

Generierung des Eingangssingals für Barrier Bucket RF Systeme and der GSI



TECHNISCHE
UNIVERSITÄT
DARMSTADT

**Jonas Christ, Artem Moskalew, Maximilian Nolte
Jens Harzheim, M.Sc.**

Projektseminar Beschleunigertechnik



Institut für Theorie
Elektromagnetischer Felder
Computational Electromagnetics
Research Group at GSCE

Outline

- 1** Einführung
 - Problemstellung
 - Zielsetzung

- 2** Erreichtes
 - Gerätekommunikation
 - Code : das Design
 - Gegeben
 - Code : Evaluierung

- 3** Ausblick

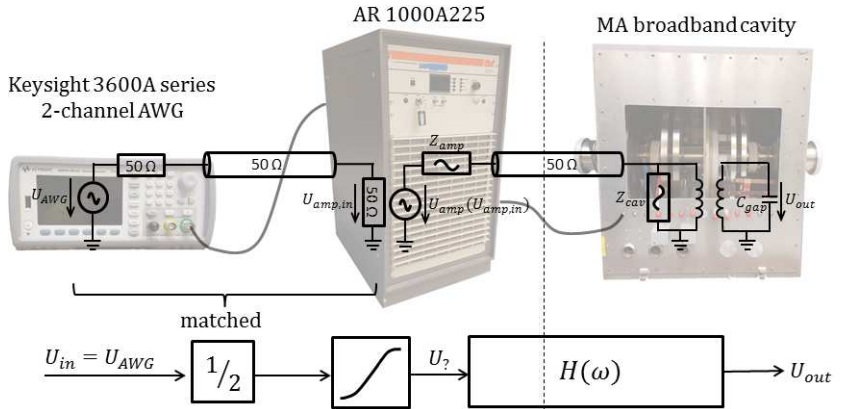


Problemstellung

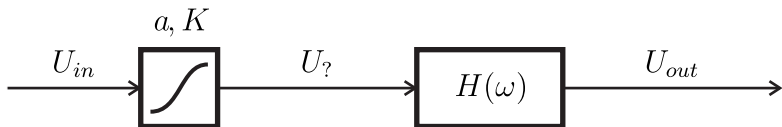
Zielsetzung

Erreichtes: das VISA-Handbuch

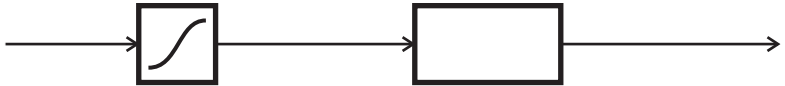
Erreichtes: Code



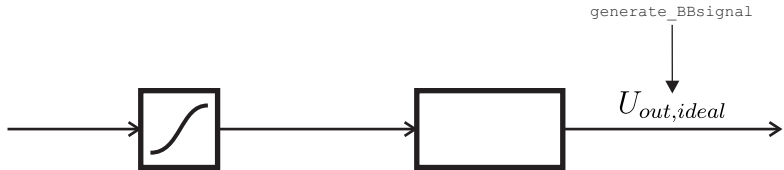
Erreichtes: Code



Erreichtes: Code

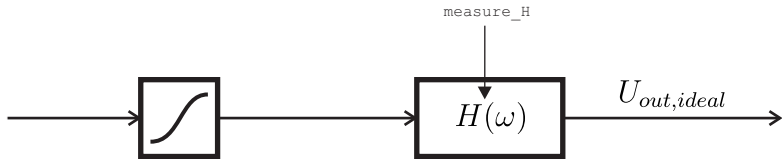


Erreichtes: Code



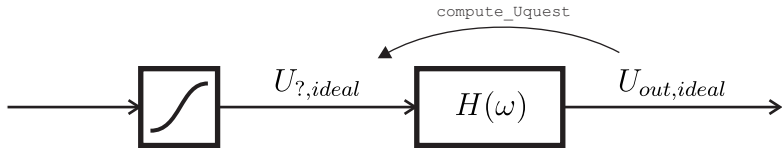
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
```

Erreichtes: Code



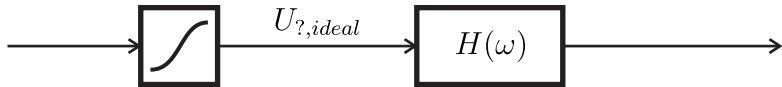
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )  
2 H = measure_H ( )
```

