

# K&K FXE / FXQE80 - USB

Company: <http://www.kplusk-messtechnik.de/index.html>

## Manuals

- FDI 120225
- FXE 160826
- FXE 081203
- Manual\_2020
- Handbuch\_2020

## Hardware

- New Counter (FXE07 & FXE04) from 2018

## Important Notes of Hardware

1. Es müssen immer **alle** acht Eingänge ein Signal bekommen. Ansonsten gibt es thermische Effekte, die die Zähler die Performance kaput macht!
  1. Auch wenn man zum Beispiel nur aus 2 Kanälen die Daten speichern möchte und diese im Programm auswählt, werden ständig alle acht Kanäle gemessen!
  2. Auch wenn zum Beispiel die Intervalldauer 100ms beträgt, werden alle 1ms die Daten vermessen und nur alle 100ms gespeichert!
2. Noisefloor des Messgeräts ist  **$5 \times 10^{-13}$  in 1s** mit Mittelung (= 12ps Zeitauflösung für 1ms Rohmessung)
3. Die acht Eingänge können:
  1. **4kHz - 132MHz** zählen
  2. **~ 0dBm - 17dBm**
    1. für 13dBm@10MHz ist der instabilitätswert bei 1s am besten
    2. unter 8dBm kann die Performance schlechter werden (bei 3dBm@10MHz beobachtet - private Kommunikation mit A. Kuhl/S.Koke)
4. Sync-Fähigkeit mit PPS
  1. Bei 1V ist Schaltschwelle
5. USB-Nummer: 000000EA (kann verändert werden)
6. Der Zähler kann als TCP-Server über LAN verwendet werden
  1. Nun können bis zu 4 unabhängige User einen Kanal benutzen
  2. Sie können unterschiedliche Modis gleichzeitig verwenden: gemittelt/ungemittelt etc.
  3. Der Server (Netzwerk) wird über die K+K Firmware (befindet sich auf der CD) ausgelesen und der Name/Nickname des Zählers kann angepasst werden

## 7. Der rote Resetknopf am Frontpanel resetet den Zähler komplett

1. startet dann mit Default
2. verwendet man nur, wenn der Computer den Zähler über USB nicht erkennt

# Software

## Bedienung der Version "FXX\_App 1.5.2" mit E-Mail Warnung

- Software Link: \\AFS\iqo.uni-hannover.de\projects\magnesium\Software\K+K Zähler mit E-Mail Meldung\FXX App Ver\_1\_4.zip
  - Vorteile:
    1. Überwachung von Eingangsfrequenzen und Referenztakt mit einstellbarer Fehlertoleranz
    2. Benachrichtigung per Email
- Read: Manual\_2020 and Handbuch\_2020
- Setups in software:
  - Root Folder: D:\Counter\_Data\
  - Debug files: C:\K+K\_Zähler\_FXX\_App\Debug\
  - Options:
    - Decimal separator window: point
    - Decimal separator logfiles: point
  - Info:
    - Mode: Phase Average
    - Interval: 1s
    - Scrambler: off
  - E-Mail Notification:
    - E-mail: 10mhz@iqo.uni-hannover.de
    - Notify: Always
    - Source: A
    - Classes:
      - All Errors, Warnings
      - All Infos
  - Config Email Server:
    - SMTP Host: mailgate.uni-hannover.de ([https://www.luis.uni-hannover.de/email\\_smtpserver.html](https://www.luis.uni-hannover.de/email_smtpserver.html))
    - Port: 587
    - Account name: 10mhz@iqo.uni-hannover.de
    - password: ask Steffen Sauer
    - Email Sender: 10mhz@iqo.uni-hannover.de
  - Clock Warnings:

- general:
  - Min. count before warning: 1
  - Next warning after **0** minutes
  - Add warnings to log files: no
- Channel 9: HMaser
  - Nominal frequency: 10,000,000.00 Hz
  - Margin: 0.000050 Hz
  - Warning Channel : Check
  - Warning reference Clock: Check
- Channel 10: Rb clock
  - Nominal frequency: 10,000,000.00 Hz
  - Margin: 0.001 Hz
  - Warning Channel : Check
  - Warning reference Clock: Check

## Bedienung der Version "K+K FXE 18-05-08"



Installation CD "K+K FXE 18-05-08" is in the ResLab (cabinet/computer manual)!



