### How to Test Quantum Causality with the Neutron Interferometer





Neutrons for Europe

**USER MEETING** 

2022

ess



- Motivation
- Preliminaries
- Quantum Causality: Delayed Choice Cheshire Cat Experiment

### **Motivation**

<u>Causality</u>

Relationship between cause and effects

Classically an effect cannot occur before the cause.

In QM this fixed order is loosened as can be demonstrated in an experiment with a delayed choice setting

### **Motivation**

<u>Causality</u>

Relationship between cause and effects

Classically an effect cannot occur before the cause.

In QM this fixed order is loosened as can be demonstrated in an experiment with a delayed choice setting

 Find experiment for the neutron interferometer to put both the concept to the test

### **Motivation**

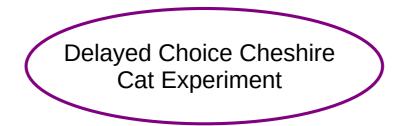
<u>Causality</u>

Relationship between cause and effects

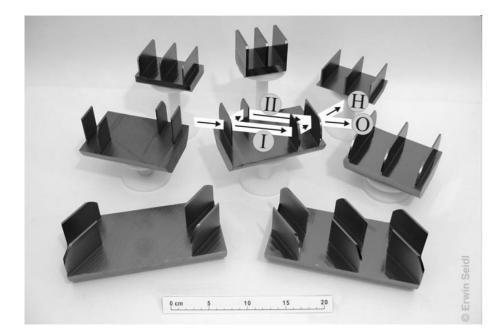
Classically an effect cannot occur before the cause.

In QM this fixed order is loosened as can be demonstrated in an experiment with a delayed choice setting

 Find experiment for the neutron interferometer to put both the concept to the test



- Neutrons have wave and particle properties  $p = mv = h/\lambda$  (L. De Broglie)
- Perfect Silicon Crystals can be utilized to build interferometer for neutrons
- Can be used to demonstrate fundamental laws of quantum mechanics (e.g.  $4\pi$  symmetry of the wave function)



#### Self-interference



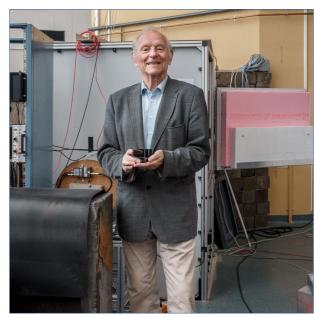
- Demonstrated successfully in 1974 by Helmut Rauch et al. at the Atominstitut Vienna, Austria
- Due to this the Atominstitut was granted 2019 the status of an

EPS (European Physical Society) Historic Site

- Soon afterwards plans for experiments at the ILL
  - ➔ today the instrument S18

7





H.Rauch 1939-2019

- Demonstrated successfully in 1974 by Helmut Rauch et al. at the Atominstitut Vienna, Austria
- Due to this the Atominstitut was granted 2019 the status of an



- EPS (European Physical Society) Historic Site
- Soon afterwards plans for experiments at the ILL
  - $\rightarrow$  today the instrument S18

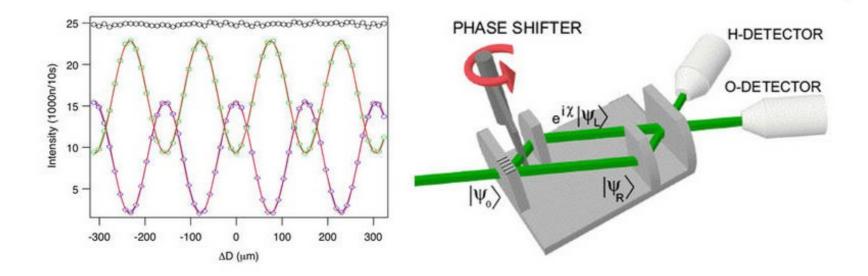


H.Rauch 1939-2019

# Anton Zeilinger made his PhD ThesisTriviaunder Rauch;<br/>worked on neutron interferometry (at ILL)<br/>before moving on to photons