Erreichtes: Code



```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )  
2 H = measure_H ( )  
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
```

Erreichtes: Code



```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )  
2 H = measure_H ( )  
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
```

Erreichtes: Code



```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )  
2 H = measure_H ( )  
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
```

Erreichtes: Code



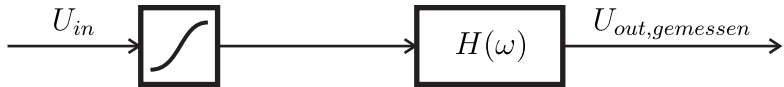
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )  
2 H = measure_H ( )  
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )  
4 Uin = Uquest_ideal
```

Erreichtes: Code



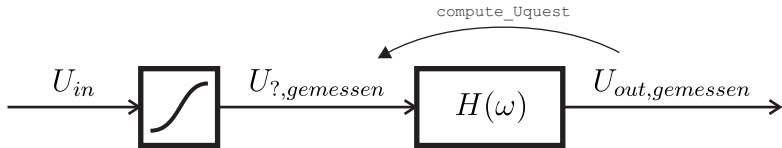
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )  
2 H = measure_H ( )  
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )  
4 Uin = Uquest_ideal  
5 Uout_measured = measure_Uout ( Uin )
```

Erreichtes: Code



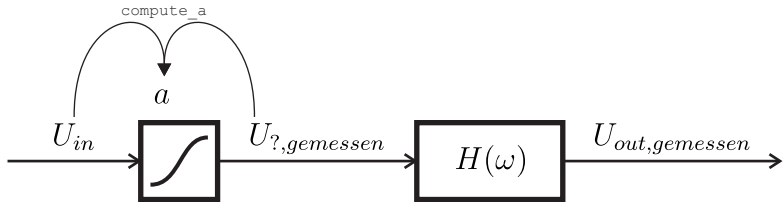
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )  
2 H = measure_H ( )  
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )  
4 Uin = Uquest_ideal  
5 Uout_measured = measure_Uout ( Uin )
```


Erreichtes: Code



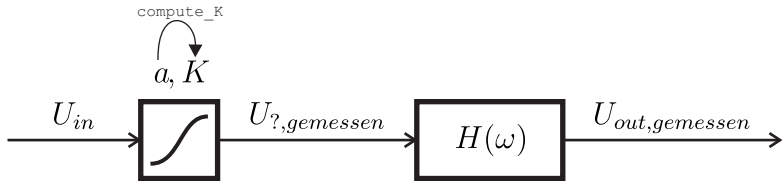
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
6 Uquest_measured = compute_Uquest ( Uout_measured , H )
```

Erreichtes: Code



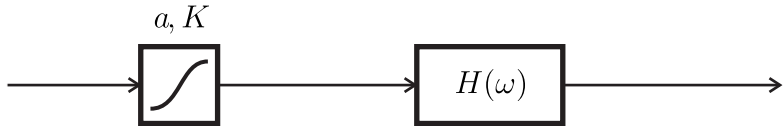
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
6 Uquest_measured = compute_Uquest ( Uout_measured , H )
7 a = compute_a ( Uin , Uquest_measured , N )
```

Erreichtes: Code



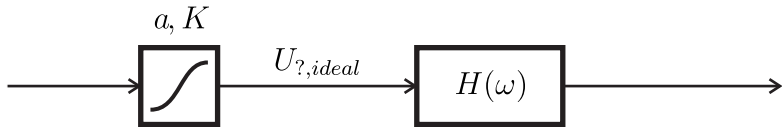
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
6 Uquest_measured = compute_Uquest ( Uout_measured , H )
7 a = compute_a ( Uin , Uquest_measured , N )
8 K = compute_K ( a )
```

