

# Gamecontroller



# Inhalt

- Definition: "Gamecontroller"
- Typen und Nutzbarkeit
- Komponenten und Nutzbarkeit
- Design-Vorschlag
- Interner Aufbau

# Definition: "Gamecontroller"

- Eingabegeräte für Spiele (PC / Konsole)
- Beispiele:
  - Tasten- oder Bewegungs-Gamepads
  - Joysticks
  - Lenkräder

# Typen und Nutzbarkeit

- Tasten-Gamepad
  - Gut geeignet für unser Projekt
- Joystick
  - Zu aufwändig / keine gute Ausnutzung
- Lenkrad
  - Ungeeignet
- Bewegungs-Gamepad (z.B. Wii-Controller)
  - ungeeignet

# Komponenten und Nutzbarkeit

- Button
  - An / Aus
- Steuerkreuz
  - Feste Anzahl an Bewegungsrichtungen
- Analog-Stick
  - Unbegrenzte Anzahl an Richtungen und variabler Neigungsgrad (z.B. Laufen / Sprinten)

# Buttons

- Essentiell für Bestätigung der Eingabe
- Nutzbar für weitere Optionen:
  - Pause?
  - Letzter Zug rückgängig?
  - ...

# Steuerkreuz

- Essentiell für '4 gewinnt': rechts / links
- Optional: klassisches Steuerkreuz für weitere Spiele (z.B. TicTacToe, ...)
- Eigenbau aus 4 Tasten (Nintendo® besitzt Patent auf "Steuerkreuz in Kreuzform")

# Analog-Stick

- Komplexeste Komponente
- Ähnlich wie Joystick: schlechte Ausnutzung der vorhandenen Möglichkeiten  
(Bewegungsrichtungen und Neigungsgrad)
- Nicht für unser Projekt geeignet