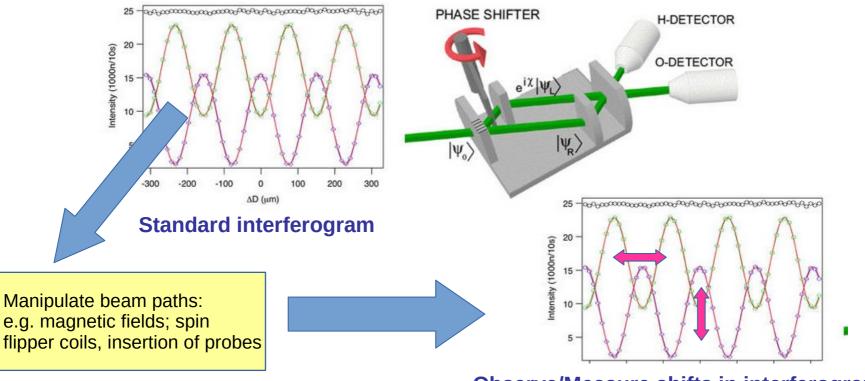


### **Basic Measuring Principle**



Standard interferogram

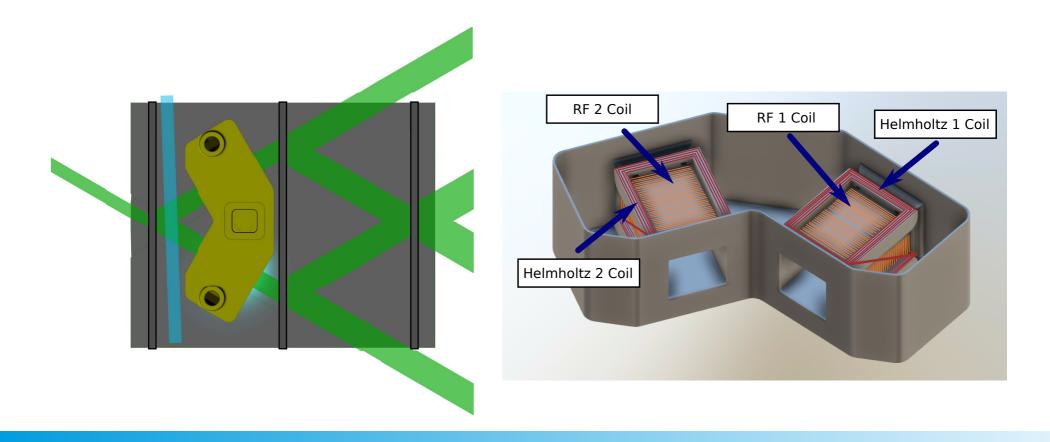
### **Basic Measuring Principle**



**Observe/Measure shifts in interferogram** 

# **Spin Manipulation**

Separate manipulation of neutron wave function in the individual paths of the interferometer



# **Spin Manipulators**

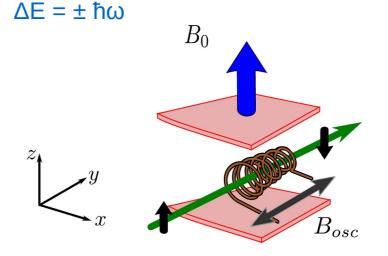
#### **Direct Current (DC) Spin Rotator**

- Change direction of the neutron spin
- Utilizing Larmor precession
- Rotation of Spin around perpendicular magnetic field

