Appendix - Decoded Instructions

This appendix contains a list of decoded instructions for the distinct ISAs we have explored. We enclose optional operands in parentheses. Operands use the names indicated by Figure 1. A specific operand name in the tables here correspond to a specific range of bits in Figure 1. We include official descriptions of the instructions from NVIDIA's CUDA Toolkit Documentation.

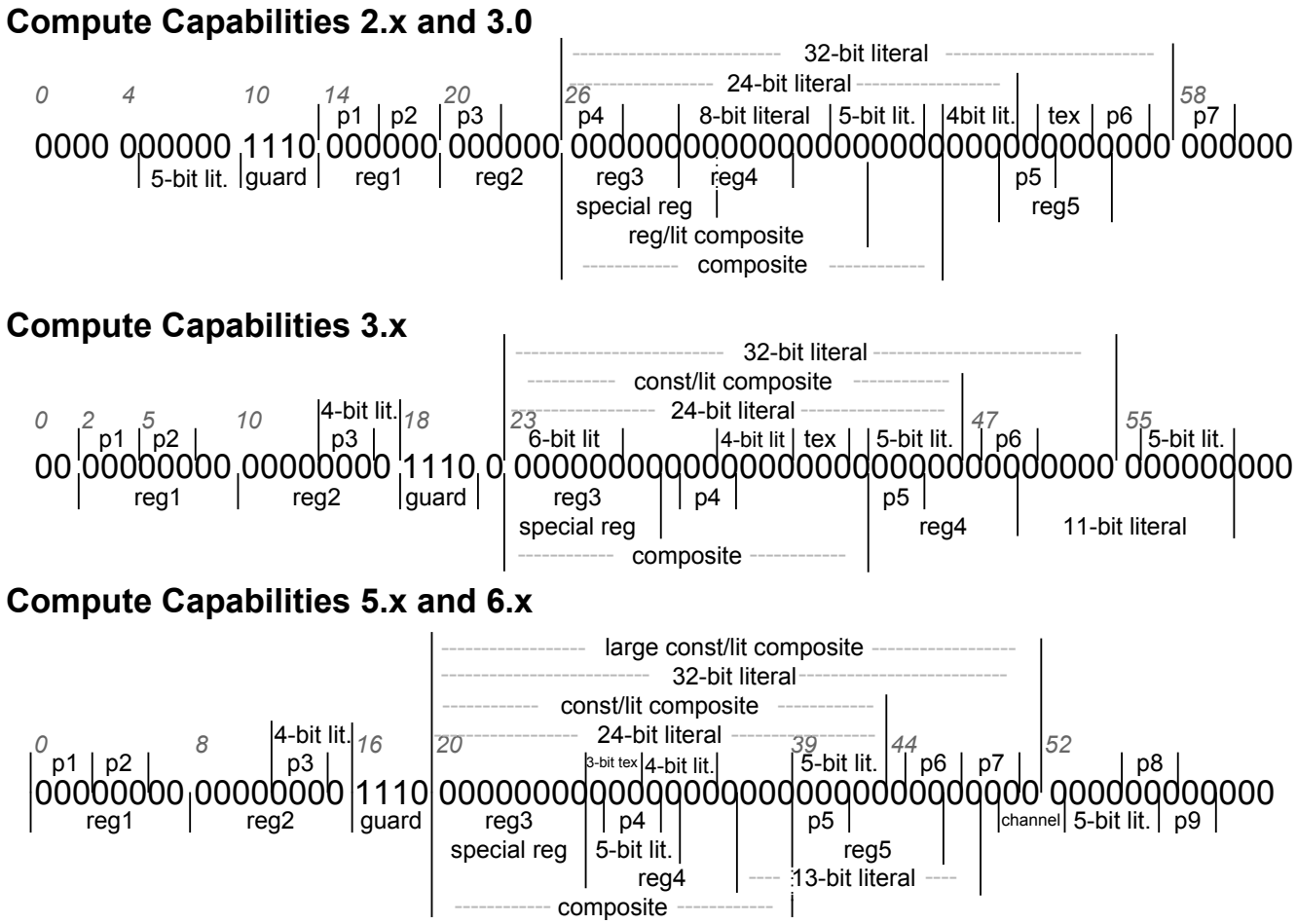

Figure 1. Common operands on different CUDA architectures.

Instructions for Compute Capabilities 2.x and 3.0

Instruction	Official Description
VOTE (reg1), p6, p3	Query condition across threads
CCTL (reg1), [reg2 + 30-bit lit]	Cache Control
MOV32I, reg1, 32-bit lit	((undocumented))
MOV reg1, composite	Move
LDC reg1, const_mem_with_reg2	Load from Constant
SEL reg1, reg2, composite, p5	Conditional Select/Move
B2R reg1, 6-bit lit, p4	Barrier to Register
S2R reg1, special reg	Special Register to Register
P2R reg1, PR, reg2, composite	Predicate to Register
R2P PR, reg2, composite	Register to Predicate
TEX reg1, reg2, (reg3), 8-bit lit, 5-bit lit, tex, 4-bit lit	Texture Fetch
TLD reg1, reg2, 8-bit lit, tex, 4-bit lit	Texture Load