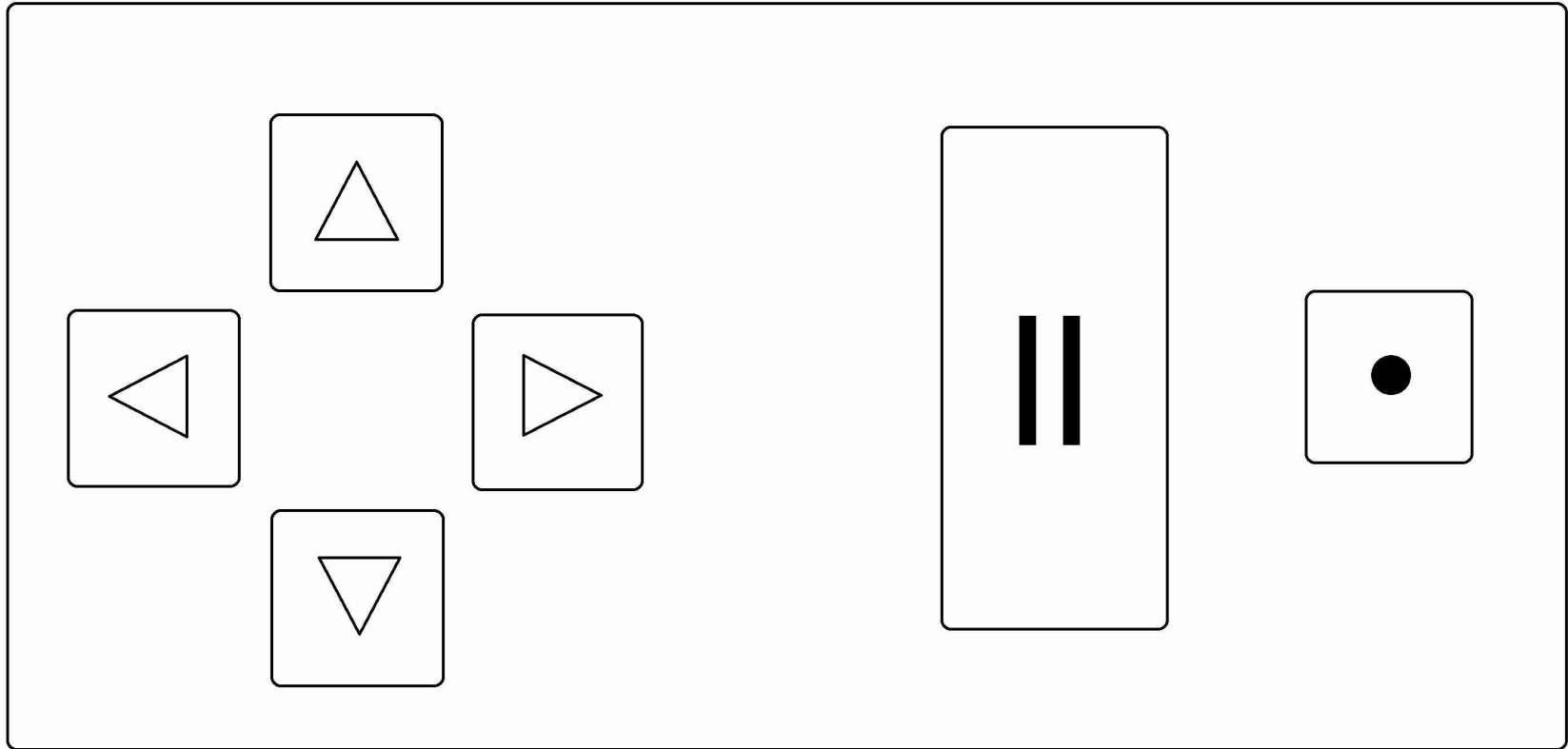


# Design-Vorschlag

- Orientierung am Super-Nintendo-Controller (Struktur)
- Komponenten:
  - Eigenbau-Steuerkreuz
  - OK-Button
  - Pause-Button



# Schematischer Aufbau



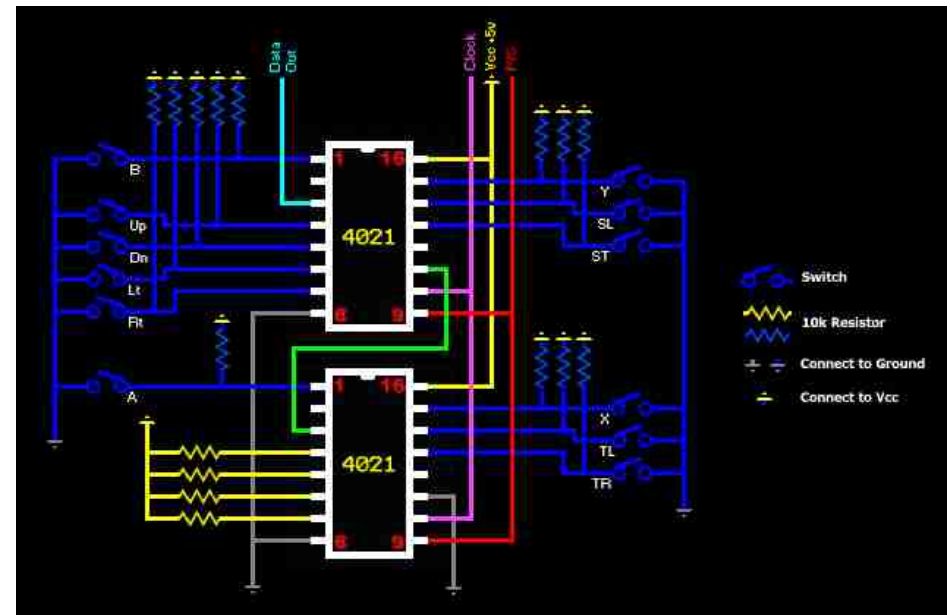
# Interner Aufbau am Beispiel des SNES (Supernintendo)-Controllers

- neben Steuerkreuz 6 Buttons (4 Buttons mehr als für unser Projekt?)



