

Die Übertragung der Daten erfolgt mit einer Datenrate von 1187,5 Bit/s. Da die Daten in Blöcke zu 16 Bit aufgeteilt werden, welche dann jeweils mit 10 Bit Prüfinformation versehen werden, liegt die Nutzdatenrate bei ≈ 731 Bit/s. Für die Übertragung im UKW-Signal wird das RDS-Signal in das Stereo-Multiplex-Signal des Senders eingefügt. Als Träger dient eine 57-kHz-Schwingung, welche 90° Phasenverschiebung zu dem ebenfalls an dieser Stelle liegenden ARI-Träger aufweist. Um Beeinflussungen des Rundfunksignals zu vermeiden, erfolgt eine Bandbegrenzung auf $(57 \pm 2,4)$ kHz. Ein RDS-Empfänger muss zunächst das binäre Datensignal aus dem empfangenen Frequenzgemisch herausfiltern. Anschließend ist die Block- und Gruppensynchronisation herzustellen, um die Daten ordnungsgemäß decodieren zu können. Hierbei sowie für die Erkennung bzw. Korrektur von Übertragungsfehlern spielt die eingesetzte Fehlerschutzcodierung eine wichtige Rolle.

3.1.2. Fehlerschutzcodierung

Zum Schutz der Daten gegen Übertragungsstörungen wird ein modifizierter verkürzter zyklischer Code eingesetzt. Die Vorschrift zur Codierung eines Informationspolynoms $a(x)$ lautet:

$$c(x) = a(x) \cdot x^{n-k} + [a(x) \cdot x^{n-k}] \bmod g(x)$$

Ordnet man die Koeffizienten der Basispolynome in einer (k,n) -Matrix, so lässt sich die Codierung auch als Multiplikation des Informationsvektors $\underline{a}$ mit dieser Matrix $\underline{G}$ (Generatormatrix) beschreiben. Für die Decodierung wird die Eigenschaft genutzt, dass das Polynom eines jeden gültigen Codeworts durch $g(x)$ teilbar sein muss. Zur Überprüfung wird das Syndrom $\underline{s}$ berechnet:

$$\underline{s} = \underline{y} \cdot \underline{H}^T \quad \text{mit} \quad \underline{G} \cdot \underline{H}^T = 0$$

Ist bei der Übertragung kein Fehler aufgetreten ($\underline{y} = \underline{c}$), oder entspricht der Fehler $\underline{e}$ einem gültigen Codewort, so ist $\underline{s} = 0$. Anderenfalls ($\underline{y} = \underline{c} + \underline{e}$) ist das Syndrom $\underline{s} \neq 0$. Soll nur Fehlererkennung durchgeführt werden, so werden die verfälschten Daten verworfen. Im Falle der Fehlerkorrektur wird versucht, an Hand des Syndroms $\underline{s}$ den aufgetretenen Fehler zu ermitteln. Ein zyklischer Blockcode kann Fehlerbündel bis zur Länge

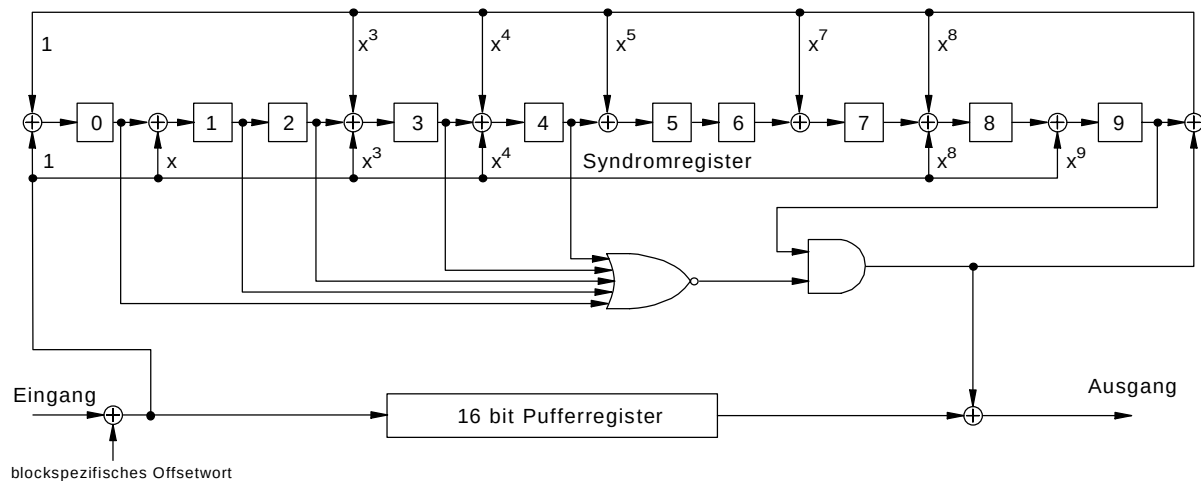
$$t \leq (n - k)/2 \quad (\text{Reiger-Schranke})$$

korrigieren.