LEPC reg1	Load Effective PC
I2F reg1, composite	Integer to Float
I2I reg1, composite	Integer to Integer
F2I reg1, composite	Float to Integer
F2F reg1, composite	Float to Float
MUFU reg1, reg2	FP Multi-Function Operator
FLO reg1, composite	Integer Find Leading One
RRO reg1, composite	FP Range Reduction Operator
ISCADD32I reg1, reg2, 32-bit lit, 5-bit lit	((undocumented))
FADD32I reg1, reg2, 32-bit lit	((undocumented))
FMUL32I reg1, reg2, 32-bit lit	((undocumented))
IADD32I reg1, reg2, 32-bit lit	((undocumented))
IMUL32I reg1, reg2, 32-bit lit	((undocumented))
LOP32I reg1, reg2, 32-bit lit	((undocumented))
FADD reg1, reg2, composite	FP32 Add
FMUL reg1, reg2, composite	FP32 Multiply
DADD reg1, reg2, composite	FP64 Add
DMUL reg1, reg2, composite	FP64 Multiply
IADD reg1, reg2, composite	Integer Add
ISUB reg1, reg2 <-> composite	((undocumented))
IMUL reg1, reg2, composite	Integer Multiply
LOP reg1, reg2, composite	Integer Logic Op
SHL reg1, reg2, composite	Integer Shift Left
SHR reg1, reg2, composite	Integer Shift Right
BFE reg1, reg2, composite	Integer Bit Field Extract
POPC reg1, reg2, reg3	Population Count
FFMA reg1, reg2, composite <-> reg5	FP32 Fused Multiply Add
DFMA reg1, reg2, composite <-> reg5	FP64 Fused Multiply Add
IMAD reg1, reg2, composite <-> reg5	Integer Multiply Add
FCMP reg1, reg2, composite, reg5	FP32 Compare
ICMP reg1, reg2, composite, reg5	Integer Compare and Select
BFI reg1, reg2, composite, reg5	Integer Bit Field Insert
PRMT reg1, reg2, composite, reg5	Permute
FSET reg1, reg2, composite, p5	FP32 Set
ISET reg1, reg2, composite, p5	Integer Set
PSET reg1, p3/p5, p4, p5/p3	Predicate Set
FMNMX reg1, reg2, composite, p5	FP32 Minimum/Maximum
DMNMX reg1, reg2, composite, p5	FP64 Minimum/Maximum
IMNMX reg1, reg2, composite, p5	Integer Minimum/Maximum
VMNMX reg1, reg2, reg3, reg5	((undocumented))
ISCADD reg1, reg2, composite, 5-bit lit	Integer Scaled Add
VADD reg1, reg2, reg/lit composite, reg5	((undocumented))
BRX reg2, 24-bit lit	Branch to Relative Indexed Address
SSY 24-bit lit	Set Sync Relative Address
BRA 24-bit lit	Branch to Relative Address
PCNT 24-bit lit	Pre-Continue Relative Address
PBK 24-bit lit	Pre-Break Relative Address
CAL 24-bit lit	Call to Relative Address
JCAL 24-bit lit	Call to Absolute Address
TEXDEPBAR composite	((undocumented))
DSETP p2, p1, reg2, composite, p5	FP64 Set Predicate
ISETP p2, p1, reg2, composite, p5	Integer Set Predicate
FSETP p2, p1, reg2, composite, p5	FP32 Set Predicate
CSETP p2, p1, CC, p5	CC Set Predicate

PSETP p2, p1, p3/p5, p4, p5/p3	Predicate Set Predicate
STSCUL p, [reg2 + ??lit], reg1	Store to Shared Memory Conditionally and Unlock
LDLK p, reg1, [reg2 + 32-bit lit]	Load and Lock
LDSLK p7, reg1, [reg2 + 24-bit lit]	Load from Shared Memory and Lock
LD reg1, [reg2 + 32-bit lit]	Load from Memory
LDU reg1, [reg2 + 32-bit lit]	Load Uniform
LDL reg1, [reg2 + 24-bit lit]	Load from Local Memory
LDS reg1, [reg2 + 24-bit lit]	Load from Shared Memory and Lock
STL [reg2 + 24-bit lit], reg1	Store to Local Memory
STS [reg2 + 24-bit lit], reg1	Store to Shared Memory and Unlock
STSUL [reg2 + 24-bit lit], reg1	Store to Shared Memory and Unlock
ST [reg2 + 32-bit lit], reg1	Store to Memory
STUL [reg2 + 32-bit lit], reg1	Store and Unlock
RED [reg2 + 32-bit lit], reg1	Atomic Memory Reduction Operation
ATOM too complicated	Atomic Memory Operation
LD_LDU reg4, reg1, [reg2 + 6-bit lit], [reg3 + 6-bit lit]	A combination of a generic load LD with a load uniform LDU
NOP (CC)	No Operation
CONT	Continue in Loop
BRK	Break from Loop
RET	Return from Call
EXIT	Exit Program
BPT	Breakpoint/Trap
MEMBAR	Memory Barrier

