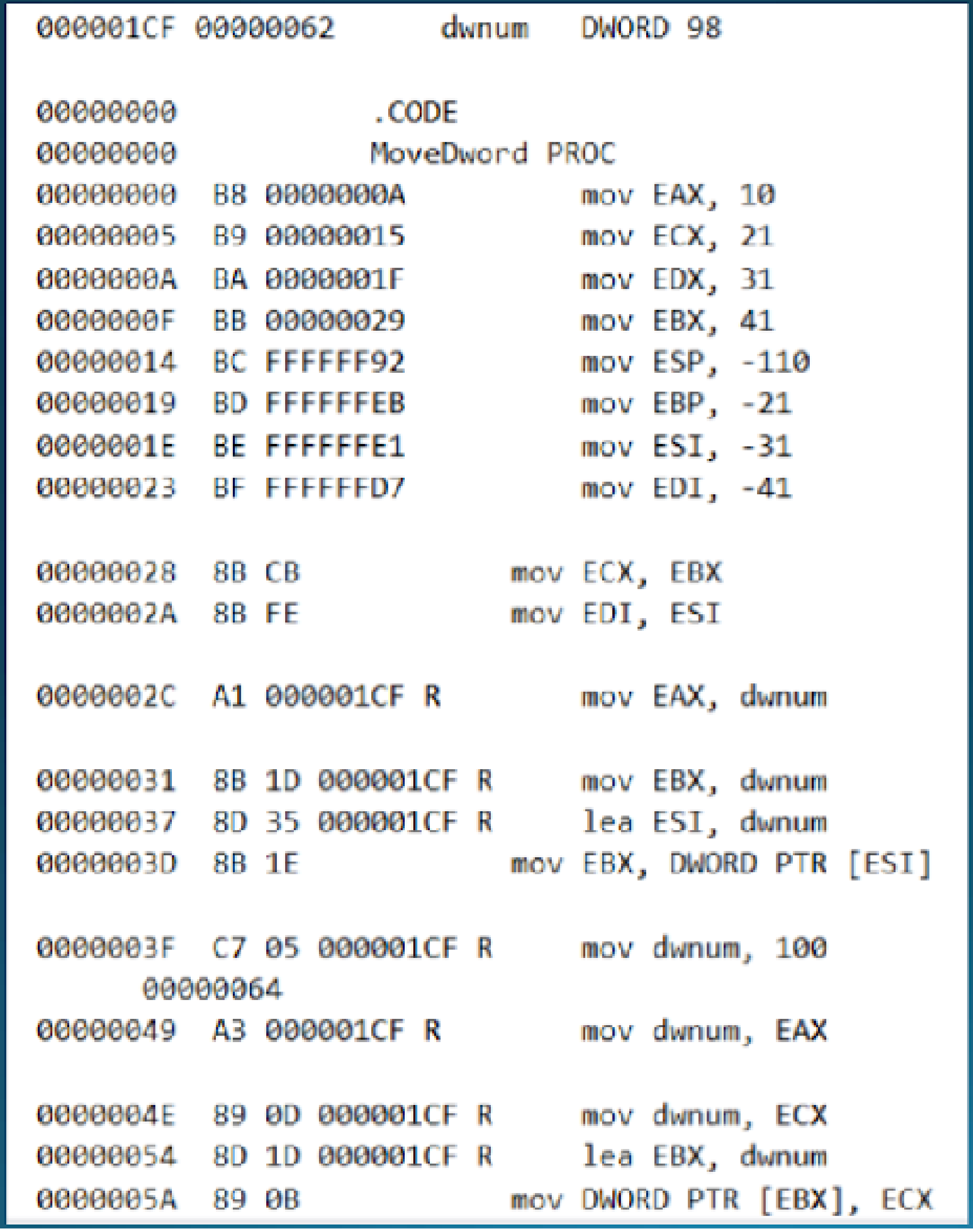
Machine Code – Move a dword

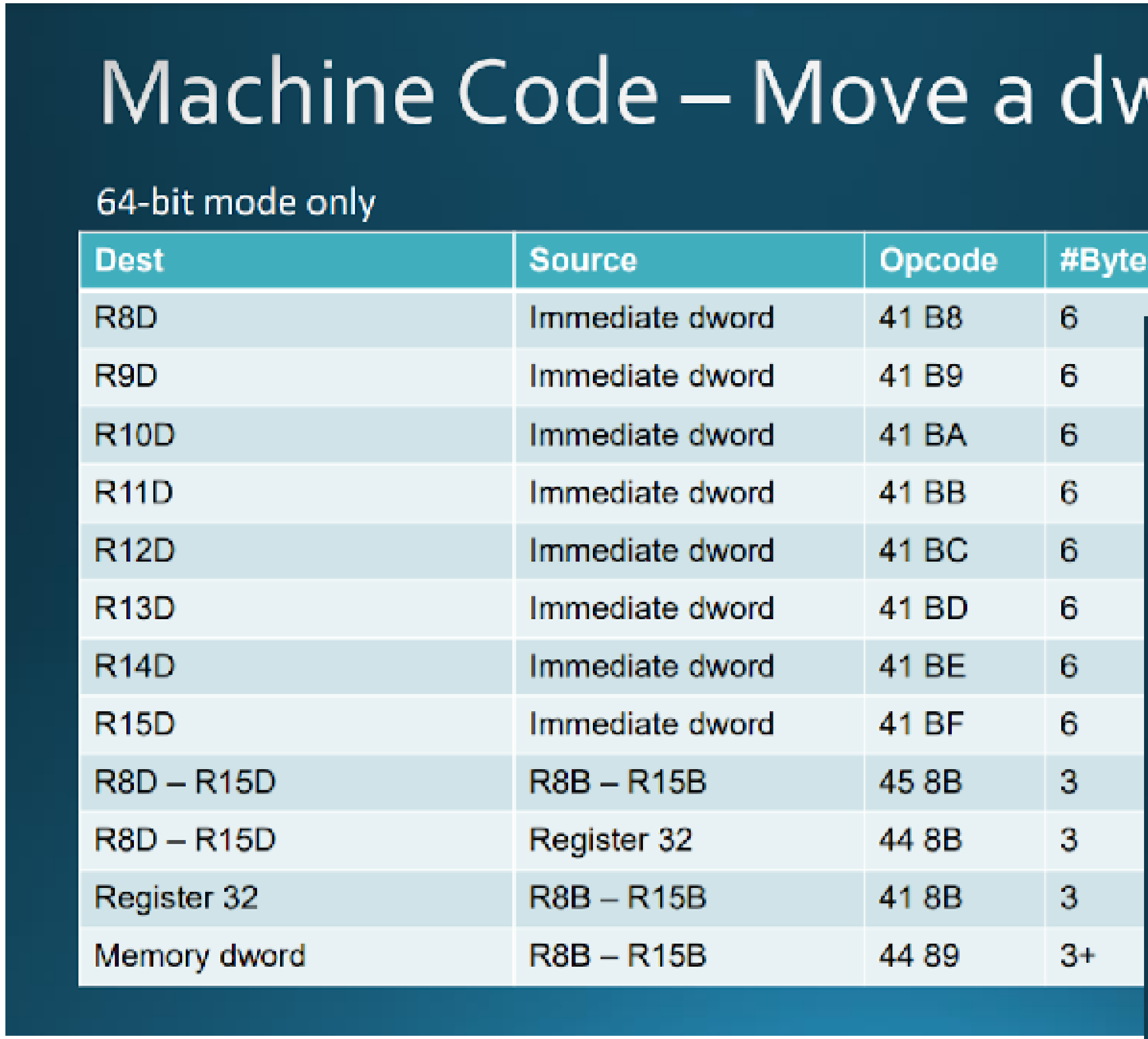
Dest	Source	Opcode	#Bytes in Obj code
EAX	Immediate dword	B8	5
ECX	Immediate dword	B9	5
EDX	Immediate dword	BA	5
EBX	Immediate dword	BB	5
ESP	Immediate dword	BC	5
EBP	Immediate dword	BD	5
ESI	Immediate dword	BE	5
EDI	Immediate dword	BF	5
Register 32 (ABOVE)	Register 32	8B	2
EAX	Memory dword direct address	A1	5
Register 32	Memory dword	8B	2+
Memory dword	Immediate dword	C7	3+
Memory dword, direct	EAX	A3	5
Memory dword	Register 32	89	2+

