AVR-8-bit-Mikrocontroller
Gruppe 500 - CodeVisionAVR C-Compiler
Inhaltsverzeichnis

Teil 601 - PB_LED

Teil 602 - 2_Draht_LCD

Teil 603 - IRDMS

Teil 604 - Pegelsonde (Noch in Arbeit)

Teil 605 - IR_Decoder (Noch in Arbeit)

3 AVR-C-Projekt IRDMS (Infrared Distance Measurement Sensor)

Zu diesem Abschnitt gehört eine Sammlung von PDF-Dateien, die die technischen Zeichnungen, die Schaltbilder und das Platinen-Layout beinhalten. Diese Dateien stehen auch als Dateien im Format ***.cdr** (CorelDRAW) zur Verfügung und können bei Bedarf vom Autor zur Verfügung gestellt werden. (Auf Grund längerer Erfahrung/Übung mit diesem Zeichenprogramm wurde es bisher immer noch der Anwendung EAGLE vorgezogen).

32 KB	603_01_ADC_Sukzessive_Approximation.pdf	01 Analog Digital Converter - Sukzessive Approximation
63 KB	603_02_ADC_Blockschaltbild.pdf	02 Analog Digital Converter - Blockschaltbild
10 KB	603_03_ADC_Register.pdf	03 Analog Digital Converter - Register-Beschreibung
32 KB	603_04_ADC_Timing.pdf	04 Analog Digital Converter - Zeitdiagramme
97 KB	603_05_IRDMS_Dimensionierung_Messstab.pdf	05 IRDMS - Ansicht und Maße der Zisterne und Messstab
83 KB	603_06_IRDMS_Anschaltung.pdf	06 IRDMS - Anschaltung und IRDMS-Charakteristik
23 KB	603_07_IRDMS_Stange.pdf	07 IRDMS - Gestänge mit Aufhängevorrichtung
49 KB	603_08_IRDMS_Rahmen.pdf	08 IRDMS - Aufhängungs-Rahmen in der Zisterne
65 KB	603_09_IRDMS_Schwimmer.pdf	09 IRDMS - Anbringung der Schwimmerscheibe
52 KB	603_10_IRDMS_Messkopf_Montage.pdf	10 IRDMS - Messkopf - Montage
201 KB	603_11_IRDMS_Messkopf_Schaltung.pdf	11 IRDMS - Messkopf - Schaltung
217 KB	603_12_RWNA_Konzept_Ueberlauf.pdf	12 RWNA - Konzept der Überlauf-Steuerung
273 KB	603_13_RWNA_Gesamt_Schaltbild.pdf	13 RWNA - Gesamt-Schaltbild der Anlage
109 KB	603_14_RWNA_Steuerung_Frontplatte.pdf	14 RWNA - Steuerung - Fronplatte
98 KB	603_15_RWNA_Steuerung_Frontplatte_Layout.pdf	15 RWNA - Steuerung - Layout der Frontplatte
204 KB	603_16_RWNA_Steuerung_Montage.pdf	16 RWNA - Steuerung - Montage
214 KB	603_17_RWNA_Steuerung_Sockel_Platine.pdf	17 RWNA - Steuerung - Layout der Sockel-Platine
230 KB	603_18_RWNA_Steuerung_Display_Platine.pdf	18 RWNA - Steuerung - Layout der Display-Platine
98 KB	603_19_RWNA_Steuerung_Platinenlayout.pdf	19 RWNA - Steuerung - Platinen-Layout zur Filmherstellung
89 KB	603_20_RWNA_Ablaufdiagramm_1.pdf	20 RWNA - Programm - Ablaufdiagramm 1
19 KB	603_21_RWNA_Ablaufdiagramm_2.pdf	21 RWNA - Programm - Ablaufdiagramm 2
28 KB	603_22_RWNA_Ablaufdiagramm_3.pdf	22 RWNA - Programm - Ablaufdiagramm 3
521 KB	603_23_RWNA_Pumpenstation.pdf	23 RWNA - Pumpenstation
26 KB	603_24_RWNA_Zisternenanlage.pdf	24 RWNA - Zisternenanlage
11 KB	603_25_RWNA_Funktionsbeschreibung.pdf	25 RWNA - Funktionsbeschreibung
12 KB	603_26_RWNA_Gebrauchsanweisung.pdf	26 RWNA - Gebrauchsanweisung
3.941 KB	603_27_RWNA_Lageplan.pdf	27 RWNA - Lageplan
106 KB	603_29_IRDMS_Simulation_Testschaltung.pdf	29 IRDMS - Simulation Testschaltung

3.1 Funktion des IRDMS

Das Projekt stellt eine Applikation zu den Abstandsmess-Sensoren der Firma SHARP dar, um Abstände optisch zu erfassen. Das Messsystem basiert auf der Triangulation mittels eines Infrarot-Lichtstrahls: Das Licht einer IR-Licht emittierenden Diode wird mittels einer Sammellinse scharf gebündelt und auf das Ziel gerichtet, dessen Entfernung ermittelt werden soll. Das Ziel-Objekt darf **keine** spiegelnde Oberfläche besitzen, sondern soll den auf das Objekt projizierten Punkt möglichst diffus reflektieren, d.h. der Punkt ist aus allen davor befindlichen Winkeln sichtbar.

Anmerkung: In den meisten Beschreibungen der IRDMS wird von einem Lichtstrahl gesprochen, der vom Objekt reflektiert wird. Das ist aus physikalischer Sicht grundsätzlich richtig, lässt aber auch sofort an das Gesetz **Einfallswinkel gleich Ausfallswinkel** denken, was in diesem Fall in eine völlig falsche Denkrichtung führt. Gerade um dieses Gesetz zu umgehen, darf das Objekt nicht wie ein Spiegel wirken, sondern soll wie eine Projektionswand nur den Lichtpunkt als "projiziertes Bild" darstellen.

Der Lichtpunkt wird mit Hilfe einer Empfangsoptik wie mit einer Kamera erfasst und nicht auf einer lichtempfindlichen Filmschicht oder auf einem CCD-Sensor (**C**harge **C**oupled **D**evice) abgebildet, sondern wird auf einen eindimensionalen optischen Positionssensor **PSD** (**Position Sensitive Device**) gelenkt und vermessen. Das PSD ist im Prinzip eine lineare Kette von dicht aufeinander folgenden Fotodioden.

Im Idealfall wird der Lichtpunkt nur von einer Fotodiode in der Kette empfangen und deren Lage entspricht einer genau bestimmbaren Entfernung des Objektes von dem IRDMS.

3.1.1 IR-Triangulation

Umfangreiche Beschreibungen der Triangulation, d.h. der Vermessungstechnik auf Grund der Bestimmung der Seiten und Winkel in einem Dreieck, werden u. a. im Wikipedia und einschlägigen Lehrbüchern der Vermessungstechnik im terrestrischen wie im astronomischen Bereich beschrieben. Hier soll die Triangulation nur soweit erläutert werden, wie sie zum Verständnis der IRDMS von Nutzen ist.

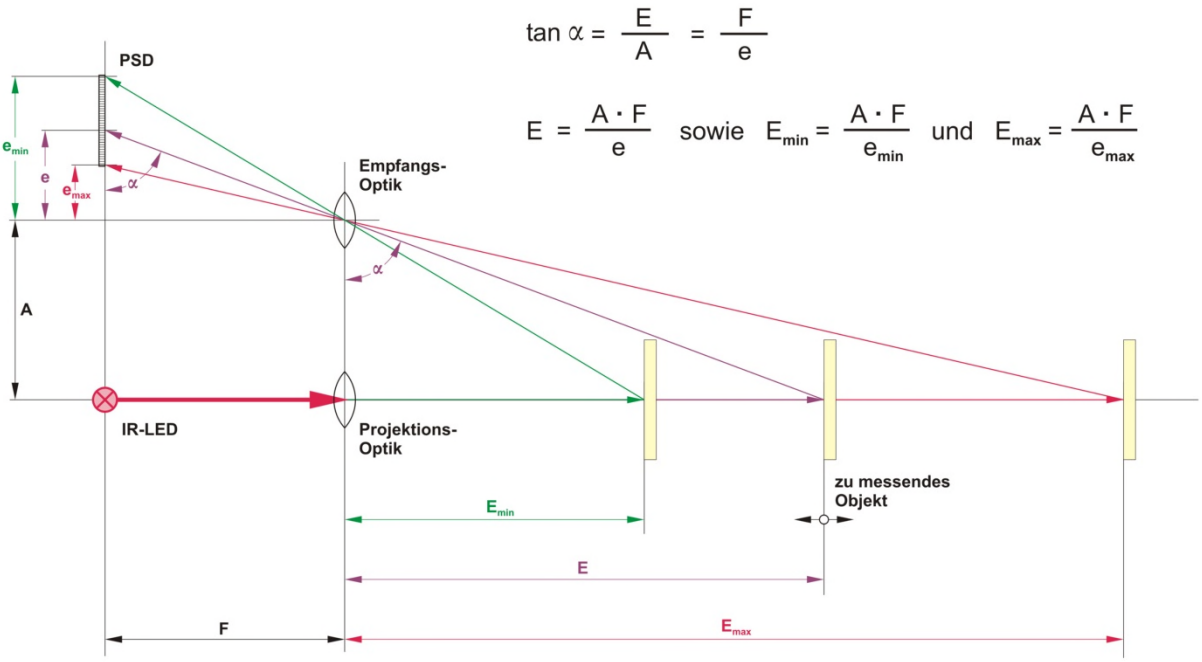


Bild 3.1.1-01: Prinzip der Triangulation eines IRDMS (Bildvergrößerung)

Natürlich sind die Maße in dem **Bild 3.1.1-01** nicht maßstabsgerecht und stellen nur ein qualitatives Abbild der Strahlengänge dar. Trotzdem sind die Zuordnungen der Strecken zum Winkel **α** exakt wiedergegeben, so dass sich gem. der Tangensfunktion (der Tangens eines Winkels ergibt sich im rechtwinkligen Dreieck aus Gegenkathete dividiert durch Ankathete) die in der Abbildung dargestellten Gleichungen aufstellen lassen.

Aus ihnen lässt sich die grundsätzliche Abhängigkeit der Entfernung des Objektes vom IRDMS-Modul herleiten

$$\text{Entfernung } \mathbf{E} = \frac{\text{Abstand der Optiken } \mathbf{A} * \text{ Brennweite } \mathbf{F} \text{ der Empfangs-Optik}}{\text{Abstand } \mathbf{e} \text{ des PSD-Lichtpunktes von der optischen Achse}}$$

oder

$$\text{Abstand } \mathbf{e} = \frac{\text{Abstand der Optiken } \mathbf{A} * \text{ Brennweite } \mathbf{F} \text{ der Empfangs-Optik}}{\text{Entfernung } \mathbf{E} \text{ des Objektes vom Sensor}}$$

wobei **A** und **F** Konstante der praktischen Ausführung des Moduls darstellen. $\mathbf{E_{min}}$ und $\mathbf{E_{max}}$ sind durch die geometrischen Abmessungen $\mathbf{e_{min}}$ und $\mathbf{e_{max}}$, also durch die Länge des PSD, als auch durch den beschränkten Blickwinkel der Empfangs-Optik bedingt.

Die Funktion

e = f (const/E) **const** ist das konstante Produkt aus **A * F**

ist eine Hyperbelfunktion, wie man auch an der tatsächlichen Kurve der Ausgabe-Charakteristik des IRDMS-Moduls ersehen kann (die Ausgangsspannung $\mathbf{U_{IR}}$ entspricht direkt dem Abstand **e** des PSD-Lichtpunktes von der optischen Achse der Empfangs-Optik):

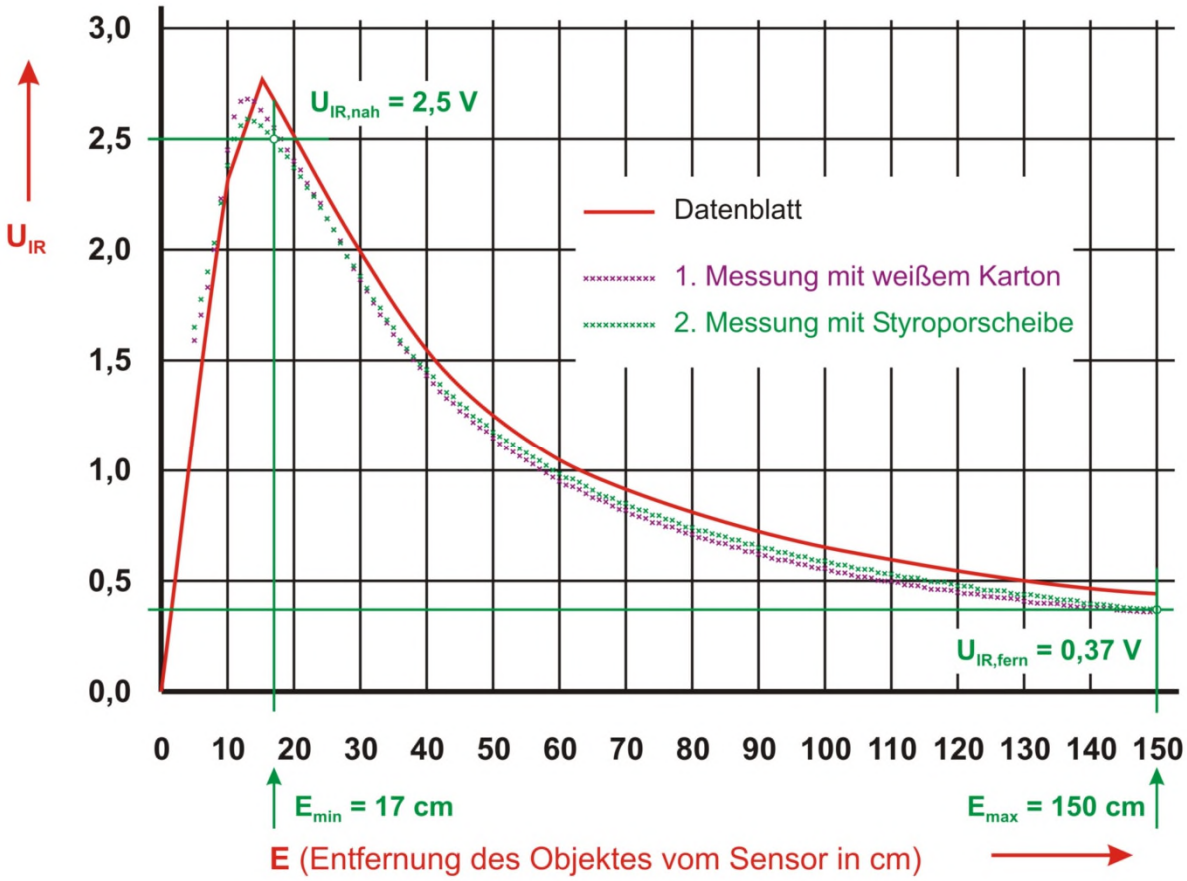


Bild 3.1.1-02: Ausgabe-Charakteristik des GP2Y0A02YK0F (Bildvergrößerung)