Instructions for Compute Capability 3.x

Instruction	Official Description
FCHK p2, reg2, composite	FP32 Division Test
VOTE (reg1), p6, p5	Query condition across threads
LDC reg1, const_mem_with_reg2	Load Constant
MOV reg1, composite, (5-bit lit)	Move
MOV32I reg1, 32-bit lit, (4-bit lit)	((undocumented))
S2R reg1, special reg	Move Special Register to Register
P2R reg1, PR, reg2, composite	Predicate to Register
R2P PR, reg1, composite	Register to Predicate
SEL reg1, reg2, composite, p5	Conditional Select/Move
TEX reg1, reg2, (reg3), 14-bit lit, tex, 4-bit lit	Texture Fetch
TLD reg1, reg2, (reg3), 15-bit lit tex, 4-bit lit	Texture Load
F2F reg1, composite	Floating Point To Floating Point Conversion
F2I reg1, composite	Floating Point To Integer Conversion
I2F reg1, composite	Integer To Floating Point Conversion
I2I reg1, composite	Integer To Integer Conversion
MUFU reg1, reg2	FP32 Multi Function Operation
FLO reg1, composite	Find Leading One
RRO reg1, composite	Range Reduction Operator FP
FADD32I reg1, reg2, 32-bit lit	((undocumented))
FMUL32I reg1, reg2, 32-bit lit	((undocumented))
IADD32I reg1, reg2, 32-bit lit	((undocumented))
IMUL32I reg1, reg2, 32-bit lit	((undocumented))
LOP32I reg1, reg2, 32-bit lit	((undocumented))
BFE reg1, reg2, composite	Integer Bit Field Extract
IADD reg1, reg2, composite	Integer Addition
IMUL reg1, reg2, composite	Integer Multiply

ISUB reg1, composite, reg2	((undocumented))
ISUB reg1, reg2, composite	((undocumented))
DADD reg1, reg2, composite	FP64 Add
DMUL reg1, reg2, composite	FP64 Multiply
FADD reg1, reg2, composite	FP32 Add
FMUL reg1, reg2, composite	FP32 Multiply
LOP reg1, reg2, composite	Integer Logic Op
POPC reg1, reg2, composite	Population Count
SHL reg1, reg2, composite	Integer Shift Left
SHR reg1, reg2, composite	Integer Shift Right
ISET reg1, reg2, composite, p5	Integer Set
PSET reg1, p3, p4, p5	Predicate Set
DFMA reg1, reg2, composite <-> reg4	FP64 Fused Multiply Add
FFMA reg1, reg2, composite <-> reg4	FP32 Fused Multiply Add
IMAD reg1, reg2, composite <-> reg4	Integer Multiply And Add
DMNMX reg1, reg2, composite, p5	FP64 Minimum/Maximum
FMNMX reg1, reg2, composite, p5	FP32 Minimum/Maximum
IMNMX reg1, reg2, composite, p5	Integer Minimum/Maximum
BFI reg1, reg2, composite, reg4	Integer Bit Field Insert
PRMT reg1, reg2, composite, reg4	Permute
SHF reg1, reg2, composite, reg4	Integer Funnel Shift
FCMP reg1, reg2, composite, reg4	FP32 Compare to Zero and Select Source
ICMP reg1, reg2, composite, reg4	Integer Compare to Zero and Select Source
ISCADD reg1, reg2, composite, 5-bit lit	Scaled Integer Addition
ISCADD32I reg1, reg2, 32-bit lit, 5-bit lit	((undocumented))
BRA (CC), const/lit composite	Relative Branch
BRX (CC), const_mem_with_reg2	Relative Branch Indirect
BRX (CC), reg2, 24-bit lit	Relative Branch Indirect
CAL const/lit composite	Relative Call
PBK const/lit composite	Pre-Break
PCNT const/lit composite	Pre-continue
SSY const/lit composite	Set Synchronization Point
TEXDEPBAR 6-bit lit	((undocumented))
DSETP p2, p1, reg2, composite, p5	FP64 Set Predicate
FSETP p2, p1, reg2, composite, p5	FP32 Set Predicate
ISETP p2, p1, reg2, composite, p5	Integer Set Predicate
CSETP p2, p1, CC, p5	CC Set Predicate
PSETP p2, p1, p3, p4, p5	Predicate Set Predicate
LDG reg1, [reg2 + 11-bit lit]	Non-coherent Global Memory Load
LD reg1, [reg2 + 32-bit lit]	Load from Memory
LDL reg1, [reg2 + 24-bit lit]	Load from Local Memory
LDS reg1, [reg2 + 24-bit lit]	Load from Shared Memory
ST [reg2 + 32-bit lit], reg1	Store to Memory
STL [reg2 + 24-bit lit], reg1	Store to Local Memory
STS [reg2 + 24-bit lit], reg1	Store to Shared Memory
BPT	Breakpoint/Trap
BRK (CC)	Break
CONT (CC)	Continue
EXIT (CC)	Exit Program
MEMBAR	Memory Barrier
NOP (CC)	No Operation
RET (CC)	Return From Subroutine