Machine Code – Move a dwor



```
000001CF 00000062       dwnum    DWORD 98

00000000              .CODE
00000000              MoveDword PROC
00000000   B8 0000000A           mov EAX, 10
00000005   B9 00000015           mov ECX, 21
0000000A   BA 0000001F           mov EDX, 31
0000000F   BB 00000029           mov EBX, 41
00000014   BC FFFFFF92           mov ESP, -110
00000019   BD FFFFFFEB           mov EBP, -21
0000001E   BE FFFFFFE1           mov ESI, -31
00000023   BF FFFFFFD7           mov EDI, -41

00000028   8B CB             mov ECX, EBX
0000002A   8B FE             mov EDI, ESI

0000002C   A1 000001CF R         mov EAX, dwnum

00000031   8B 1D 000001CF R      mov EBX, dwnum
00000037   8D 35 000001CF R      lea ESI, dwnum
0000003D   8B 1E             mov EBX, DWORD PTR [ESI]

0000003F   C7 05 000001CF R      mov dwnum, 100
       00000064
00000049   A3 000001CF R         mov dwnum, EAX

0000004E   89 0D 000001CF R      mov dwnum, ECX
00000054   8D 1D 000001CF R      lea EBX, dwnum
0000005A   89 0B             mov DWORD PTR [EBX], ECX
```

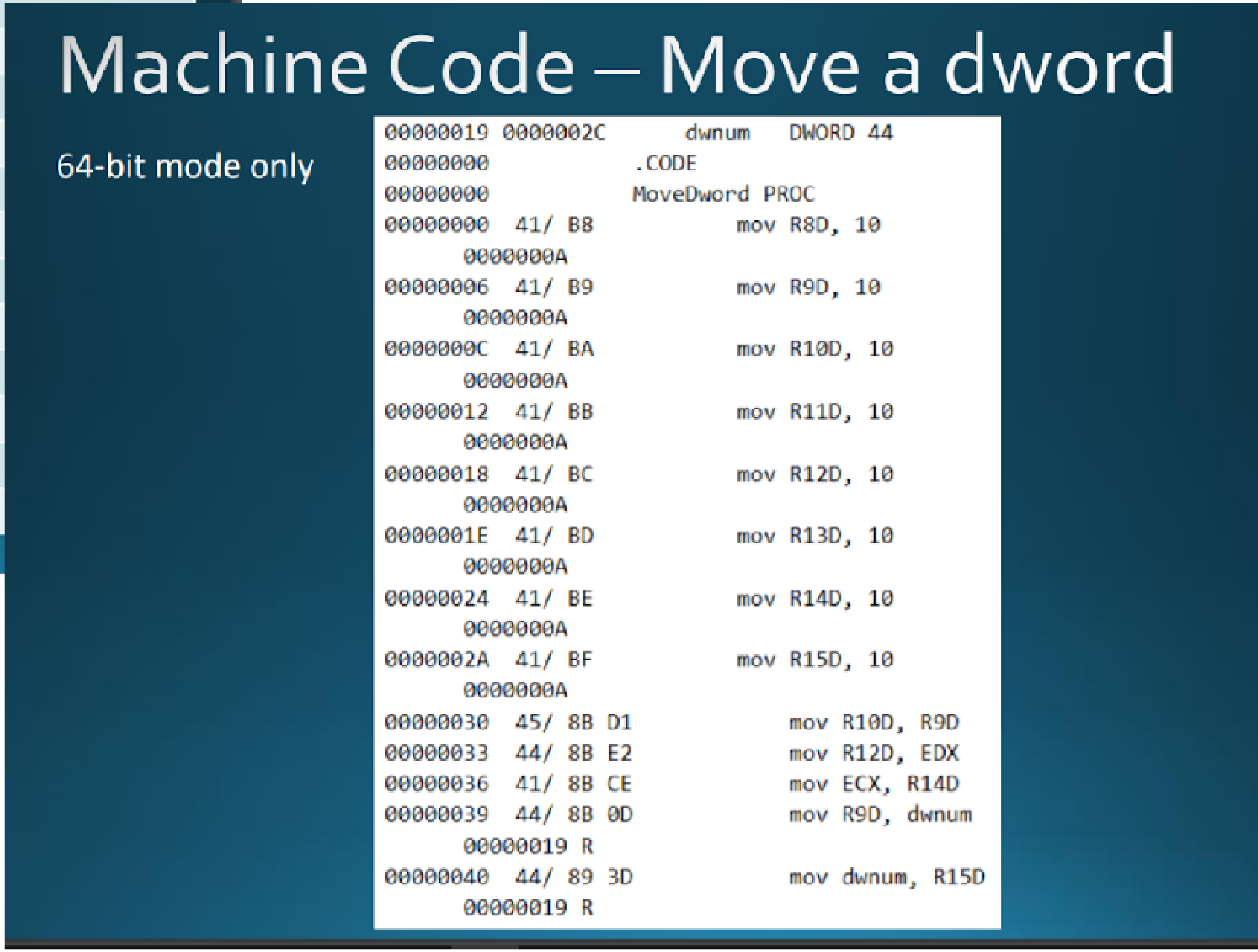
41/38|41 55



Machine Code – Move a dword

64-bit mode only

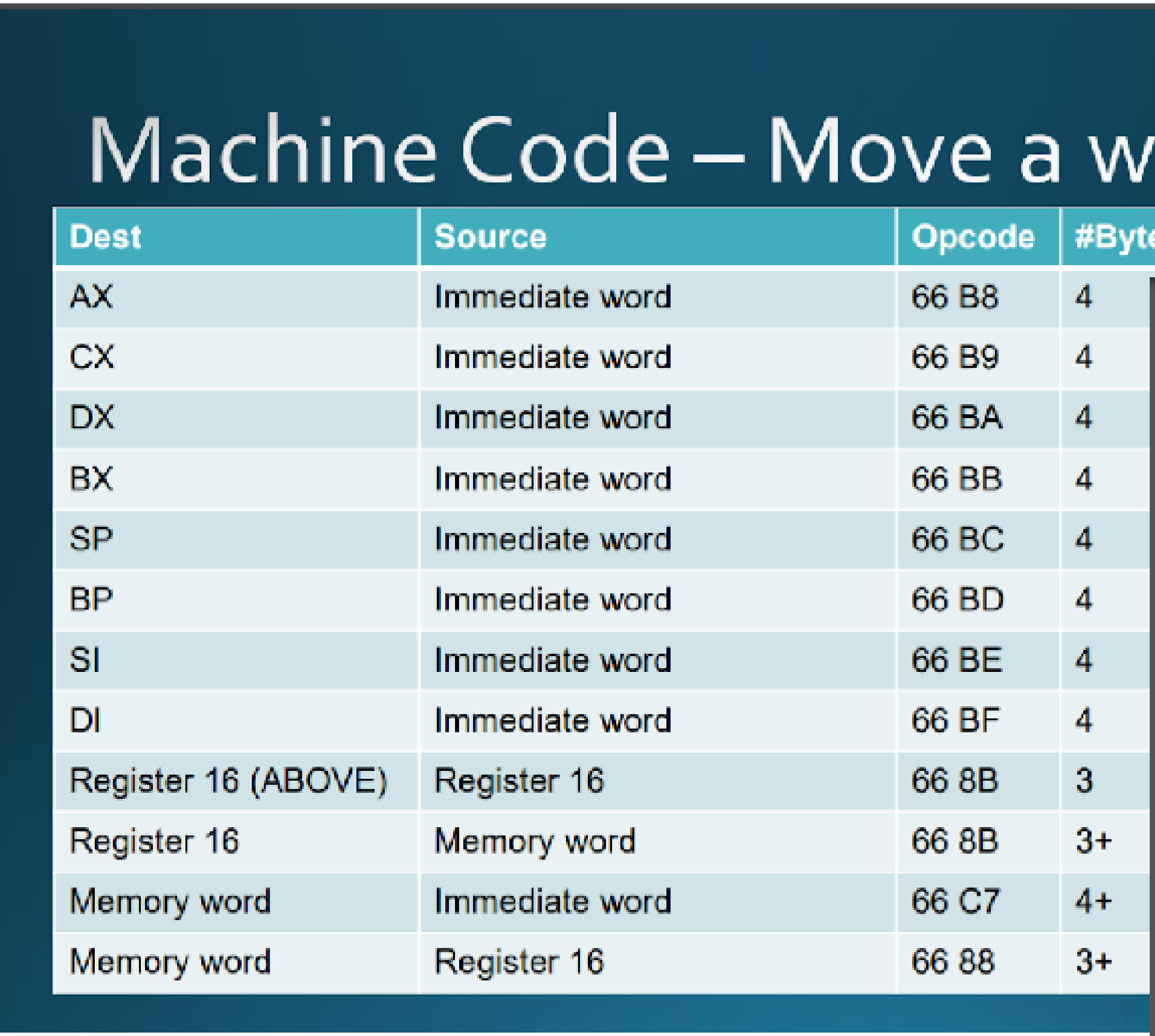
Dest	Source	Opcode	#Bytes in Obj code
R8D	Immediate dword	41 B8	6
R9D	Immediate dword	41 B9	6
R10D	Immediate dword	41 BA	6
R11D	Immediate dword	41 BB	6
R12D	Immediate dword	41 BC	6
R13D	Immediate dword	41 BD	6
R14D	Immediate dword	41 BE	6
R15D	Immediate dword	41 BF	6
R8D – R15D	R8B – R15B	45 8B	3
R8D – R15D	Register 32	44 8B	3
Register 32	R8B – R15B	41 8B	3
Memory dword	R8B – R15B	44 89	3+



