

Table of Contents

[The aims for the engine](#)

[Playing strength](#)

[Parts](#)

[Features](#)

[To do](#)

[Known deficiencies](#)

[See also](#)

[Forum Posts](#)

[External Links](#)

[References](#)

[What links here?](#)

[Home](#) * [Engines](#) * **Chessprogramming-wiki engine**

The **CPW-Engine** is being under development by the [wiki members Pawel Koziol](#) and [Edmund Moshhammer](#), who try to put all the theory presented on this page into practice. When finished the CPW will be a fully functional chess engine that should provide guidance to new programmers and exemplify some ideas. If You have any ideas how to simplify it, feel free to use "discussion" option. See also [CPW history](#) on how the engine has evolved. A revised 2014 version 1.1 was provided Pawel Koziol ^[1], full code is further available at [GitHub](#) ^[2].

The aims for the engine

- readability
- annotation
- playing strength is less important
- modularization

Playing strength

Version 1.0 of the CPW-Engine (CPW 1.00 beta 1) scored 5.5 out of 11 in [ChessWar](#) XIII F, earning there a rating of 1828. For the sake of comparison the same number of points has been gained by [Gerbil](#) (1894) and [Faile](#) (1900). Crosstables and games can be found at the [ChessWar](#) site.

Version 1.1 is probably about 2200 Elo on CCRL scale, but this needs to be tested.

The following game represents the current strength of the engine:

```
[Event "Test game"]
[Date "2008.08.19"]
[Round "7"]
[White "Tscpl81"]
[Black "CPW 1.0"]
[Result "0-1"]
[TimeControl "300"]
[Termination "adjudication"]
[PlyCount "67"]
```

```
1. d4 Nf6 2. Nf3 e6 3. c4 c5 4. d5 exd5 5. cxd5 d6 6. Nc3 g6 7. e4 Bg7
8. Bb5+ Bd7
9. Qd3 O-O 10. Bf4 Nxe4 11. Nxe4 Qa5+ 12. Qd2 Qxb5 13. O-O-
O Qa6 14. Bxd6 Re8 15. Qc2
Bf5 16. Nfg5 Bh6 17. Nf6+ Kh8 18. Qxf5 gxf5 19. Nxe8 Bxg5+ 20. Kb1 Nd7
21. Be5+ Nxe5
22. Nc7 Qf6 23. Rhe1 Rc8 24. Nb5 Nc4 25. Re2 Qa6 26. Nc3 Bf6 27. Rde1 N
a3+ 28. Ka1 Qc4
29. d6 Bxc3 30. d7 Rd8 31. Re8+ Kg7 32. R8e4 fxe4 33. Rc1 Bxb2+ 34. Kxb
2
{0-1 Arena Adjudication} 0-1
```

This is it: lousy positional play and some luck aided by king tropism evaluation. Currently the engine scores above 90% against [TSCP](#).

Parts

- [CPW-Engine_stdafx_h](#) (constants and most of function prototypes)
- [CPW-Engine_book](#)
- [CPW-Engine_chronos](#) (timing functions)
- [CPW-Engine_console_ui](#)
- [CPW-Engine_main](#)

- [CPW-Engine_com](#)
- [CPW-Engine_eval_init](#)
- [CPW-Engine_quiescence](#)
- [CPW-Engine_recognize](#)
- [CPW-Engine_root](#) (obsolete)
- [CPW-Engine_search](#)
- [CPW-Engine_transposition](#)
- [CPW-Engine_0x88_math](#)
- [CPW-Engine_board\(0x88\)](#)
- [CPW-Engine_movegen\(0x88\)](#)
- [CPW-Engine_move\(0x88\)](#)
- [CPW-Engine_algebraic](#)
- [CPW-Engine_eval](#)
- [CPW-Engine_attacks](#)
- [CPW-Engine_utils](#)

Features

- [0x88](#) board
- [Alpha-beta](#) with [PVS](#)
- [Quiescence search](#) with [delta pruning](#)
- [Adaptive null move pruning](#)
- [Futility pruning](#)
- [Evaluation](#): [material](#) (with some scaling), [piece-square tables](#), [weak](#) and [passed pawns](#), [enemy king tropism](#)
- [UCI](#) support

To do

This list represents things that are to be done in order to be ready for a release

- create a text interface for testing and debugging
- comment the source
- get some more speed
- add a winboard command parser

Known deficiencies

Though CPW is a didactic program, it has a couple of features that should not be repeated in Your programs

- lack of a separate capture generator

- hash table does not save the move itself, but only its position on the list, which limits development options

See also

- [Glass](#), the new [UCI engine](#) by the authors of the CPW-Engine.

Forum Posts

- [Cpw : encore un UCI Engine sans intérêt](#) by [Patrick Buchmann](#), [Le Fou numérique Forum](#), August 08, 2008 (French)
- [fixing CPW-engine](#) by [Pawel Koziol](#), [CCC](#), December 30, 2014

External Links

- [nescitus/cpw-engine · GitHub](#)

References

1. [fixing CPW-engine](#) by [Pawel Koziol](#), [CCC](#), December 30, 2014
2. [nescitus/cpw-engine · GitHub](#)

What links here?

Page	Date Edited
0x88	Nov 28, 2016
CCCP	Jan 10, 2013
CPW history	Jan 1, 2015
CPW-Engine	Dec 31, 2014
CPW-Engine_0x88_math	Dec 30, 2014
CPW-Engine_algebraic	May 15, 2011
CPW-Engine_attacks	Dec 30, 2014
CPW-Engine_board(0x88)	Dec 30, 2014
CPW-Engine_book	May 8, 2017
CPW-Engine_book_h	Sep 27, 2008
CPW-Engine_chronos	Dec 30, 2014
CPW-Engine_com	Dec 30, 2014
CPW-Engine_console_ui	Dec 30, 2014
CPW-Engine_constants	Jun 13, 2011
CPW-Engine_eval	Jul 26, 2016
CPW-Engine_eval_h	Dec 30, 2014
CPW-Engine_eval_init	Dec 30, 2014
CPW-Engine_main	Jun 13, 2011

Page	Date Edited
CPW-Engine_move(0x88)	Dec 30, 2014
CPW-Engine_movegen(0x88)	Dec 30, 2014
CPW-Engine_movegen_h	Dec 30, 2014
CPW-Engine_quiescence	Dec 30, 2014
CPW-Engine_recognize	Dec 30, 2014
CPW-Engine_root	Sep 27, 2008
CPW-Engine_search	Feb 4, 2017
CPW-Engine_search_h	Dec 30, 2014
CPW-Engine_stdafx_h	Dec 30, 2014
CPW-Engine_transposition	Dec 30, 2014
CPW-Engine_transposition_h	Dec 30, 2014
CPW-Engine_utils	Dec 30, 2014
CPW-Engine_variables	Mar 13, 2009
CPW_King	Sep 27, 2008
Delta Pruning	Dec 2, 2014
Edmund Moshammer	Sep 23, 2017
Engine releases	Apr 23, 2018
Engines	Mar 10, 2018
Gavon	Apr 30, 2018
Glass	May 24, 2017
Home	Jul 2, 2018
King Safety	Feb 14, 2018
Open Source Engines	Jul 14, 2015
Pawel Koziol	Dec 17, 2017
Programming	Dec 16, 2017
Rodent	Jan 11, 2018

[Up One Level](#)