Aus einem gegebenen (n,k) -Code kann durch das Streichen der führenden i Stellen jedes Codewortes ein $(n-i,k-i)$ -Code erzeugt werden. Ein solcher Code wird verkürzter zyklischer Code genannt. Die Fehlerkorrekturleistung des Codes wird durch die Verkürzung nicht beeinträchtigt. Es ist nur notwendig, vor dem Decodieren jedem verkürzten Codewort i Nullen voranzustellen. Der beim RDS verwendete $(26,16)$ -Code¹ wird durch die Verkürzung eines $(341,331)$ -Codes um $i = 315$ Stellen gebildet.

Das Generatorpolynom $g(x) = x^{10} + x^8 + x^7 + x^5 + x^4 + x^3 + 1$ ist ein irreduzibler Teiler von $x^{341} - 1$. Für die Decodierung wird ein Meggitt-Decoder empfohlen, welcher die Korrektur von maximal 5 aufeinanderfolgenden Fehlern ermöglicht (siehe Bild2).

¹ $\underline{G}$ und $\underline{H}^T$ liegen am Versuchsplatz aus (' bedeutet eine mögliche Form der Prüfmatrix)

**Bild 2** Meggitt-Decoder

Dieser Aufbau führt, wenn möglich, die sequentielle Korrektur eines maximal 5 Bit langen Fehlerbündels durch. Die empfangenen 16 Informationsbit werden sowohl in das Pufferregister als auch in das Syndromregister geschoben. Im Syndromregister werden Vormultiplikation ($x^9 + x^8 + x^4 + x^3 + x + 1$) und Division ($x^{10} + x^8 + x^7 + x^5 + x^4 + x^3 + 1$) ausgeführt, so dass nach Einlesen der empfangenen 26 Information- und Prüfbits das Syndrom $\underline{s} = [s_9 s_8 \dots s_1 s_0]$ entsteht. Sind alle Registerstellen $s_v = 0$, so wird die Nutzinformation unverändert aus dem Puffer weitergegeben, ansonsten wird parallel zum bitweisen Auslesen des Puffers die Fehlerkorrektur durchgeführt. Dabei werden folgende Schritte ausgeführt:

1. Steht in den 5 linken Zellen des Syndromregisters eine 0 ($s_0 \dots s_4 = 0$) und in der äußeren rechten Zelle eine 1 ($s_9 = 1$), so werden der Ausgang von Register s_9 und der Ausgang des Pufferregisters y_{25} durch Addition einer 1 mod 2 negiert.
2. Anlegen eines Taktimpulses bewirkt, dass alle an den Registereingängen anliegenden logischen Pegel in die Register übernommen werden und bis zum nächsten Takt statisch an deren Ausgängen vorliegen. Im Pufferregister gelangt y_{24} in das letzte Register. Nun wird Schritt 1. wiederholt.
3. Werden die Schritte 1. und 2. sechzehnmal ausgeführt und ist dann das Syndrom $\underline{s} \neq 0$, so liegt ein nicht korrigierbarer Fehler oder ein Fehler in den Prüfbits vor.

Diesem Vorgehen liegt die Tatsache zugrunde, dass die Fehlerbits e_v des „rotierten“ Syndroms zu den ebenso „rotierten“ empfangenen Codebits $c_v + e_v$ korrespondieren. Eine Rotation des Syndroms entspricht demnach der Rotation des zugehörigen Fehlermusters.

Die Anzahl der Schritte bis zum Auftreten des Syndroms $\underline{s} = [1XXXX00000]^2$ gibt Auskunft über die Lage des maximal 5 Bit langen Fehlerbündels im Codewort. So wird jedes Fehlerbit $e_v = 1$ am Ausgang des Syndromregisters im nächsten Takt im Syndromregister als $e_v \oplus \hat{e}_v = 1 \oplus 1 = 0$ korrigiert weiter geschoben (rotiert) und am Ausgang des Pufferregisters als $c_v \oplus e_v \oplus \hat{e}_v = c_v$ korrigiert ausgelesen. Bei erfolgreicher Korrektur wird $\underline{s} = 0$ angezeigt.