Machine Code – Move a dword

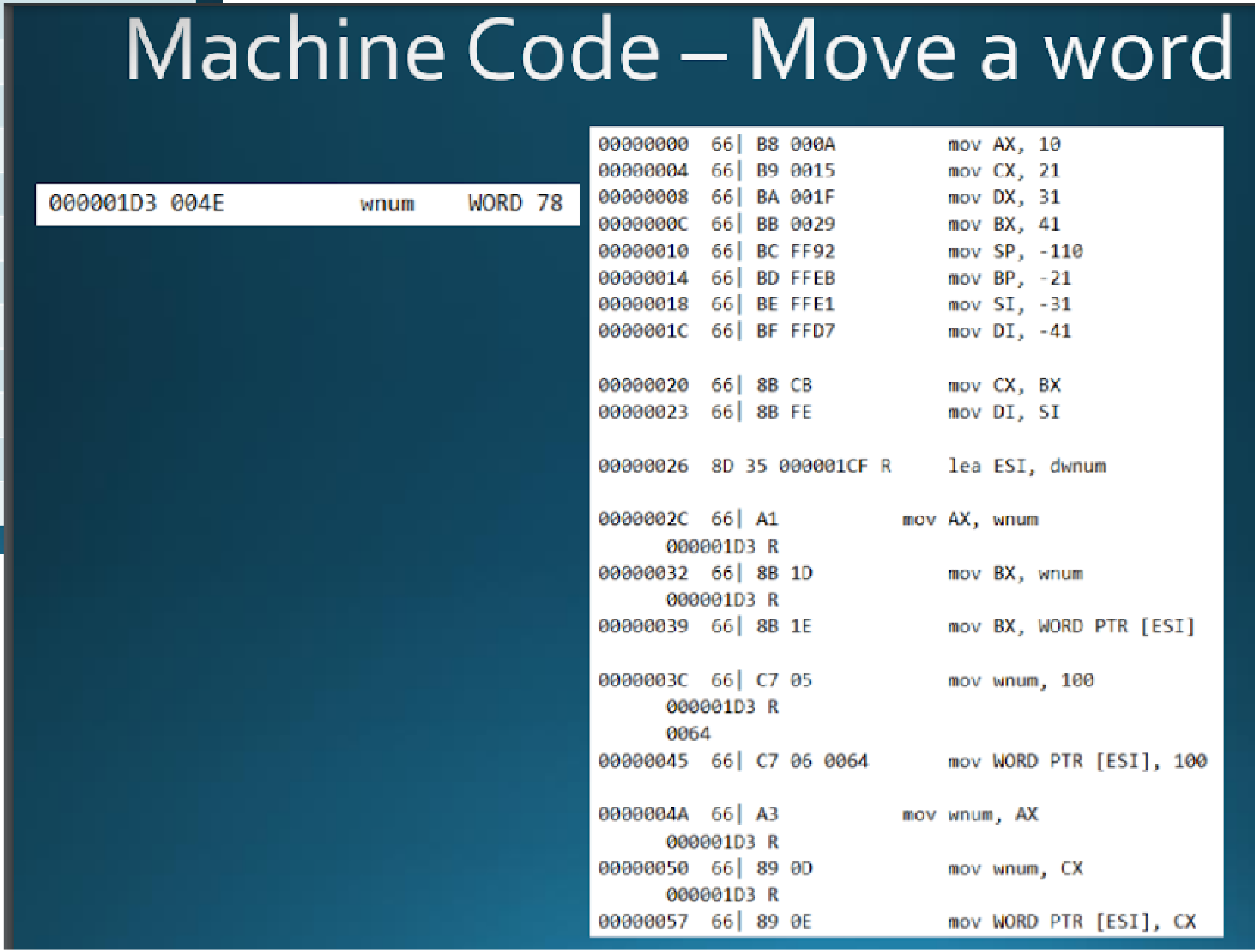
64-bit mode only

```
00000019 0000002C       dwnum   DWORD 44
00000000               .CODE
00000000               MoveDword PROC
00000000  41/ B8                mov R8D, 10
      0000000A
00000006  41/ B9                mov R9D, 10
      0000000A
0000000C  41/ BA                mov R10D, 10
      0000000A
00000012  41/ BB                mov R11D, 10
      0000000A
00000018  41/ BC                mov R12D, 10
      0000000A
0000001E  41/ BD                mov R13D, 10
      0000000A
00000024  41/ BE                mov R14D, 10
      0000000A
0000002A  41/ BF                mov R15D, 10
      0000000A
00000030  45/ 8B D1                 mov R10D, R9D
00000033  44/ 8B E2                 mov R12D, EDX
00000036  41/ 8B CE                 mov ECX, R14D
00000039  44/ 8B 0D                 mov R9D, dwnum
      00000019 R
00000040  44/ 89 3D                 mov dwnum, R15D
      00000019 R
```