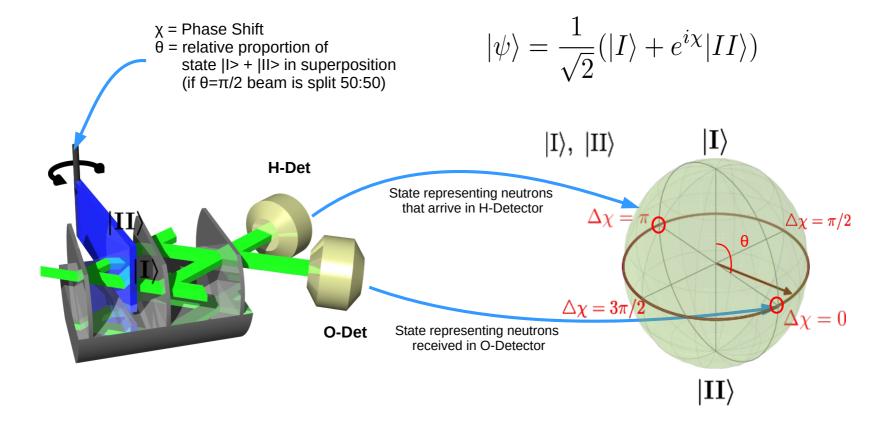
# GF B<sub>z</sub> B<sub>y</sub> t y

#### **Radio Frequency (RF) Spin Rotator**

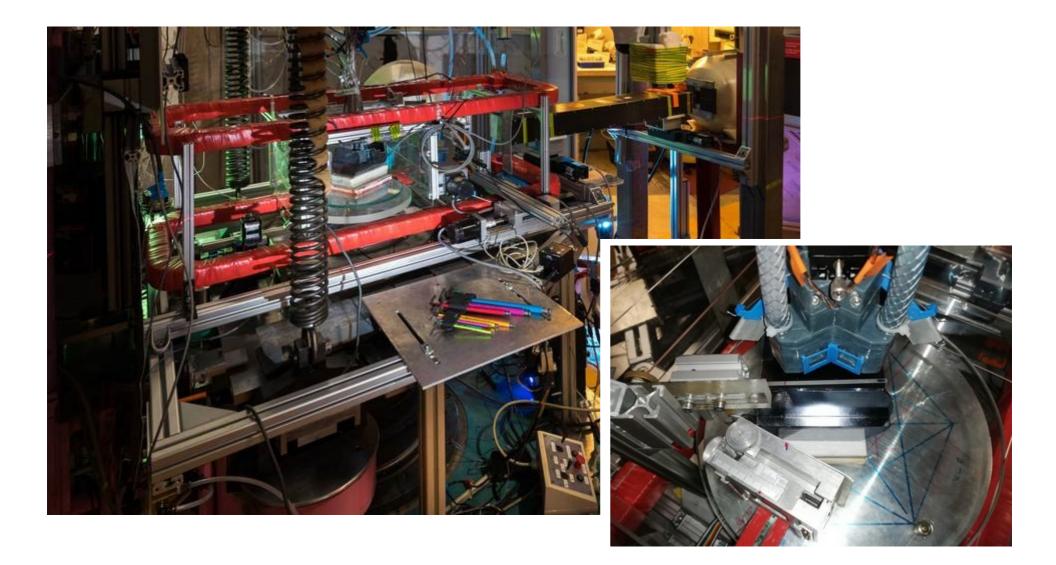
- Small oscillation field  $B_{osc}$  at resonance frequency  $\omega$  of guide field  $B_0$
- Neutrons end up rotated with changed energy



• For the following experiments we utilize the properties of the interferometer as a two level system



### Typical Experimental Setup at the S18



#### Weak Measurement - Weak Value

- Proposed by Yakir Aharonov, David Z. Albert and Lev Vaidman (AAV) (1988)
- Describes the effect a quantum system has on a measuring device (probe system) when weakly interacting
- Coupling with minimal disturbance

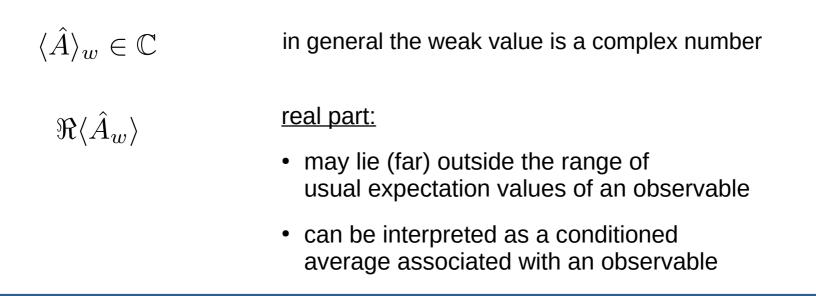
Definition: 
$$\langle \hat{A} \rangle_w = \frac{\langle \psi_f | \hat{A} | \psi_i \rangle}{\langle \psi_f | \psi_i \rangle}$$

$$|\phi_{\rm f}\rangle \simeq \int dp \, e^{-\Delta^2 (p - \langle A \rangle_{\rm w})^2} |p
angle$$

WV appears shift in the gaussian of the meter system during a weak measurement

#### Weak Value – Weak Measurement

#### • Properties of Weak Value

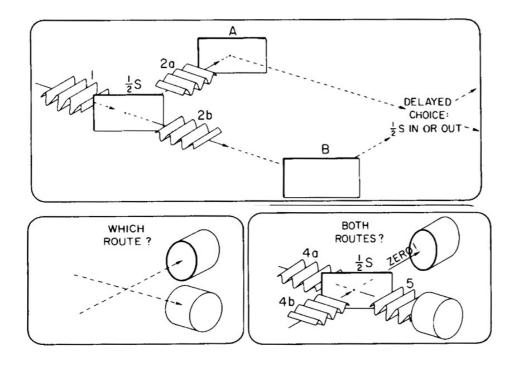


- Weak measurement in neutron interferometer
  - Induce path (I and II) dependent spin rotations α
  - Coupling of path and spin
  - If  $\alpha$  small  $\Rightarrow$  weak measurement criteria fulfilled