Von links nach rechts



## Data Transmission

### • Device

1. Ein Computer muss mit dem Zähler über USB verbunden sein
2. drei weitere Computer/Nutzer können den Zähler als TCP-Server im Netzwerk auslesen

### • User

1. Welcher User man ist: 1-4 möglich

### • Chnls = Channels

1. Anzahl der zu speicherten Channels

### • Intrvl = Intervalllength

1. Messdauer zwischen zwei gespeicherten Werten

1. You can change the length of the *interval*:

- Range: ~1 ms - 20 s
- For shorter length [shortest: 1ms] the memory would fill up very fast.
- Don't forget to adapt the length in the program *Allan-Live with time graph*!

## Local TCP Server

- **IP**

1. Auslesen mit K+K Firmware

## Log Files

- **Phase**

1. Speicherung der Phasenwerte

- **Frequency**

1. Speicherung der Frequenzwerte

- **Phase Difference**

1. Messung der Phasendifferenz
  1. Ch(n)-Ch(1): Bei allen Kanälen wird die Phasendifferenz zu Kanal 1 aufgenommen
  2. Ch(n)-Ch(n-1): gegen seinen Vorgängerkanal

- **Debug**

## Scramble

- **off** = off



- **Auto**

1. geht nur für "Average"-Mode

- **Trimm** = ?

## Report Mode = Select your measuring method

- Unordered List ItemDifference between  $\pi$ - and  $\Lambda$ -measurement:
  - $\pi$ -measurement = *Phase*
  - $\Lambda$ -measurement = *Phase Average* <color #ffa500>(Meistens)</fc>

Please note that I distinguish between the raw measurement results generated by the hardware, and the reports to the PC which are produced by the microprocessor.

Internally, the FXE counters continuously take raw measurements of the phase (relative to the 10MHz reference) of each channel every millisecond.

The on-board microprocessor can evaluate these raw measurements in different modes for presentation to the PC. I shall try to describe this by giving examples for each mode, where  $P_c(i)$  shall be the current raw 'P'hase measurement of channel 'c' at millisecond 'i', and I assume a report interval of 1s (=1000ms):

**a) Instantaneous phase report mode:** The on-board microprocessor reports to the PC at the selected report interval only the latest raw measurement result. All other intermediate results are

ignored. Measurements reported (@ interval 1s) will be

```
Pc( i )
Pc(i+1000)
Pc(i+2000)
Pc(i+3000) ...
```

**b) Averaged phase report mode** <color #ffa500>(meistens)</fc>: The on-board microprocessor calculates the average of all raw measurements taken during the report interval and reports to the PC at the end of the interval. All raw measurements contribute to the reported result. This helps to reduce measurement noise. Results reported (@ interval 1s) will be

```
Sum(n=0..999){Pc( i + n )}/1000
Sum(n=0..999){Pc(i+1000+n)}/1000
Sum(n=0..999){Pc(i+2000+n)}/1000
Sum(n=0..999){Pc(i+3000+n)}/1000 ...
```

Beispiel zur **Phase Average** Mode:

1. Mittelung über Zeit
  1. Zähler zählt alle 1ms
  2. Beispiel: 1s → Mittelung über 1000 Messungen → Damit wird durch die Mittelung die Messung einen Faktor  $\sqrt{N=1000}=30$  besser als nicht "Average"

**c) Phase difference report mode:** Inter-channel phase differences Pc-P1 are reported. These differences are the 'best possible' measurements, as any uncertainties of the reference cancel and most temperature effects cancel, as they are common to all channels. Measurements reported (@ interval 1s) will be

```
Pc( i ) - P1( i )
Pc(i+1000) - P1(i+1000)
Pc(i+2000) - P1(i+2000)
Pc(i+3000) - P1(i+3000) ...
```

**d) Averaged phase difference report mode:** Inter-channel differences of averaged phase Pc-P1 are reported. Measurements reported (@ interval 1s) will be

```
Sum(n=0..999){Pc( i + n ) - P1( i + n )}/1000
Sum(n=0..999){Pc(i+1000+n) - P1(i+1000+n)}/1000
Sum(n=0..999){Pc(i+2000+n) - P1(i+2000+n)}/1000
Sum(n=0..999){Pc(i+3000+n) - P1(i+3000+n)}/1000 ...
```