Machine Code – Move a word

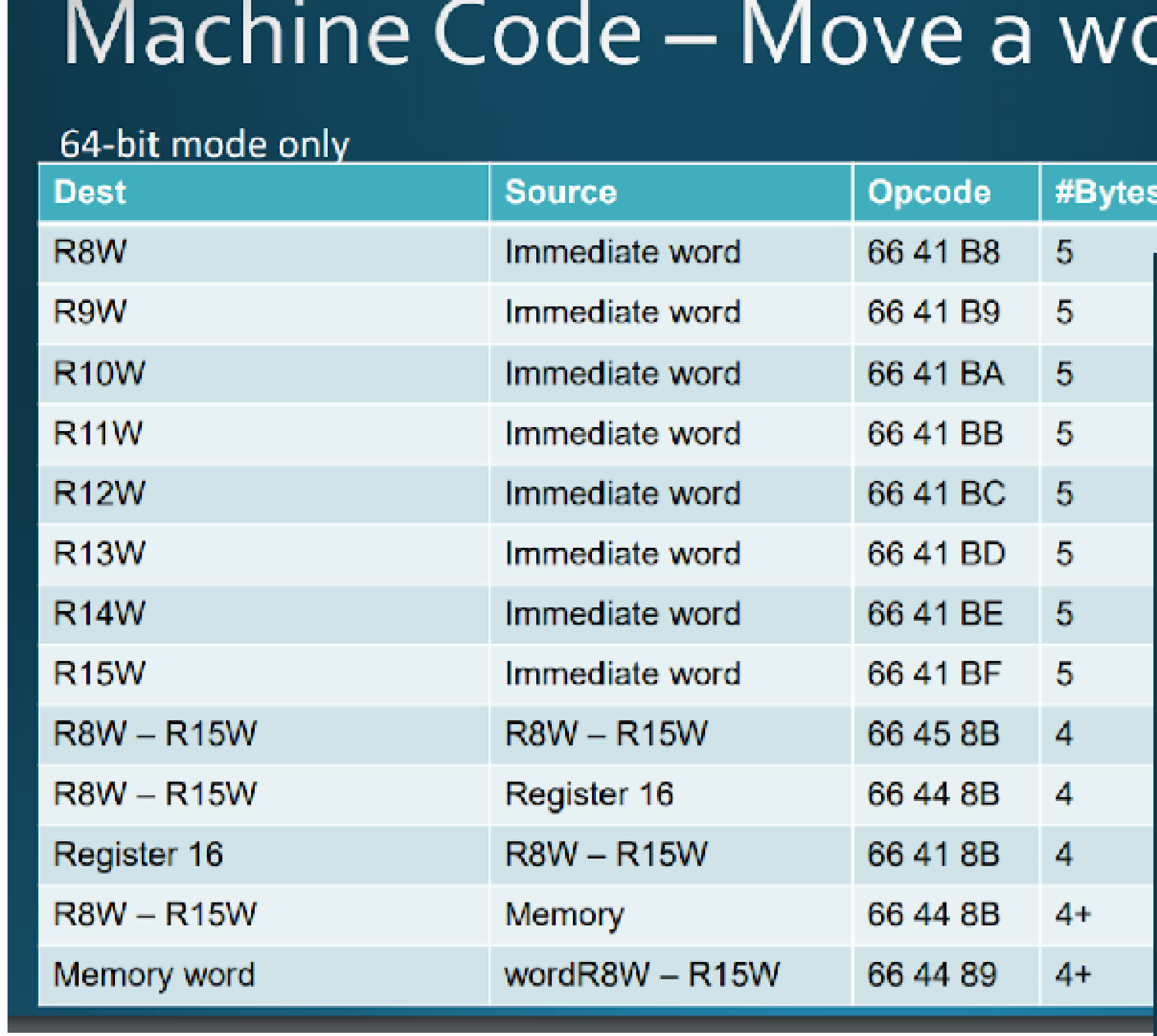
Dest	Source	Opcode	#Bytes in Obj code
AX	Immediate word	66 B8	4
CX	Immediate word	66 B9	4
DX	Immediate word	66 BA	4
BX	Immediate word	66 BB	4
SP	Immediate word	66 BC	4
BP	Immediate word	66 BD	4
SI	Immediate word	66 BE	4
DI	Immediate word	66 BF	4
Register 16 (ABOVE)	Register 16	66 8B	3
Register 16	Memory word	66 8B	3+
Memory word	Immediate word	66 C7	4+
Memory word	Register 16	66 88	3+



Machine Code – Move a word

```
000001D3 004E          wnum     WORD 78
```

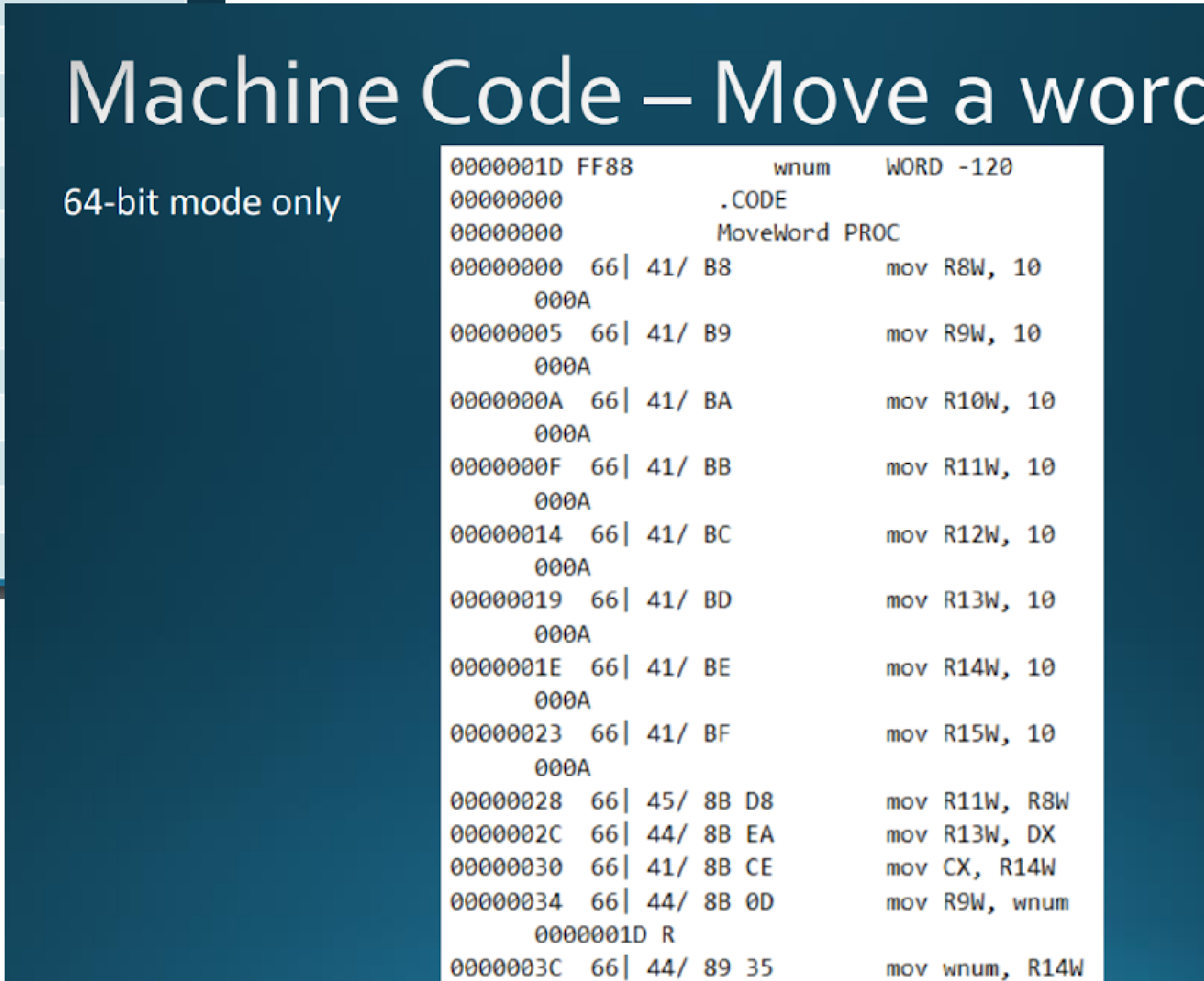
```
00000000  66| B8 000A           mov AX, 10
00000004  66| B9 0015           mov CX, 21
00000008  66| BA 001F           mov DX, 31
0000000C  66| BB 0029           mov BX, 41
00000010  66| BC FF92           mov SP, -110
00000014  66| BD FFEB           mov BP, -21
00000018  66| BE FFE1           mov SI, -31
0000001C  66| BF FFD7           mov DI, -41

00000020  66| 8B CB             mov CX, BX
00000023  66| 8B FE             mov DI, SI

00000026  8D 35 000001CF R      lea ESI, dwnum

0000002C  66| A1            mov AX, wnum
       000001D3 R
00000032  66| 8B 1D             mov BX, wnum
       000001D3 R
00000039  66| 8B 1E             mov BX, WORD PTR [ESI]

0000003C  66| C7 05             mov wnum, 100
       000001D3 R
       0064
00000045  66| C7 06 0064        mov WORD PTR [ESI], 100

0000004A  66| A3            mov wnum, AX
       000001D3 R
00000050  66| 89 0D             mov wnum, CX
       000001D3 R
00000057  66| 89 0E             mov WORD PTR [ESI], CX
```