Bei dieser Messmethode ist die Beschaffenheit und Lage des Objektes relativ ohne Belang: Ob sich die Oberfläche des Objektes schräg zum auftreffenden IR-Lichtstrahl befindet oder genau senkrecht zu ihm, ob sie weiß oder grau ist, ergibt innerhalb der gesetzten Grenzen stets einen Lichtpunkt, der von der Empfangs-Optik erkannt und auf dem PSD abgebildet wird. Als Störfaktoren kommen hauptsächlich Streulicht, ggf. weitere Reflexionen des IR-Stahls und Bewegungen des Objektes (abgesehen von denen in Richtung Sensor) während der Messung in Betracht. Der Sensor soll grundsätzlich so angeordnet werden, dass eine mögliche Objekt-Bewegungsrichtung quer zum Sensor im rechten Winkel zu diesem erscheint (also senkrecht zur Bildebene von **Bild 3.1.1-01**).

3.1.2 Signalverarbeitung innerhalb des IRDMS-Moduls

Das Modul erzeugt als Ausgabesignal eine zur Entfernung proportionale Spannung. Leider - wie oben beschrieben - steht diese jedoch nicht in einem linearen Verhältnis zur Entfernung, sondern bildet im praktischen Bereich der Messung annähernd eine Hyperbel.

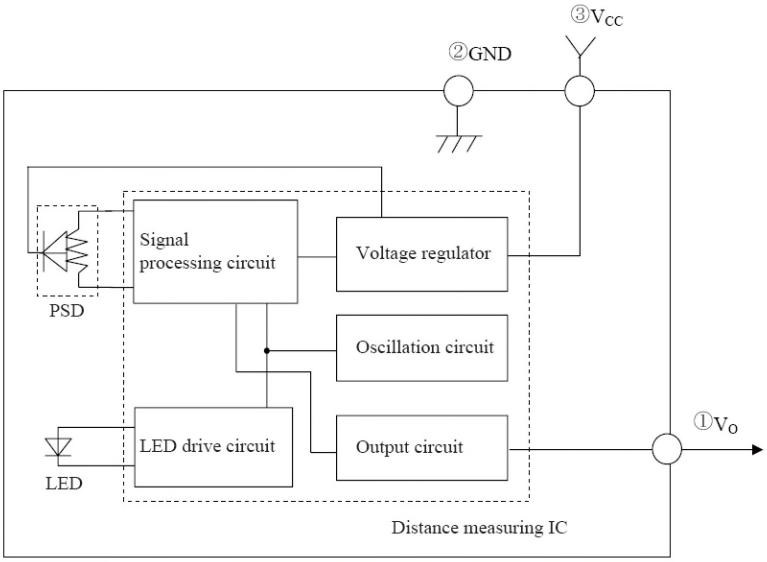


Bild 3.1.2-01: Block-Diagramm (Bildvergrößerung)

① Ausgangsspannung $\mathbf{V_0}$
② Masse (GND; Ground)
③ Spannungsversorgung $\mathbf{V_{cc}}$ = +5 V

LED Drive Circuit

In diesem Block wird die LED angeregt, während eines 38,3 ms dauernden Messzyklusses 32 sehr kurze und sehr starke IR-Lichtimpulse auszusenden. Der Empfang am PSD-Empfänger wird mit diesen 32 Impulsen synchronisiert und der Mittelwert der Einzelmessungen wird dem **Output Circuit** für die Ausgabe zugeführt. Der **Oscillation Circuit** (Oszillator) steuert den gesamten zeitlichen Ablauf des Moduls einschließlich der Synchronisation zwischen **Led Drive Circuit** und dem **Signal Processing Circuit**.

Signal Processing Circuit

In diesem Block sind mehrere Schaltkreise unterschiedlicher Funktionen untergebracht. Da ist 1. ein Verstärker zur Erkennung und zur Beseitigung von Störungen auf Grund von umgebenden Interferenz-Licht während des Empfangs der IR-Lichtimpulse und 2. ein Verstärker zur Verstärkung des PSD-Ausgabe-Pegels. Eine besondere Funktion dient der Umwandlung der an den verschiedenen Fotodioden des PSD empfangenen Lichtimpulse in entsprechende Spannungswerte und schließlich dient eine andere Rechenfunktion dazu, den Mittelwert aus den 32 Einzelmessungen zu berechnen.

3.2 Anschaltung des IRDMS an den Mikrocontroller

Da die sehr intensiven IR-Lichtimpulse Stromspitzen von 230 mA bei einer Grundbelastung von 10 mA verursachen, wird die Stromversorgung ebenfalls pulsweise stark belastet. Die Firma SHARP empfiehlt daher, RC-Filter im Stromkreis vorzusehen. Die Versorgungsspannung $\mathbf{V_{cc}}$ soll mit einem Kondensator von $\mathbf{C_1}$ = 2200 :F (unmittelbar zwischen Pin ③ und ②) geglättet werden. Auch das Ausgangssignal weist eine Restwelligkeit von ca. 20 mV auf, so dass ein RC-Glied als Tiefpass aus $\mathbf{C_2}$, $\mathbf{R_1}$ und $\mathbf{R_2}$ nachgeschaltet wird. Der Spannungsteiler aus $\mathbf{R_1}$ und $\mathbf{R_2}$ dient gleichzeitig zur Anpassung an die Eingangsspannung $\mathbf{U_{mess}}$ des Mikrocontrollers (vergl. **Bild 3.2-01**):

$$U_{mess} = \frac{R_1 * U_{IR}}{(R_1 + R_2)} \qquad U_{mess,nah} = \frac{R_1 * U_{IR,nah}}{(R_1 + R_2)} \qquad U_{mess,fern} = \frac{R_1 * U_{IR,fern}}{(R_1 + R_2)}$$

Da die interne Mikrocontroller-Referenzspannung von 1,1 Volt benutzt wird, darf die Eingangsspannung U_{mess} nicht höher als diese Spannung werden. Es wird ein Spannungsteiler wie folgt gewählt:

R_1 = 4k7 (1% Toleranz)

R_2 = 6k8 (1% Toleranz)

$$\frac{R_1}{(R_1 + R_2)} = 0{,}4087$$

so dass sich ein oberer und unterer Grenzwert für U_{mess} ergibt:

$$U_{mess,nah} = \frac{4k7 * 2{,}5\ V}{(4k7 + 6k8)} = 1{,}022\ V$$

$$U_{mess,fern} = \frac{4k7 * 0{,}37\ V}{(4k7 + 6k8)} = 0{,}151\ V$$

Das Schaltbild für den Messkopf sieht dann so aus, wobei die Eingangsspannung U_{mess} an **AD5**, dem Eingang des Analog-Digital-Umsetzers des Mikrocontrollers **ATmega88**, anliegt:

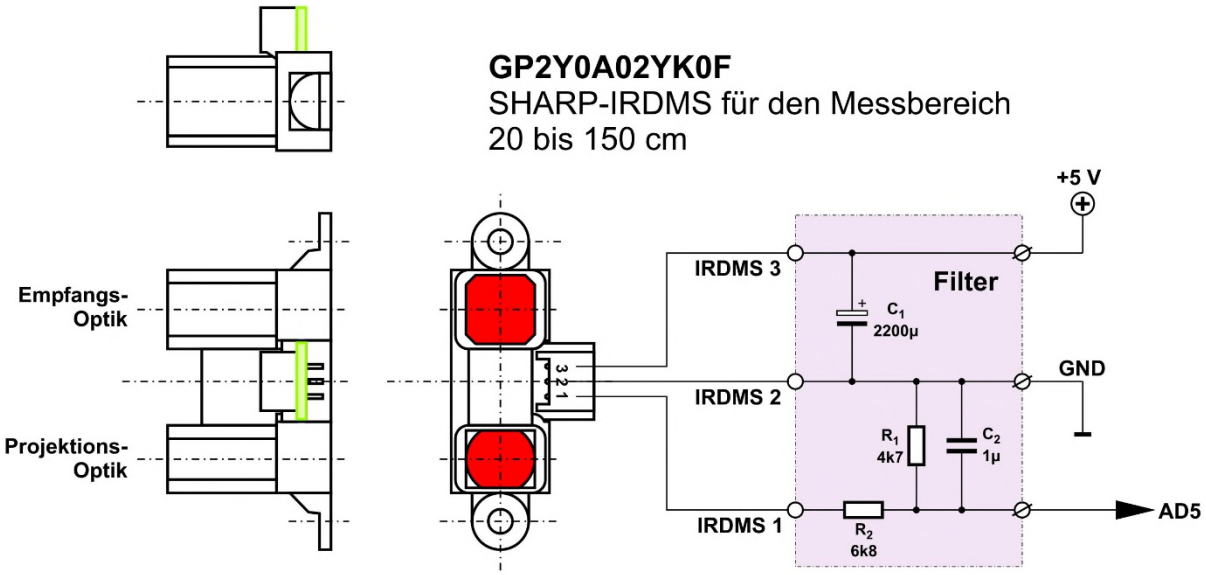


Bild 3.2-01: Anschaltung des IRDMS (Bildvergrößerung)

3.2.1 Der Analog-Digital-Umsetzer (ADC - Analog Digital Converter)

Zur Auswertung des analogen Signals des IRDMS wird der Analog-Digital-Umsetzer (ADC) des Mikrocontrollers **ATmega88** verwendet.

Mittels des ADC's wird die Eingangsspannung quantisiert, d.h. die Amplitude wird innerhalb sehr kurzer Zeitabschnitte ermittelt und diesen Zeitabschnitten zugeordnet. Jedes Signal stellt also nach der Umsetzung in einem Amplituden-Zeit-Diagramm eine **treppenförmige** Kurve dar. Die Hauptparameter eines ADC's sind seine **Auflösung in Bit** und seine **Abtastrate in Sampels pro Sekunde** (**SPS** bzw.

kSPS - kilo-Samples per Second) wovon die maximale **Umsetzungsrate** abhängt. Die Auflösung ist gleichzeitig die Genauigkeitsgrenze für die Auswertung.

Der ADC braucht zur Umsetzung Zeit. Je kürzer diese ist, desto höher kann die Abtastrate sein. Die Quantisierung wird in einer endlichen Anzahl **Quantierungsstufen** vorgenommen. Auch wenn dafür 10 Bits zur Verfügung stehen, bedingt sie eine reduzierte Auflösung, wie man im Schema-Bild ablesen kann.

