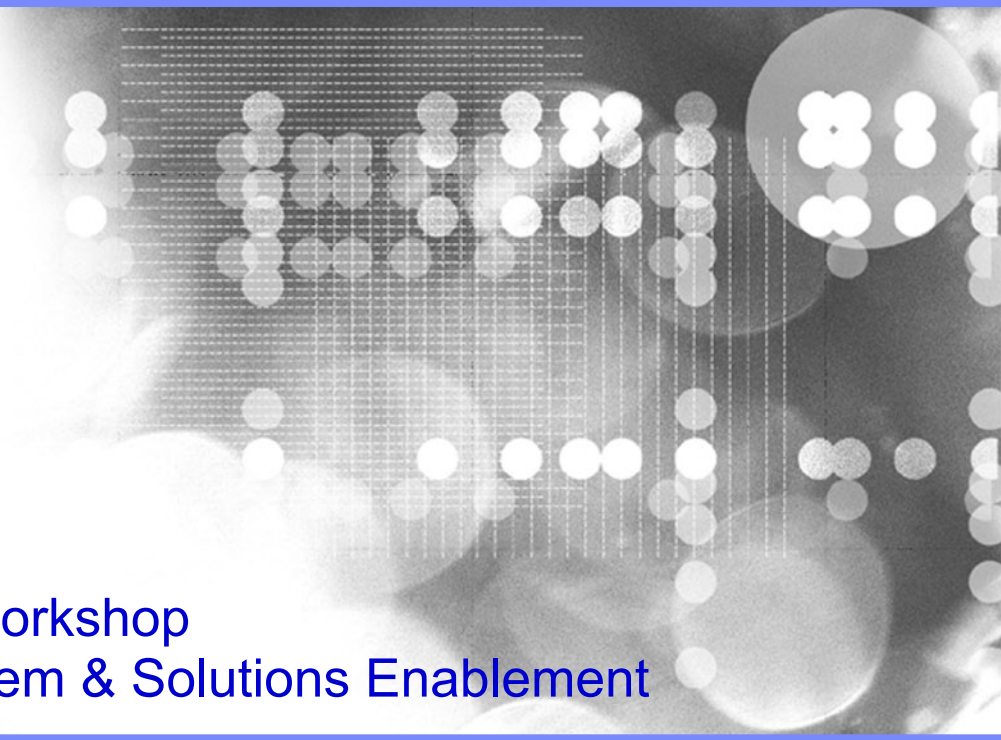
Hands-on DMA 1



Cell Programming Workshop
Cell/Quasar Ecosystem & Solutions Enablement

Class Objectives

- **At the end of this class you should know how to use DMA to transfer data between SPU and PPU, back and forth, using a buffer**

Class agenda

- **DMA transfer into an SPU from a PPU**
- **Example of mfc_get**
- **DMA transfer from an SPU into a PPU**
- **Example of mfc_put**

  – /opt/cell_class/Hands-on-21/basicDMA

DMA data into an SPU - mfc_get

DMA Example: Read into Local Store

```
void dma_mem_to_ls(unsigned int mem_addr,
         volatile void *ls_addr,unsigned int size)
{
  unsigned int tag = 0;
  unsigned int mask = 1;
  mfc_get(ls_addr,mem_addr,size,tag,0,0);
  mfc_write_tag_mask(mask);
  mfc_read_tag_status_all();
}
```

Read contents of mem_addr into ls_addr

Set tag mask

Wait for all tag DMA completed

DMA data out of the SPU into the PPU – mfc_put

DMA Example: Write to Main Memory

```
void dma_ls_to_mem(unsigned int mem_addr,volatile void
*ls_addr, unsigned int size)
{
  unsigned int tag = 0;
  unsigned int mask = 1;
  mfc_put(ls_addr,mem_addr,size,tag,0,0);
  mfc_write_tag_mask(mask);
  mfc_read_tag_status_all();
}
```

Write contents of mem_addr into ls_addr

Set tag mask

Wait for all tag DMA completed

Using libspe 2.x – synchronous thread model

The PPU program

```
#include <stdio.h>
//#include <libspe.h>
//#include <libmisc.h>
#include <string.h>
#include <libspe2.h>

//spu program
extern spe_program_handle_t getbuf_spu;
//local buffer
unsigned char buffer[128] __attribute__
    ((aligned(128)));
//spe context
spe_context_ptr_t speid;
unsigned int flags = 0;
unsigned int entry = SPE_DEFAULT_ENTRY;
spe_stop_info_t stop_info;
int rc;
```

```
int main (void)
{
    strcpy (buffer, "Good morning!");
    printf("Original buffer is %s\n", buffer);
    speid = spe_context_create(flags, NULL);
        spe_program_load(speid, &getbuf_spu);
    rc = spe_context_run(speid, &entry, 0, buffer,
    NULL, &stop_info);
    spe_context_destroy(speid);

    printf("New modified buffer is %s\n", buffer);
    return 0;
}
```

```
DIRS                    = spu
PROGRAM_ppu             = getbuf_dma
IMPORTS  = -lspe2 -lpthread -lmisc \
                        spu/getbuf_spu.a
include $(CELL_TOP)/make.footer
```

