

## Basic flow charts for the CT800

The following pages contain some basic control flow charts in a simplified version, just for getting some orientation:

- What each source text file contains
- How the overall control flow works
- How a response move is calculated (main flow in bold lines)

Time keeping, hash tables, menu logic and HMI are not shown in this overview. The menu logic is simple enough anyway because the tree structure of the menu is directly reflected in the call tree of the menu logic.

Note that this software is single-threaded; the only thing in parallel is the timer interrupt on the target platform, which also handles the keypad input. Usually, this does not interfere with the regular control flow, with one notable exception:

If the computer is calculating and the "GO!"-key is pressed, the keypad evaluation routine will set up the "time is up" flag for the ongoing computation, simulating a time-over condition.

Rasmus Althoff, November 2021

## Overall Software Architecture

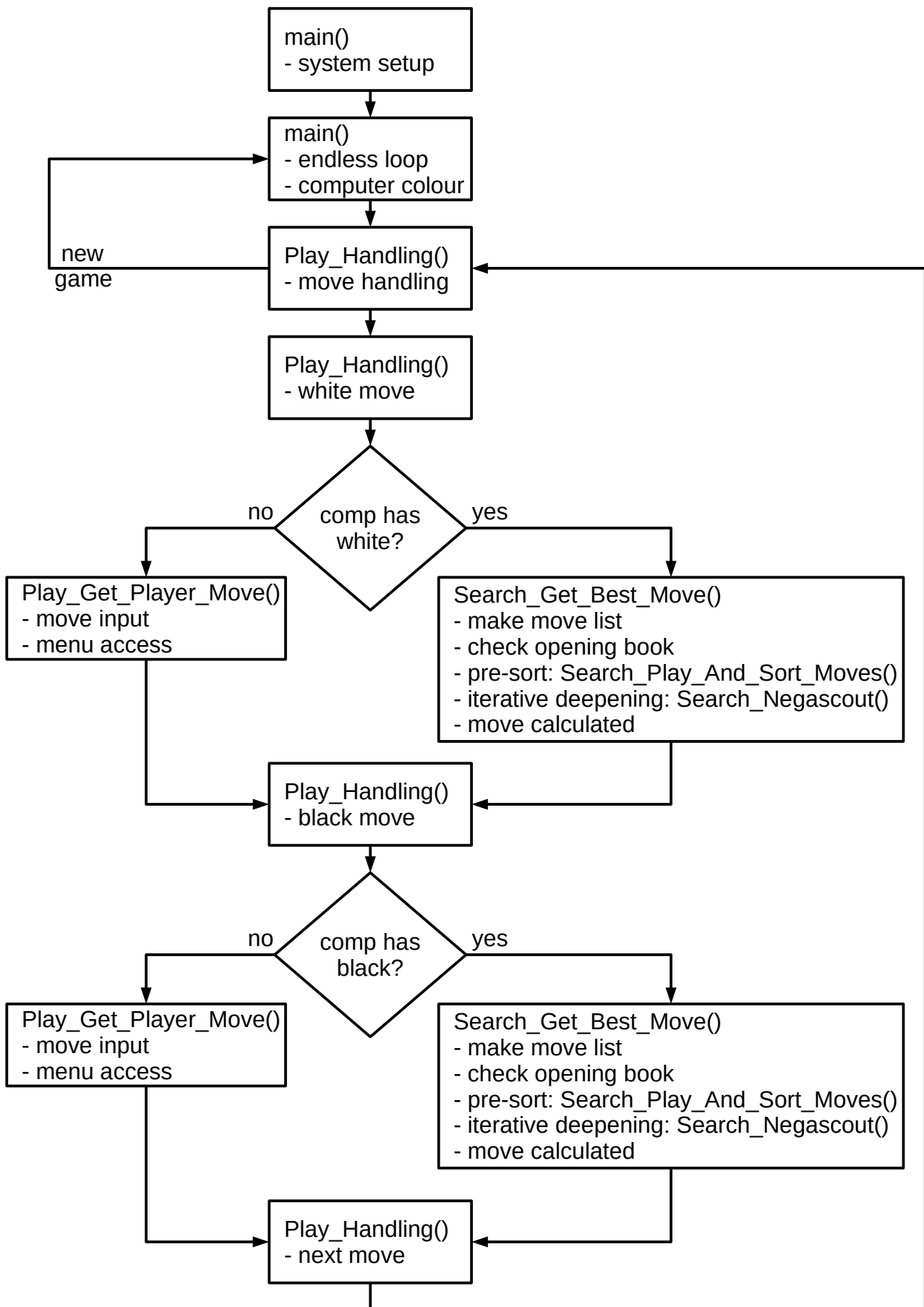
Chess Application Layer		
module	func. prefix	purpose
play	Play_	main(), overall game handling, move entering, menu access
search	Search_	search tree including quiescence
move_gen	Mvgen_	move generator, in-check info
eval	Eval_	static position evaluation
book	Book_	opening book
hashtables	Hash_	hashtable handling
kpk	Kpk_	king+pawn vs. king endgame table
timekeeping	Time_	time controls implementation

HMI Layer (Human-Machine Interface)		
module	func. prefix	purpose
hmi	Hmi_	dialogue system, game screen display, notation and position viewer
menu	Menu_	menu system
posedit	Pos_	position editor

Library Layer		
module	func. prefix	purpose
util	Util_	various library utility functions
ctdefs	-	project-wide definitions of constants and data types
confdefs	-	definition of the configuration options and macros

Driver Layer			
module	func. prefix	purpose	
boot_stm32f405	-	startup code, interrupt tables, RAM-test	(ARM only)
hardware_arm	Hw_	high level hardware interface	(ARM only)
hardware_arm _disp	Hw_Disp_	display driver	(ARM only)
hardware_arm _keybd	Hw_Keybd_	keypad driver	(ARM only)
hardware_arm _signal	Hw_Sig_	LED and beeper driver	(ARM only)
arm_driver	Drv_	low level CPU hardware interface	(ARM only)
hardware_pc	Hw_	replacement for hardware_arm	(PC only)

## Simplified Control Flow Chart



# Simplified Move Computation Chart