Erreichtes: Code



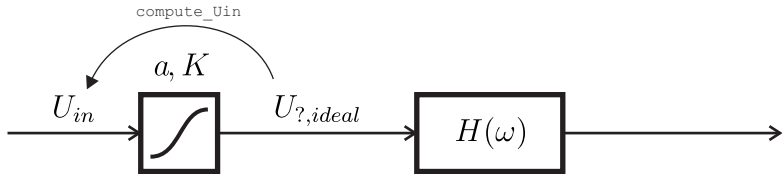
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )  
2 H = measure_H ( )  
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )  
4 Uin = Uquest_ideal  
5 Uout_measured = measure_Uout ( Uin )  
6 Uquest_measured = compute_Uquest ( Uout_measured , H )  
7 a = compute_a ( Uin , Uquest_measured , N )  
8 K = compute_K ( a )
```

Erreichtes: Code



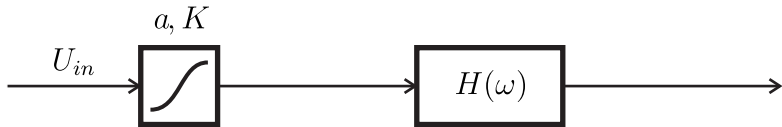
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
6 Uquest_measured = compute_Uquest ( Uout_measured , H )
7 a = compute_a ( Uin , Uquest_measured , N )
8 K = compute_K ( a )
```

Erreichtes: Code



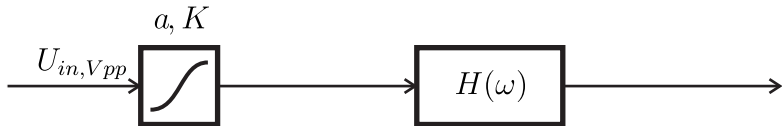
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
6 Uquest_measured = compute_Uquest ( Uout_measured , H )
7 a = compute_a ( Uin , Uquest_measured , N )
8 K = compute_K ( a )
9 Uin = compute_Uin ( Uquest_ideal , K )
```

Erreichtes: Code



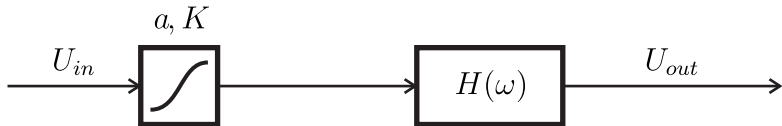
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
6 Uquest_measured = compute_Uquest ( Uout_measured , H )
7 a = compute_a ( Uin , Uquest_measured , N )
8 K = compute_K ( a )
9 Uin = compute_Uin ( Uquest_ideal , K )
```

Erreichtes: Code



```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
6 Uquest_measured = compute_Uquest ( Uout_measured , H )
7 a = compute_a ( Uin , Uquest_measured , N )
8 K = compute_K ( a )
9 Uin = compute_Uin ( Uquest_ideal , K )
10 Uin = set_Vpp ( Uin , Vpp )
```


Erreichtes: Code



```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
6 Uquest_measured = compute_Uquest ( Uout_measured , H )
7 a = compute_a ( Uin , Uquest_measured , N )
8 K = compute_K ( a )
9 Uin = compute_Uin ( Uquest_ideal , K )
10 Uin = set_Vpp ( Uin , Vpp )
11 Uout = measure_Uout ( Uin )
```