Instructions for Compute Capabilities 5.x and 6.x

Instruction	Official Description
FCHK p2, reg2, composite	Single Precision FP Divide Range Check
VOTE (reg1), p6, p5	Vote Across SIMD Thread Group
LDC reg1, const_mem_with_reg2	Load Constant
MOV reg1, composite, (5-bit lit)	Move
MOV32I reg1, 32-bit lit, (4-bit lit)	((undocumented))
S2R reg1, special_reg	Move Special Register to Register
SEL reg1, reg2, composite, p5	Select Source with Predicate
TEX reg1, reg2, (reg3), 13-bit lit, 3-bit tex, 4-bit lit	Texture Fetch
TEXS reg4, reg1, reg2, (reg3), 13-bit lit, 1-bit tex, channel	Texture Fetch with scalar/non-vec4 source/destinations
TLDS reg4, reg1, reg2, (reg3), 13-bit lit, 1-bit tex, channel	Texture Load with scalar/non-vec4 source/destinations
I2I reg1, composite	Integer To Integer Conversion
I2F reg1, composite	Integer To Floating Point Conversion
F2I reg1, composite	Floating Point To Integer Conversion
F2F reg1, composite	Floating Point To Floating Point Conversion
MUFU reg1, reg2	Multi Function Operation
FLO reg1, composite	Find Leading One
POPC reg1, composite	Population count
RRO reg1, composite	Range Reduction Operator FP
FADD32I reg1, reg2, 32-bit lit	((undocumented))
FMUL32I reg1, reg2, 32-bit lit	((undocumented))
IADD32I reg1, reg2, 32-bit lit	((undocumented))
LOP32I reg1, reg2, 32-bit lit	((undocumented))
BFE reg1, reg2, composite	Bit Field Extract
DADD reg1, reg2, composite	FP64 Add
DMUL reg1, reg2, composite	FP64 Multiply
FADD reg1, reg2, composite	FP32 Add
FMUL reg1, reg2, composite	FP32 Multiply
IADD reg1, reg2, composite	Integer Addition
SHL reg1, reg2, composite	Shift Left
SHR reg1, reg2, composite	Shift Right
LOP (p7), reg1, reg2, composite	Logic Operation
BFI reg1, reg2, composite, reg5	Bit Field Insert
PRMT reg1, reg2, composite, reg5	Permute Register Pair
DFMA reg1, reg2, composite <-> reg5	FP64 Fused Mutiply Add
FFMA reg1, reg2, composite <-> reg5	FP32 Fused Multiply and Add
FCMP reg1, reg2, composite, reg5	FP32 Compare to Zero and Select Source
IADD3 reg1, reg2, composite, reg5	3-input Integer Addition
ICMP reg1, reg2, composite, reg5	Integer Compare to Zero and Select Source
XMAD reg1, reg2, composite <-> reg5	Integer Short Multiply Add
SHF reg1, reg2, composite, reg5	Funnel Shift
VABSDIFF reg1, reg2, composite, reg5	((undocumented))
IMNMX reg1, reg2, composite, p5	Integer Minimum/Maximum
DMNMX reg1, reg2, composite, p5	FP64 Minimum/Maximum
FMNMX reg1, reg2, composite, p5	FP32 Minimum/Maximum
ISCADD reg1, reg2, composite, 5-bit lit	Scaled Integer Addition
ISCADD32I reg1, reg2, 32-bit lit, 5-bit lit	((undocumented))
DSET reg1, reg2, composite, p5	FP64 Compare And Set
FSET reg1, reg2, composite, p5	FP32 Compare And Set
ISET reg1, reg2, composite, p5	Integer Compare And Set
PSET reg1, p3, p4, p5	Combine Predicates and Set
LOP3 (p7), reg1, reg2, reg3, reg5, 8-bit lit	3-input Logic Operation

LOP3 reg1, reg2, composite, reg5, 9-bit lit	3-input Logic Operation
LEA (p7), reg1, reg2, 19-bit lit, (5-bit lit)	Compute Effective Address
LEA (p7), reg1, reg2, reg3, reg5, (5-bit lit)	Compute Effective Address
LEA (p7), reg1, reg2, reg3, (5-bit lit)	Compute Effective Address
PSETP p2, p1, p3, p4, p5	Combine Predicates and Set Predicate
DSETP p2, p1, reg2, composite, p5	FP64 Compare And Set Predicate
FSETP p2, p1, reg2, composite, p5	FP32 Compare And Set Predicate
ISETP p2, p1, reg2, composite, p5	Integer Compare And Set Predicate
CSETP p2, p1, CC, p5	Test Condition Code and Set Predicate
BAR 8-bit lit, (19-bit lit)	Barrier Synchronization
BRA (CC), const/lit composite	Relative Branch
CAL const/lit composite	Relative Call
PBK const/lit composite	Pre-Break
PCNT const/lit composite	Pre-continue
SSY const/lit composite	Set Synchronization Point
JCAL large const/lit composite	Absolute Call
BRX (CC), const_mem_with_reg2	Relative Branch Indirect
BRX (CC), reg2, 24-bit lit	Relative Branch Indirect
DEPBAR (sb, 6-bit lit), (bitfield)	((undocumented))
LD reg1, [reg2 + 32-bit lit], p8	Load from generic Memory
ST [reg2 + 32-bit lit], reg1, p9	Store to generic Memory
LDG reg1, [reg2 + 24-bit lit]	Load from Global Memory
LDL reg1, [reg2 + 24-bit lit]	Load within Local Memory Window
LDS reg1, [reg2 + 24-bit lit]	Local within Shared Memory Window
STG [reg2 + 24-bit lit], reg1	Store to global Memory
STL [reg2 + 24-bit lit], reg1	Store within Local or Shared Window
STS [reg2 + 24-bit lit], reg1	Store within Local or Shared Window
RED [reg2 + 20-bit lit], reg1	Reduction Operation on generic Memory
NOP (CC), 16-bit lit	No Operation
BRK (CC)	Break
CONT (CC)	Continue
EXIT (CC)	Exit Program
RET (CC)	Return From Subroutine
SYNC (CC)	Converge threads after conditional branch