#### Quantum Causality

### Delayed Choice Cheshire Cat Experiment

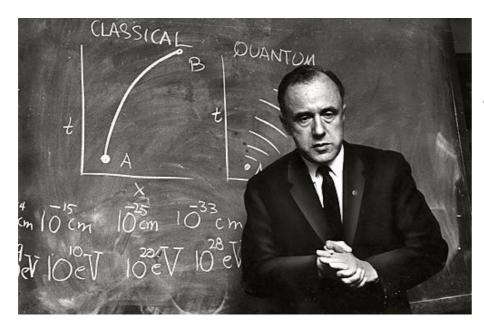
### **Delayed Choice Experiment I**



(Wheeler, 1984 – Law without Law)

- Interference or Which-Way (path taken) information depends whether beam splitter (BS) is inserted or not
- Delay Choice whether to insert BS until particle has entered interferometer
- But: → delayed choice is not altering result of measurement
- By putting in the BS we have an effect on an event in the past
- Retro causal effect

### **Delayed Choice Experiment II**



• Wheeler: "No phenomenon is a phenomenon until it is an observed phenomenon."

- Delayed Choice Experiments have been realized with photons
- Only known delayed choice experiment with Neutrons: Kawai et. al. (1998) "Realization of a delayed choice experiment using a multilayer cold neutron pulser"

#### **Quantum Cheshire Cat**

- Cheshire cat is a character from Lewis Carrols novel 'Alice's Adventures in Wonderland'
  - Cat leaves the scene but grin stays behind
- Aharonov and Rohrlich (2005) asked the question: Can a property be separated ('disembodied') from an object?
- For classical systems impossible
- But in the quantum realm
  - ➔ Formulation of the Quantum Cheshire Cat (qCC)



PHYSICS TEXTBOOK Yakir Aharonov Daniel Rohrlich

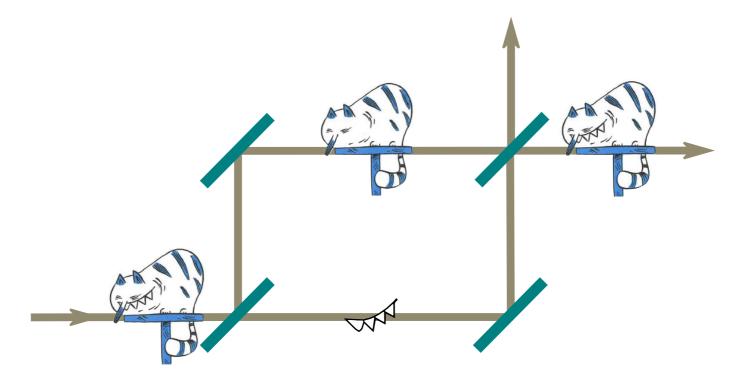
Quantum Paradoxes

Quantum Theory for the Perplexed



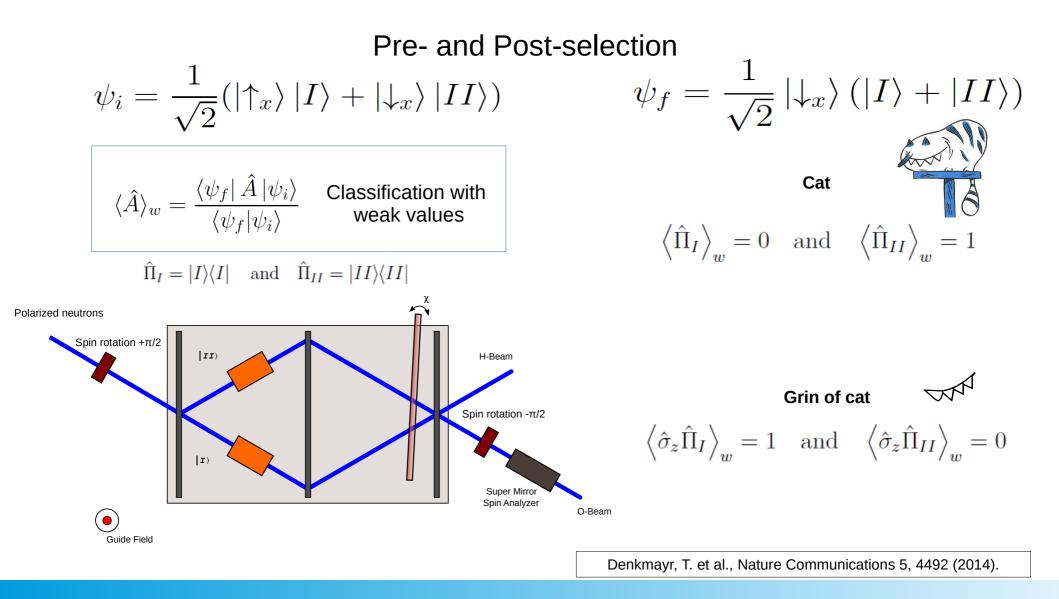
#### **Quantum Cheshire Cat**

- In an interferometer experiment the cat is represented by the neutron and the grin by its magnetic moment/spin (property).
- qCC is realized if neutron and spin take different paths in the interferometer and hence are spatially separated