The SPU program

```
#include <stdio.h>
#include <string.h>
//#include <libmisc.h>
#include <spu_mfcio.h>
unsigned char buffer[128] __attribute__ ((aligned(128)));
int main(unsigned long long speid, unsigned long long argp, unsigned long long envp)
{
    int tag = 31, tag_mask = 1<<tag;
    // DMA in buffer from PPE
    mfc_get(buffer, (unsigned long long)argp, 128, tag, 0, 0);
    mfc_write_tag_mask(tag_mask);
    mfc_read_tag_status_any();
    printf("SPE received buffer \"%s\"\n", buffer);
    // modify buffer
    strcpy (buffer, "Guten Morgen!");
    printf("SPE sent to PPU buffer \"%s\"\n", buffer);
    // DMA out buffer to PPE
    mfc_put(buffer, (unsigned long long)argp, 128, tag, 0, 0);
    mfc_write_tag_mask(tag_mask);
    mfc_read_tag_status_any();
    return 0;
```

```
PROGRAM_spu      := getbuf_spu
LIBRARY_embed    := getbuf_spu.a
IMPORTS          = -lmisc
include $(CELL_TOP)/make.footer
```

Using libspe 2.x – asynchronous thread model

The PPU program

```
#include <stdio.h>
//#include <libspe.h>
//#include <libmisc.h>
#include <string.h>
#include <libspe2.h>
#include <pthread.h>
typedef struct ppu_pthread_data{
   spe_context_ptr_t context;
   pthread_t pthread;
   unsigned int entry;
   unsigned int flags;
   void *argp;
   void *envp;
   spe_stop_info_t stopinfo;
}  ppu_pthread_data_t;
void *ppu_pthread_function(void *arg)
{
   ppu_pthread_data_t *datap = (ppu_pthread_data_t *)arg;
   int rc;
      rc = spe_context_run(datap->context, &datap->entry, datap-
     >flags, datap->argp, datap->envp, &datap->stopinfo);
   pthread_exit(NULL);
}
//spu program
extern spe_program_handle_t getbuf_spu;
//local buffer
unsigned char buffer[128] __attribute__ ((aligned(128)));
```

```
//spe context
//spe_context_ptr_t speid;
//unsigned int flags = 0;
//unsigned int entry = SPE_DEFAULT_ENTRY;
//spe_stop_info_t stop_info;
//int rc;
int main (void)
{
strcpy (buffer, "Good morning!");
      printf("Original buffer is %s\n", buffer);
//    speid = spe_context_create(flags, NULL);
//    spe_program_load(speid, &getbuf_spu);
//    rc = spe_context_run(speid, &entry, 0, buffer, NULL, &stop_info);
//    spe_context_destroy(speid);
   ppu_pthread_data_t data;
   data.context = spe_context_create(0, NULL);
   spe_program_load(data.context, &getbuf_spu);
   data.entry = SPE_DEFAULT_ENTRY;
   data.flags = 0;
   data.argp = buffer;
   data.envp = NULL;
   pthread_create(&data.pthread, NULL, &ppu_pthread_function, &data);
   pthread_join(data.pthread, NULL);
   spe_context_destroy(data.context);
      printf("New modified buffer is %s\n", buffer);
      return 0;
}
```

```
DIRS                          = spu
PROGRAM_ppu                   = getbuf_dma
IMPORTS       = -lspe2 -lpthread -lmisc  spu/getbuf_spu.a
include $(CELL_TOP)/make.footer
```

The SPU program

```
#include <stdio.h>
#include <string.h>
//#include <libmisc.h>
#include <spu_mfcio.h>

unsigned char buffer[128] __attribute__ ((aligned(128)));

int main(unsigned long long speid, unsigned long long argp, unsigned long long envp)
{
    int tag = 31, tag_mask = 1<<tag;
    // DMA in buffer from PPE
    mfc_get(buffer, (unsigned long long)argp, 128, tag, 0, 0);
    mfc_write_tag_mask(tag_mask);
    mfc_read_tag_status_any();
    printf("SPE received buffer \"%s\"\n", buffer);

    // modify buffer
    strcpy (buffer, "Guten Morgen!");
    printf("SPE sent to PPU buffer \"%s\"\n", buffer);
    // DMA out buffer to PPE
    mfc_put(buffer, (unsigned long long)argp, 128, tag, 0, 0);
    mfc_write_tag_mask(tag_mask);
    mfc_read_tag_status_any();
    return 0;
}
```

```
PROGRAM_spu      := getbuf_spu
LIBRARY_embed    := getbuf_spu.a
IMPORTS          = -lmisc
include $(CELL_TOP)/make.footer
```

Special Notices -- Trademarks

This document was developed for IBM offerings in the United States as of the date of publication. IBM may not make these offerings available in other countries, and the information is subject to change without notice. Consult your local IBM business contact for information on the IBM offerings available in your area. In no event will IBM be liable for damages arising directly or indirectly from any use of the information contained in this document.