**e) Frequency report mode:** Frequency is Phase advance per time. The on-board microprocessor calculates the difference of the current phase measurement and the measurement taken one report interval earlier, and divides it by the report interval. Measurements reported (@ interval 1s) will be

```
[Pc( i ) - Pc(i-1000)]/1000ms
[Pc(i+1000) - Pc( i )]/1000ms
[Pc(i+2000) - Pc(i+1000)]/1000ms
```

$$[Pc(i+3000) - Pc(i+2000)]/1000ms \dots$$

**f) Averaged frequency report mode:** Averaged frequency is advance of averaged phase per time. The on-board microprocessor calculates the difference of the currents averaged phase and the averaged phase calculated one report interval earlier. Measurements reported (@ interval 1s) will be

$$\begin{aligned} & [Sum(n=0..999)\{Pc(i+n)\} - Sum(n=0..999)\{Pc(i-1000+n)\}]/1000ms \\ & [Sum(n=0..999)\{Pc(i+1000+n)\} - Sum(n=0..999)\{Pc(i+n)\}]/1000ms \\ & [Sum(n=0..999)\{Pc(i+2000+n)\} - Sum(n=0..999)\{Pc(i+1000+n)\}]/1000ms \\ & [Sum(n=0..999)\{Pc(i+3000+n)\} - Sum(n=0..999)\{Pc(i+2000+n)\}]/1000ms \dots \end{aligned}$$

By just re-arranging these sums you can also write

$$\dots = Sum(n=0..999)\{[Pc(i+n) - Pc(i-1000+n)]/1000\}$$

Hence, averaged frequency can be thought of either as 'advance of averaged phase' or as the average over 1000 non-averaged frequencies (@ interval 1s), in 1ms spacing.

Consequently, data measured during two(!) report intervals contribute to the averaged frequency value reported.

## Freq. Conv. = Frequency Conversion

- **off**

1. Ist eine Spezialfallanwendung, damit man die Phase zwischen 5MHz und 10MHz direkt vermessen kann

## Dec. Conv. = Decimal Separator

Wird zur Separierung der gespeicherten Daten verwendet.

- **W/N** für Windows
- **“,”**
- **“.”**

## Graph

Mit rechtsklick kann die Farbe und der Name der Channels angepasst werden.

- **ps/div**
  1. Achsenskalierung
- **PhDff**
  1. Phasendifferenz
- **relFre**

1. Mittelwert zu jedem sich selbst gemessener Frequenz

1. reCenter ab und zu betätigen, damit der Mittelwert wieder passt (Drift)

- **absFre**

1. Absolute Abweichung zum Mittelwert

## **Lin. Comb. = Lineare Comination**

Es ist möglich zu den 8 Kanälen eine lineare Kombination aus den Messdaten der Kanälen zu erstellen/berechnen und zu speichern. Sinnvoll für ein Transferbeat.

## **Persist**

Wird Persist aktiviert, bleibt die Graphik stehen und die nächsten Graphen werden überlagert.

## **Start a measurement**

Push buttons before measurement to erase RAM:

1. *Clear Buffer*-button = Reset
2. Rename the data file in KK\_FXE Folder
3. Start new measurement via "Häckchen" at *Frequency*



# Upgrade der Zählerkarten 2018

1. Die Zählerkarte **FXE02 (alt)** wurde zu einer **FXE07 (neu)** ersetzt. Die FXE02 war zu alt und konnte nicht upgegradet werden.
  1. Sync-Fähigkeit mit PPS
    1. Bei 1V ist Schaltschwelle
  2. Wird als Master-Zähler verwendet
2. Die Zählerkarte **FXE04** wurde upgegradet:
  1. USB-LAN-Buchse
    1. USB-Nummer: 000000EA (kann verändert werden)
  2. Ergänzung der Sync-Verkettung
  3. Wird weiterhin als Slave-Zählerkarte verwendet
3. Allgemein:
  1. Die acht Eingänge können:
    1. 4kHz - 132MHz zählen anstatt bis 65MHz
    2. ~ 0dBm - 17dBm
  2. Der Zähler kann als TCP-Server über LAN verwendet werden
    1. Nun können bis zu 4 unabhängige User einen Kanal benutzen
      1. Sie können unterschiedliche Modis gleichzeitig verwenden:  
gemittelt/ungemittelt etc
      2. Der Server (Netzwerk) wird über die K+K Firmware (befindet sich auf der CD) ausgelesen und der Name/Nickname des Zählers kann angepasst werden
  3. Verbindung mit dem Computer ist mit USB anstatt Serial möglich
    1. Dadurch können die Kanäle schneller ausgelesen werden. Anstatt 1 Kanal 1ms (bei Serial) bis zu 8 Kanäle jeweils 8 Kanäle 1ms (USB).
  4. Man hat nun die Möglichkeit, dass man einen zeitlichen Verlauf graphisch anzeigen kann

## Old Counter (FXE02 & FXE04)

- Possible range: 4kHz - 65MHz
- Recommended Input: 4-17 dBm (1-5V)

## Program

Default Settings: 8 Com Ports  
Trimmer: Auto Trimmer  
Measurement Method: Phase