Gegeben

Gegeben

generate_BBsignal

measure_H

compute_Uquest

compute_Uin

measure_Uout

compute_a

compute_K

Gegeben

generate_BBsignal : musste implementiert werden

measure_H

compute_Uquest

compute_Uin

measure_Uout

compute_a

compute_K

Gegeben

`generate_BBsignal` : musste implementiert werden

`measure_H` : bereits gegeben in Python

`compute_Uquest`

`compute_Uin`

`measure_Uout`

`compute_a`

`compute_K`

Gegeben

`generate_BBsignal` : musste implementiert werden

`measure_H` : bereits gegeben in Python

`compute_Uquest` : bereits gegeben in Matlab und Python

`compute_Uin`

`measure_Uout`

`compute_a`

`compute_K`

Gegeben

`generate_BBsignal` : musste implementiert werden

`measure_H` : bereits gegeben in Python

`compute_Uquest` : bereits gegeben in Matlab und Python

`compute_Uin` : bereits gegeben in Matlab

`measure_Uout`

`compute_a`

`compute_K`

Gegeben

`generate_BBsignal` : musste implementiert werden

`measure_H` : bereits gegeben in Python

`compute_Uquest` : bereits gegeben in Matlab und Python

`compute_Uin` : bereits gegeben in Matlab

`measure_Uout` : `writeAWG.py` und `writeDSO.py` waren gegeben

`compute_a`

`compute_K`

Gegeben

`generate_BBsignal` : musste implementiert werden

`measure_H` : bereits gegeben in Python

`compute_Uquest` : bereits gegeben in Matlab und Python

`compute_Uin` : bereits gegeben in Matlab

`measure_Uout` : `writeAWG.py` und `writeDSO.py` waren gegeben

`compute_a` : geben in Matlab

`compute_K`

Gegeben

`generate_BBsignal` : musste implementiert werden

`measure_H` : bereits gegeben in Python

`compute_Uquest` : bereits gegeben in Matlab und Python

`compute_Uin` : bereits gegeben in Matlab

`measure_Uout` : `writeAWG.py` und `writeDSO.py` waren gegeben

`compute_a` : geben in Matlab

`compute_K` : bereits gegeben in Matlab und Python

Gegeben

`generate_BBsignal` : musste implementiert werden

`measure_H` : bereits gegeben in Python

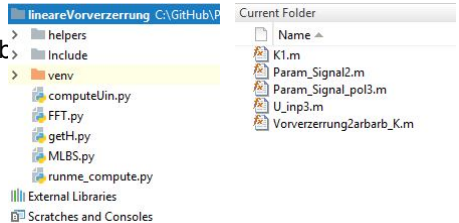
`compute_Uquest` : bereits gegeben in Matlab und Python

`compute_Uin` : bereits gegeben in Matlab

`measure_Uout` : `writeAWG.py` und `writeDSO.py` waren gegeben

`compute_a` : geben in Matlab

`compute_K` : bereits gegeben in Matlab





Vorgehensweise

Vorgehensweise

- Refactoring / Anpassung der Matlab-Funktionen an unser Design

Vorgehensweise

- Refactoring / Anpassung der Matlab-Funktionen an unser Design
- Portierung der Matlab-Funktionen nach Python

Vorgehensweise

- Refactoring / Anpassung der Matlab-Funktionen an unser Design
- Portierung der Matlab-Funktionen nach Python
- Überprüfung der portierten Funktionen mithilfe von **TDD**

Vorgehensweise

- Refactoring / Anpassung der Matlab-Funktionen an unser Design
- Portierung der Matlab-Funktionen nach Python
- Überprüfung der portierten Funktionen mithilfe von **TDD**
 - Momentan jeweils 10 korrespondierende **Unit Tests** in Matlab und Python