Appendix - Opcodes

In this Appendix we list the opcodes for decoded instructions on most of the distinct ISAs we have explored. We omit Compute Capabilities 2.x and 3.0, which have previously had their opcodes detailed in Asfermi [1].

The first column in each table is the opcode's string and the operand types. We use 'reg' for general register operands, 'p' for predicate registers, 'lit' for literal (hexadecimal or decimal) values, 'mem' for memory, 'const' for constant memory, 'tex' for texture shapes, 'channel' for texture channels, 'bitfield' for bitfield operands, and 'misc' for operands without a clear type.

The second column is the binary, with zeros and ones for opcode bits, and the character 'x' in place of bits that are not controlled by the opcode, with the least significant bit on the left. Bits which affect the types of the operands are treated as opcode bits.

Opcodes on Compute Capability 3.x

Table 4. Opcode lookup table: opcodeTable3x

Instruction	Bits
MOV reg reg lit	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1100100111
MOV reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1111xxxxxxxx1100100111
MOV reg const lit	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1100100110
MOV reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1111xxxxxxxx1100100110
MOV32I reg lit lit	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx000101110
MOV32I reg lit	01xxxxxxxxxxxx1111xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx000101110
LD reg mem	00xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx100100011
LDL reg mem	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001011110
LDS reg mem	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1001011110
LDC reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0100111110
ST mem reg	00xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx100100111
STL mem reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0101011110
STS mem reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1101011110
FADD reg reg lit	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1101010011
FADD reg reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1101000110
FADD reg reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1101000111
FADD32I reg reg lit	00xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx000000010
FMUL reg reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1011000110
FMUL reg reg lit	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1011000011
FMUL reg reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1011000111
FMUL32I reg reg lit	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx000000100
FFMA reg reg const reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx00110010
FFMA reg reg reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000110011
FFMA reg reg lit reg	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000111001
FFMA reg reg reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx000110001
FSETP p p reg const p	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx110111010
FSETP p p reg reg p	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110111011
FSETP p p reg lit p	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110101101
DSETP p p reg lit p	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000101101
DSETP p p reg const p	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx000111010
DSETP p p reg reg p	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1000111011
FCMP reg reg reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010111011
FCMP reg reg const reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx010111010
MUFU reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000100001
DADD reg reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0111000110
DADD reg reg lit	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0111000011
DADD reg reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0111000111
DMUL reg reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000100111
DMUL reg reg lit	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000110011

DMUL reg reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000100110
DFMA reg reg lit reg	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0111001101
DFMA reg reg reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx111011001
DFMA reg reg const reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx111011010
DFMA reg reg reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0111011011
IADD reg reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0100000110
IADD reg reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0100000111
IADD reg reg lit	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0100000011
IADD32I reg reg lit	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx000000010
IMNMX reg reg lit p	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010000011
IMNMX reg reg reg p	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010000111
IMNMX reg reg const p	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010000110
IMUL reg reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110000110
IMUL reg reg lit	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110000011
IMUL reg reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110000111
IMUL32I reg reg lit	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx001110100
IMAD reg reg lit reg	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010000101
IMAD reg reg const reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx10001010
IMAD reg reg reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010001011
IMAD reg reg reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx010001001
ISCADD reg reg lit lit	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1100000011
ISCADD reg reg reg lit	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1100000111
ISCADD reg reg const lit	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1100000110
ISET reg reg const p	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx101011010
ISETP p p reg lit p	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1011001101
ISETP p p reg reg p	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1011011011
ISETP p p reg const p	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx011011010
ICMP reg reg lit reg	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1001011101
ICMP reg reg reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001011011
ICMP reg reg const reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx001011010
I2F reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110100110
I2F reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110100111
I2I reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001100110
I2I reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001100111
I2I reg lit	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001110011
F2I reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0110100111
F2I reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0110100110
F2F reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1010100111
F2F reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1010100110
LOP reg reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001000110
LOP reg reg lit	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001000011
LOP reg reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001000111
LOP32I reg reg lit	00xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx000000100
SHL reg reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1001000110
SHR reg reg lit	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1010000011
SHR reg reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1010000110
BFE reg reg lit	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000000011
BFE reg reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000000110
BFI reg reg lit reg	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0111101101
SEL reg reg reg p	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010100111
SEL reg reg lit p	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010100011
SEL reg reg const p	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010100110
SSY const	00xxxxx1xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx100101000
SSY lit	00xxxxx0xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0100101000

