

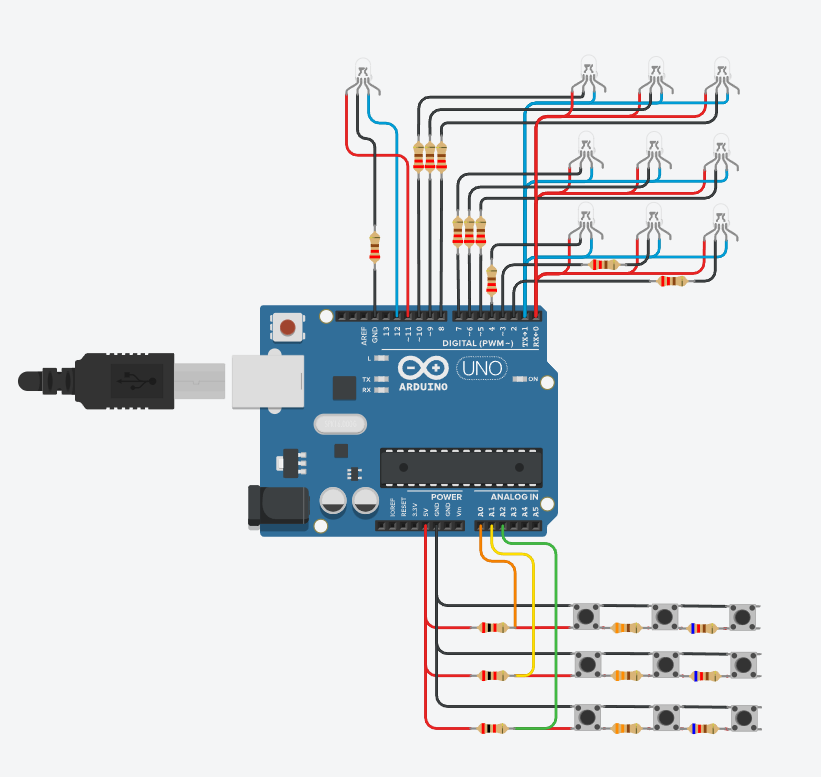
[ЛИНК ЗА ПРОЕКТА](https://www.tinkercad.com/things/fitrjgXMuKn?sharecode=Plr28TCbAgu5sNUZOT5d-OtP0QWWA-XonTewF9AwG1A)

СЪДЪРЖАНИЕ:

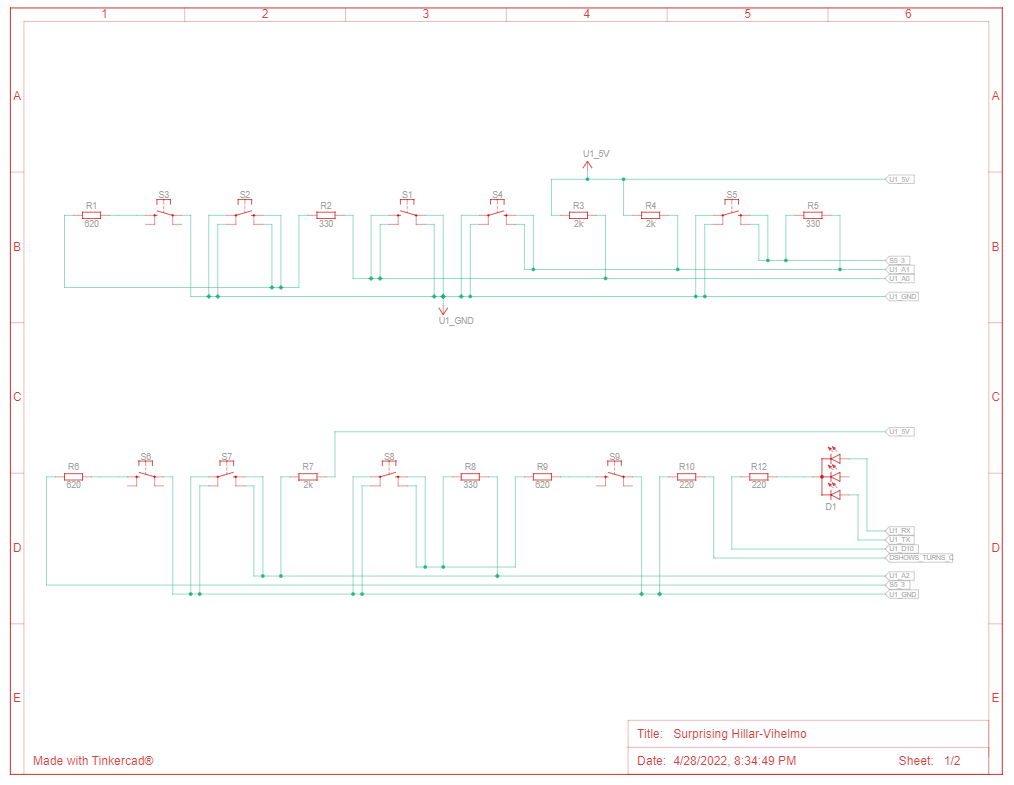
1. Описание на проекта
2. Блокова схема
3. Електрическа схема
4. Списък съставни части
5. Сорс код – описание на функционалността
6. Заключение
7. **Описание**

