

Table of Contents

[Book Transpositions](#)

[Flipping an 8x8 Board](#)

[Looking for Bugs](#)

[Monochrome](#)

[MicroChess](#)

[Quad-Bitboards](#)

[Six-Two Bitboards](#)

[Keeping two Boards](#)

[See also](#)

[Publications](#)

[Forum Posts](#)

[External Links](#)

[References](#)

[What links here?](#)

[Home](#) * [Chess](#) * [Position](#) * **Color Flipping**

Color Flipping a chess position refers to the [vertical flipping](#) or mirroring of all [pieces](#) along the horizontal axis between the 4th and 5th [rank](#) also swapping the [color](#) of the flipped pieces from White to Black and vice versa, the [side to move](#), the [castling rights](#) and the rank of a possible [en passant](#) target square from six to three or vice versa accordantly. The resulting position is mirror equivalent, pawn and pieces have the same [span length](#) or home rank relation. A white pawn on the light square c2 becomes a black pawn on the dark square c7. Color flipping a vertical symmetrical or mirror position (same castling rights, no en passant possible) results in the same position as if the side to move performed a [null move](#):

Giuoco Piano

Giuoco Piano Symmetrical

r1bqk1nr/pppp1ppp/2n5/2b1p3/2B
1P3/5N2/PPPP1PPP/RNBQK2R
w KQkq -

r1bqk2r/pppp1ppp/2n2n2/2b1p3/2
B1P3/2N2N2/PPPP1PPP/R1BQK
2R w KQkq -

rnbqk2r/pppp1ppp/5n2/2b1p3/2B1
P3/2N5/PPPP1PPP/R1BQK1NR b
KQkq -

r1bqk2r/pppp1ppp/2n2n2/2b1p3/2
B1P3/2N2N2/PPPP1PPP/R1BQK
2R b KQkq -

Book Transpositions

During the [opening](#), in some lines, White may intentionally lose a [tempo](#), to [transpose](#) into a color flipped position of an opening White likes to play the "Black" side, which might be considered in chess programs [book](#) access code, see for instance [WMCCC 1988, Pandix-Y!88](#).

Flipping an 8x8 Board

An [8x8 Board](#) with a [rank-file mapping](#), needs to perform an [exclusive or](#) with 56 (A8 in [LERF](#)) to vertically flip square coordinates. Assuming piece color is least significant bit and empty square is zero, a pure 8x8 Board may be color flipped that way in [C](#):

```
int board[64], sq, s, f;

for (sq = 0; sq < 32; ++sq) {
    s = board[sq];
    f = board[sq^56];
    board[sq] = f ^ (f != 0);
    board[sq^56] = s ^ (s != 0);
}
```

Looking for Bugs

A side to move relative, bug free static [evaluation](#) should have the same [score](#) for both positions. [Search](#) is a different issue, since [move ordering](#) might be slightly different for instance in bitboard programs due to different order in [bitboard serialization](#) if one doesn't use a [symmetrical bitscan](#) by side to move. Similar to

the [butterfly effect](#) in [chaos theory](#) a minor initial difference might amplify to enormous effects during the search. However, if the search behavior of a [tactical test position](#) of a most deterministic chess engine (e.g. no [parallel search](#)), varies by magnitudes in solution [depth](#) and time between original and color flipped position, one may take a closer look performing some [logging](#) and/or [debugging](#).

Monochrome

An idea to unify and simplify [move generation](#) and to get rid of order dependencies, is to perform a color flip of the position and its internal [board representation](#) directly after [making a move](#), to make the engine white to move only, that is only generating white moves, or similar, to keep a normal and flipped board to [update them incrementally](#).

MicroChess

[Jennings' approach](#) with [MicroChess](#) was similar ^[1], but instead of vertical flipping (xor 56) he reversed or rotated the board by 180 degrees (xor 63), like players changing places and looking the board alternately from White's or Black's point of view, with some considerations of the additional horizontal mirroring, concerning [castling](#) and [en passant](#).

Quad-Bitboards

[Gerd Isenberg](#) proposed a make move - color flipping approach using [Quad-Bitboards](#) ^[2]. The black [Zobrist keys](#) were the byte-swapped white Zobrist keys from the vertically flipped squares (xor 56). In symmetrical positions this gains some additional hits while probing the [transposition table](#). But symmetrical mirror positions, or symmetrical subsets have the property, their [xored](#) hashkey has only 32 distinct bits since hashkey equals [flipVertical](#)(hashkey), for instance:

```
01020304|50607080 // white pieces
^ 80706050|04030201 // black pieces of a mirror position
= 81726354|54637281
```

Six-Two Bitboards

The common [six piece kind and two color bitboard](#) board representation takes eight byte swaps to color flip the whole board.