² X bedeutet kann 0 oder 1 sein

3.1.3. Synchronisation

Der eindeutige Zusammenhang zwischen der Verfälschung eines Codewortes und dem berechneten Syndrom wird benutzt, um dem RDS-Empfänger die Herstellung von Block- und Gruppensynchronisation zu ermöglichen. Jedem Codewort wird ein so genanntes Offsetwort $\underline{o}_v$ hinzugefügt. Das bedeutet eine künstlichen Verfälschung des Codewortes. Dem Empfänger sind die Syndrome $\underline{s}(\underline{o}_v)$ für die einzelnen Offsetwörter bekannt. Für den einlaufenden Datenstrom wird nach jedem neuen Bit das Syndrom berechnet und mit den Syndromen der Offsetwörter verglichen. Werden im Abstand von 26 Bit die Syndrome zweier aufeinanderfolgender Blöcke (A-B, B-C, C-D oder D-A) gefunden, so kann der Empfänger die Grenzen eines Blockes und einer Gruppe bestimmen und beginnen, die empfangenen Daten gezielt auszuwerten.

Offsetwörter zur Block- und Gruppensynchronisation:

Block v	Offsetwort $\underline{o}_v$	Syndrom $\underline{s}(\underline{o}_v)$
A	0000000000000000 0011111100	1111011000
B	0000000000000000 0110011000	1111010100
C	0000000000000000 0101101000	1001011100
C'	0000000000000000 1101010000	1111001100
D	0000000000000000 0110110100	1001011000

Im weiteren Verlauf des Empfangs hat der Empfänger dafür zu sorgen, dass ein Verlust der Synchronisation in Folge von Senderwechsel, Empfangsverschlechterung, Bitversatz o.ä. schnell erkannt wird, um geeignete Maßnahmen bis hin zum vollständigen Empfängerreset einzuleiten. Dafür gibt es verschiedene Möglichkeiten:

Führen einer Fehlerstatistik:

- Abbruch bei einer zu hohen Fehlerrate oder sprunghaftem Anstieg der Fehler

Überwachung des PI-Codes in Block A:

- da dieser Code in jeder Gruppe gesendet wird und sich nicht ändert, kann durch Vergleich des empfangenen PI-Codes mit dem bereits gespeicherten PI-Code ein eventueller Bit-Versatz ermittelt werden