BRA const	00111101xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx001001000
BRA misc const	00xxxxx1xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx001001000
BRA lit	00111100xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001001000
BRA misc lit	00xxxxx0xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001001000
BRX misc reg lit	00xxxxx0xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0101001000
BRX reg lit	00111100xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0101001000
BRX const	00111101xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx101001000
BRX misc const	00xxxxx1xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx101001000
PCNT const	00xxxxx1xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx110101000
PCNT lit	00xxxxx0xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0110101000
CONT misc	00xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0101011000
CONT	0011110xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0101011000
PBK lit	00xxxxx0xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010101000
PBK const	00xxxxx1xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx010101000
BRK misc	00xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001011000
BRK	0011110xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001011000
CAL const	00xxxxx1xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx011001000
CAL lit	00xxxxx0xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0011001000
RET misc	00xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010011000
RET	0011110xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010011000
EXIT misc	00xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000011000
EXIT	0011110xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000011000
NOP misc lit	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0110100001
NOP lit	01xxxxxxxx11110xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0110100001
BPT lit	00xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000000000
S2R reg sreg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1001100001
PSETP p p p p p	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0100100001
PSET reg p p p	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1000100001
FLO reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0110000111
FLO reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0110000110
P2R reg misc reg lit	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1001100011
P2R reg misc reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1001100110
R2P misc reg lit	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx000101100011
R2P misc reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx100101100110
TEX reg reg lit tex lit	10xxxxxxxxxxxxxxxxxxxx11111111xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx110
TEX reg reg reg lit tex lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx110
TEXDEPBAR lit	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0011101110
RRO reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0100100111
RRO reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0100100110
PRMT reg reg lit reg	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001101101
DMNMX reg reg reg p	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0101000111
DMNMX reg reg const p	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0101000110
FMNMX reg reg lit p	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0011000011
FMNMX reg reg reg p	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0011000111
FMNMX reg reg const p	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0011000110
VOTE p p	0111111111xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1101100001
VOTE reg p p	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1101100001
POPC reg reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1000000111
POPC reg reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1000000110
MEMBAR	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1100111110
CSETP p p misc p	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010100001
ISCADD32I reg reg lit lit	00xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0xxxxx101
TLD reg reg lit tex lit	01xxxxxxxxxxxxxxxxxxxx11111111xxxxxxxxxxxxxxxxxxxxxxxxxxxx1110
TLD reg reg reg lit tex lit	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110

SHF reg reg reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1111111011
SHF reg reg lit reg	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1111101101
FCHK p reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1000100111
FCHK p reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1000100110
ISUB reg const reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx01x0100000110
ISUB reg reg const	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx10x0100000110
ISUB reg reg reg	01xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx10x0100000111
ISUB reg lit reg	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx01x0100000011
ISUB reg reg lit	10xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx10x0100000011
LDG reg mem	10xxxxxxxxxxxxxxxxxxxx111111110xx1111xxx000100011xxxxxxxxxxx110

Opcodes on Compute Capabilities 5.x and 6.x

Table 5. Opcode lookup table: opcodeTable5x6x

Instruction	Bits
MOV reg const	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1111xxxxxxxx1100100110010
MOV reg const lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1100100110010
MOV reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1111xxxxxxxx1100100111010
MOV reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1100100111010
MOV32I reg lit	xxxxxxxxxxxx1111xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx000010000000
MOV32I reg lit lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx000010000000
LD reg mem p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1001xxxxxxx1
LDL reg mem	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001011110111
LDS reg mem	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1001011110111
LDC reg const	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0100111110111
ST mem reg p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx101
STL mem reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0101011110111
STS mem reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1101011110111
FADD reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1101000111010
FADD reg reg const	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1101000110010
FADD reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1101010011100
FADD32I reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx000000010000
FMUL reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1011000111010
FMUL reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1011000011100
FMUL reg reg const	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1011000110010
FMUL32I reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx000001111000
FFMA reg reg reg const	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000110001010
FFMA reg reg lit reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000101001100
FFMA reg reg const reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000110010010
FFMA reg reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000110011010
FSET reg reg lit p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1001100001100
FSET reg reg reg p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1000100011010
FSETP p p reg lit p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110101101100
FSETP p p reg reg p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110111011010
FSETP p p reg const p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0110111010010
DSETP p p reg reg p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1000111011010
DSETP p p reg lit p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000101101100
DSETP p p reg const p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1000111010010
FCMP reg reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010111011010
FCMP reg reg const reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1010111010010
FCMP reg reg lit reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1010101101100
MUFU reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000100001010
DADD reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0111000111010