Machine Code – Move a word

64-bit mode only

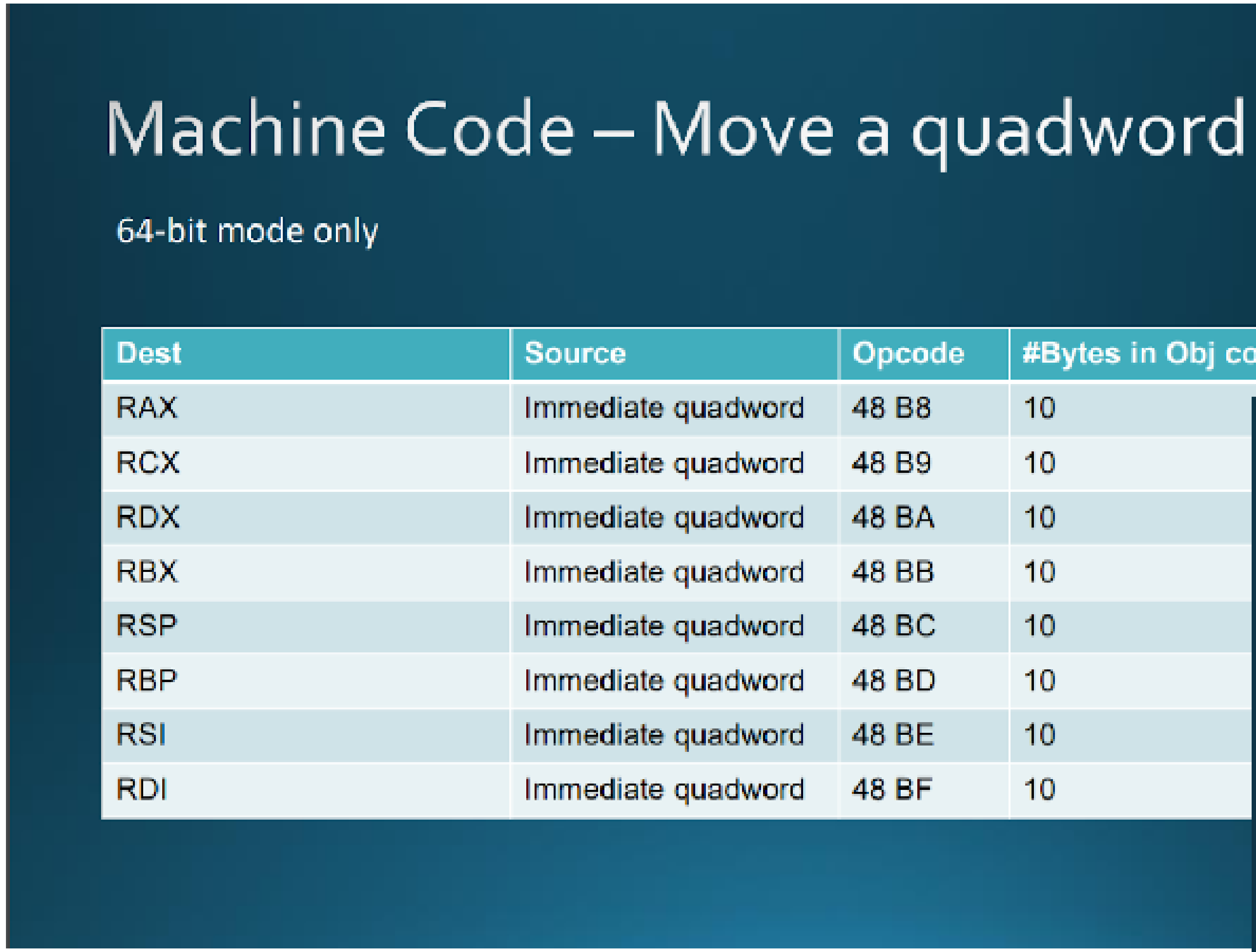
Dest	Source	Opcode	#Bytes in Obj code
R8W	Immediate word	66 41 B8	5
R9W	Immediate word	66 41 B9	5
R10W	Immediate word	66 41 BA	5
R11W	Immediate word	66 41 BB	5
R12W	Immediate word	66 41 BC	5
R13W	Immediate word	66 41 BD	5
R14W	Immediate word	66 41 BE	5
R15W	Immediate word	66 41 BF	5
R8W – R15W	R8W – R15W	66 45 8B	4
R8W – R15W	Register 16	66 44 8B	4
Register 16	R8W – R15W	66 41 8B	4
R8W – R15W	Memory	66 44 8B	4+
Memory word	wordR8W – R15W	66 44 89	4+



Machine Code – Move a wor

64-bit mode only

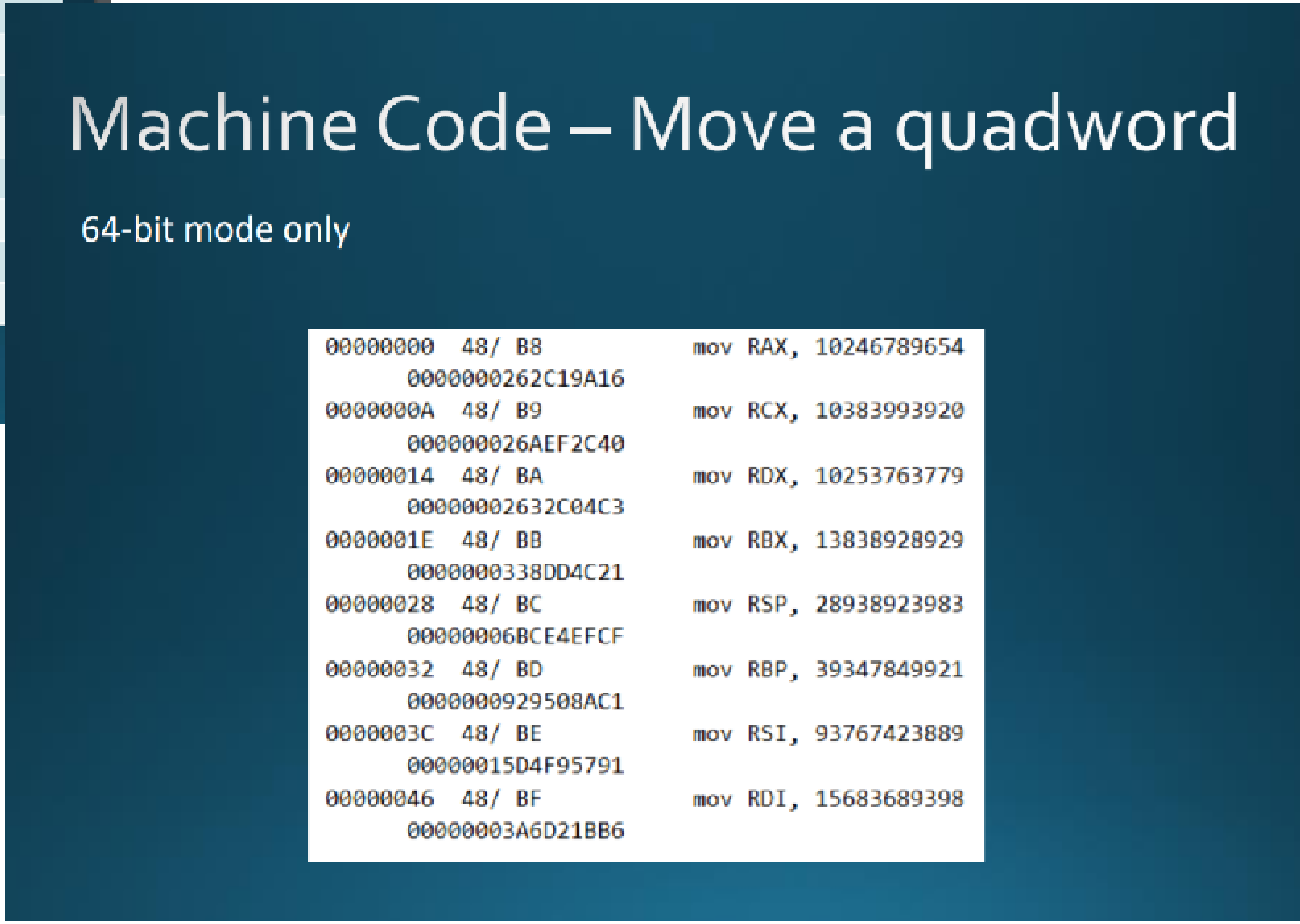
```
0000001D FF88           wnum    WORD -120
00000000              .CODE
00000000              MoveWord PROC
00000000  66| 41/ B8            mov R8W, 10
       000A
00000005  66| 41/ B9            mov R9W, 10
       000A
0000000A  66| 41/ BA            mov R10W, 10
       000A
0000000F  66| 41/ BB            mov R11W, 10
       000A
00000014  66| 41/ BC            mov R12W, 10
       000A
00000019  66| 41/ BD            mov R13W, 10
       000A
0000001E  66| 41/ BE            mov R14W, 10
       000A
00000023  66| 41/ BF            mov R15W, 10
       000A
00000028  66| 45/ 8B D8         mov R11W, R8W
0000002C  66| 44/ 8B EA         mov R13W, DX
00000030  66| 41/ 8B CE         mov CX, R14W
00000034  66| 44/ 8B 0D         mov R9W, wnum
       0000001D R
0000003C  66| 44/ 89 35         mov wnum, R14W
```