3.2. Versuchsaufbau

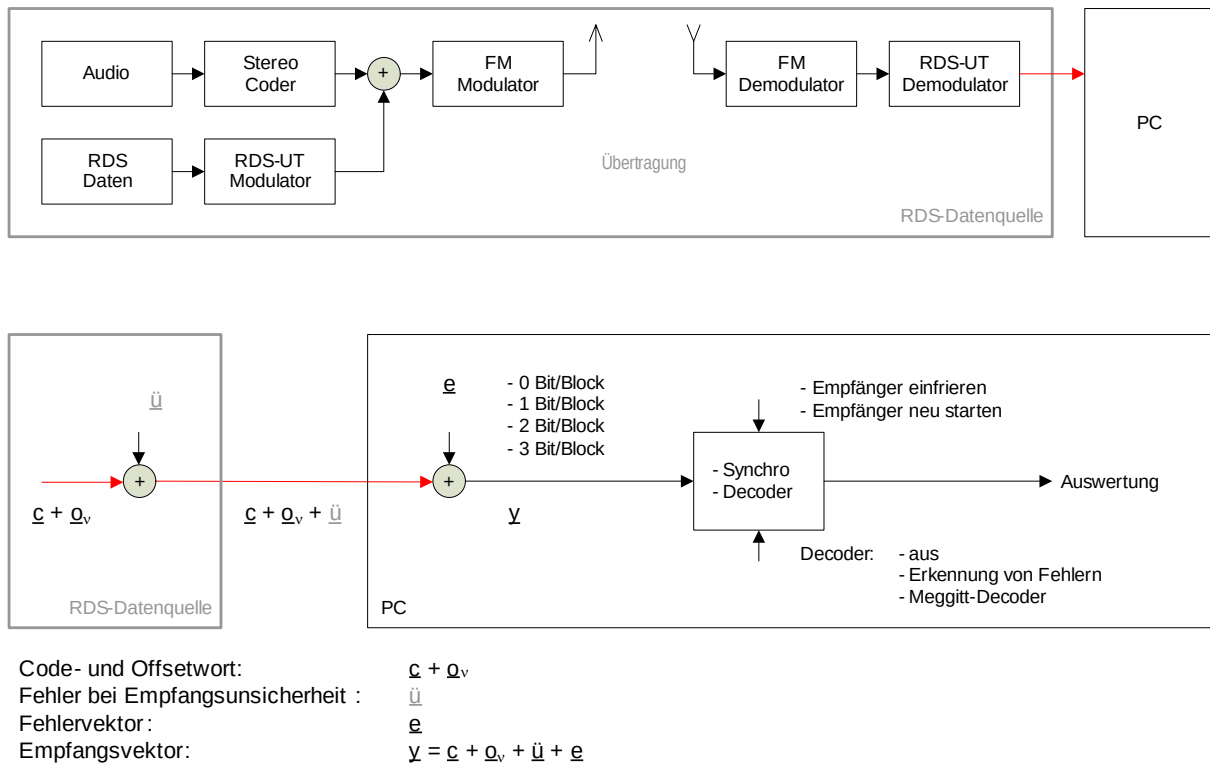


Bild 3 PC-gestützter RDS-Empfänger

Bild 3 zeigt den schematischen Aufbau des RDS-Empfangs. Die einzelnen Komponenten haben die folgenden Aufgaben zu erfüllen:

RDS-Datenquelle:

Sendeseite

Audioquelle, Stereo-Coder, RDS-Datenquelle mit Coder und Addition des Offsetwortes Q_v , Unterträgermodulator, Addition im Basisband, FM-Modulator, Sendeantenne

Empfangsseite

Empfangsantenne, FM-Demodulator (UKW-Empfänger), RDS-Unterträgerdemodulator (Anpassung des Datensignals an die Pegel der seriellen PC-Schnittstelle)

PC:

Demo-Kanal

Einlesen des Datensignals $C + Q_v + \underline{U}$ über eine serielle Schnittstelle (bei empfangssicheren Sendern $\underline{U} = 0$)

Überlagerung des Fehlervektors $\underline{e}$ mit maximal 1, 2 oder 3 Fehlerbits je Codewort

Synchro und Decoder

Decodieren

Anzeigen der RDS-Daten, Fehlererkennung, Fehlerstatistik

Daten zur weiteren Bearbeitung abspeichern

3.3. Softwarebeschreibung zum RDS-Empfänger

Zu Beginn ermittelt das Programm, an welcher seriellen Schnittstelle die RDS-Daten anliegen. Dazu wird zunächst an COM2 und bei Misserfolg auch an COM1 nach dem RDS-Datentakt gesucht. Wird dieser nicht festgestellt, so ist entweder die Verbindung nicht ordnungsgemäß hergestellt, oder UKW-Empfänger bzw. Spannungsversorgung sind nicht eingeschaltet. Sonst wird damit begonnen, das Datensignal auszuwerten, um die Block- und Gruppensynchronisation herzustellen. Gelingt die Synchronisation nicht innerhalb von 10 Sekunden, so wird der Empfänger gestoppt. In diesem Fall ist die Qualität der RDS-Daten unzureichend, und es sollte versucht werden, eine bessere Abstimmung des UKW-Empfängers durchzuführen oder einen anderen Sender zu suchen.

Ein Indikator für die Qualität der RDS-Daten ist eine rote Leuchtdiode am Gehäuse des Demodulators. Sie leuchtet, wenn die vom Demodulator gelieferten Daten unsicher sind. Die Chancen auf korrekten RDS-Empfang sinken also, wenn die LED stark flackert.

Konnte die Block- und Gruppensynchronisation hergestellt werden, dann erfolgt der Übergang in den synchronisierten Zustand und das Programm beginnt, die empfangenen Daten auszuwerten und anzuzeigen.

Die angezeigten decodierten Daten sind im einzelnen:

- **Sendername (PS):** Der Name des Programms wird in 4 Segmenten mit je 2 alphanumerischen Zeichen übertragen, das Empfangsprogramm ordnet die Segmente und zeigt sie an.
- **Programmcode (PI):** Diese 16 Bit-Zahl dient der eindeutigen Kennzeichnung des Programms, sie enthält codierte Informationen über Herkunftsland und Sendebereich des Programms. Diese Informationen werden nicht weiter ausgewertet.
- **Liste alternativer Frequenzen (AF):** Die Informationen über alternative Sendefrequenzen für das empfangene Programm sind nach festgelegten Protokollen codiert, für die Anzeige erfolgt eine Umsetzung in entsprechende Texte. Da für einen Sender mehrere Listen gesendet werden können, ist ein ständiges Wechseln der angezeigten Daten möglich.
- **Verkehrssender (TP):** Die Codierung für die Kennzeichnung des Programms als Verkehrssender wird wie folgt umgesetzt:

<u>Wert von TP</u>	<u>Anzeige</u>	<u>Bedeutung</u>
0	nein	es werden keine Verkehrsdurchsagen gesendet
1	ja	es werden Verkehrsdurchsagen gesendet

- **Programmart (PTY):** Die Codierung dieser Information wird durch die im RDS-Standard empfohlene alphanumerische Ausgabe ersetzt (0=KEINE,1=NACHRICHT etc.).
- **Verkehrsdurchsage (TA):** Bei laufender Verkehrsdurchsage wird der Text "läuft" ausgegeben. Werden Verkehrsdurchsagen anderer Programme genutzt, so erscheint "über EON", ansonsten wird "keine" ausgegeben.