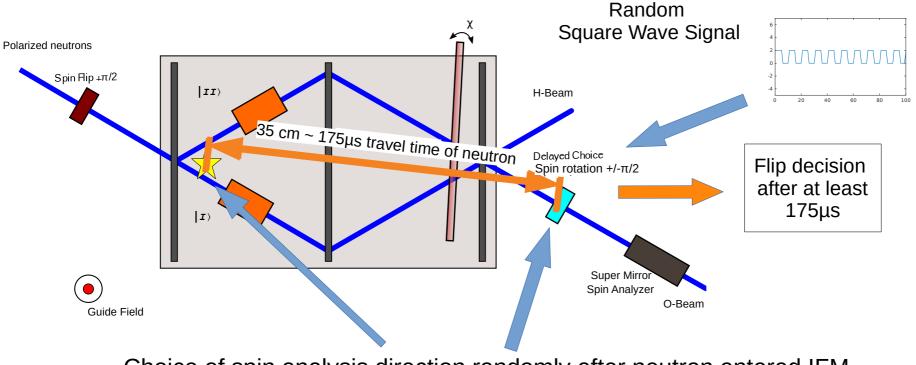


Cat © N.Mahler

#### **Quantum Cheshire Cat**

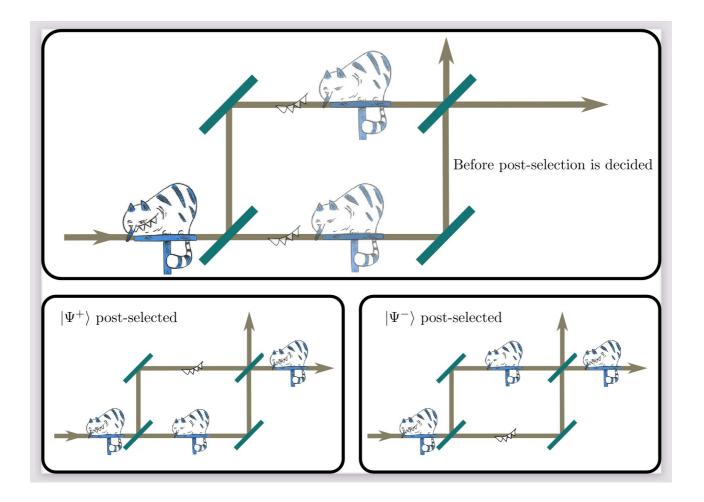


• Delayed Choice Extension ("delayed post-selection")



Choice of spin analysis direction randomly after neutron entered IFM

- Behavior of qCC is switched with state of Delayed Choice Coil
- Separation of neutron and its property occurs at the moment of post-selection



#### Random Number Generator

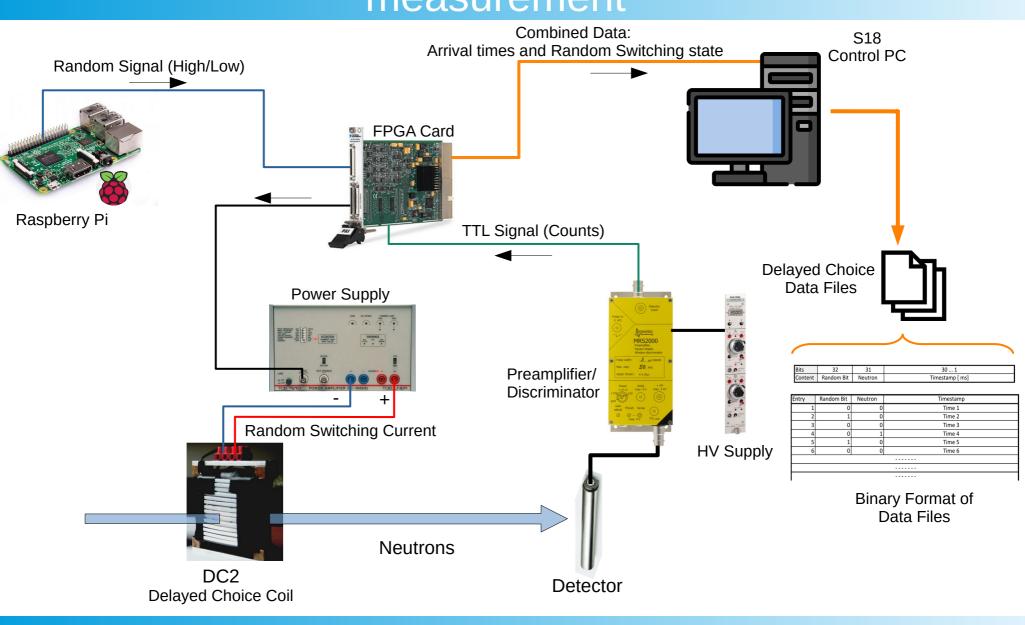
- Realization with Raspberry Pi HWRNG on Pi's SoC
- With rng-tools-package