Keeping two Boards

Rather than flipping the board after each make, [Reinhard Scharnagl](#) proposed monochrome move generation by keeping a normal and a color flipped board, both [incremental updated](#) ^[3]. He further suggested asymmetrical [piece coding](#) with less piece codes (only one or two) for the pieces of the side not to move. His approach has also the advantage to keep independent Zobrist keys for all squares and pieces.

See also

- [Color](#)
- [Diagonal Mirroring](#)
- [Flipping, Mirroring and Rotating](#) of [Bitboards](#)
- [Horizontal Mirroring](#)
- [Side to move](#)
- [Vertical Flipping](#)

Publications

- [Fernand Gobet](#) (1993). [A Computer Model of Chess Memory](#). Proceedings of the 15th Annual Meeting of the Cognitive Science Society, pp. 463-468.

Forum Posts

- [Conversion to dual positions for EPD testsuites?](#) by [Leen Ammeraal](#), [CCC](#), November 26, 2001
- [fen to fen functions](#) by [Uri Blass](#), [CCC](#), May 21, 2007
- [Comparison data of Crafty vs Rybka that I promised 2 days ago](#) by Les, [CCC](#), June 01, 2007
- [color-flipped perft](#) by [Gerd Isenberg](#), November 04, 2007
- [Board without color flags](#) by [Fritz Reul](#), [CCC](#), October 31, 2009
- [Position flipping](#) by [Steven Edwards](#), [CCC](#), August 05, 2013
- [Black/White symmetry in move generation](#) by Jeffery A Esposito, [CCC](#), November 25, 2014 » [Move Generation](#)
- [Symmetric move generation using bitboards](#) by [Lasse Hansen](#), [CCC](#), December 20, 2014
- [reversed-color transpositions](#) by [John Fishburn](#), [CCC](#), March 11, 2016
- [FEN - Flipper for Windows](#) by [Matthias Gemuh](#), [CCC](#), May 17, 2017 » [FEN](#), [EPD](#)

External Links

- [A "Mirror" Chess Problem](#) from [Jim Loy's Chess Pages](#)
- [Playing Chess Against Its Mirror Image](#)

References

1. [^ Peter Jennings \(1976\). MicroChess, a Chess playing program for the 6502 Microcomputer. pdf](#), Courtesy of [Peter Jennings, The Computer History Museum](#)
2. [^ color-flipped perft](#) by [Gerd Isenberg](#), November 04, 2007
3. [^ Computerschach mit monochromen Methoden ?](#) by [Reinhard Scharnagl](#), [OpenChess - Deutsches Forum](#), January 27, 2011 (German)

What links here?

Page	Date Edited
6502	Dec 25, 2017
AlphaZero	Feb 10, 2018
Bitboard Board-Definition	Jun 23, 2014
Bitboard Serialization	Dec 24, 2014
Chess Position	Sep 10, 2017
Chessmaps Heuristic	Aug 27, 2015
Color	Mar 12, 2014
Color Flipping	May 17, 2017
CTG	Aug 22, 2016
Debugging	Jan 20, 2018
Diagonal Mirroring	Jun 29, 2013
DirGolem	Jun 5, 2016
ECO	Nov 2, 2017
Excel	Dec 5, 2016
Extended Position Description	Dec 2, 2017
Flipping Mirroring and Rotating	Oct 14, 2016
Forsyth-Edwards Notation	Sep 10, 2017
Fritz Reul	Nov 28, 2016
Horizontal Mirroring	Jun 29, 2013
Incremental Updates	Sep 6, 2017
John Philip Fishburn	May 3, 2017
Leen Ammeraal	Nov 3, 2014
Matthias Gemuh	Jan 21, 2018
MicroChess	Jul 14, 2015
Move Generation	Jan 29, 2018
Opening Book	Feb 26, 2018
Pawn chain	Jan 17, 2018
Phalanx	May 2, 2016
Pigeon	Nov 2, 2016
Quad-Bitboards	Jan 30, 2017
Side to move	Oct 29, 2017
Teki	Mar 29, 2018
Transposition	Jun 8, 2015
Vertical Flipping	Oct 10, 2013

Page

[Vice](#)

[WMCCC 1988](#)

Date Edited

Mar 8, 2016

Oct 27, 2016

[Up one Level](#)