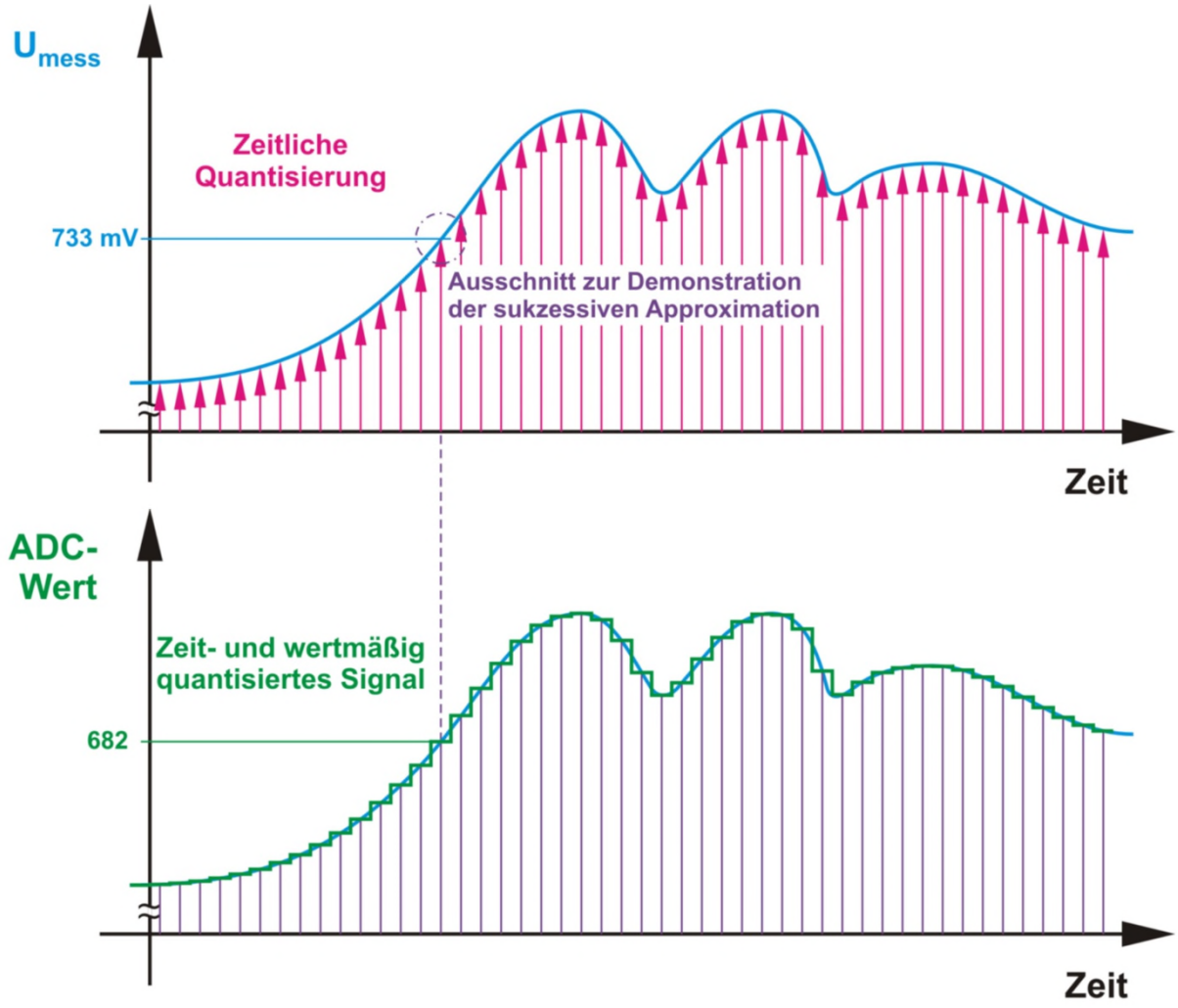


Bild 3.2.1-01: Schema der Umsetzung (Sampeln) der Eingangsspannung U_{mess}
(Bildvergrößerung)

Der ADC des ATmega88 ermittelt den **ADC-Wert** durch eine 10-stufige **sukzessive Approximation** (10 aufeinander folgende Annäherungen) an die Eingangsspannung, die an einem der 8 vorhandenen Analog-Multiplexer-Eingänge anliegt. Gewählt wird hier der Eingang **ADC5** (PC5). Die Eingangsspannung ist auf **0 Volt** (GND) bezogen.

Besonderheiten der sukzessiven Approximation sind:

- ihre Anwendung ist besonders für abtastende Messungen geeignet
- feste Abgleichdauer von **n** Taktperioden; hier: **n = 10**
- die Eingangsspannung muss während des Abgleichs konstant bleiben; hier: **Sample&Hold**
- die Referenz legt die maximale Amplitude des Eingangssignals fest; hier: **U_{ref} = 1100 mV**

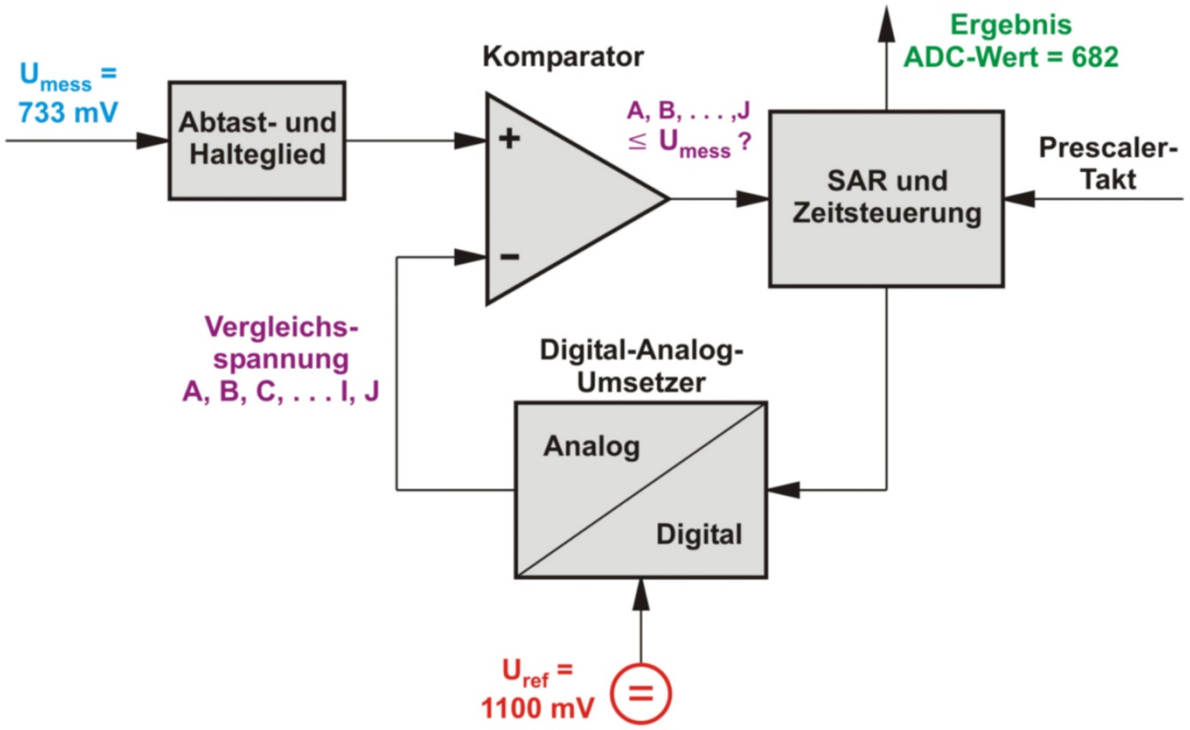


Bild 3.2.1-02: Aufbau des ADC's (Bildvergrößerung)

Das Messsignal $\mathbf{U_{mess}}$ (hier beispielhaft mit dem Momentan-Wert **733 mV** angenommen) wird in 10 Schritten digitalisiert, wobei die Genauigkeit bei jedem Schritt um 1 Bit steigt. Bei jedem Schritt wird die Eingangsspannung mit einem konstanten Spannungswert **A**, **B**, **C**, . . . **I, J** verglichen, den ein Digital-Analog-Umsetzer (**DAC**) aus der konstanten Referenzspannung $\mathbf{U_{ref}}$ erzeugt:

Tabelle 3.2.1-01: Analog-Digital-Umsetzung

$A = U_{ref}/2$	=	550,00000000	< 733 mV	1
$B = A \pm U_{ref}/4$	=	825,00000000	> 733 mV	0
$C = B \pm U_{ref}/8$	=	687,50000000	< 733 mV	1
$D = C \pm U_{ref}/16$	=	756,25000000	> 733 mV	0
$E = D \pm U_{ref}/32$	=	721,87500000	< 733 mV	1
$F = E \pm U_{ref}/64$	=	739,06250000	> 733 mV	0
$G = F \pm U_{ref}/128$	=	730,46875000	< 733 mV	1
$H = G \pm U_{ref}/256$	=	734,76562500	> 733 mV	0
$I = H \pm U_{ref}/512$	=	732,61718750	< 733 mV	1
$J = I \pm U_{ref}/1024$	=	733,69140625	> 733 mV	0
Ergebnis der sukzessiven Approximation:				1010101010
Das entspricht dem dezimalen ADC-Wert (abgerundet):				682

Je nachdem, ob die Vergleichsspannung des DAC's kleiner/gleich oder größer als $\mathbf{U_{mess}}$ ist, steuert das **S**uccessive **A**pproximation **R**egister (**SAR**) den Spannungswert im nächsten Schritt um die halbe Schrittweite des letzten Schritts nach oben oder nach unten. Dadurch nähert sich die Spannung des DAC's immer mehr der momentanen Eingangsspannung (siehe **Bild 3.2.1-03: Sukzessive Approximation**).

Zum Schluss, wenn das 10. Bit des DAC's gesetzt ist, **entspricht** der Wert des DAC's der momentanen Eingangsspannung und die registrierten Bits ergeben den **ADC-Wert**.

Die einzelnen 10 Approximations-Stufen werden mit **A**, **B**, **C**, **D**, **E**, **F**, **G**, **H**, **I** und **J** bezeichnet; jeder Stufe ist ein Bit zugeordnet, mit dem **MSB** beginnend. Die Addition/Subtraktion (±) in der Stufen-Gleichung wird wie folgt durchgeführt:

- wenn **A**, **B**, **C**, . . . ≤ $\mathbf{U_{mess}}$ dann ist das Bit **ADC-Wert**$_{\mathbf{A, B, C, \ldots}}$ = **1** andernfalls **0**
- es folgt eine Addition, wenn das Bit der vorhergehenden Stufe **1** ergab und
- es folgt eine Subtraktion, wenn das Bit der vorhergehenden Stufe **0** ergab

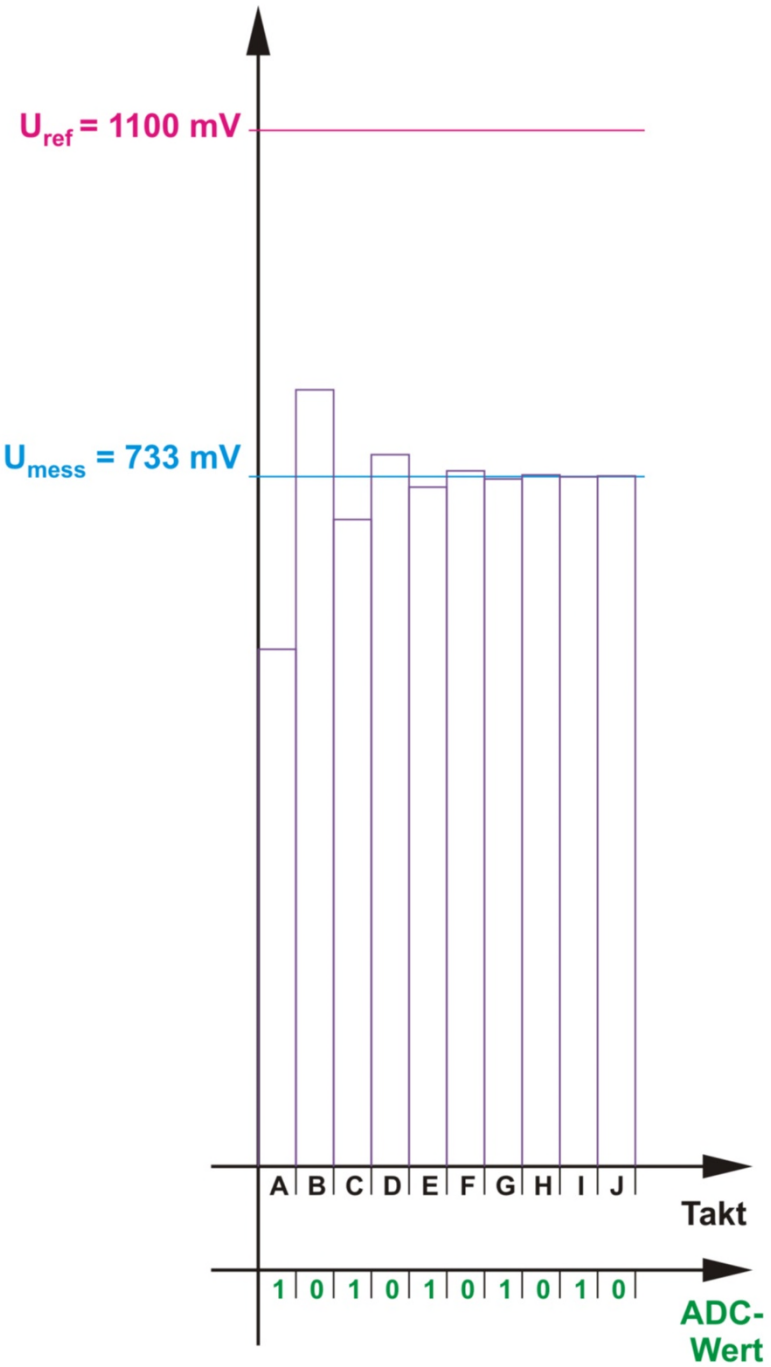


Bild 3.2.1-03: Sukzessive Approximation

Der **ADC-Wert** errechnet sich allgemein nach der Formel (bei 10-bittiger Auflösung; Faktor 1024):

$$\textbf{ADC-Wert} = \frac{\mathbf{U_{mess} * 1024}}{\mathbf{U_{ref}}}$$

Das ergibt für $\mathbf{U_{mess}}$ **= 733 mV** und $\mathbf{U_{ref}}$ **= 1100 mV** den dimensionslosen **ADC-Wert = 682**

Der ADC benötigt während des Umsetzungsvorgangs ein konstantes Eingangssignal, um zu korrekten Ergebnissen zu kommen, sonst sinkt seine Auflösung. Es ist daher die Verwendung eines Abtast- und Haltegliedes (**Sample&Hold**) notwendig. Unter konstantem Eingangssignal ist ein Signal gemeint, das sich innerhalb der Umsetzungszeit maximal um die halbe Höhe der kleinsten Stufe des ADC's ändert.