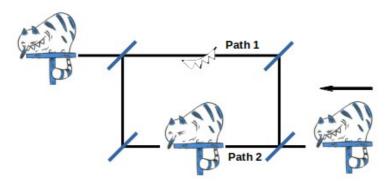
   → True Random (!)

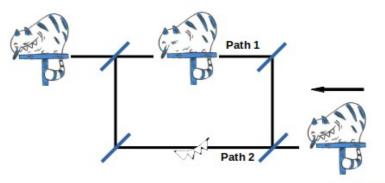


<pre>\$</pre>
<pre>2 # 3 import RPi.GPIO as GPIO 4 import time 5 import sys 6 import random 7 import os 8 import wiringpi 9 10 GPIO.setmode(GPIO.BOARD) 11 #OutputPin 11 == GPIO 17 12 GPIO.setup(11, GPIO.OUT) 13 14 #use hardware random generator 15 #rnd = random.SystemRandom() 16 f = os.open('/dev/hwrng', os.O_RDONLY)</pre>
<pre>3 import RPi.GPIO as GPIO 4 import time 5 import sys 6 import random 7 import os 8 import wiringpi 9 10 GPIO.setmode(GPIO.BOARD) 11 #OutputPin 11 == GPIO 17 12 GPIO.setup(11, GPIO.OUT) 13 14 #use hardware random generator 15 #rnd = random.SystemRandom() 16 f = os.open('/dev/hwrng', os.O_RDONLY)</pre>
<pre>4 import time 5 import sys 6 import random 7 import os 8 import wiringpi 9 10 GPIO.setmode(GPIO.BOARD) 11 #OutputPin 11 == GPIO 17 12 GPIO.setup(11, GPIO.OUT) 13 14 #use hardware random generator 15 #rnd = random.SystemRandom() 16 f = os.open('/dev/hwrng', os.O_RDONLY)</pre>
<pre>5 import sys 6 import random 7 import os 8 import wiringpi 9 10 GPI0.setmode(GPI0.BOARD) 11 #OutputPin 11 == GPIO 17 12 GPI0.setup(11, GPI0.OUT) 13 14 #use hardware random generator 15 #rnd = random.SystemRandom() 16 f = os.open('/dev/hwrng', os.0_RDONLY)</pre>
<pre>6 import random 7 import os 8 import wiringpi 9 10 GPIO.setmode(GPIO.BOARD) 11 #OutputPin 11 == GPIO 17 12 GPIO.setup(11, GPIO.OUT) 13 14 #use hardware random generator 15 #rnd = random.SystemRandom() 16 f = os.open('/dev/hwrng', os.O_RDONLY)</pre>
<pre>7 import os 8 import wiringpi 9 10 GPI0.setmode(GPI0.BOARD) 11 #OutputPin 11 == GPIO 17 12 GPI0.setup(11, GPI0.OUT) 13 14 #use hardware random generator 15 #rnd = random.SystemRandom() 16 f = os.open('/dev/hwrng', os.0_RDONLY)</pre>
<pre>8 import wiringpi 9 10 GPI0.setmode(GPI0.BOARD) 11 #OutputPin 11 == GPI0 17 12 GPI0.setup(11, GPI0.OUT) 13 14 #use hardware random generator 15 #rnd = random.SystemRandom() 16 f = os.open('/dev/hwrng', os.0_RDONLY)</pre>
<pre>9 10 GPI0.setmode(GPI0.BOARD) 11 #OutputPin 11 == GPI0 17 12 GPI0.setup(11, GPI0.OUT) 13 14 #use hardware random generator 15 #rnd = random.SystemRandom() 16 f = os.open('/dev/hwrng', os.0_RDONLY)</pre>
<pre>10 GPIO.setmode(GPIO.BOARD) 11 #OutputPin 11 == GPIO 17 12 GPIO.setup(11, GPIO.OUT) 13 14 #use hardware random generator 15 #rnd = random.SystemRandom() 16 f = os.open('/dev/hwrng', os.O_RDONLY)</pre>
<pre>11 #OutputPin 11 == GPIO 17 12 GPIO.setup(11, GPIO.OUT) 13 14 #use hardware random generator 15 #rnd = random.SystemRandom() 16 f = os.open('/dev/hwrng', os.0_RDONLY)</pre>
<pre>12 GPI0.setup(11, GPI0.OUT) 13 14 #use hardware random generator 15 #rnd = random.SystemRandom() 16 f = os.open('/dev/hwrng', os.0_RDONLY)</pre>
<pre>13 14 #use hardware random generator 15 #rnd = random.SystemRandom() 16 f = os.open('/dev/hwrng', os.0_RDONLY)</pre>
<pre>14 #use hardware random generator 15 #rnd = random.SystemRandom() 16 f = os.open('/dev/hwrng', os.O_RDONLY)</pre>
<pre>15 #rnd = random.SystemRandom() 16 f = os.open('/dev/hwrng', os.O_RDONLY)</pre>
16 f = os.open('/dev/hwrng', os.O_RDONLY)
[/ #print(os, read(t, 1))
18 #define delay time of (endless) while loop
19 delay_microseconds = 33 # equals 50 $\mu$ s
20 # together with while loop execution time
20 # together with while loop execution time
22 while True:
23 #read 1 byte from hardware-rng-device
24 #test on first bit -> our random bit
25 random bit = $(ord(os.read(f,1)) \& 1)$
26 #set output pin to random bit state
27 GPIO.output(11, <i>bool</i> (random_bit))
28 #waitwiringpi.delayMicroseconds(delay ms) for delay microseconds
29 wiringpi.delayMicroseconds(delay_microseconds)
30
31 f = os.close()