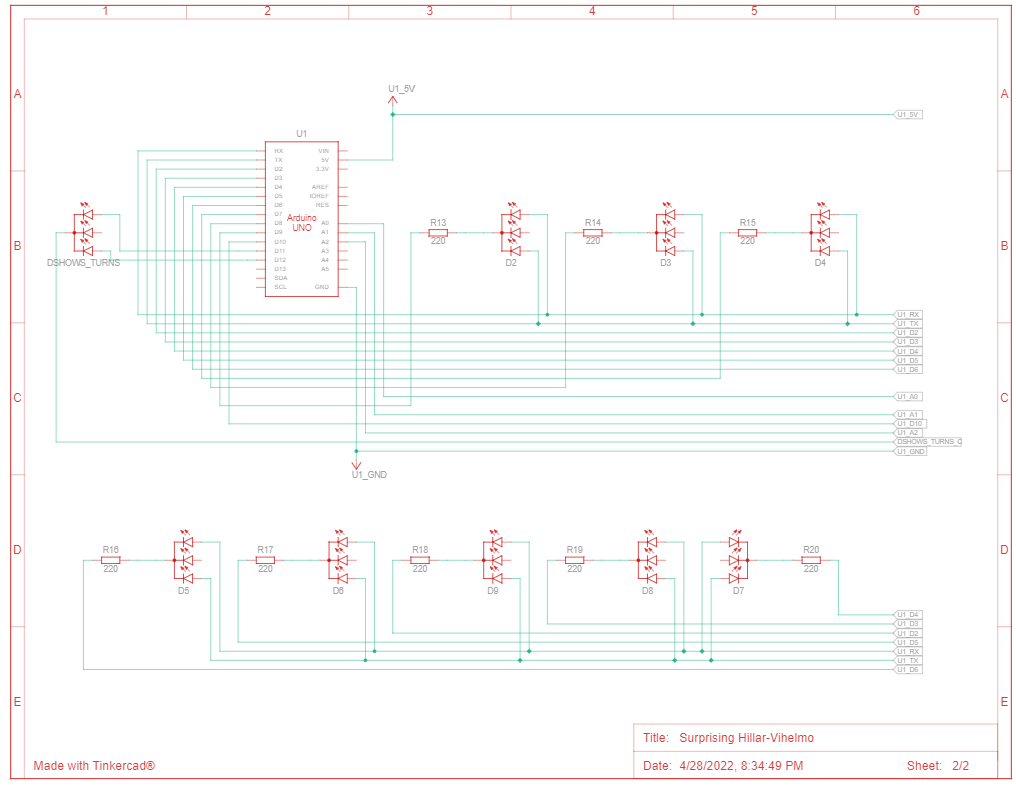
Проектът „Морски шах“ е буквално имплементация на световноизвестната игра на хартия морски шах. За проекта е използвана онлайн платформата Tinkercad, която поддържа симулация на Arduino. Играта може да бъде играна както от двама души, така и от срещу компютър с избираема трудност – лесно и трудно.