Machine Code – Move a quadword

64-bit mode only

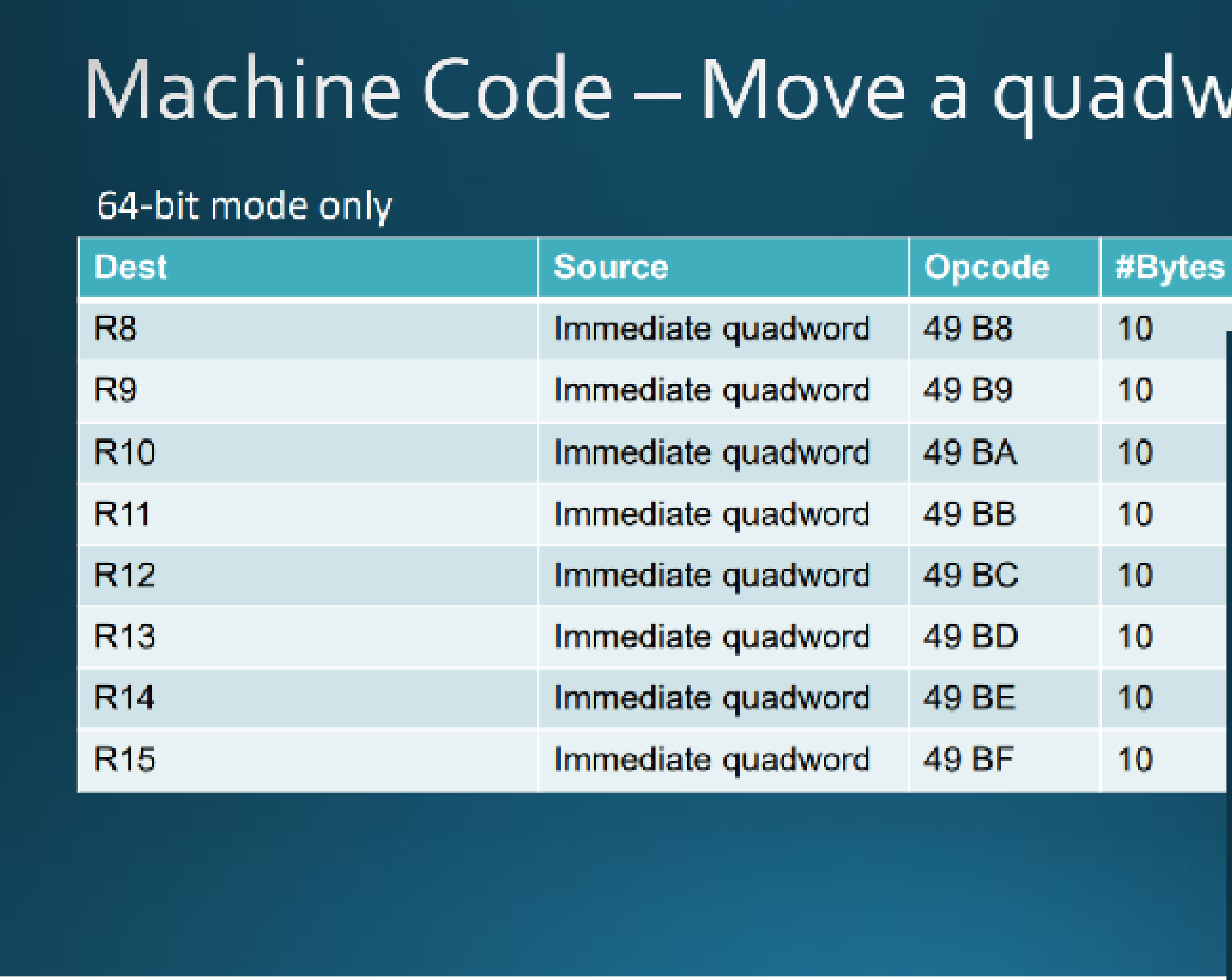
Dest	Source	Opcode	#Bytes in Obj code
RAX	Immediate quadword	48 B8	10
RCX	Immediate quadword	48 B9	10
RDX	Immediate quadword	48 BA	10
RBX	Immediate quadword	48 BB	10
RSP	Immediate quadword	48 BC	10
RBP	Immediate quadword	48 BD	10
RSI	Immediate quadword	48 BE	10
RDI	Immediate quadword	48 BF	10