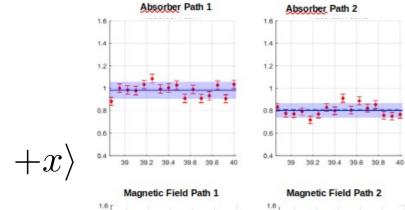
# Schematic Overview – Time resolved measurement

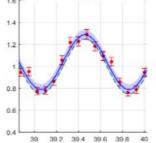


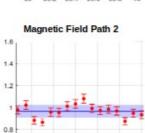




Cat © N.Mahler



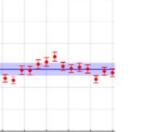




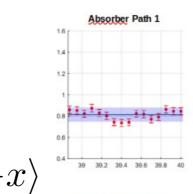
39 39.2 39.4 39.6 39.8

0.6

0.4



40



1.6

1.4

1.2

0.8

0.6

0.4

39

39.2 39.4 39.6 39.8

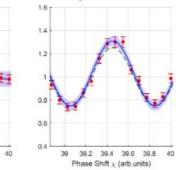
Phase Shift x (arb.units)

0.6 39 39.2 39.4 39.6 39.8 40 Magnetic Field Path 1 Magnetic Field Path 2 1.6

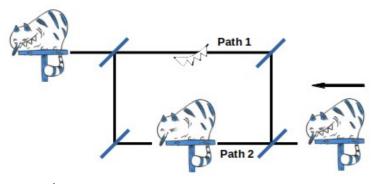
1.6

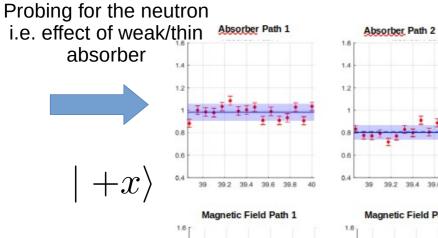
1.4

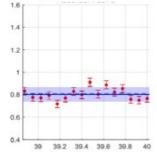
12

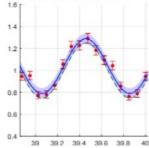


Absorber Path 2



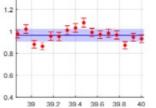


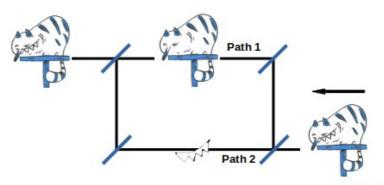




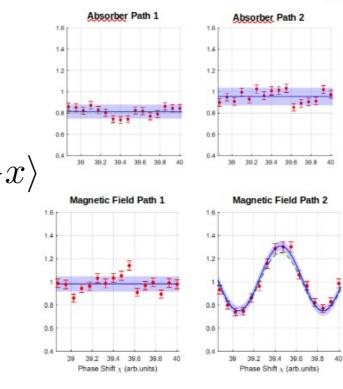


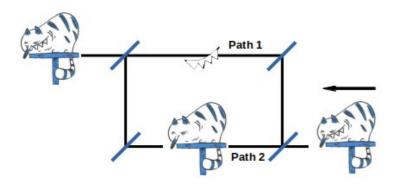
1.4

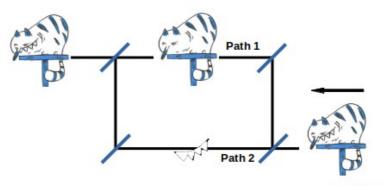




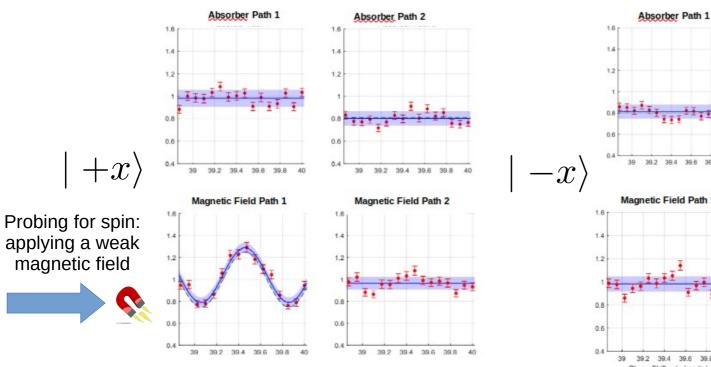
Cat © N.Mahler

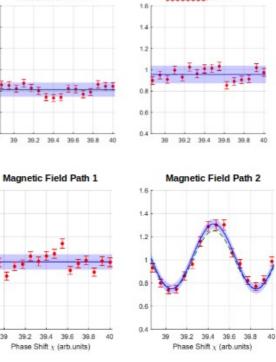