3.2.2 Benutzte Register und Register-Bits

Im folgenden Bild ist das Zusammenspiel der einzelnen Blöcke mit den Registern dargestellt.

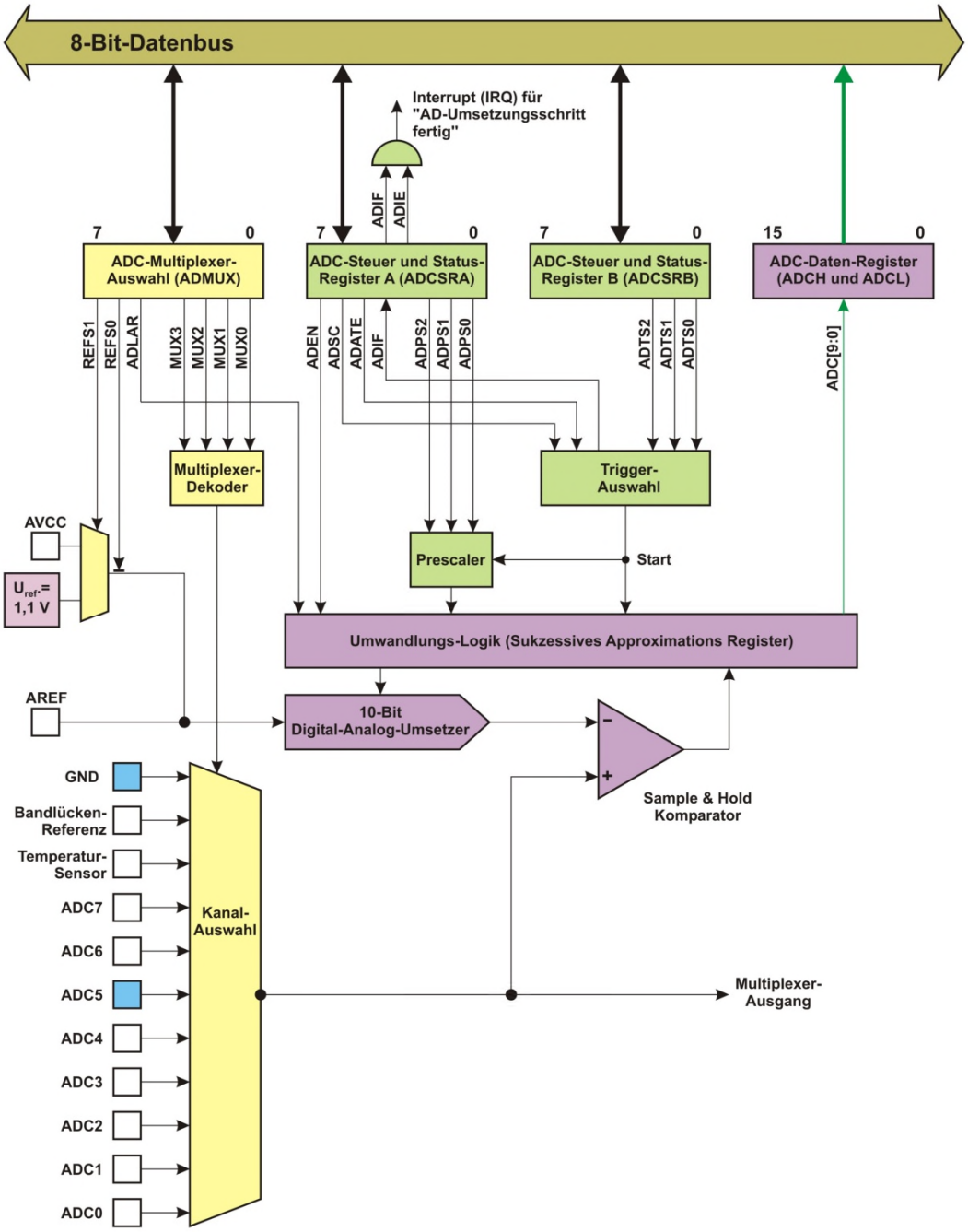


Bild 3.2.2-01: Blockschaltbild des AD-Umsetzers (Bildvergrößerung)

Nicht jeder Block und nicht jedes Bit wird in diesem Projekt benötigt und manche können nur alternativ benutzt werden. Für das **AVR-Projekt IRDMS** als Grundlage für die Anwendung **RWNA-Pegel** (zur Messung, Anzeige und Berechnung der Wasserentnahmen in einer **R**egen**W**asser**N**utzungs**A**nlage) werden die Einstellungen in den Registern wie folgt vorgenommen.

Zuerst wird der Analog-Digital-Umsetzer initialisiert, d.h. er wird in Grundstellung gebracht:

ADCSRA - ADC-Steuer- und Status-Register A

(**AD**C **C**ontrol and **S**tatus **R**egister **A**)

Bit-Nummer:		7	6	5	4	3	2	1	0
Adresse:	0x7A	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0
Lesen/Schreiben:		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Grundwert:		0	0	0	0	0	0	0	0

Bild 3.2.2-02: Register ADCSRA

1. Die Bits im Register **ADCSRA** werden gesetzt:

- Bit7 **ADEN** = **1** der ADC wird aktiviert
- Bit6 **ADSC** = **0** zunächst **0**, es wird im **Single Conversion Mode** vor jeder Abfrage per Programm auf **1** und nach erfolgter Umsetzung automatisch wieder auf **0** gesetzt
- Bit5 **ADATE** = **0** keine automatische Triggerung
- Bit4 **ADIF** = **0** ADC-Interrupt-Flag wird ignoriert
- Bit3 **ADIE** = **0** der ADC-Interrupt wird nicht aktiviert

Bit2 bis Bit0 bestimmen den Divisionsfaktor. Der eingestellte System-Takt wird durch diesen Teiler (Prescaler) dividiert und ergibt den Eingabe-Takt für den ADC:

- Bit2 **ADPS2** = **1**
- Bit1 **ADPS1** = **1**
- Bit0 **ADPS0** = **1** Der Teiler ist **128**

Wenn der System-Takt z.B. auf **8 MHz** eingestellt ist, wird der ADC mit einer Taktrate von **62,5 kHz** betrieben, d. h. ein Takt dauert **16 µs**. Für jeden Mess-Zyklus werden 10 Taktperioden für die sukzessive Approximation und 3 Perioden für die Steuerung benötigt, so dass der **ADC-Wert** erst nach **208 µs** zur Verfügung steht (siehe **TIMING**).

ADCSRB - ADC-Steuer- und Status-Register B

(**AD**C **C**ontrol and **S**tatus **R**egister **B**)

Bit-Nummer:		7	6	5	4	3	2	1	0
Adresse:	0x7B	-	ACME	-	-	-	ADTS2	ADTS1	ADTS0
Lesen/Schreiben:		R	R/W	R	R	R	R/W	R/W	R/W
Grundwert:		0	0	0	0	0	0	0	0

Bild 3.2.2-03: Register ADCSRB

2. Alle Bits im Register **ADCSRB** werden auf **0** gesetzt:

- Bit6 **ACME** = **0** Das Bit wird vom ADC nicht verwendet (siehe Analog Comparator)

Bit2 bis Bit0 bestimmen die Auswahl des Trigger-Verfahrens bzw. des Triggerverursachers. Durch Setzen aller Bits auf **0** wird der **Free Running Mode** eingestellt, d.h. es erfolgt keine externe Triggerung:

- Bit2 **ADTS2** = **0**
- Bit1 **ADTS1** = **0**
- Bit0 **ADTS0** = **0** Free Running Mode

DIDR0 - Pin-Deaktivierungs-Register 0

(**D**igital **I**nput **D**isable **R**egister **0**)

Bit-Nummer:		7	6	5	4	3	2	1	0
Adresse:	0x7E	-	-	ADC5D	ADC4D	ADC3D	ADC2D	ADC1D	ADC0D
Lesen/Schreiben:		R	R	R/W	R/W	R/W	R/W	R/W	R/W
Grundwert:		0	0	0	0	0	0	0	0

Bild 3.2.2-04: Register DIDR0

3. Die Pins **ADC6** und **ADC7** sind bei der PDIP-Version (28-Pin-Ausführung) des ATmega88 nicht nach außen geführt worden, so dass der Eingangs-Pin **ADC5** benutzt werden soll. Um den Stromverbrauch des digitalen Eingabe-Puffers zu reduzieren, sollten alle Eingangs-Pins, die nicht benötigt werden, deaktiviert werden. Das geschieht in dem Register **DIDR0**. Jedes hier mit einem Eingang korrespondierende Bit wird auf **1** gesetzt, um den entsprechenden Eingangs-Pin tot zu legen. Die Pins **ADC6** und **ADC7** haben keinen digitalen Eingangs-Puffer und benötigen somit keine korrespondierenden Bits. Bit5 bis Bit0 werden besetzt:

- Bit5 **ADC5D** = **0** Freigeschalteter Eingangs-Pin **ADC5**
- Bit4 **ADC4D** = **1** **ADC4** deaktiviert
- Bit3 **ADC3D** = **1** **ADC4** deaktiviert
- Bit2 **ADC2D** = **1** **ADC4** deaktiviert
- Bit1 **ADC1D** = **1** **ADC4** deaktiviert
- Bit0 **ADC0D** = **1** **ADC4** deaktiviert

Die Initialisierung für den ADC besteht also nur aus 3 **C**-Anweisungen (man kann die Anweisungen natürlich auch vereinfacht hinschreiben; hier wurde die Darstellung so gewählt, wie sie sich aus den Definitionen der Header-Dateien ergibt):

```
void adc_init (void)                          // ADC-Initialisierungs-Funktion
{
ADCSRA = 0x80 | 0x04 | 0x02 | 0x01;           // ADCSRA = 10000111
ADCSRB = 0;                                   // ADCSRB = 00000000
DIDR0  = 0x10 | 0x08 | 0x04 | 0x02 | 0x01;    // DIDR0  = 00011111
}
```

Damit ist die Initialisierung des ADC abgeschlossen und es kann ein Mess-Zyklus (ein Lesevorgang) durch Setzen der entsprechenden Bits im Register **ADMUX** eingeleitet werden:

ADMUX - ADC-Multiplexer-Auswahl-Register

(**AD**C **MU**ltiple**X**er Selection Register)

Bit-Nummer:		7	6	5	4	3	2	1	0
Adresse:	0x7C	REFS1	REFS0	ADLAR	-	MUX3	MUX2	MUX1	MUX0
Lesen/Schreiben:		R/W	R/W	R/W	R	R/W	R/W	R/W	R/W
Grundwert:		0	0	0	0	0	0	0	0

Bild 3.2.2-05: Register ADMUX

Bit7 und Bit6 bestimmen die Auswahl der Referenzspannung. Wenn beide Bits gesetzt sind, wird die interne Referenzspannung $\mathbf{U_{ref}}$ von **1,1 V** verwendet. Da die interne Referenz benutzt werden soll, ist ein externer Kondensator zwischen Pin **AREF** und Masse zu schalten (auf dem Testboard bereits vorhanden). Bit 5 legt die Ablage des Messergebnisses im Doppel-Register **ADCH** und **ADCL** fest; **0** für rechtsbündig (entspricht auch der Voreinstellung) und **1** für linksbündig:

- Bit7 **REFS1** = **1**
- Bit6 **REFS0** = **1** Einstellung auf die interne Referenzspannung von **1,1 Volt**;
- Bit5 **ADLAR** = **0** Ergebnis wird rechtsbündig im Doppel-Register **ADCH** und **ADCL** abgelegt

Bit3 bis Bit0 bestimmen den Eingabe-Pin. Von den vielen Möglichkeiten wird der Pin **ADC5** in der Header-Datei **`application.h`** der aktuellen Anwendung festgelegt:

- Bit3 **MUX3** = **0**
- Bit2 **MUX2** = **1**
- Bit1 **MUX1** = **0**
- Bit0 **MUX0** = **1** Der Eingangs-Pin ist auf **ADC5** eingestellt

Mit Setzen des Bit6 **ADSC** im Register **ADCSRA** (**S**ingle **C**onversion Mode) wird der Mess-Zyklus eingeleitet. Die Kodierung eines Messvorganges würde dann so aussehen:

```
void adc_read(void)

{
  ADMUX    = 0x80 | 0x40 | 0x05;              // ADMUX   = 11000101

  ADCSRA  |= 0x04;                            // ADCSRA  = 11000111
                                              // nur ADSC wird gesetzt!
  while (ADCSRA & 0x40)                       // Warte bis der ADC automatisch
                                              // das Bit ADSC nach der
                                              // Umsetzung auf 0 gesetzt hat
  {
  }

  adc-mess = ADCL + ADCH * 256;               // Uebergabe des ADC-Mess-Wertes
                                              // als 16-Bit Gleitkomma-Ergebnis
}
```