Machine Code – Move a quadword

64-bit mode only

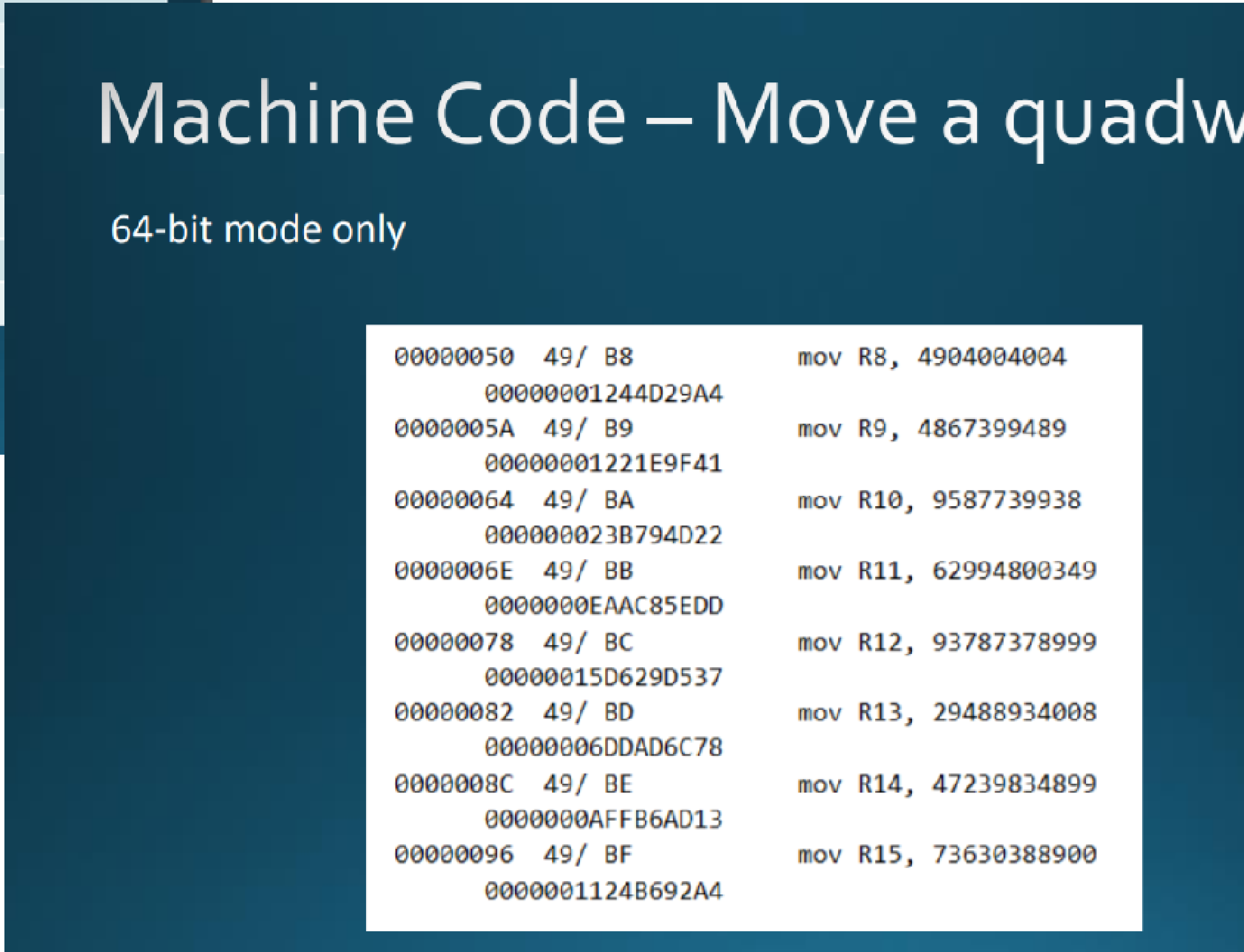
```
00000000  48/ B8                    mov RAX, 10246789654
      0000000262C19A16
0000000A  48/ B9                    mov RCX, 10383993920
      000000026AEF2C40
00000014  48/ BA                    mov RDX, 10253763779
      00000002632C04C3
0000001E  48/ BB                    mov RBX, 13838928929
      0000000338DD4C21
00000028  48/ BC                    mov RSP, 28938923983
      00000006BCE4EFCF
00000032  48/ BD                    mov RBP, 39347849921
      0000000929508AC1
0000003C  48/ BE                    mov RSI, 93767423889
      00000015D4F95791
00000046  48/ BF                    mov RDI, 15683689398
      00000003A6D21BB6
```



Machine Code – Move a quadword

64-bit mode only

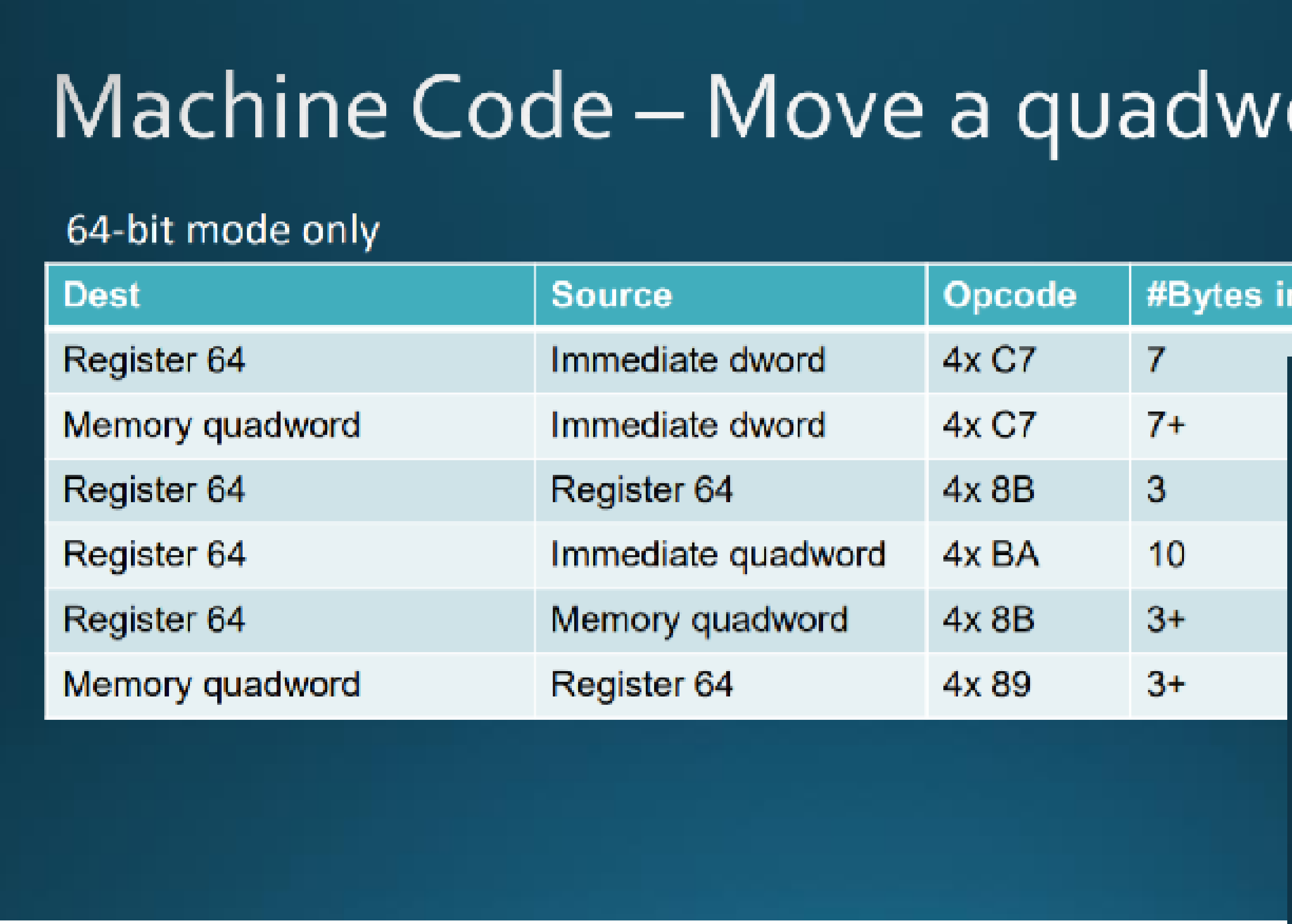
Dest	Source	Opcode	#Bytes in Obj code
R8	Immediate quadword	49 B8	10
R9	Immediate quadword	49 B9	10
R10	Immediate quadword	49 BA	10
R11	Immediate quadword	49 BB	10
R12	Immediate quadword	49 BC	10
R13	Immediate quadword	49 BD	10
R14	Immediate quadword	49 BE	10
R15	Immediate quadword	49 BF	10