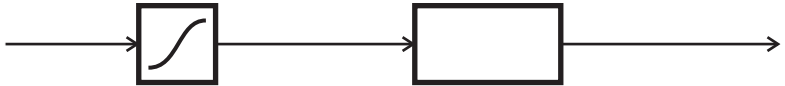
Vorgehensweise

- Refactoring / Anpassung der Matlab-Funktionen an unser Design
- Portierung der Matlab-Funktionen nach Python
- Überprüfung der portierten Funktionen mithilfe von **TDD**
 - Momentan jeweils 10 korrespondierende **Unit Tests** in Matlab und Python
- Maximale Vorbereitung der Funktionen ohne die Geräte dank **TDD**

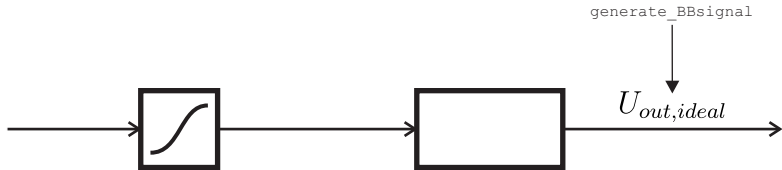
Vorgehensweise

- Refactoring / Anpassung der Matlab-Funktionen an unser Design
- Portierung der Matlab-Funktionen nach Python
- Überprüfung der portierten Funktionen mithilfe von **TDD**
 - Momentan jeweils 10 korrespondierende **Unit Tests** in Matlab und Python
- Maximale Vorbereitung der Funktionen ohne die Geräte dank **TDD**
 - Nur fürs Testen von `measure_Uout` sind Geräte notwendig

Erreichtes: Code

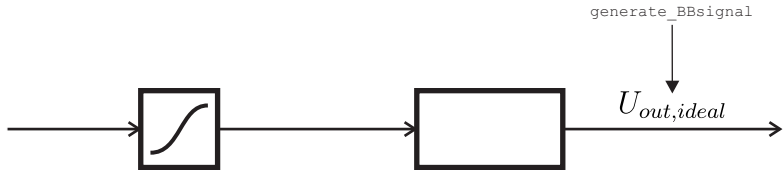


Erreichtes: Code

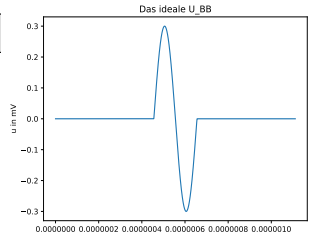


```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
```

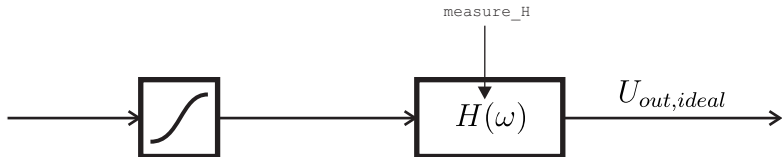
Erreichtes: Code



```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
```

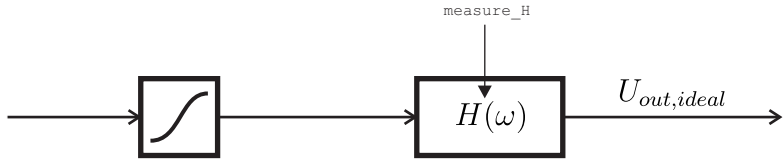


Erreichtes: Code

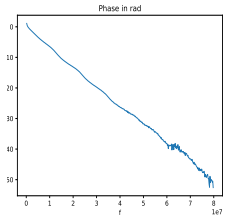
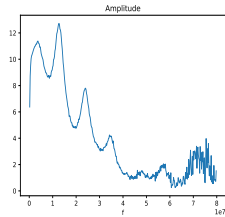


```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )  
2 H = measure_H ( )
```

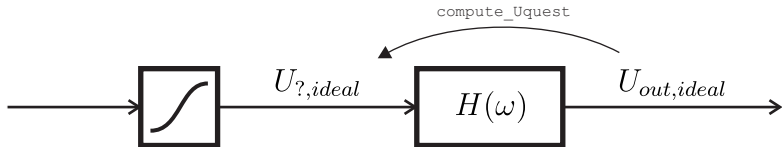
Erreichtes: Code



```
1 Uout_ideal = generate_BBsignal ( fq_rep ,  
2 H = measure_H ( )
```