3.2.3 Timing

Bei einem Messvorgang der Eingangsspannung $\mathbf{U_{mess}}$ beginnt die Umsetzung bei der nächsten steigenden Flanke der Taktperiode des Prescalers. Die erste Umsetzung nach der Initialisierung des ADC's dauert 25 Taktperioden, da zu Beginn der Multiplexer, die $\mathbf{U_{ref}}$-Einstellung und das Abtast- und Halteglied (Sample&Hold) bereits 13,5 Perioden benötigen:

ADC-Zeitdiagramm für die 1. Umsetzung im Single Conversion Mode

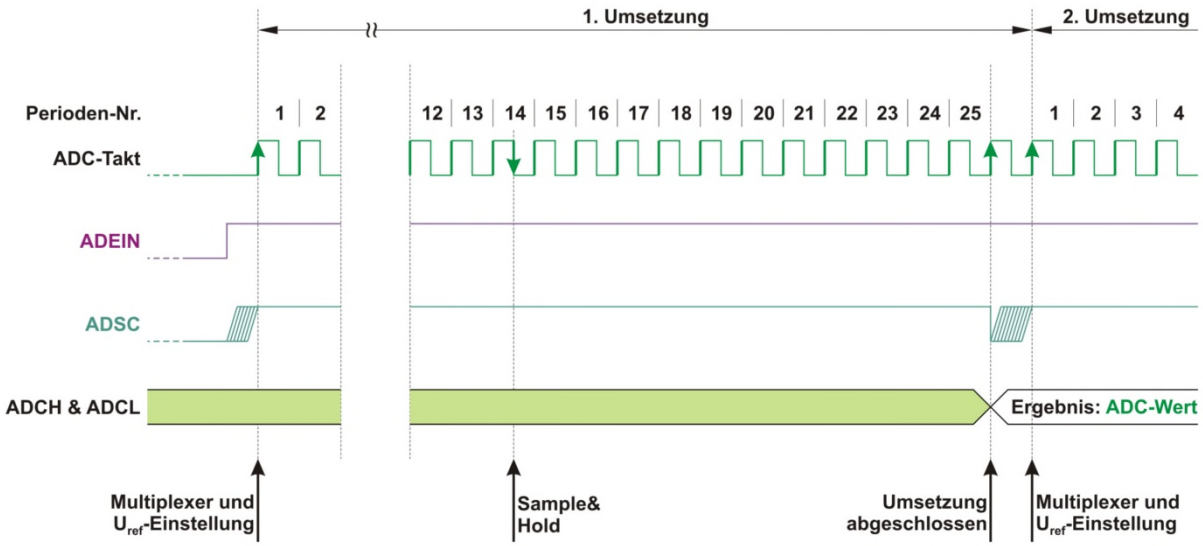


Bild 3.2.3-01: TIMING der ersten Initialisierungs-Umsetzung (Bildvergrößerung)

Bei den folgenden Umsetzungen gibt es durch Sample&Hold eine Verzögerung von 1,5 Perioden, so dass jede weitere Umsetzung nur 13 Taktperioden dauert:

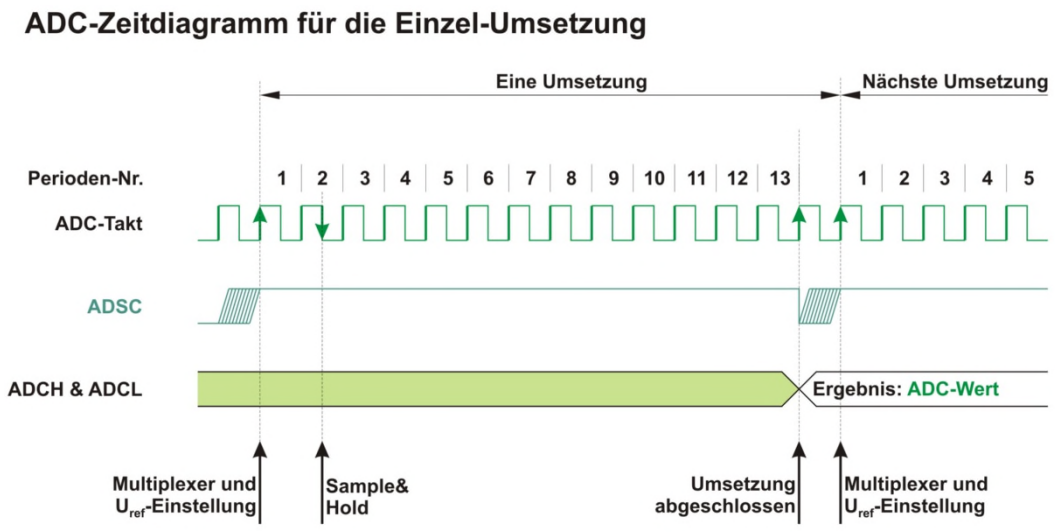


Bild 3.2.3-02: TIMING einer nachfolgenden Umsetzung (Bildvergrößerung)

3.3 Praktische Anwendung "RWNA"

(http://www.alenck.de/Regenwassernutzung.html)

Damit die lange Einleitung und Erklärung zum ADC nicht nackte Theorie bleibt und auch nicht zur bloßen Spielerei dient (bei der man bekanntlich ebenfalls viel lernen kann), soll eine praktische Anwendung beschrieben werden:

Um in einer **R**egen**W**asser**N**utzungs**A**nlage (**RWNA**) den **Wasser-Pegel** in einer vergrabenen Zisterne zu ermitteln und zu kontrollieren, soll mittels des oben beschriebenen IR-Sensors IRDMS der Abstand zwischen Wasseroberfläche und IR-Sensor gemessen werden. Und da die Auswertung mit dem beispielhaften Mikrocontroller ATmega88 vorgenommen werden soll, wird die Messung nicht nur zur Anzeige gebracht, sondern es wird der Überlauf und der Leerlaufschutz für die Motorpumpe gesteuert und zusätzlich werden die Wasserentnahmen berechnet und aufsummiert.

Eine Besonderheit weist die RWNA auf: Die Zisterne hat keinen Überlauf in die Kanalisation, so dass überschüssiges Wasser bei starken Regenfällen automatisch in einen Gartenteich geleitet werden muss. Der Teich besitzt einen natürlichen "Überlauf in die Natur".

Die Zisterne mit einem Fassungsvermögen von 3.500 Litern wird als **ABSOLUT VOLL** betrachtet, wenn der Pegel eine Höhe von **120 cm** erreicht hat. Der Pegelstand **115 cm** wird dagegen als normal **VOLL** gewertet. Die Grundüberlegung ist, dass beim Erreichen von **ABSOLUT VOLL** ein Magnetventil geöffnet wird, welches über ein Rohr mit dem Gartenteich verbunden ist. Die Gartenwasserpumpe spricht darauf durch den entstehenden Unterdruck an und pumpt so das überschüssige Wasser in den Teich. Wenn der Pegel dann die Marke von **115 cm** erreicht hat, soll das Magnetventil wieder schließen, damit die Zisterne nicht leer gepumpt wird.

Bild 3.3-01: Schwimmerschalter

Bei der Einrichtung der RWNA wurde diese Prozedur durch 11 Schwimmerschalter gesteuert, die in 12 cm Abstand an einem Stab in der Zisterne befestigt waren. Schwimmerschalter bestehen aus einem in einem Glasröhrchen eingeschweißten Reed-Kontakt (im Bild: hellgraues Röhrchen) und einem Ring-Permanent-Magneten, der in einem ringförmigen Styropor eingebettet ist und den Reed-Kontakt umschließt (im Bild: dunkler Zylinder). Im Ruhezustand (Wasser-Pegel unterhalb des Auftriebs des Ringkörpers) ist der Reed-Kontakt geöffnet und wenn das Wasser über einen bestimmten Punkt angestiegen ist, schließt der Kontakt und zeigt damit einen bestimmten Pegelwert an.

Der Nachteil dieser Schalter ist - abgesehen vom hohen Preis - dass die Reed-Kontakte offensichtlich beim Einsatz in rauer Umgebung im Freien häufig ihren Geist aufgeben: Die Kontaktgabe ist nicht mehr einwandfrei! Ein weiterer Nachteil ist, dass die Lötstellen der Anschlüsse zwangsweise untertauchen und - trotz erheblicher Isolation mit Epoxydharz - dem "sauren" Regenwasser ausgesetzt werden und korrodieren: Kupfer und Zinn und Säure ergeben ein "herrliches" galvanisches Element!

3.3.1 Zur Hardware

RWNA-Steuerung-NEU ersetzt die bisherige **RWNA-Steuerung-ALT**.

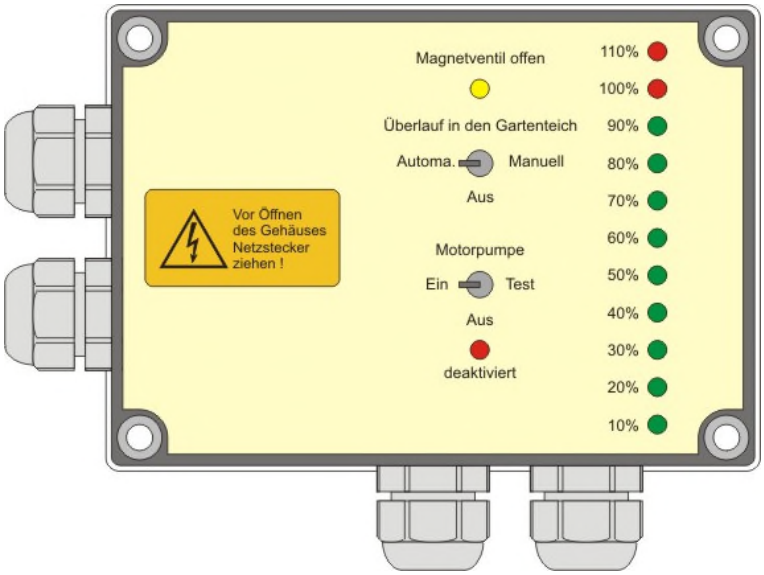


Bild 3.3.1-01:
RWNA-Steuerung - ALT

Die Wasserstandsanzeige wird in der neuen RWNA-Steuerung nicht mehr mit LEDs angezeigt, sondern erhält nun eine LCD-Anzeige, mit erheblich mehr Informationen:

- Pegel,
- Füllstand,
- Verbrauch und
- Überlauf

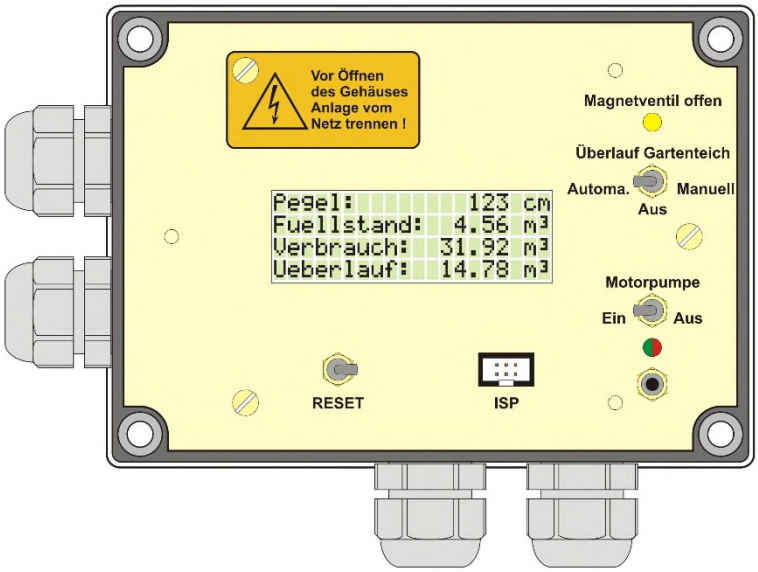


Bild 3.3.1-02
RWNA-Steuerung - NEU

Darüber hinaus sieht sie 4 Schalter für grundlegende Funktionen und eine ISP-Schnittstelle vor:

1. Schalter zur manuellen Steuerung des Magnetventils (z.B. zum Leeren der Zisterne für die Winterpause)

2. Schalter zum Schalten der Motorpumpe (Stilllegung für die Winterpause)

3. Reset-Taster (für die Wiederinbetriebnahme nach der Winterpause)

4. Unterhalb der Motor-Anzeige-LED ein Tastschalter, um für die Winterpause eine vollständige Entleerung der Zisterne bei Pegelstellung 0 cm zu gewährleisten (vergl. RWNA - Konzept der Überlauf-Steuerung und RWNA - Gesamt-Schaltbild der Anlage).

5. ISP-Schnittstelle zur Programmierung des Mikrocontrollers innerhalb des Systems

Neue Hardware

Der grundsätzliche Aufbau ist in **Bild 3.3.1-03: Überlaufsteuerung und Leerlaufschutz der Motorpumpe** dargestellt.

Die manuelle Steuerung, die Überlaufsteuerung und der Leerlaufschutz sollen zunächst ausführlich beschrieben werden, um dann auf die Programmierung einzugehen.

In diesem Block wird die LED angeregt, während eines 38,3 ms dauernden Messzyklusses 32 sehr Der Messstab mit den 11 Schwimmerschaltern wird durch einen Stab mit einem einfachen Styropor-Teller ersetzt. Der Styropor-Teller schwimmt auf der Wasseroberfläche und sorgt für eine "saubere" Reflexion des IR-Strahls.