DADD reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0111000011100
DADD reg reg const	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0111000110010
DMUL reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000100111010
DMUL reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000110011100
DMUL reg reg const	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000100110010
DFMA reg reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0111011011010
DFMA reg reg lit reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0111001101100
DFMA reg reg const reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0111011010010
DFMA reg reg reg const	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0111011001010
IADD reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0100000111010
IADD reg reg const	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0100000110010
IADD reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0100000011100
IADD32I reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx000000111000
IMNMX reg reg const p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010000110010
IMNMX reg reg reg p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010000111010
IMNMX reg reg lit p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010000011100
ISCADD reg reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1100000111010
ISCADD reg reg lit lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1100000011100
ISCADD reg reg const lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1100000110010
ISET reg reg reg p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0101011011010
ISET reg reg const p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1101011010010
ISET reg reg lit p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0101001101100
ISETP p p reg reg p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1011011011010
ISETP p p reg lit p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1011001101100
ISETP p p reg const p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1011011010010
ICMP reg reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001011011010
ICMP reg reg lit reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1001011101100
ICMP reg reg const reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1001011010010
I2F reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110100111010
I2F reg const	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110100110010
I2I reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0011110011100
I2I reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0011100111010
I2I reg const	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0011100110010
F2I reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0110100111010
F2F reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1010100111010
F2F reg const	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1010100110010
F2F reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1010100011100
LOP reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110001000011100
LOP p reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1000111010
LOP reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110001000111010
LOP p reg reg const	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1000110010
LOP reg reg const	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110001000110010
LOP p reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1000011100
LOP32I reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx000000100000
SHL reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1001000011100
SHL reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1001000111010
SHR reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1010000011100
SHR reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1010000111010
BFE reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000000011100
BFI reg reg lit reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0111101101100
SEL reg reg reg p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010100111010
SEL reg reg lit p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010100011100
SEL reg reg const p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010100110010
SSY lit	xxxxx0xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0100101000111

SSY const	xxxxx1xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0100101000111
BRA const	111101xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001001000111
BRA misc const	xxxxx1xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001001000111
BRA lit	111100xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001001000111
BRA misc lit	xxxxx0xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001001000111
BRX const	111101xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0101001000111
BRX misc const	xxxxx1xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0101001000111
BRX reg lit	111100xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0101001000111
BRX misc reg lit	xxxxx0xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0101001000111
PCNT lit	xxxxx0xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0110101000111
PCNT const	xxxxx1xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0110101000111
CONT	11110xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0101011000111
CONT misc	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0101011000111
PBK lit	xxxxx0xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010101000111
PBK const	xxxxx1xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010101000111
BRK	11110xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001011000111
BRK misc	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001011000111
CAL lit	xxxxx0xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0011001000111
CAL const	xxxxx1xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0011001000111
RET	11110xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010011000111
RET misc	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010011000111
EXIT	11110xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000011000111
EXIT misc	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000011000111
NOP lit	xxxxxxxx11110xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0110100001010
NOP misc lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0110100001010
NOP	xxxxxxxx11110xxxxxxx0000000000000000xxxxxxxxxxxxxxx0110100001010
NOP misc	xxxxxxxxxxxxxxxxxxx0000000000000000xxxxxxxxxxxxxxx0110100001010
BAR lit	xxxxxxxxxxxxxxxxxxx00000000000000000011101100000010101000011111
BAR lit lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110110000001010100001111
S2R reg sreg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1001100001111
PSETP p p p p p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0100100001010
PSET reg p p p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1000100001010
FLO reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0110000111010
TEX reg reg lit tex lit	xxxxxxxxxxxxxxxxxxx11111111xxxxxxxxxxxxxxxxxxxxx1110000000011
TEX reg reg reg lit tex lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110000000011
RRO reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0100100111010
PRMT reg reg lit reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001101101100
DMNMX reg reg reg p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0101000111010
DMNMX reg reg lit p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0101000011100
FMNMX reg reg reg p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0011000111010
FMNMX reg reg lit p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0011000011100
RED mem reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1111111010111
VOTE reg p p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1101100001010
VOTE p p	11111111xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1101100001010
POPC reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1000000111010
CSETP p p misc p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010100001010
ISCADD32I reg reg lit lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0xxxxx101000
SHF reg reg lit reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1111101101100
SHF reg reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1111111011010
FCHK p reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1000100111010
JCAL lit	xxxxx0xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx010001000111
JCAL const	xxxxx1xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010001000111
LDG reg mem	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0101101110111
ATOM reg mem reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0111101110111

