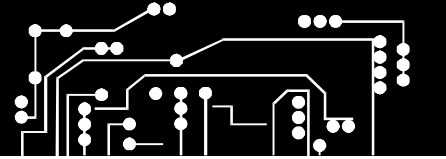


# **PowerPC Embedded Processors**

## **Application Note**



## **Using Multiple Banks of Self-Refresh DRAM**

June 26, 1996

IBM Microelectronics  
Dept D95/Bldg 060  
3039 Cornwallis Road  
Research Triangle Park, NC 27709

*This application provides a method for multiple banks of DRAM to enter and exit self-refresh mode with the PPC403Gx processor.*

### **1 Introduction**

Self-refresh DRAM provide the system designer with the capability to reduce the power dissipation of the DRAM device to 2-4% of the power required when the DRAM is active. This allows applications which run on battery power to achieve significant power savings during long intervals when the DRAM is not accessed. Data can be stored in DRAM with reduced power consumption when the application is in a lower power state. This also allows quick access to the data when the application returns to full power.

The PPC403Gx supports self-refresh DRAM by providing system software the capability to control RAS and CAS through the bank control registers. These registers and functions are described in the PPC403 Embedded Controller User's Manual for each specific device.

### **2 Entering Self-Refresh Mode Sequence**

#### **2.1 Initial Conditions and Assumptions**

Two banks of self-refresh DRAM attached to the PPC403Gx processor as Bank 7 and Bank 6. Both banks are initialized and can be accessed by the processor. Each bank has initially been set with a regular refresh interval.

The self-refresh DRAM described in this application note requires CAS to be active before RAS. CAS and RAS are held active in the self-refresh state. Refer to IBM part number IBM0117800AT3, 2Mx8 Fast Page Mode DRAM with Self Refresh.

Once any bank of DRAM is set into self-refresh mode no DRAM accesses are possible. The CAS lines are shared between the DRAM banks.

The code that is executing to control the self-refresh DRAM must execute from ROM/RAM or be contained entirely in the instruction cache before starting the sequence to enter self-refresh mode.

General purpose registers (GPR) r6, r7, r16, and r17 are used throughout to modify the bank control registers for Bank 6 and Bank 7. GPRs r30 and r31 are used to store the original values.



## 2.2 Overview of Sequence

1. Set Bank 7 into immediate refresh and then no refresh.
2. Set Bank 6 into immediate refresh and then no refresh.
3. Set Bank 7 for self-refresh, Hold CAS.
4. Set Bank 7 for self-refresh, Hold CAS, Hold RAS
5. Set Bank 6 for self-refresh, Hold CAS, Hold RAS

### 2.3 Set Bank 7 into Immediate Refresh and then No Refresh.

Bank 7 is set into immediate refresh mode by setting bits 19 of Bank Register 7 (BR7) to '1', alternate refresh mode, and bits 27:30 to '0001', immediate refresh. This guarantees that the DRAM controller will execute one refresh operation to that DRAM bank before entering self-refresh mode. The bank register only needs to be set for one cycle. The refresh interval will be maintained if the DRAM controller was close to initiating a refresh operation. Refresh control for Bank 7 is then turned off so that Bank 6 can be set up for Immediate Refresh. Bits 19 and 27:30 are all set to "0" in BR7.

#### 2.3.1 Code Example

```
#-----
# Bank 7 - Immediate Refresh and No Refresh
#-----
mfbr7    r31                /* save old value of Bank 7
mfbr7    r7                 /* get working copy of Bank 7
or       r17,r7,r7          /* copy r7 to r17
ori      r17,r17,0x1002     /* set bits 19,30
andi     r17,r17,0xFFE3     /* reset bits 27,28,29
andi     r7,r7,0xEFE1       /* reset bits 19,27,28,29,30
sync                                /* ensure no DRAM accesses while setting
                                /* bank register
mtbr7    r17                /* set bank register for immediate refresh
mtbr7    r7                 /* set bank register for no refresh

nop                                /* allow time for refresh
nop
nop
nop
```

### 2.4 Set Bank 6 into Immediate Refresh and then No Refresh.

Bank 6 is set into immediate refresh mode by setting bits 19 of BR6 to '1', alternate refresh mode, and bits 27:30 to '0001', immediate refresh. This guarantees that the DRAM controller will execute one refresh operation to that DRAM bank before entering self-refresh mode. The bank register only needs to be set for one cycle. The refresh interval will be maintained if the DRAM controller was close to initiating a refresh operation. Refresh control for Bank 6 is then turned off. Bits 19 and 27:30 of BR6 are all set to "0" in the bank control register. The refresh control to both banks is now turned off.

### 2.4.1 Code Example

```
#-----
# Bank 6 - Immediate Refresh and No Refresh
#-----
mfbr6    r31          /* save old value of Bank 6
mfbr6    r6           /* get working copy of Bank 6
or       r16,r6,r6     /* copy r6 to r16
ori      r16,r16,0x1002 /* set bits 19,30
andi     r16,r16,0xFFE3 /* reset bits 27,28,29
andi     r6,r6,0xEFE1  /* reset bits 19,27,28,29,30
sync                    /* ensure no DRAM accesses while setting
                    /* bank register
mtbr6    r16          /* set bank register for immediate refresh
mtbr6    r6           /* set bank register for no refresh

nop                      /* allow time for refresh
nop
nop
nop
```

### 2.5 Set Bank 7 for Self-Refresh, Hold CAS.

Bank 7 is set to alternate refresh mode, bit 19 set to “1”, and hold CAS active, bits 27:30 set to “0100”. This activates all four CAS lines and holds them active. The CAS lines are connected to both banks of DRAM.