1. **Блокова схема**



1. **Електрическа схема**





Линк за изтегляне на електрическата схема: [Електрическа схема: МОРСКИ ШАХ](https://drive.google.com/file/d/1XuPTO5SMDypJGZxdfu_yvFEWdHY2_UmH/view?usp=sharing)

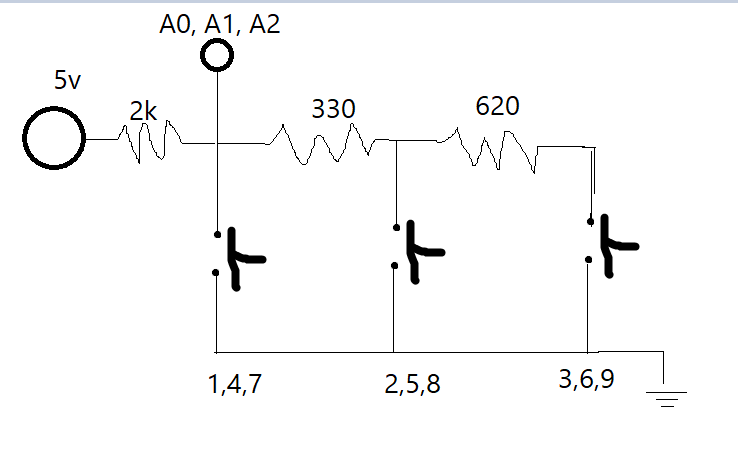
1. **Списък на съставни части**



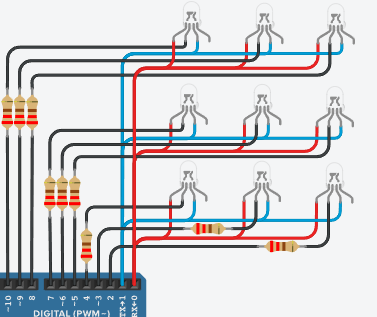
1. **Сорс код – описание на функционалността**

[**МОРСКИ ШАХ - АРДУИНО КОД**](https://pastebin.com/nXffewvP)

Поради недостиг на дигитални портове, се наложи бутоните да бъдат свързани към аналоговите портове и чрез ADC (Analog to Digital Converter) се получава INPUT.



Всички червени и сини краища на RGB светодиодите са свързани заедно, а само катодите им са вързани към отделни дигитални портове, за да могат да бъдат активирани един по един.



За да бъдат визуализирани едновременно, ардуиното включва и изключва всеки светодиод поне 30 пъти в секундата, за да бъде невидимо изгасването на светлините за човешкото око.

АЛГОРИТЪМ:

Кодът e реализиран чрез следните методи:

* Setup() – начален метод при включване на ардуиното
* Loop() – метод, който се повтаря безкрайно
* ChooseGamemode() – метод, чрез който се избира каква вариация на игра да започне
* DrawBoard() – чертае дъската
* GetInput() – получава входни данни от аналоговите портове
* CheckForWin() – проверява и обявава при наличност край на играта
* BlinkBoard() – пресвятка светодиодите
* ResetGame() – връща до началния екран
* ChechMovesLeft() – проверя дали има още празни полета на дъската
* EvaluateBoard() – пресмята в каква позиция се намира компютърът
* MiniMaxAlgorithm() – анализира всички ходове в дълбочина и избира най-добрия
* AITurn() – прави ход за компютъра
* RandomAITurn() – прави случаен ход за компютъра
* ChangeAllBoard() – променя цялата дъска в съответен цвят/спира всички светодиоди.

За логиката на компютъра е имплементиран алгоритъма MINIMAX и в него е вложен и ALPHA BETA PRUNING за допълнителна скорост при компютъране на резулата.

1. **Заключение**

Проектът беше доста приятен за правене, научихме много, макар и по трудния начин.