XMAD reg reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000011011010
XMAD reg reg lit reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0010001101100
XMAD reg reg reg const	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0110010001010
XMAD reg reg const reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000001110010
SYNC	11110xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1111100001111
SYNC misc	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1111100001111
STG mem reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1101101110111
IADD3 reg reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001100111010
IADD3 reg reg lit reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001100011100
IADD3 reg reg const reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001100110010
VABSDIFF reg reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx10010000101010
VABSDIFF reg reg lit reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx00010000101010
DEPBAR sb lit bitfield	xxxxxxxxxxxxxxxxxxxxxxxxxxxx1xxxxxxxxxxxxxxxxxxx0111100001111
DEPBAR sb lit	000000xxxxxxxxxxxxxxxxxxxxxx1xxxxxxxxxxxxxxxxxxx0111100001111
DEPBAR bitfield	xxxxxxxxxxxxxxxxxxxxxxxxxxxx0xxxxxxxxxxxxxxxxxxx0111100001111
DEPBAR	000000xxxxxxxxxxxxxxxxxxxxxx0xxxxxxxxxxxxxxxxxxx0111100001111
LOP3 reg reg reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110011111011010
LOP3 p reg reg reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx11111011010
LOP3 reg reg const reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1000000
LOP3 reg reg lit reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx111100
TLDS reg reg reg lit tex channel	xxxxxxxxxxxxxxxxxxx11111111xxxxxxxxxxxxxxxxxxxxxxxxxxx1011011
TLDS reg reg reg reg lit tex channel	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1011011
TEXS reg reg reg lit tex channel	xxxxxxxxxxxxxxxxxxx11111111xxxxxxxxxxxxxxxxxxxxxxxxxxx0011011
TEXS reg reg reg reg lit tex channel	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0011011
LEA reg reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110101111011010
LEA reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx00000xxxx1110101111011010
LEA p reg reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx101111011010
LEA p reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx00000xxxxxxxx101111011010
LEA reg reg reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1111101111011010
LEA reg reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1111101111011010
LEA p reg reg reg reg	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1101111011010
LEA p reg reg reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1101111011010
LEA reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110101101101100
LEA reg reg lit lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1110101101101100
LEA p reg reg lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx101101101100
LEA p reg reg lit lit	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx101101101100
DSET reg reg reg p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1000010011010
DSET reg reg const p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0000010010010
DSET reg reg lit p	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx0001001001100

Special Registers

In this appendix we list all special registers we have observed. The first column is the encoded value. The second column contains the name of the special register, which nearly always begins with the text "SR_". In cases where different architectures disagree regarding the name associated with an encoding, we list the older name in the third column.

Not all special registers are available on every GPU architecture. Additionally, the names produced by the disassembler will vary in formatting depending on its version. In particular, capitalization of the letters may differ, and dots may be replaced by underscores.

Encoding	Special Register	Alternate SR
0	SR_LANEID	
1	SR_CLOCK	
2	SR_VIRTCFG	
3	SR_VIRTID	
4	SR_PM0	
5	SR_PM1	
6	SR_PM2	
7	SR_PM3	
8	SR_PM4	
9	SR_PM5	
10	SR_PM6	
11	SR_PM7	
16	SR_PRIM_TYPE	
17	SR_INVOCATION_ID	
18	SR_Y_DIRECTION	
19	SR_THREAD_KILL	
20	SM_SHADER_TYPE	
21	SR_DIRECTCBEWRITEADDRESSLOW	
22	SR_DIRECTCBEWRITEADDRESSHIGH	
23	SR_DIRECTCBEWRITEENABLED	
24	SR_MACHINE_ID_0	
25	SR_MACHINE_ID_1	
26	SR_MACHINE_ID_2	
27	SR_MACHINE_ID_3	
28	SR_AFFINITY	
29	SR_INVOCATION_INFO	
30	SR_WSCALEFACTOR_XY	
31	SR_WSCALEFACTOR_Z	
32	SR_TID	
33	SR_TID.X	
34	SR_TID.Y	
35	SR_TID.Z	
36	SR_CTA_PARAM	
37	SR_CTAID.X	
38	SR_CTAID.Y	
39	SR_CTAID.Z	
40	SR_NTID	
41	SR_CIRQUEUEINCRMINUSONE	SR_NTID.X
42	SR_NLATC	SR_NTID.Y
43	SR_NTID.Z	
44	SR_GRIDPARAM	
45	SR_NCTAID.X	
46	SR_NCTAID.Y	
47	SR_NCTAID.Z	
48	SR_SWINLO	

49	SR_SWINSZ
50	SR_SMEMSZ
51	SR_SMEMBANKS
52	SR_LWINLO
53	SR_LWINSZ
54	SR_LMEMLOSZ
55	SR_LMEMHIOFF
56	SR_EQMASK
57	SR_LTMASK
58	SR_LEMASK
59	SR_GTMASK
60	SR_GEMASK
61	SR_REGALLOC
62	SR_CTXADDR
64	SR_GLOBALERRORSTATUS
66	SR_WARPERRORSTATUS
67	SR_WARPERRORSTATUSCLEAR
72	SR_PM_HI0
73	SR_PM_HI1
74	SR_PM_HI2
75	SR_PM_HI3
76	SR_PM_HI4
77	SR_PM_HI5
78	SR_PM_HI6
79	SR_PM_HI7
80	SR_CLOCKLO
81	SR_CLOCKHI
82	SR_GLOBALTIMERLO
83	SR_GLOBALTIMERHI
96	SR_HWTASKID
97	SR_CIRCULARQUEUEENTRYINDEX
98	SR_CIRCULARQUEUEENTRYADDRESSLOW
99	SR_CIRCULARQUEUEENTRYADDRESSHIGH

References

[1] Hou, Y., Lai, J., and Mikushin, D. Asfermi: An assembler for the nvidia fermi instruction set. *URL: http://code.google.com/p/asfermi/(accessed: 03.12. 2012)* (2011).