Machine Code – Move a quadw

64-bit mode only

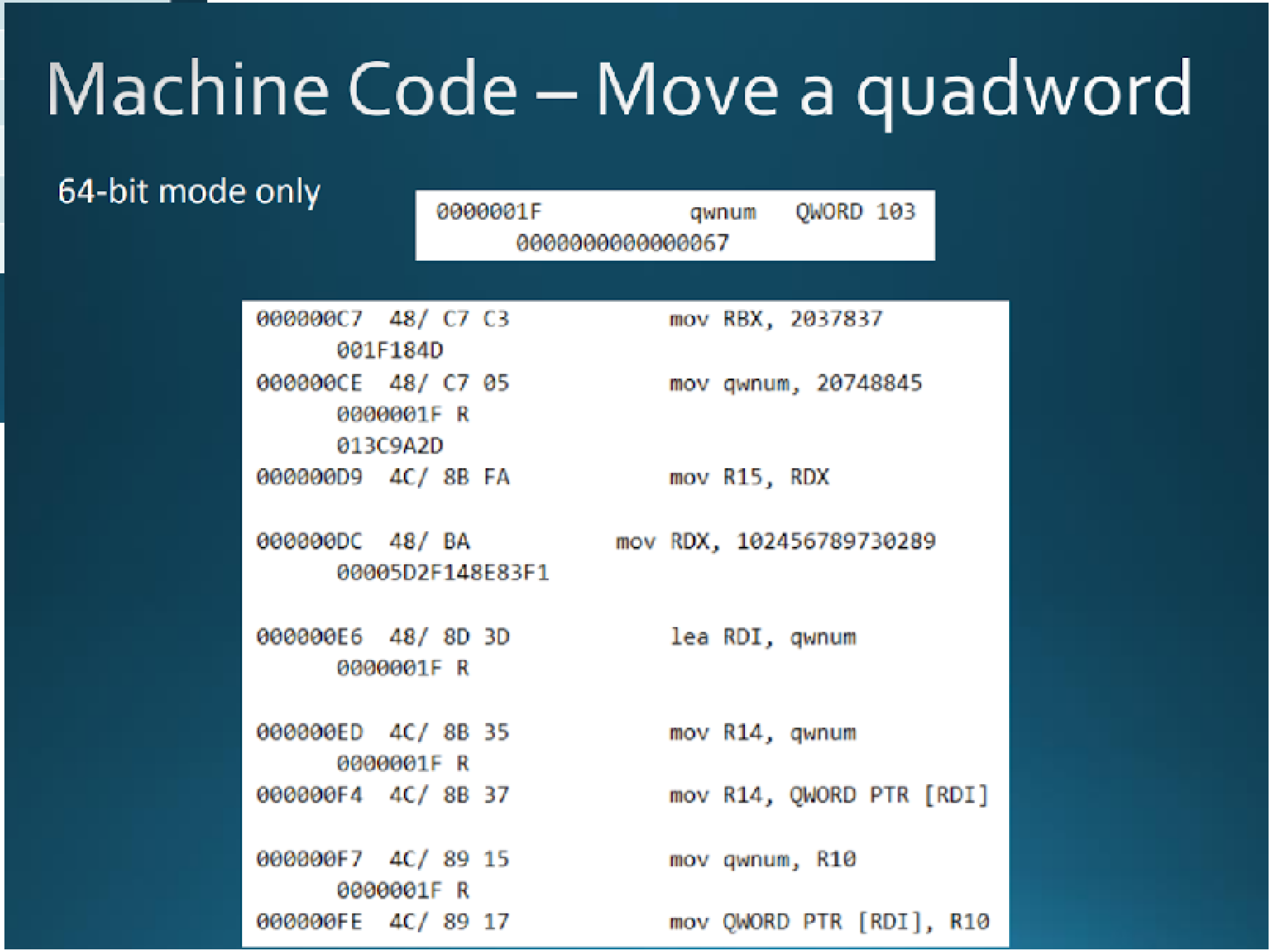
```
00000050  49/ B8                mov R8, 4904004004
      00000001244D29A4
0000005A  49/ B9                mov R9, 4867399489
      0000001221E9F41
00000064  49/ BA                mov R10, 9587739938
      000000023B794D22
0000006E  49/ BB                mov R11, 62994800349
      0000000EAAC85EDD
00000078  49/ BC                mov R12, 93787378999
      00000015D629D537
00000082  49/ BD                mov R13, 29488934008
      00000006DDAD6C78
0000008C  49/ BE                mov R14, 47239834899
      0000000AFFB6AD13
00000096  49/ BF                mov R15, 73630388900
      0000001124B692A4
```



Machine Code – Move a quadword

64-bit mode only

Dest	Source	Opcode	#Bytes in Obj code
Register 64	Immediate dword	4x C7	7
Memory quadword	Immediate dword	4x C7	7+
Register 64	Register 64	4x 8B	3
Register 64	Immediate quadword	4x BA	10
Register 64	Memory quadword	4x 8B	3+
Memory quadword	Register 64	4x 89	3+



Machine Code – Move a quadword

64-bit mode only

```
0000001F             qwnum    QWORD 103
      0000000000000067
```

```
000000C7  48/ C7 C3               mov RBX, 2037837
      001F184D
000000CE  48/ C7 05               mov qwnum, 20748845
      0000001F R
      013C9A2D
000000D9  4C/ 8B FA               mov R15, RDX

000000DC  48/ BA              mov RDX, 102456789730289
      00005D2F148E83F1

000000E6  48/ 8D 3D               lea RDI, qwnum
      0000001F R

000000ED  4C/ 8B 35               mov R14, qwnum
      0000001F R
000000F4  4C/ 8B 37               mov R14, QWORD PTR [RDI]

000000F7  4C/ 89 15               mov qwnum, R10
      0000001F R
000000FE  4C/ 89 17               mov QWORD PTR [RDI], R10
```

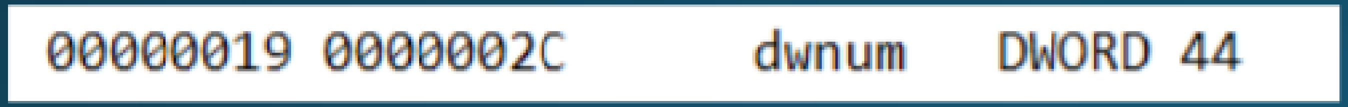
xchg Instruction

- xchg <operand1> <operand2>
- Swaps the values referenced by its two operands
  - Can’t have both operands in memory
- Does not alter any flag
- Example:

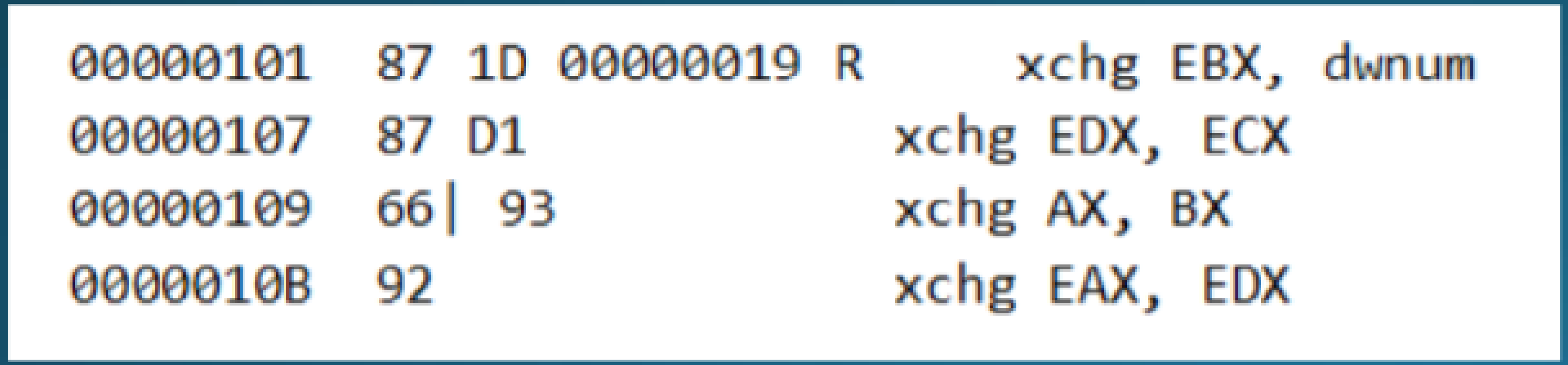
mov	xchg
mov ecx, eax mov eax, ebx mov ebx, ecx	xchg eax, ebx

xchg Instruction

- Four specific forms:
  - xchg <reg>, <mem>
  - xchg <reg>, <reg>
  - xchg ax, <reg16>
  - xchg eax, <reg32>



```
00000019 0000002C       dwnum   DWORD 44
```



```
00000101  87 1D 00000019 R      xchg EBX, dwnum
00000107  87 D1               xchg EDX, ECX
00000109  66| 93              xchg AX, BX
0000010B  92                  xchg EAX, EDX
```