- **Musik/Sprache (MS):** Bisher wurde kein Sender empfangen, welcher dieses Bit zur Unterscheidung von Musik- und Sprachsendungen nutzt, im ungenutzten Zustand wird dauerhaft der Wert "1" angezeigt.

- **Radiotext (RT):** Der Übertragung dieser alphanumerischen Nachrichten erfolgt in Segmenten mit je 4 Zeichen, wobei die maximale Nachrichtenlänge bei 64 Zeichen liegt. An der Anzeige lässt sich gut der segmentweise Empfang verfolgen, da die vollständige Nachricht über mehrere Sekunden verteilt empfangen wird.

Die Tabelle gibt einen Überblick über **Sender** und deren **Programmcode (PI)**.

Sendername	Programmcode (PI)	Sendefrequenz
R.SA	0001 0100 1100 0010	88,3/89,2 MHz
MDR.JUMP	1101 0011 1100 0010	90,1 MHz
BBC-RFI	0001 0000 1100 1011	91,1 MHz
MDR SACH (U4)	1101 0100 1100 0001	92,2 MHz
D-KULTUR (U3)	1101 0010 0010 0000	93,2 MHz
MDR.FIGARO (U2)	1101 0011 1100 0011	95,4 MHz
DLF (U1)	1101 0010 0001 0000	97,3 MHz
APOLLO	0001 1010 1100 0101	98,4/99,2 MHz
ENERGY	1101 0011 1100 1110	100,2 MHz

Statistik:

- Empfänger aktiv:** seit dem Start des Empfängers vergangene Zeit
- empfangene Bits:** insgesamt eingelesene Datenbits
- empfangene Blöcke:** Zähler für die empfangenen Blöcke (y)
- fehlerhafte Blöcke:** Zähler für die fehlerhaften Blöcke (vor Korrektur $s \neq 0$)
- korrigierte Blöcke:** Zähler für die korrigierten Blöcke (nach Korrektur $s = 0$)
- Blockfehlerrate:** Verhältnis von Zahl fehlerhafter Blöcke zu Zahl empfangener Blöcken in der vergangenen Sekunde
- addierte Blockfehler:** Zähler für nachträglich verfälschte Blöcke (e)
- fehlerhaft korrigiert:** Zähler für falsch korrigierte Blöcke (nach Korrektur $s = 0$)

Die Aktualisierung der Anzeige erfolgt im Sekundentakt, da ein unmittelbares Auswerten und Anzeigen der empfangenen Daten nicht möglich ist. Ebenso überprüft das Empfangsprogramm im Sekundentakt, ob die Synchronisation noch gewährleistet ist. Lag die Blockfehlerrate in der vergangenen Sekunde über 60% oder wurden mindestens 45 fehlerhafte Blöcke in Folge empfangen, so wird der Empfang abgebrochen und in den unsynchronisierten Zustand übergegangen.

Voraussetzung für alle weiteren Ausführungen ist, dass ein Sender mit stabilem RDS-Empfang gefunden werden kann. Hierfür ist z.T. ein mehrmaliger Senderwechsel notwendig, bevor die Synchronisation erfolgreich durchgeführt werden kann.

Möglichkeiten zur Beeinflussung der Arbeitsweise bei laufendem Empfang:

1. Fehlerbehandlung: Über Radiobuttons unterhalb der Statistikanzeige kann die Art der Fehlerbehandlung variiert werden.

- "*keine*": Empfangene Blöcke werden nicht auf Fehler überprüft, sondern unverändert decodiert.
- "*nur Erkennung*": Für jeden Block wird das Fehlersyndrom berechnet und mit dem erwarteten Syndrom verglichen. Als fehlerhaft erkannte Blöcke werden nicht weiterverarbeitet.
- "*Meggitt-Decoder*": Wird ein Fehler erkannt, so erfolgt der Korrekturversuch mit einem Meggitt-Decoder gemäß der RDS-Spezifikation [DIN93]. Dieser Decoder kann die Korrektur von maximal 5 aufeinanderfolgenden Fehlern durchführen.