Der IR-Sensor IRDMS wird in einem wasserdichten Gehäuse mit einem Acrylglas-Fensterchen untergebracht, so dass spannungsführende Teile nicht mehr mit dem Wasser in Berührung kommen können. Das Filter zur Weiterleitung der Messspannung $\mathbf{U_{mess}}$ an den AVR-Mikrocontroller ist ebenfall in dem Gehäuse untergebracht (siehe IRDMS - Messkopf - Schaltung).

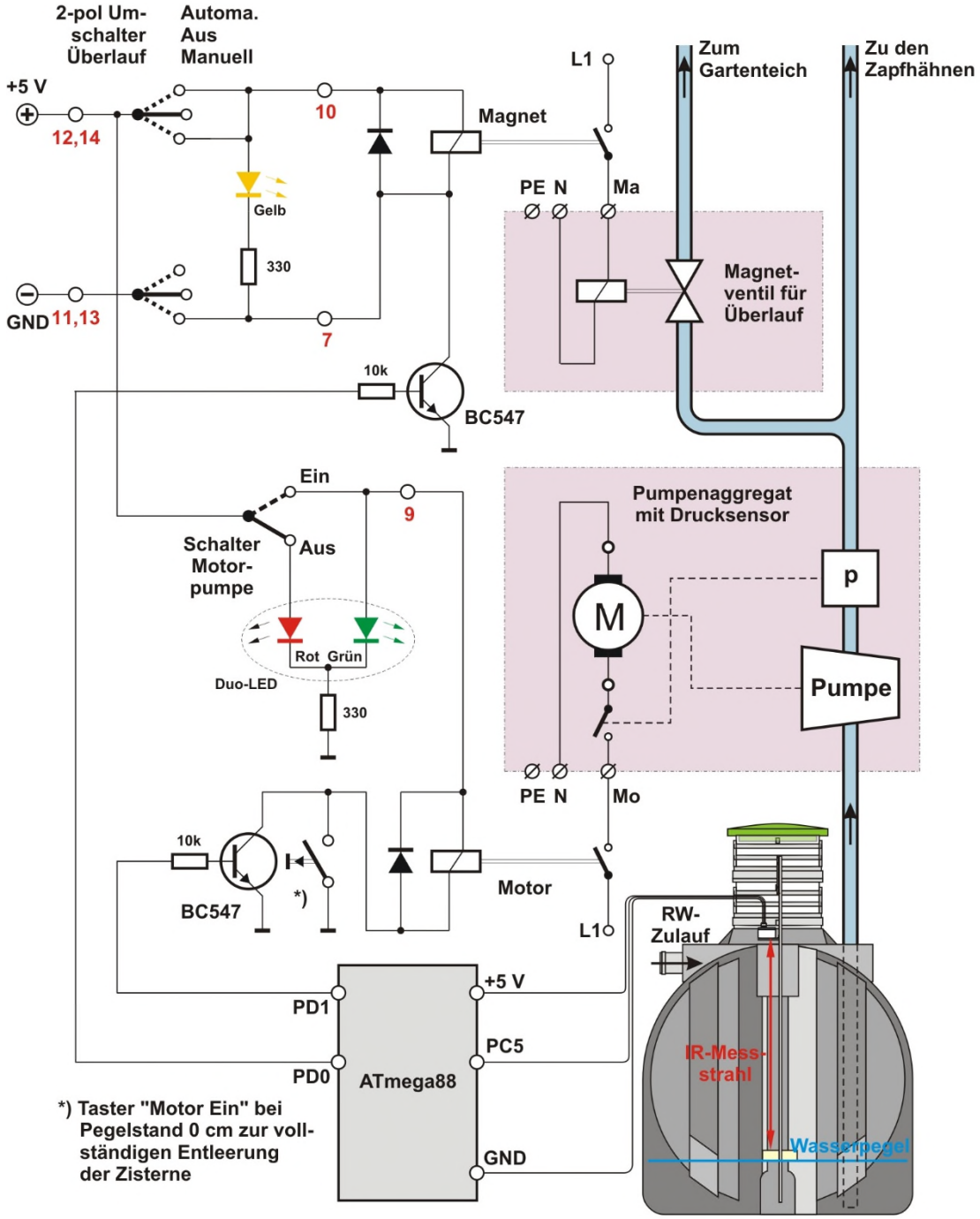


Bild 3.3.1-03: Leerlaufschutz für die Motorpumpe und Überlaufsteuerung für die Zisterne
(Bildvergrößerung)

Simulation der RWNA

Um die Software auch "im Trocknen", d.h. ohne IRDMS und Starkstromtechnik, simulieren zu können, wurde ein kleiner Adapter (IRDMS-Simulation) entworfen. Dieser stellt in der nachfolgenden Beschaltung die veränderliche Spannung **U_{mess}/mV** im Bereich von **110** bis **1030 mV** zur Verfügung.

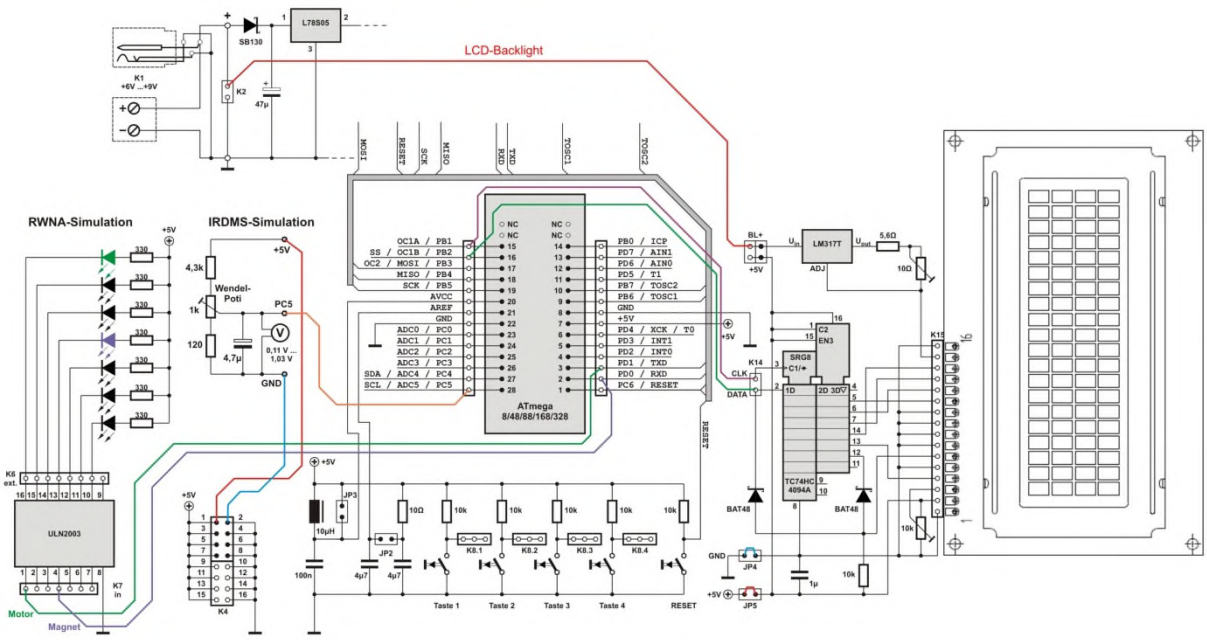


Bild 3.3.1-04: Beschaltung zur Simulation der RWNA (Bildvergrößerung)

Das Magnet-Relais und das Relais für die Motor-Pumpe werden dabei durch zwei LEDs simuliert:

- **PD0** (violett) simuliert das Magnet-Relais
- **PD1** (grün) simuliert das Relais für die Motor-Pumpe

Durch Verändern der Spannung am Wendel-Poti werden die zugehörigen Pegel-Schaltpunkte auf dem Display abgelesen.

3.3.2 Leerlaufschutz für die Motorpumpe

Der Mikrocontroller-Ausgang **PD1** steuert über den Transistor **BC547** das Motor-Relais bei Pegel-Minimum auf **AUS**, bevor die Pumpe über das Ansaugsieb Luft ansaugt und die Abschaltautomatik des Pumpenaggregats anspricht. Dieser Zusatz ist hilfreich, da sich die Abschaltautomatik des Pumpenaggregats nicht selbsttätig wieder aktiviert. Das Aggregat muss in diesem Fall manuell wieder eingeschaltet werden. Das Pegel-Minimum wird durch die Abmessungen des Ansaugsiebes in der Zisterne bestimmt und beträgt etwa 20 cm.

Im Regelfall liegt am Ausgang **PD1** ein **HIGH**-Signal an, so dass der Schalt-Transistor durchgeschaltet und das Motor-Relais angezogen ist (es muss natürlich auch der Schalter **Motorpumpe** auf **EIN** stehen). Das Pumpenaggregat liegt dann unmittelbar an der Netzspannung von 230 V, so dass die Motorpumpe losläuft, sobald der Drucksensor eine Druckminderung im Rohrsystem feststellt, entweder dadurch, dass ein Zapfhahn geöffnet wird oder dass das Magnetventil anspricht. Wenn der Pegel auf gemessene **0 cm** abgesunken ist, schaltet **PD1** auf **LOW**, der Transistor wird gesperrt, das Motor-Relais fällt ab und das Pumpenaggregat ist stromlos. Es wird erst dann wieder automatisch eingeschaltet, wenn der Pegel deutlich über **8 cm** angestiegen ist. Die Motorpumpe läuft dann kurz an und erneuert den Druck im Rohrsystem.

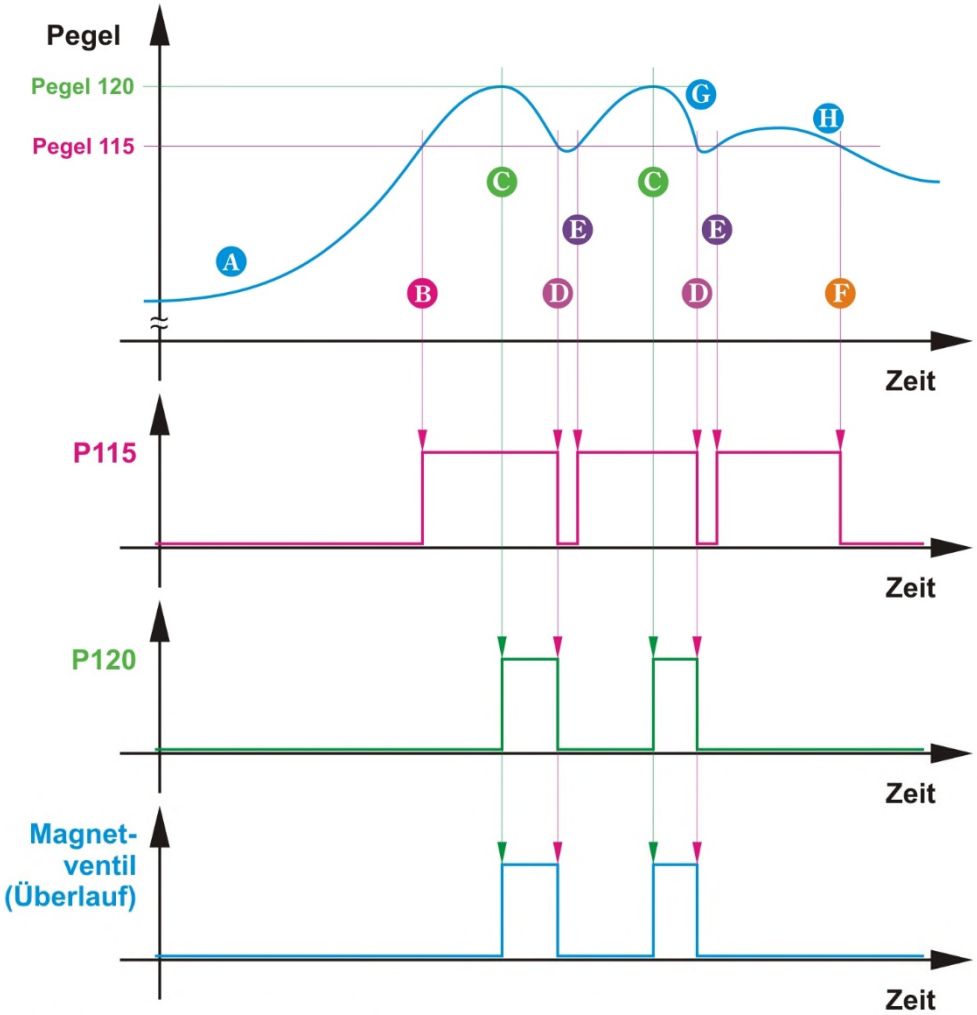
Um für die Winterpause die Zisterne nahezu vollständig zu entleeren - obgleich **PD1** auf **LOW** steht - wurde ein Tastschalter (Schließkontakt) parallel zum Transistorschalter ergänzt. Die Betätigung des Tasters versorgt das Pumpenaggregat wieder mit 230 Volt.

3.3.3 Überlaufsteuerung

Wenn starker Regen einsetzt, wird die sehr wichtige Funktion der RWNA in Gang gesetzt: Der Überlauf von der Zisterne in den Gartenteich. Gesteuert wird das Magnetventil über den Ausgang **PD0** und dem Schalt-Transistor **BC547** im Relais-Kreis des Magneten.

PD0 ist **HIGH** wenn der Pegel **120 cm oder mehr** erreicht hat und schaltet über den Transistor das Relais **Magnet** ein.

Sinkt der Pegel auf **unter 115 cm** wird **PD0** wieder auf **LOW** gesteuert und der Transistor schaltet das Magnetventil ab. So wird verhindert, dass die Zisterne vollständig entleert wird.