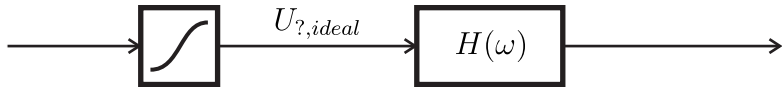


Erreichtes: Code

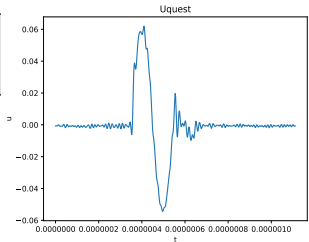


```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )  
2 H = measure_H ( )  
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
```


Erreichtes: Code



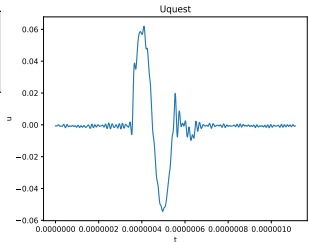
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )  
2 H = measure_H ( )  
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
```



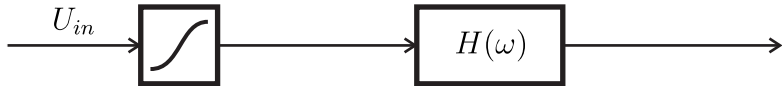
Erreichtes: Code



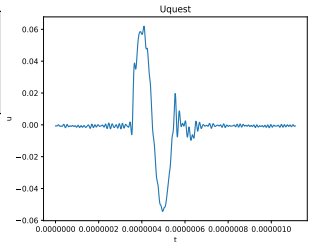
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )  
2 H = measure_H ( )  
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
```



Erreichtes: Code



```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )  
2 H = measure_H ( )  
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )  
4 Uin = Uquest_ideal
```

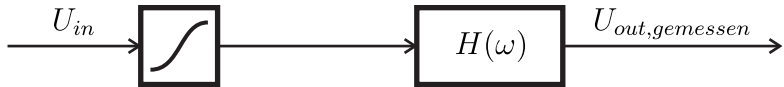


Erreichtes: Code



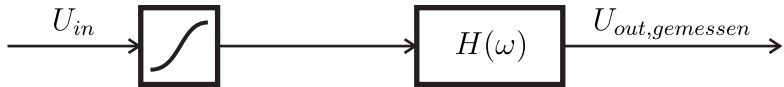
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
```

Erreichtes: Code

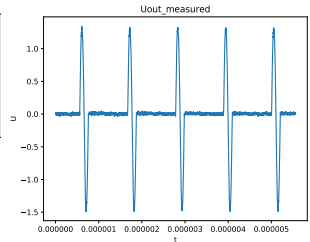


```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
```

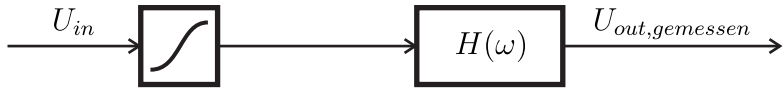
Erreichtes: Code



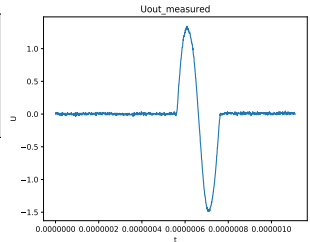
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
```



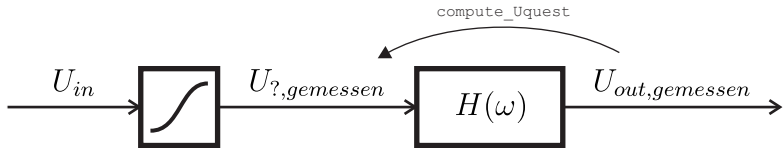
Erreichtes: Code



```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )  
2 H = measure_H ( )  
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )  
4 Uin = Uquest_ideal  
5 Uout_measured = measure_Uout ( Uin )
```