2. Additive Fehler: Ebenfalls über Radiobuttons unterhalb der Statistikanzeige kann das Hinzufügen von Fehlern zu den eingelesenen Daten veranlasst werden. Dabei gibt es folgende Möglichkeiten:

- "*max. 1 Bit / Block*": durch die Überlagerung einer um den Faktor 27 gedehnten PN-Folge wird in jedem Block ein Bit mit der Wahrscheinlichkeit $p=0,5$ verfälscht
- "*max. 2 Bit / Block*": durch die Überlagerung einer um den Faktor 15 gedehnten PN-Folge werden in jedem Block maximal 2 Bit jeweils mit der Wahrscheinlichkeit $p=0,5$ verfälscht
- "*max. 3 Bit / Block*": durch die Überlagerung einer um den Faktor 9 gedehnten PN-Folge werden in jedem Block 3 Bit jeweils mit der Wahrscheinlichkeit $p=0,5$ verfälscht

Unter einer um den Faktor n gedehnten PN-Folge soll eine Folge verstanden werden, die durch das Einschieben von $n-1$ Nullen nach jedem Element der Originalfolge gebildet wird. Der überlagerte Fehlervektor wird gespeichert, um nach erfolgter Fehlerkorrektur etwaige Decodierfehler (ermitteltes Korrekturmuster überlagerten Fehlervektor) erkennen zu können.

In dieser Betriebsart ist die automatische Erkennung eines Synchronisationsverlustes ausgeschaltet, weil sonst das vermehrte Auftreten von Fehlern zu einer Neusynchronisation des Empfängers führen würde.

3. Empfänger einfrieren: Ein Betätigen des "Einfrieren"-Buttons in der Werkzeugleiste stoppt die sekundenweise Aktualisierung der Anzeige. Gleichzeitig wird am oberen Bildschirmrand ein Fenster geöffnet, in welchem der aktuelle Empfängerstatus betrachtet werden kann. Aus Platzgründen kann dieses Fenster nur 5 Zeilen gleichzeitig anzeigen, für weitere Daten ist die Bildlaufleiste am rechten Rand des Fensters zu betätigen. Zur Anzeige gelangen dann ebenfalls die Originaldaten der letzten 2 Sekunden vor dem Einfrieren sowie die verfälschten Daten, falls gerade mit additiven Fehlern gearbeitet wird. Das Empfangsprogramm läuft unterdessen im Hintergrund weiter, um die Synchronisation aufrecht zu erhalten. Im Falle des Synchronisationsverlustes wird das Daten-Anzeigefenster automatisch geschlossen und der Empfänger zurückgesetzt. Im eingefrorenen Zustand ist es nicht möglich, die Fehlerbehandlung oder den additiven Fehler zu verändern (die Radiobuttons sind deaktiviert). Eine Fortsetzung der normalen Arbeitsweise wird dadurch erreicht, dass der "Einfrieren"-Button erneut betätigt wird.

4. Empfänger neu starten: Ein vollständiges Zurücksetzen des RDS-Empfängers kann durch die Betätigung des "Neustart"-Buttons erreicht werden. Alle Variablen werden zurückgesetzt und der gesamte Programmablauf mit dem Schnittstellentest neu gestartet. Dies ist z.B. notwendig, wenn der Empfänger nach nicht gelungener Synchronisation angehalten wurde.

Bei einer Änderung der Fehlerbehandlung oder des additiven Fehlers werden die Zähler für empfangene Blöcke, fehlerhafte Blöcke, korrigierte Blöcke, additive Blockfehler und fehlerhaft korrigierte Blöcke zurückgesetzt, so dass die angezeigten Werte sich immer auf die momentan eingestellten Verarbeitungsbedingungen beziehen.

Die nach dem Einfrieren des Empfängers angezeigten Daten werden automatisch in der Datei "RDSDATEN.FRS" im Arbeitsverzeichnis abgespeichert. Jedes erneute Einfrieren überschreibt die vorher abgespeicherten Werte. Es besteht jedoch die Möglichkeit, die Datei nachträglich umzubenennen, um eine spätere Auswertung der abgespeicherten Werte durchzuführen.

3.4. Versuchsdurchführung

3.4.1. Versuchsabläufe

Für die praktische Untersuchung des Radio-DatenSystems ist der RDS-Empfänger gemäß Bild 3 aufzubauen und in Betrieb zu nehmen. Eine Beeinflussung der Arbeitsweise des Empfängers ist am UKW-Empfänger (Senderwechsel, Abstimmung) und am PC möglich. Das Empfangsprogramm wird über die Oberfläche der Windows-Anwendung „FRS für Windows“ gestartet.