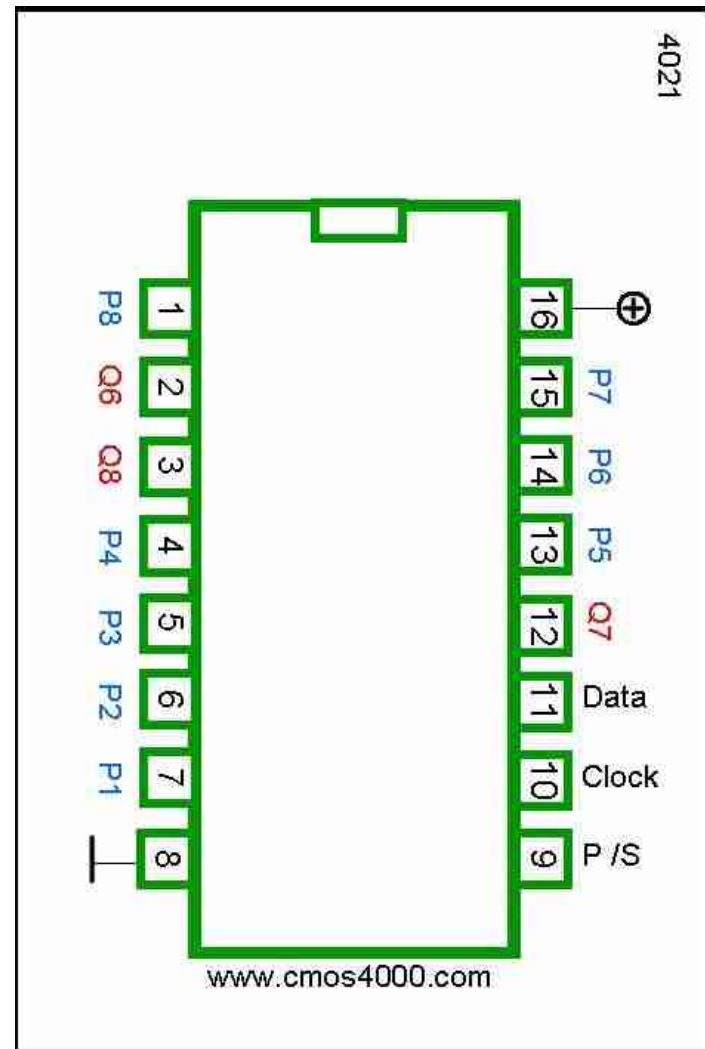
# Komponenten

- Schalter für Buttons
- 2 x CMOS 4021
- 10 kOhm-  
Widerstände
- GND und Vcc



# CMOS 4021

- 8-stufiges Schieberegister
- P/S low: Daten von Data zu Out (Q6 bis Q8)
- P/S high: 8-Bit-Datenwort an P1-P8 (wie bei Super-nintendo) unmittelbar laden
- bei  $V_{cc} = 5V$ : 2.5 Mhz Taktrate



# Datenformat Supernintendo

- 2 4021-Register ->  $2 \cdot 8 = 16$  Bit nutzbar

SNES Data Format

Operation	Latch	CLK	CLK	CLK	CLK	CLK	CLK	CLK	CLK	CLK	CLK	CLK	CLK	CLK	CLK	CLK
	bit 0	bit 1	bit 2	bit 3	bit 4	bit 5	bit 6	bit 7	bit 8	bit 9	bit 10	bit 11	bit 12	bit 13	bit 14	bit 15
Data	B	Y	Select	Start	Up	Down	Left	Right	A	X	L	R	1	1	1	1

# Verwendung für unser Projekt

- 4 Buttons weniger (nur OK und Pause) -> 4 Schalter weniger
- benötigte Leitungen:
  - 6: 6 Buttons
  - 2: GND und Vcc
  - 2: Clock und P/S
  - Summe: 10 Leitungen -> auch 2 x CMOS 4021 ()

# Alternativen

- einfacheres Konzept:
  - pro Button eine 1-Bit-Leitung (low / high)
  - left / right, up / down, OK, Pause -> 6 1-Bit-Leitungen als Out



# Quellenangaben

[http://www.ps3-repairparts.com/photo/pl596673-replacement\\_dualshock\\_sixaxis\\_2\\_4ghz\\_rf\\_wireless\\_game\\_controller\\_with\\_receiver\\_for\\_ps3.jpg](http://www.ps3-repairparts.com/photo/pl596673-replacement_dualshock_sixaxis_2_4ghz_rf_wireless_game_controller_with_receiver_for_ps3.jpg)

<http://www.techfuels.com/attachments/gaming-accessories/5933d1222511480-saitek-industries-x52-flight-control-system-game-controller-saitek-industries-x52-flight-control-system-game-controller.jpg>

[http://image.dhgate.com/albu\\_262590082\\_00-1.0x0/steering-wheel-game-controller-for-nintendo.jpg](http://image.dhgate.com/albu_262590082_00-1.0x0/steering-wheel-game-controller-for-nintendo.jpg)

<http://images.amazon.com/images/G/01/videogames/detail-page/WiiRemote1.jpg>

[http://saschak.ch/wp-content/uploads/2010/04/snes\\_controller.gif](http://saschak.ch/wp-content/uploads/2010/04/snes_controller.gif)

<http://de.wikipedia.org/wiki/Gamecontroller>

<http://tk-421projects.blogspot.de/2012/12/snes-controller-protocol.html>

<http://www.cmos4000.de/cmos/4021.html>

<http://www.cmos4000.de/media/cmos/ic-cmos-4021.pdf>