## Intervall length

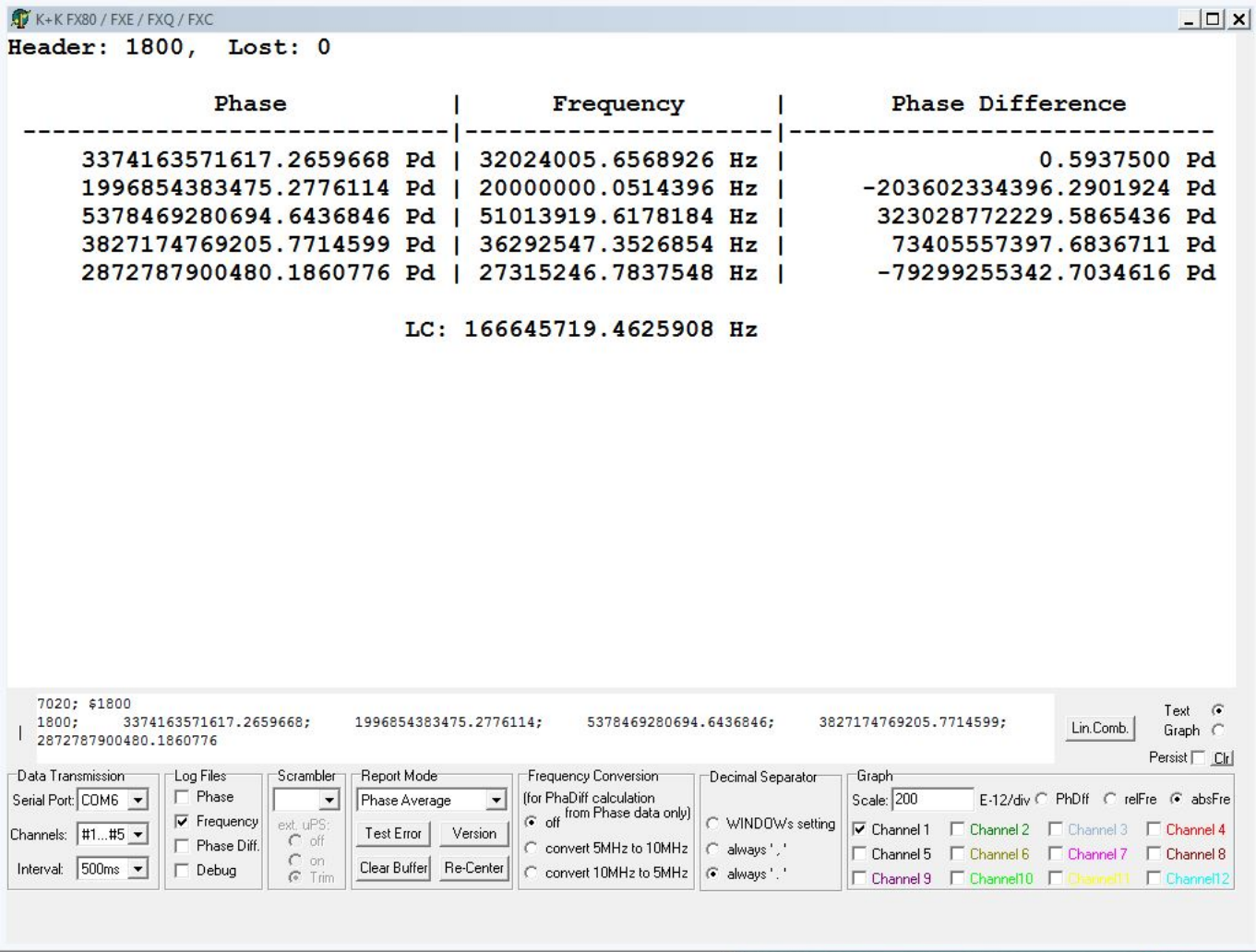
You can change the length of the *interval*:

- Range: ~1 ms - 1 s
- For shorter length [shortest: 1ms] the memory would fill up very fast.
- Don´t forget to adapt the length in the program *Allan-Live with time graph*!

Start a measurement

Push buttons before measurement to erase RAM:

1. *Clear Buffer*-button = Reset
2. Rename the data file in KK\_FXE Folder
3. Start new measurement via “Häckchen” at *Frequency*



From:  
<https://iqwiki.iqo.uni-hannover.de/> - IQwiki

Permanent link:  
[https://iqwiki.iqo.uni-hannover.de/doku.php?id=groups:mg:private:resonatoren:k\\_k\\_counter:start&rev=1710927339](https://iqwiki.iqo.uni-hannover.de/doku.php?id=groups:mg:private:resonatoren:k_k_counter:start&rev=1710927339)

Last update: 2024/03/20 09:35