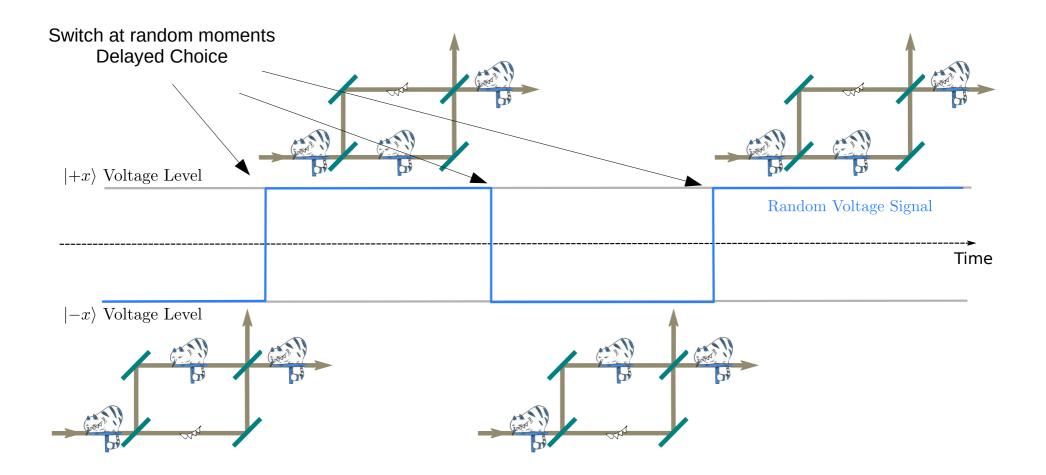
Cat © N.Mahler





Absorber Path 2

#### Evolvement of the qCC state over time



# Conclusion

- Realization of a qCC: neutron and one of its properties are spatially separated inside the interferometer
- Observation of the influence of an in the future lying random delayed choice
- The random selection/choice decides on the location of the neutron and the property
- Demonstration of quantum causality  $\rightarrow$  conventional notion of causal order not valid anymore

#### Quantum Causality emerging in a Delayed-Choice Quantum Cheshire-Cat Experiment with Neutrons

Soon to be published

Richard Wagner<sup>1,2,\*</sup>, Wenzel Kersten<sup>1</sup>, Hartmut Lemmel<sup>1,2</sup>, Stephan Sponar<sup>1</sup>, and Yuji Hasegawa<sup>1,3</sup>

<sup>1</sup>Atominstitut, TU Wien, Stadionallee 2, 1020 Vienna, Austria
 <sup>2</sup>Institut Laue Langevin, 38000 Grenoble, France
 <sup>3</sup>Department of Applied Physics, Hokkaido University, Kita-ku, Sapporo 060-8628, Japan

\*wagnerrichard@ill.fr

#### Thank you for your attention!