#### 2.5.1 Code Example

```
#-----
# Bank 7 - Self-Refresh, Hold CAS
#-----
ori      r7,r7,0x1008  /* set bits 19,28
andi     r7,r7,0xFFE9  /* reset bits 27,29,30
mtbr7    r7           /* set bank register
```

### 2.6 Set Bank 7 for Self-Refresh, Hold CAS, Hold RAS

Bank 7 is set to alternate refresh mode, bit 19 set to “1”, and hold CAS and RAS active, bits 27:30 set to “0110”. This activates the RAS line for this bank and holds it active. CAS and RAS are now both active to this bank and the DRAM will perform self-refresh.

#### 2.6.1 Code Example

```
#-----
# Bank 7 - Self-Refresh, Hold CAS, Hold RAS
#-----
ori      r7,r7,0x100C  /* set bits 19,28,29
andi     r7,r7,0xFFED  /* reset bits 27,30
mtbr7    r7           /* set bank register
```

### 2.7 Set Bank 6 for Self-Refresh, Hold CAS, Hold RAS

Bank 6 is set to alternate refresh mode, bit 19 set to “1”, and hold CAS and RAS active, bits 27:30 set to “0110”. This activates the RAS line for this bank and holds it active. CAS and RAS are now both active to this bank and the DRAM will perform self-refresh. The CAS lines have already been activated when Bank 7 was set into self-refresh.

### 2.7.1 Code Example

```
#-----  
# Bank 6 - Self-Refresh, Hold CAS, Hold RAS  
#-----  
ori      r6,r6,0x100C    /* set bits 19,28,29  
andi     r6,r6,0xFFED    /* reset bits 27,30  
mtbr6    r6              /* set bank register
```

## 3 Exiting Self-Refresh Mode Sequence

### 3.1 Initial Conditions and Assumptions

Two banks of self-refresh DRAM attached as Bank 7 and Bank 6 in self-refresh mode.

### 3.2 Overview of Sequence

1. Set Bank 7 for no refresh.
2. Set Bank 6 for no refresh.
3. Set Bank 6 into immediate refresh.
4. Set Bank 6 for regular refresh.
5. Set Bank 7 into immediate refresh.
6. Set Bank 7 for regular refresh.

### 3.3 Set Bank 7 for No Refresh.

Refresh control for Bank 7 is turned off and RAS is set inactive for this bank. The CAS lines are still being held active by Bank 6. Bits 19 and 27:30 are all set to "0" in BR7. The refresh control to both banks is now turned off.

#### 3.3.1 Code Example

```
#-----  
# Bank 7 - No Refresh  
#-----  
andi     r7,r7,0xEFE1    /* reset bits 19,27,28,29,30  
mtbr6    r7
```

### 3.4 Set Bank 6 for No Refresh.

Refresh control for Bank 6 is turned off. RAS and CAS are set inactive and both banks are now out of self-refresh mode. Bits 19 and 27:30 are all set to "0" in BR6. The refresh control to both banks is now turned off.

#### 3.4.1 Code Example

```
#-----  
# Bank 6 - No Refresh  
#-----  
andi     r6,r6,0xEFE1    /* reset bits 19,27,28,29,30  
mtbr6    r6
```

### 3.5 Set Bank 6 into Immediate Refresh.

Bank 6 is set into immediate refresh mode. This guarantees that the DRAM controller will initiate a refresh operation to this DRAM bank as soon as possible. The refresh interval will be maintained if the self-refresh DRAM was close to initiating a refresh operation. See section 2.4 for exact bits in bank control register and code example.

### 3.6 Set Bank 6 for Regular Refresh.

Set Bank 6 back to the original refresh interval. The microprocessor can now access this DRAM.

#### 3.6.1 Code Example

```
#-----  
# Bank 6 - Restore previous bank register value  
#-----  
          mtbr6      r30          /* restore bank register
```

### 3.7 Set Bank 7 into Immediate Refresh.

Bank 7 is set into immediate refresh mode. This guarantees that the DRAM controller will initiate a refresh operation to this DRAM bank as soon as possible. The refresh interval will be maintained if the self-refresh DRAM was close to initiating a refresh operation. See section 2.3 for exact bits in bank control register and code example.

### 3.8 Set Bank 7 for Regular Refresh.

Set Bank 7 back to the original refresh interval. The microprocessor can now access both banks of DRAM.

#### 3.8.1 Code Example

```
#-----  
# Bank 7 - Restore previous bank register value  
#-----  
          mtbr7      r31          /* restore bank register
```

IBM will continue to enhance products and services as new technologies emerge. Therefore, IBM reserves the right to make changes to its products, other product information, and this publication without prior notice. Please contact your local IBM Microelectronics representative on specific standard configurations and options.

IBM assumes no responsibility or liability for any use of the information contained herein. Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of IBM or third parties. NO WARRANTIES OF ANY KIND, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE ARE OFFERED IN THIS DOCUMENT.