extension mode	1	1	1	0	0	1	1	1	0	0	0	0	0	0	1	1	0
RTC	Reset test/control flag	1	1	1	0	0	1	1	1	0	0	0	1	1	0	0	1	0
SC	Set carry bit	1	1	1	0	0	1	1	1	0	0	0	1	1	0	0	0	1
SHM	Set hold mode	1	1	1	0	0	1	1	1	0	0	0	1	1	1	0	0	1
SOVM	Set overflow mode	1	1	1	0	0	1	1	1	0	0	0	0	0	0	0	1	1
SST	Store status register ST0	1	0	1	1	1	1	0	0	0	M	← D →						
SST1†	Store status register ST1	1	0	1	1	1	1	0	0	1	M	← D →						
SSXM†	Set sign-extension mode	1	1	1	0	0	1	1	1	0	0	0	0	0	0	1	1	1
STC	Set test/control flag	1	1	1	0	0	1	1	1	0	0	0	1	1	0	0	1	1
TRAP†	Software interrupt	1	1	1	0	0	1	1	1	0	0	0	0	1	1	1	1	0