A Starker Regen setzt ein und die Zisterne wird kontinuierlich gefüllt, der Pegel steigt.

B Der Pegelwert 115 cm wird erreicht, der Pegel steigt weiter.

C Der Pegelwert 120 cm wird erreicht (P120 = TRUE), das Magnetventil öffnet.

D Der Pegel sinkt unter 115 cm (P120 = FALSE), das Magnetventil schließt.

E Der Pegel steigt wieder und erreicht erneut den Pegelwert 115 cm.

F Der Pegel erreicht keine 120 cm mehr, sondern sinkt wieder unter 115 cm.

G Der Regen lässt langsam nach, so dass der Überlauf stärker auf die Entleerung wirkt.

H Normale Wasserentnahme für die Gartenbewässerung.

Bild 3.3.3-01: Diagramm der Überlaufsteuerung (Bildvergrößerung)

3.3.4 Programmierung der Anwendung "RWNA"

Der Zeitpunkt ist gekommen und es geht zur Programmierung, d.h. zur programmtechnischen Umsetzung der Logik und Steuerung. Dazu werden zunächst einige **Variablen** benannt und **in Kurzform** in einem Ablaufdiagramm logisch miteinander verknüpft. Die Kurzform wurde nur für die Darstellung der Abläufe gewählt. Die übliche ausführlichere Bezeichnung der Variablen in Kleinschrift wird im Programm verwendet und ist hier nur der Vollständigkeit halber mit aufgeführt:

- **i** und **j** `i, j` Zähler
- **VOL** `vol` Logischer Schalter für **VOL**umenminderung Verbrauch/Überlauf
- **P120** `p_120` Logischer Schalter für **P**egelwert **120** cm
- **AE** `adc_einzel` ADC-Einzelmesswert vom IRDMS
- **AM** `adc_mittel` ADC-Mittelwert gemittelt aus 32 ADC-Einzelmesswerten
- **NL** `n_lesen` Neuer ADC-Lesewert gemittelt aus 8 aktuellen ADC-Mittelwerten
- **W1** `w1_tab` Unterer Interpolationswert aus der ADC-**W**erte-Tabelle
- **W2** `w2_tab` Oberer Interpolationswert aus der ADC-**W**erte-Tabelle
- **W[i]** `w[i]` ADC-Tabellenwert aus der ADC-**W**erte-Tabelle mit 25 abgestuften ADC-Werten (i = 0 bis 24)
- **P1** `p1_tab` Unterer Interpolationswert aus der **P**egel-Tabelle
- **P2** `p2_tab` Oberer Interpolationswert aus der **P**egel-Tabelle
- **P[i]** `p[i]` **P**egel-Tabellenwert aus der **P**egel-Tabelle mit 25 abgestuften Pegelwerten (i = 0 bis 24)
- **V1** `v1_tab` Unterer Interpolationswert aus der **V**olumen-Tabelle
- **V2** `v2_tab` Oberer Interpolationswert aus der **V**olumen-Tabelle
- **V[i]** `v[i]` **V**olumen-Tabellenwert aus der **V**olumen-Tabelle mit 25 abgestuften Volumenwerten (i = 0 bis 24)
- **pro** `prop_fak` Proportionalitätsfaktor zwischen den Tabellen
- **NP** `n_pegel` **N**euer Mittelwert in **P**egel-cm (aus der Pegel-Tabelle)
- **AP** `a_pegel` **A**lter Mittelwert in **P**egel-cm (**A**nzeige-**P**egelwert)
- **NV** `n_volumen` **N**euer Mittelwert in **V**olumen-m^3 (aus der Volumen-Tabelle)
- **AV** `a_volumen` **A**lter Mittelwert in **V**olumen-m^3 (**A**nzeige **V**olumenwert)
- **DV** `d_volumen` **D**ifferenz zwischen dem alten und dem neuen **V**olumenwert (in m^3)
- **TV** `t_volumen` Summe **T**eich-Überlauf in **V**olumen-m^3
- **GV** `g_volumen` Summe **G**artenwasser-Verbrauch in **V**olumen-m^3

Dreh- und Angelpunkt ist eine **Werte-Tabelle**, die jeweils den Pegel **P** in **cm**, den Füllstand **V** in **m^3** und den **ADC-Wert** (ohne Dimension) eindeutig einander zuordnet. Nach der Fertigstellung des Mess-Kopfes und der Aufhängevorrichtung werden die Werte der Mess-Spannung und der IRDMS-Spannung durch Verschieben des Styropor-Tellers auf verschiedene Pegelwerte **P** (bzw. Abstands-Werte **E**) gemessen.

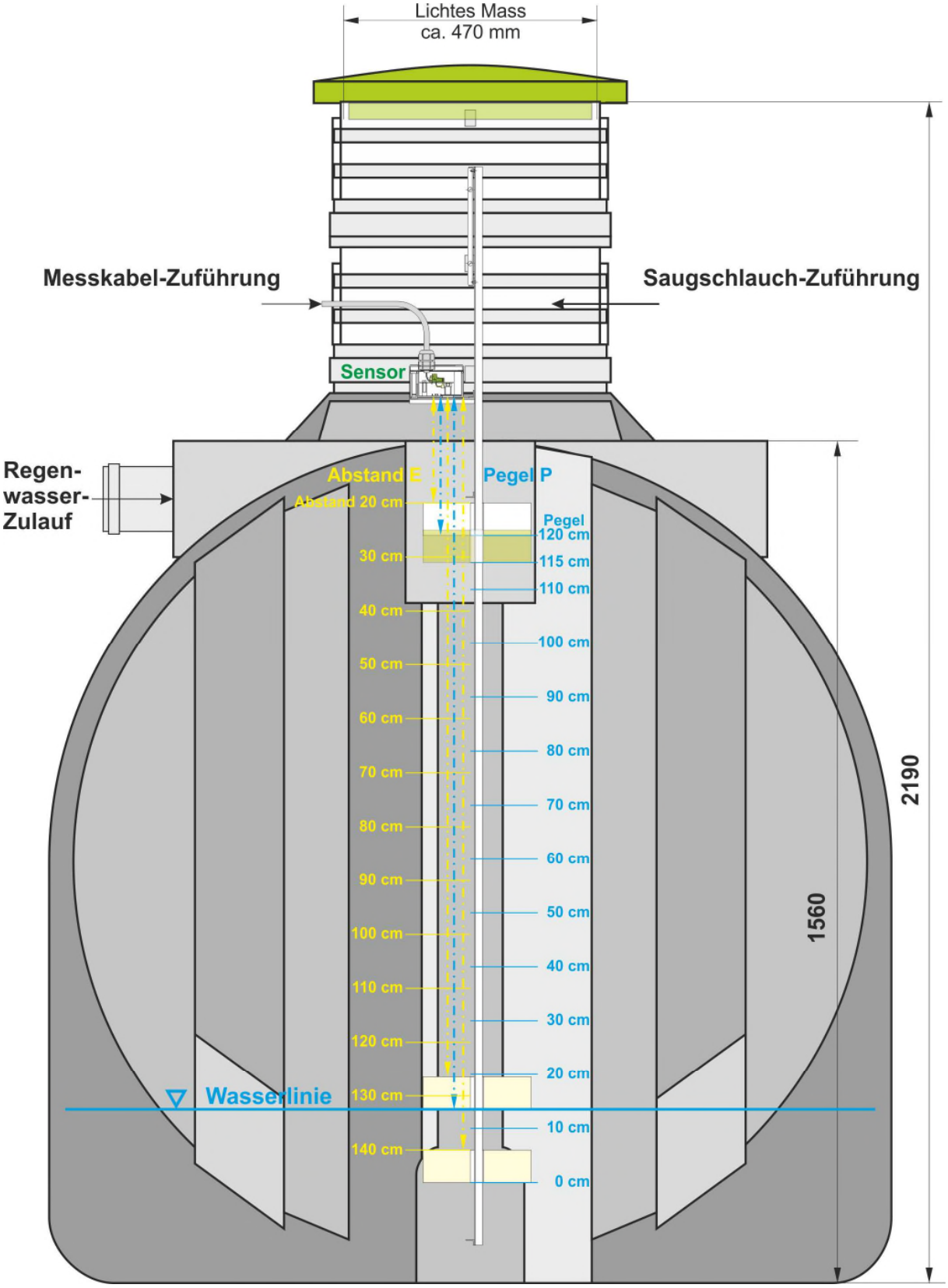


Bild 3.3.4-01: Anordnung und Dimensionierung des Messstabes (Bildvergrößerung)

Erst direkt am "Objekt" werden die Füllstände in Abhängigkeit der Mess-Spannungen ermittelt. Dafür muss die Zisterne zunächst leer gepumpt werden. Dann wird Zug um Zug die Zisterne mit Leitungswasser gefüllt, wobei jeweils die Volumen-Menge am Garten-Wasserzähler abgelesen und der Mess-Spannung zugeordnet wird.

Tabelle 3.3.4-01: Werte-Tabelle (Werte am Objekt in 5cm-Abständen ermittelt)

Index **i**	Abstand **E** in **cm**	Pegel **p[i]** in **cm**	Füllstand **v[i]** in **m³**	IRDMS **U_{IR}/mV**	Mess-Spannung **U_{mess}/mV**	**ADC-Wert w[i]**
0	140	0	0,000	458	188	177
1	135	5	0,134	476	195	180
2	130	10	0,272	496	203	190
3	125	15	0,431	515	211	196
4	120	20	0,585	553	227	213
5	115	25	0,748	571	233	216
6	110	30	0,908	590	242	228
7	105	35	1,069	610	250	236
8	100	40	1,232	629	257	244
9	95	45	1,397	667	273	258
10	90	50	1,560	708	289	273
11	85	55	1,723	758	314	294
12	80	60	1,887	811	331	310
13	75	65	2,054	838	343	322
14	70	70	2,216	890	363	346
15	65	75	2,375	972	397	374
16	60	80	2,540	1039	424	397
17	55	85	2,699	1075	439	412
18	50	90	2,843	1198	490	462
19	45	95	2,987	1364	556	525
20	40	100	3,127	1456	593	561
21	35	105	3,259	1660	677	640
22	30	110	3,375	1877	765	724
23	25	115	3,481	2169	883	837
24	20	120	3,581	2385	971	924

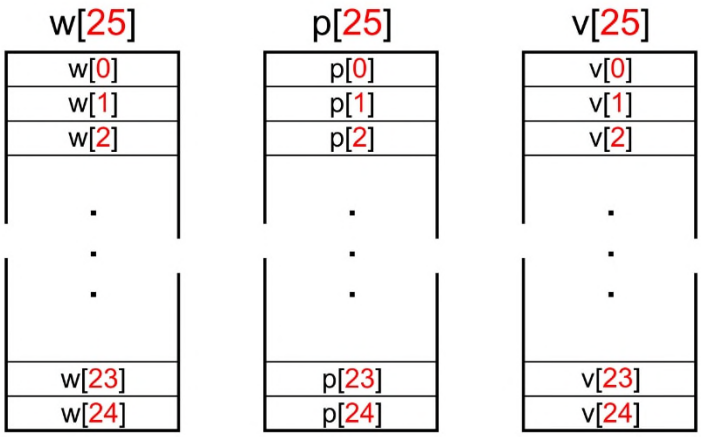


Tabelle 3.3.4-02: Tabellen-Arrays

Programmtechnisch werden die Werte auf 3 Arrays aufgeteilt, ohne ihre Abhängigkeit voneinander aufzulösen. Dadurch ist es leichter und überschaubarer zwischen den Werten zu interpolieren (siehe weiter unten):

w[25] ist die Werte-Tabelle der 25 **ADC-Messwerte** für die Abstände **E** von 140 bis 20 cm in Abständen von 5 cm.

p[25] ist die Pegel-Tabelle der 25 **Pegelwerte P** von 0 bis 120 cm analog zu den Abstandswerten **E**.

v[25] ist die Volumen-Tabelle der 25 **Volumenwerte** für die Füllstände **V**, die auf Grund der manuellen Füllung mit Leitungswasser am Gartenwasser-Zähler abgelesen und eingesetzt wurden.

