Outflank

# Outflank up

## Carry Bit

outflank = ((O| ~mask) + GetLSB(mask)) & P & mask;

1 x OR

1 x ADD

2 x AND

4 clock cycles

## BitScan

outflank = GetLSB(~O & mask) & P;

1 x NOT

2 x AND

1 x BLSI

4 clock cycles

# Outflank down

## BitScan

outflank = (0x8000000000000000ULL >> CountLeadingZeros(~O & mask)) & P;

1 x NOT

2 x AND

1 x LZCNT (3 clock cycles)

1 x SHIFT

7 clock cycles

# Outflank Look Up Mask Shift

Byte outflank\_n[64]; // This is one cache line

outflank = outflank\_n[BEXTR(O, start1, 6)] & BEXTR(P, start2, 8);

# Outflank Look Up Mask Multiply Shift

Byte outflank\_n[64]; // This is one cache line

outflank = outflank\_n[((O & mask1)\*0x0101010101010101ULL) >> 57]

& (((P & mask2)\*0x0101010101010101ULL) >> 56);

# Outflank Look Up PExt

outflank = outflank\_n[PExt(O, mask1)] & PExt(P, mask1);

2 x PExt

1 x LookUp

1 x AND

5 clock cycles + 1 Look Up

Flips

# Flips up

flips = (outflank – (outflank != 0)) & mask;

1 x CMP

1 x ADD

1 x AND

3 clock cycles

# Flips Down

flips = (-outflank \* 2) & mask;

1 x NEG

1 x ADD

1 x AND

3 clock cycles

# Flips Look Up 1

unsigned long long flips\_n[256]; 32 cache lines

flips = flips\_n[outflank] & mask;

1 x AND

1 x Look Up

1 clock cycle + 1 Look Up

# Flips Look Up 2

Byte flips\_n[256]; 4 cache lines

flips = PDep(flips\_n[outflank], mask);

1 x PDep

1 x Look Up

3 clock cycle + 1 Look Up

Hacks

# Hack 1

flips = ((P & mask1) >> 7 | (P & mask2) << 1) & O;

3 x AND

2 x SHIFT

1 x OR

6 clock cycles

# Hack 2

flips = PDep(PExt(P, mask1), mask2) & O;

1 x PExt

1 x PDep

1 x AND

7 clock cycle