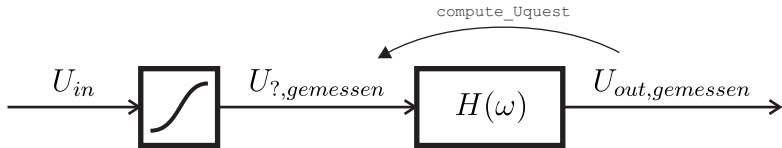


Erreichtes: Code

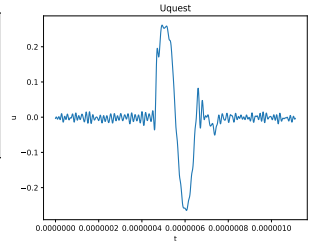


```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
6 Uquest_measured = compute_Uquest ( Uout_measured , H )
```

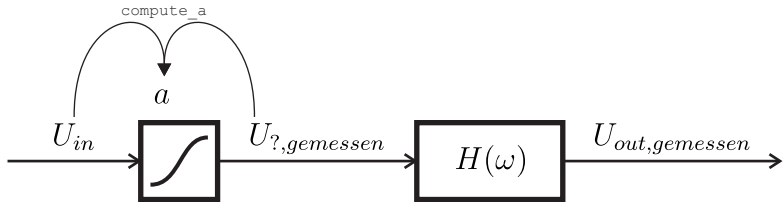

Erreichtes: Code



```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
6 Uquest_measured = compute_Uquest ( Uout_measured , H )
```

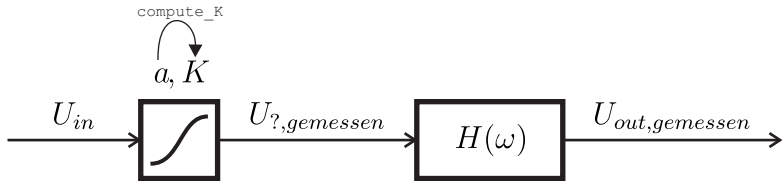


Erreichtes: Code



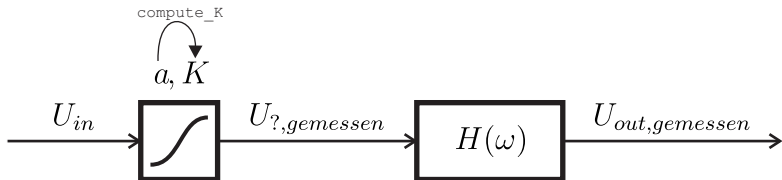
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
6 Uquest_measured = compute_Uquest ( Uout_measured , H )
7 a = compute_a ( Uin , Uquest_measured , N )
```

Erreichtes: Code

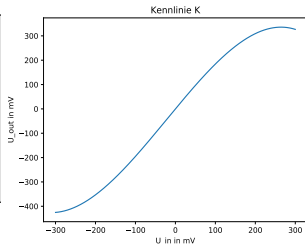


```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
6 Uquest_measured = compute_Uquest ( Uout_measured , H )
7 a = compute_a ( Uin , Uquest_measured , N )
8 K = compute_K ( a )
```

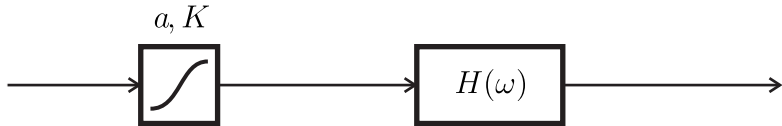
Erreichtes: Code