Nach erfolgreich durchgeführter Synchronisation werden im Programmfenster die decodierten Daten und eine Empfangsstatistik angezeigt. Die sekundenweise Ausgabe der Empfangsdaten gibt Aufschluss über die zeitliche Verteilung bestimmter Informationen (z.B. der 64-stellige alphanumerische Radiotext RT). Für Sender mit Verkehrsdurchsagen kann die Veränderung des Verkehrsdurchsagebits TA korrespondierend zur laufenden Verkehrsdurchsage beobachtet werden. Eine Verschlechterung des Empfangs äußert sich in einer erhöhten Blockfehlerrate.

Standardmäßig führt der RDS-Empfänger eine Fehlererkennung durch. Fehlerhafte Blöcke werden nicht ausgewertet. Bei ausgeschalteter Fehlererkennung werden die Daten unbehandelt angezeigt. Die Fehlerkorrektur mit einem Meggitt-Decoder ist in der Lage, maximal 5 aufeinanderfolgende Fehler zu korrigieren. Für einen Sender mit leicht gestörtem Empfang (Blockfehlerrate $\neq 0$) soll beobachtet werden, wie sich die unterschiedliche Fehlerbehandlung auf die Qualität der decodierten Daten auswirkt.

Das Empfangsprogramm bietet die Möglichkeit, den empfangenen Datenstrom nachträglich mit einem Störsignal zu überlagern. Hierfür ist ein Sender mit sehr gutem Empfang zu suchen. Für die einstellbaren Fehler ist die resultierende Blockfehlerrate zu notieren und das Fehlerbild in Abhängigkeit von der Fehlerbehandlung zu beobachten.

Die Originaldaten werden in einer Textdatei (RDSDATEN.FRS) abgelegt und in einem zusätzlichen Fenster angezeigt, sobald der RDS-Empfänger eingefroren wird. Für ausgewählte Datenblöcke ist das Fehlersyndrom zu berechnen. Durch Vergleich mit der Syndromtabelle der Offsetwörter kann entschieden werden, ob ein Übertragungsfehler aufgetreten ist bzw. welcher Block vorliegt (A, B, C oder D). Die hierfür benötigten Informationen (Generatormatrix, Prüfmatrix, Syndromtabelle) sind am Versuchsort zu finden.

3.4.2. Versuchsaufgaben

1. Empfangstest

Machen Sie sich mit der Versuchsanordnung vertraut !

- a) Mit ein oder zwei empfangssicheren Sendern ist das Synchronisations- und das Erkennungsverhalten zu beobachten und die Sender-Kenndaten sind zu notieren.

2. Codierung und Decodierung für fehlerfreie Übertragung

Für den/die empfangssicheren Sender stehen die aufgezeichneten RDS-Daten als Textdatei zur Verfügung.

- a) Finden Sie die Blockstruktur für die Blöcke A, B, C, D und prüfen Sie die Codierung der Informationsbits mit Hilfe der Generatormatrix $\underline{G}$ und die Addition der Offsetworte $\underline{o}_v$ mit $v = \{A, B, C, D\}$ am Beispiel nach !
- b) Berechnen Sie mit Hilfe der Prüfmatrix $\underline{H}^T$ die durch die überlagerten Offsetworte $\underline{o}_v$, $v = \{A, B, C, D\}$, entstehenden Syndrome !
- c) Welche Funktion kann mit den Syndromen realisiert werden ?

Der Zusammenhang zwischen den Matrizen $\underline{G}$, $\underline{H}^T$ und $\underline{H}^T$ wird während des Versuches erörtert.

3. Codierung und Decodierung für fehlerhafte Übertragung

In diesem Teil sollen Funktion und Wirksamkeit der fehlerkorrigierenden Codierung untersucht werden (Meggitdecoder zugeschaltet).