Information in this document concerning non-IBM products was obtained from the suppliers of these products or other public sources. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. Send license inquires, in writing, to IBM Director of Licensing, IBM Corporation, New Castle Drive, Armonk, NY 10504-1785 USA.

All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

The information contained in this document has not been submitted to any formal IBM test and is provided "AS IS" with no warranties or guarantees either expressed or implied.

All examples cited or described in this document are presented as illustrations of the manner in which some IBM products can be used and the results that may be achieved. Actual environmental costs and performance characteristics will vary depending on individual client configurations and conditions.

IBM Global Financing offerings are provided through IBM Credit Corporation in the United States and other IBM subsidiaries and divisions worldwide to qualified commercial and government clients. Rates are based on a client's credit rating, financing terms, offering type, equipment type and options, and may vary by country. Other restrictions may apply. Rates and offerings are subject to change, extension or withdrawal without notice.

IBM is not responsible for printing errors in this document that result in pricing or information inaccuracies.

All prices shown are IBM's United States suggested list prices and are subject to change without notice; reseller prices may vary.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

Many of the features described in this document are operating system dependent and may not be available on Linux. For more information, please check: http://www.ibm.com/systems/p/software/whitepapers/linux_overview.html

Any performance data contained in this document was determined in a controlled environment. Actual results may vary significantly and are dependent on many factors including system hardware configuration and software design and configuration. Some measurements quoted in this document may have been made on development-level systems. There is no guarantee these measurements will be the same on generally-available systems. Some measurements quoted in this document may have been estimated through extrapolation. Users of this document should verify the applicable data for their specific environment.

Revised January 19, 2006

Special Notices (Cont.) -- Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States and/or other countries: alphaWorks, BladeCenter, Blue Gene, ClusterProven, developerWorks, e business(logo), e(logo)business, e(logo)server, IBM, IBM(logo), ibm.com, IBM Business Partner (logo), IntelliStation, MediaStreamer, Micro Channel, NUMA-Q, PartnerWorld, PowerPC, PowerPC(logo), pSeries, TotalStorage, xSeries; Advanced Micro-Partitioning, eServer, Micro-Partitioning, NUMACenter, On Demand Business logo, OpenPower, POWER, Power Architecture, Power Everywhere, Power Family, Power PC, PowerPC Architecture, POWER5, POWER5+, POWER6, POWER6+, Redbooks, System p, System p5, System Storage, VideoCharger, Virtualization Engine.

A full list of U.S. trademarks owned by IBM may be found at: http://www.ibm.com/legal/copytrade.shtml.

Cell Broadband Engine and Cell Broadband Engine Architecture are trademarks of Sony Computer Entertainment, Inc. in the United States, other countries, or both.
Rambus is a registered trademark of Rambus, Inc.
XDR and FlexIO are trademarks of Rambus, Inc.
UNIX is a registered trademark in the United States, other countries or both.
Linux is a trademark of Linus Torvalds in the United States, other countries or both.
Fedora is a trademark of Redhat, Inc.
Microsoft, Windows, Windows NT and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries or both.
Intel, Intel Xeon, Itanium and Pentium are trademarks or registered trademarks of Intel Corporation in the United States and/or other countries.
AMD Opteron is a trademark of Advanced Micro Devices, Inc.
Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States and/or other countries.
TPC-C and TPC-H are trademarks of the Transaction Performance Processing Council (TPPC).
SPECint, SPECfp, SPECjbb, SPECweb, SPECjAppServer, SPEC OMP, SPECviewperf, SPECapc, SPEChpc, SPECjvm, SPECmail, SPECimap and SPECsfs are trademarks of the Standard Performance Evaluation Corp (SPEC).
AltiVec is a trademark of Freescale Semiconductor, Inc.
PCI-X and PCI Express are registered trademarks of PCI SIG.
InfiniBand™ is a trademark the InfiniBand® Trade Association
Other company, product and service names may be trademarks or service marks of others.

Revised July 23, 2006

Special Notices - Copyrights

The following are trademarks of International Business Machines Corporation in the United States, or other countries, or both.

IBM IBM Logo Power Architecture

Other company, product and service names may be trademarks or service marks of others.

All information contained in this document is subject to change without notice. The products described in this document are NOT intended for use in applications such as implantation, life support, or other hazardous uses where malfunction could result in death, bodily injury, or catastrophic property damage. The information contained in this document does not affect or change IBM product specifications or warranties. Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of IBM or third parties. All information contained in this document was obtained in specific environments, and is presented as an illustration. The results obtained in other operating environments may vary.

While the information contained herein is believed to be accurate, such information is preliminary, and should not be relied upon for accuracy or completeness, and no representations or warranties of accuracy or completeness are made.

THE INFORMATION CONTAINED IN THIS DOCUMENT IS PROVIDED ON AN "AS IS" BASIS. In no event will IBM be liable for damages arising directly or indirectly from any use of the information contained in this document.

IBM Microelectronics Division
1580 Route 52, Bldg. 504
Hopewell Junction, NY 12533-6351

The IBM home page is http://www.ibm.com
The IBM Microelectronics Division home page is
http://www.chips.ibm.com