```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
6 Uquest_measured = compute_Uquest ( Uout_measured , H )
7 a = compute_a ( Uin , Uquest_measured , N )
8 K = compute_K ( a )
```

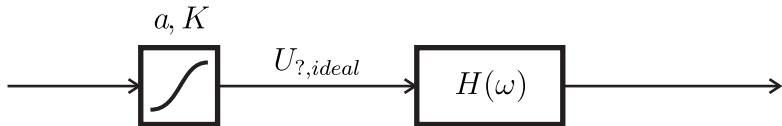


Erreichtes: Code



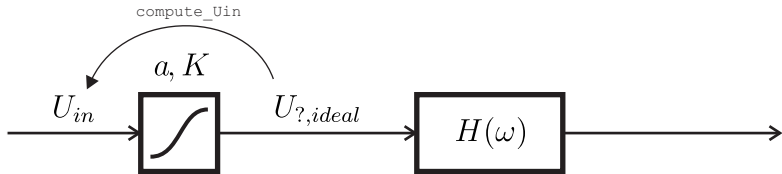
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
6 Uquest_measured = compute_Uquest ( Uout_measured , H )
7 a = compute_a ( Uin , Uquest_measured , N )
8 K = compute_K ( a )
```

Erreichtes: Code



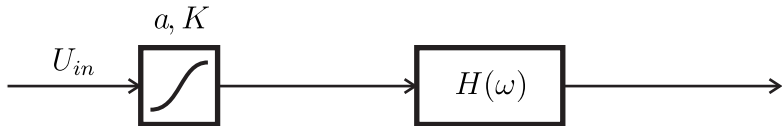
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
6 Uquest_measured = compute_Uquest ( Uout_measured , H )
7 a = compute_a ( Uin , Uquest_measured , N )
8 K = compute_K ( a )
```

Erreichtes: Code



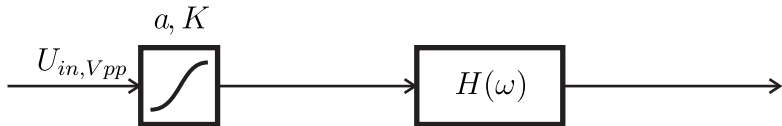
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
6 Uquest_measured = compute_Uquest ( Uout_measured , H )
7 a = compute_a ( Uin , Uquest_measured , N )
8 K = compute_K ( a )
9 Uin = compute_Uin ( Uquest_ideal , K )
```

Erreichtes: Code



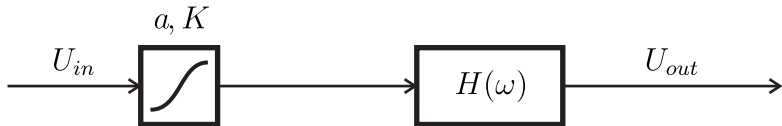
```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
6 Uquest_measured = compute_Uquest ( Uout_measured , H )
7 a = compute_a ( Uin , Uquest_measured , N )
8 K = compute_K ( a )
9 Uin = compute_Uin ( Uquest_ideal , K )
```


Erreichtes: Code



```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
6 Uquest_measured = compute_Uquest ( Uout_measured , H )
7 a = compute_a ( Uin , Uquest_measured , N )
8 K = compute_K ( a )
9 Uin = compute_Uin ( Uquest_ideal , K )
10 Uin = set_Vpp ( Uin , Vpp )
```

Erreichtes: Code



```
1 Uout_ideal = generate_BBsignal ( fq_rep , fq_bb , vpp )
2 H = measure_H ( )
3 Uquest_ideal = compute_Uquest ( Uout_ideal , H )
4 Uin = Uquest_ideal
5 Uout_measured = measure_Uout ( Uin )
6 Uquest_measured = compute_Uquest ( Uout_measured , H )
7 a = compute_a ( Uin , Uquest_measured , N )
8 K = compute_K ( a )
9 Uin = compute_Uin ( Uquest_ideal , K )
10 Uin = set_Vpp ( Uin , Vpp )
11 Uout = measure_Uout ( Uin )
```

Ausblick