- a) Berechnen Sie allgemein die Syndrome $\underline{s} = \underline{e} \cdot \underline{H}^T$, die zu Beginn des Decodiervorganges auch so in den Registern des Meggitdecoders stehen !
- b) Verdeutlichen Sie sich die Funktion des Meggitdecoders für den Fall, dass sich der Bündelfehler der Länge $t \leq 5$ zunächst exakt am Anfang des empfangenen Codewortes befindet (höchstwertige Stellen) ! Die Funktion des Decoders für den Fall, das sich das Fehlerbündel nicht am Anfang des empfangenen Codewortes befindet, wird während des Versuches erläutert.
- c) Wie verändern sich die Syndrome und deren Auswertbarkeit, wenn $t > 5$ wird ?
- d) Untersuchen Sie in den aufgezeichneten Dateien, wie viele Fehler in einem Block max. auftreten und ob diese korrigiert werden können !
- e) Führen Sie eine Messung mit einem empfangssicheren Sender durch und stellen Sie in einem Diagramm in %
 - die fehlerhaften Blöcke ($\circ$),
 - die Blöcke, die einer Korrektur unterzogen werden (x),
 - die fehlerhaft korrigierten Blöcke (*)
 - die verbleibenden Blockfehler ($\square$)
 für eine maximale Fehlerzahl von 1, 2 und 3 pro Block dar ! Interpretieren Sie die Kurven ! Geben Sie das Gewicht der sicher korrigierbaren Fehler sowie die Länge der korrigierbaren Bündelfehler an !
- f) Beobachten Sie das Decoderverhalten für einen sicher empfangbaren Sender und verstimmen Sie den UKW-Empfänger ! Was beobachten Sie ?

Vorbereitungsaufgaben

Grundlagen (Gegeben $\underline{G}$ oder $g(x)$.)

1. Wie wird aus einem Informationsvektor $\underline{a}$ das Codewort $\underline{c}$ erhalten ?
2. Wie wird aus einem Informationspolynom $a(x)$ das Codepolynom $c(x)$ erhalten ?
3. Wie erfolgt die Darstellung eines empfangenen, durch Fehler beeinflussten Codewortes $\underline{y}$ am Ausgang des Übertragungskanals ?
4. Wie erfolgt die Darstellung eines empfangenen, durch Fehler beeinflussten Codepolynoms $y(x)$ am Ausgang des Übertragungskanals ?
5. Wie sind Generatormatrix $\underline{G}$ und Prüfmatrix $\underline{H}^T$ miteinander verknüpft ? Wie kann man $\underline{H}$ aus $\underline{G}$ berechnen ?
6. Wie erfolgt die Korrektur des empfangenen, verfälschten Codewortes $\underline{y}$; wie erhält man die Syndrome ?
7. Wie erfolgt die Korrektur des empfangenen, verfälschten Codepolynoms bei gegebenem Generatorpolynom $g(x)$? Wie erhält man ein Polynom, welches die Syndrome enthält ?
8. Wie wirkt sich die Benutzung eines $(n-k)$ -stelligen Schiebregisters bei der Polynomdivision aus ? Wie ist eine Codeverkürzung bei der Empfangs-Schieberegisterschaltung zu berücksichtigen ?
9. Welcher Zusammenhang besteht zwischen $d_{\min}$ und dem Gewicht eines korrigierbaren Fehlers ?
10. Welcher Zusammenhang besteht zwischen Codewortlänge n , Anzahl der Informationsbits k und Länge t eines sicher korrigierbaren Bündelfehlers ?
11. Verdeutlichen Sie sich nochmals den Unterschied zwischen einem beliebigen Fehler des Gewichtes v und einem Fehler der Länge v in einem Codewort der Länge n !

RDS

12. Wie ist das RDS-Datenprotokoll aufgebaut ?
13. Welcher Block (A, B, C, D) besitzt immer die gleichen Informationsbits ?
14. Wie werden die RDS-Daten im Signal für den UKW-Rundfunk untergebracht ?
15. In den Registern des Meggittdecoders stehen genau die Syndrome, die man auch mit Hilfe der Operation $\underline{s} = \underline{e} \cdot \underline{H}^T$ berechnen kann. Berechnen Sie diese Syndrome! Ein Hauptanliegen des Versuches ist, sich die Funktion des Meggittdecoders für $t \leq 5$ sowie für $t > 5$ mit Hilfe dieser Syndrome zu verdeutlichen.
16. Durch welche Operation wird aus einem Codewort $\underline{c}$ und einem Offsetwort $\underline{o}_v$, $v = \{A, B, C, D\}$ ein Block des RDS-Rahmens erhalten ?
17. Welche Funktion erfüllen Programmidentifikation PI und das Offsetwort ? Verdeutlichen Sie sich die Schritte für das
 - Herstellen der Rahmensynchronisation,
 - Halten der Rahmensynchronisation und
 - Korrigieren von Fehlern !
18. Wie wird aus dem Codewort mit überlagertem Offsetwort $\underline{c} + \underline{o}_v$ das zugehörige Syndrom erhalten ?
19. Wie werden bei Kenntnis der Rahmenstruktur die Offsetworte eliminiert; was verbleibt ?