3.3.4.1 Das Hauptprogramm

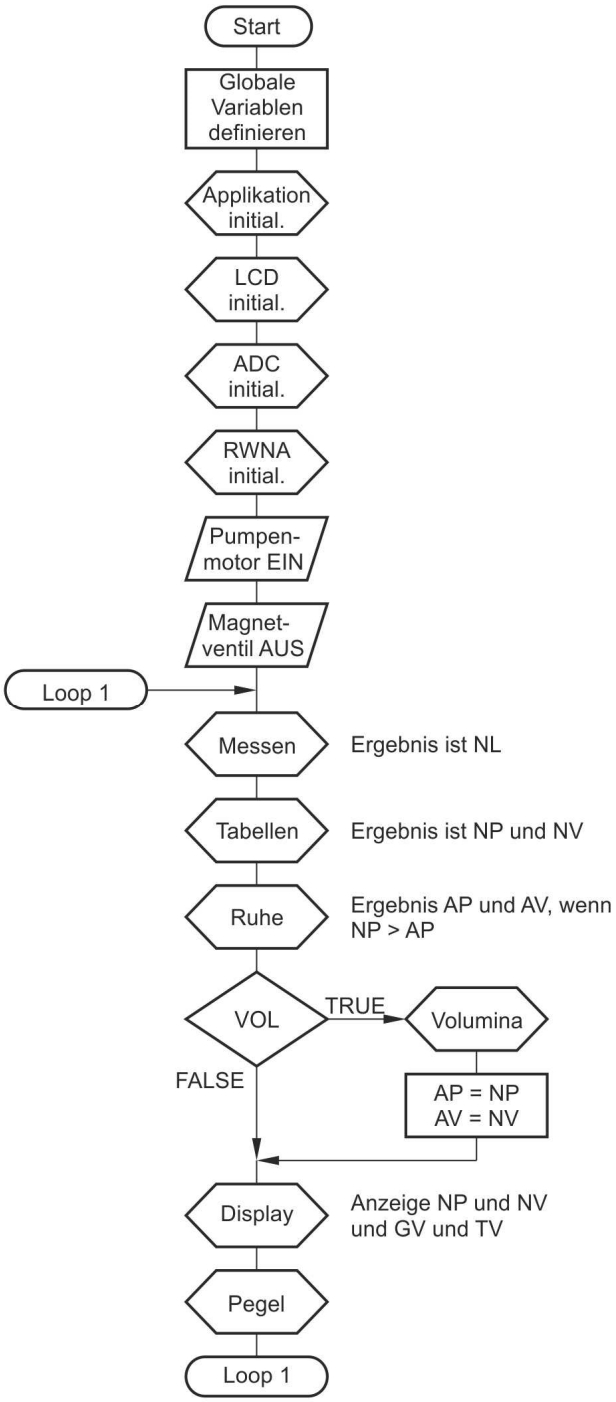


Bild 3.3.4.1-01: Start

Beim Einschalten bzw. **RESET** werden die **Variablen global definiert**, die **Applikation**, das **LCD**, das **ADC** (wie oben beschrieben) und die **RWNA** werden **initialisiert**.

Durch die Initialisierung der **RWNA** werden der Pegel **NP** gleich **AP** sowie das Volumen **NV** gleich **AV** vorbesetzt.

Das **Pumpenaggregat** wird auf **EIN** und das **Magnetventil** wird auf **AUS** gestellt.

Der Vorgang läuft in einer Endlosschleife **Loop 1** bis zum nächsten **RESET** (oder nach einem Netzausfall).

Genereller Ablauf (Loop 1):

Es wird ein neuer ADC-Messwert **NL** erfasst.

Mit dem neuen ADC-Messwert **NL** wird gem. Tabelle ein neuer Pegelwert **NP** und ein neuer Volumenwert **NV** ermittelt.

Der logische Schalter **VOL** wird nur gesetzt, wenn der neue Messwert **NP** zur Ruhe gekommen ist und wegen der Pegelschwankungen genügend kleiner als der alte vorherige Wert **AP** ist.

Dann werden die **Volumina** für den Gartenwasser-**Verbrauch** bzw. für den **Überlauf** in den Gartenteich ermittelt und auf dem **Display** geändert zur Anzeige gebracht.

Andernfalls werden nur der Anzeige-Pegelwert **AP** gleich **NP** sowie der Anzeige-Volumenwert **AV** gleich **NV** auf dem **Display** geändert zur Anzeige gebracht.

Pegel-Abfrage, ob ein Überlauf eingeleitet werden muss und ggf. Setzen des logischen Schalters **P120** sowie Überwachung des Leerlaufschutzes für das Pumpenaggregat.

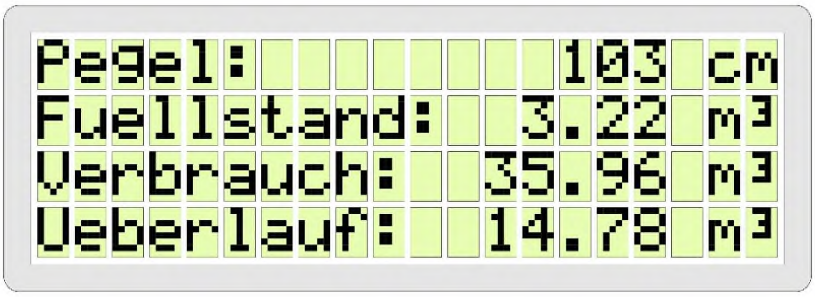


Bild 3.3.4.1-02: Anzeige der Pegel- und Verbrauchswerte

3.3.4.2 Messwert-Erfassung

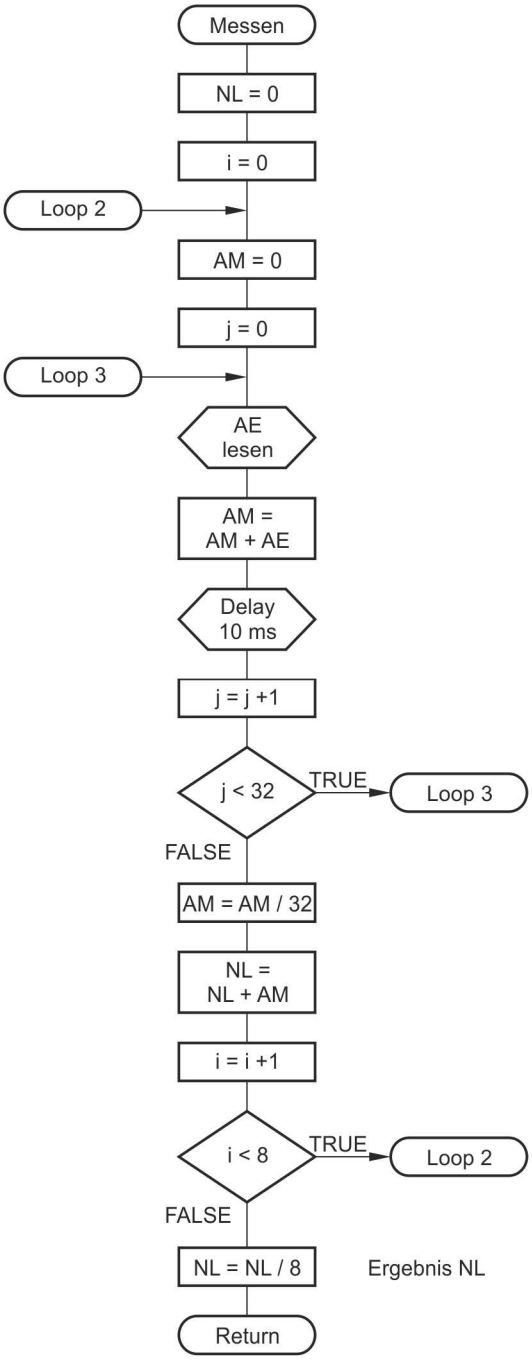


Bild 3.3.4.2-01: Messen

Loop2 (äußere Schleife)

Loop3 (innere Schleife)

Es hat sich bereits beim Messen der Charakteristik herausgestellt, dass auch ohne "Schwell" in der Wasseroberfläche die Messwerte sehr "unruhig" sind, d.h. sie schwanken auch bei feststehendem Pegel um ein paar mV. Auch der Sieb-Kondensator von 47 :F am Eingang des Mikrocontrollers kann diese Schwankungen nicht aufheben.

Es werden deshalb zunächst **32** Einzelmessungen **M** (mit einer empirisch ermittelten Verzögerung von **10 ms**) vorgenommen.

Aus ihnen wird ein Mittelwert **W** errechnet.

...

...

...

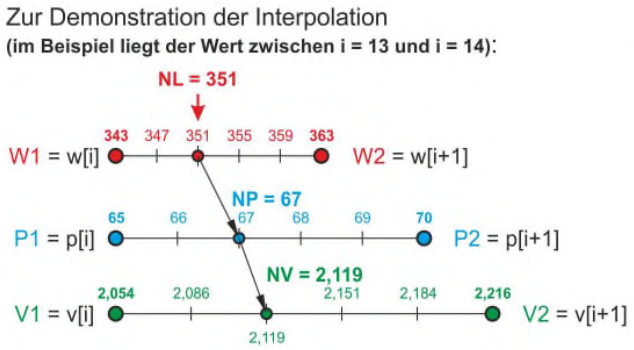
Im **Loop 2** werden nacheinander 8 solcher Mittelwerte aufaddiert, um schließlich den neuen Mittelwert **NL** aus 256 aktuellen Messwerten zu ermitteln.

Mit diesem ADC-Mittelwert **NL** wird in die Tabelle eingestiegen.

3.3.4.3 Interpolation aus einer Messwerte-Tabelle

Aus den oben beschriebenen Tabellen werden die neuen Mittelwerte für den Pegel und das Volumen durch Interpolation ermittelt. Die Interpolation sieht nur auf den ersten Blick sehr aufwendig aus, ist aber für alle möglichen linearen Interpolationen anwendbar.

Nimmt man an, dass der **NL**-Wert aus der Messung **351** ist, so ergeben sich aus den Intervallen **W1** (343) bis **W2** (363), **P1** (65) bis **P2** (70) und **V1** (2,054) bis **V2** (2,216) auf sehr anschauliche Weise die zugehörigen Werte für **NP** (67) und **NV** (2,119).



Zu einem bestimmten Messwert NL stehen die zugehörigen Pegel- und Volumenwerte in fester Proportion (Strecken-Verhältnisse):

```
pro =  (NL-W1)/(W2-W1)
    =  (NP-P1)/(P2-P1)
    =  (NV-V1)/(V2-V1)
```

Daraus folgt wenn man `pro = (NL-W1)/(W2-W1)` einsetzt:

```
NP  = P1 + pro*(P2-P1)
NV  = V1 + pro*(V2-V1)
```

Man beachte im Coding die unterschiedlichen Typen Integer und Floatingpoint !!!

Bild 3.3.4.3-01: Interpolation

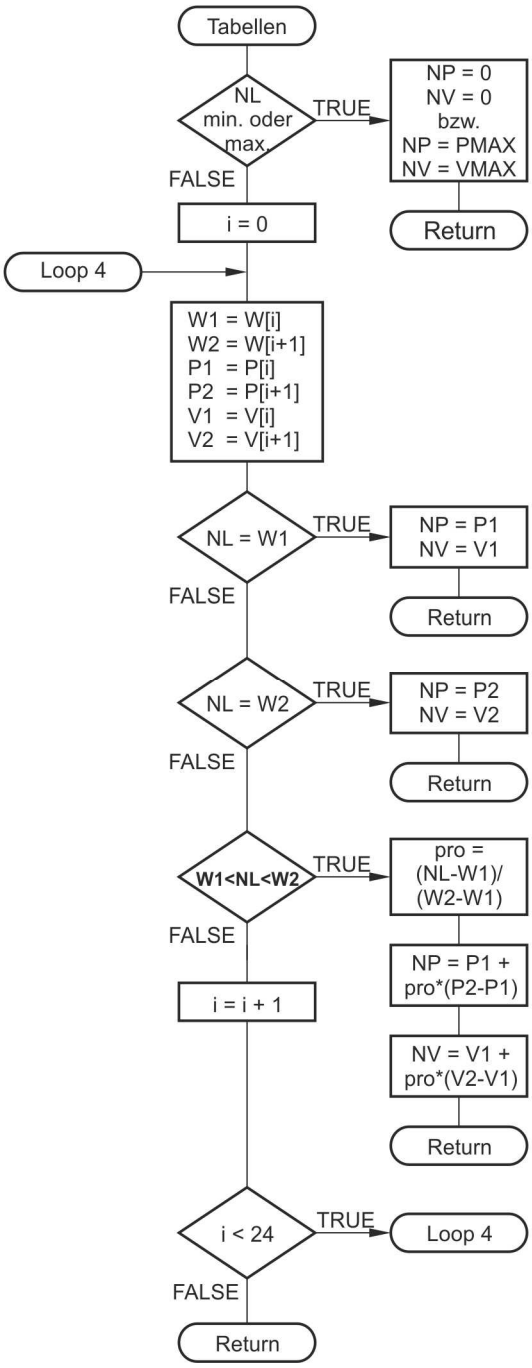


Bild 3.3.4.3-02: Tabellen

Zunächst wird ermittelt, ob der Messwert außerhalb der Tabelle liegt. Wenn das der Fall ist, werden Null-Werte bzw. die maximalen Werte unterstellt.

Loop 4

Es werden der Reihe nach alle Tabellenwerte "zeilenweise" eingelesen.

Es werden immer gleichzeitig die Werte aus 2 "Zeilen" (**i** und **i+1**) übernommen und ausgewertet; das sind die jeweils benachbarten Tabellenwerte.

Wird Gleichheit mit dem unteren **oder** oberen Tabellenwert festgestellt, so sind bereits die gesuchten Werte für **NP** und **NV** gefunden und die Funktion kann verlassen werden.

Hier: oberer Wert

Ist das nicht der Fall, dann kann der Wert **NL** entweder zwischen dem jeweils unteren und oberen Tabellenwert liegen und es ist die Interpolation durchzuführen oder er liegt noch oberhalb der betrachteten zwei "Tabellen-Zeilen" (also oberhalb des Wertes mit dem Index **i+1**) und die Schleife ist erneut zu durchlaufen:

i wird um 1 erhöht.

Solange **i < 24** ist, wird die Schleife fortgesetzt.

3.3.4.4 "Beruhigung" des Messwertes

Diese Maßnahme wird ergriffen, damit messtechnisch bedingte Schwankungen (auf Grund der sehr flachen Charakteristik im unteren Pegelbereich), die trotz der beschriebenen Mittelwert-Bildung und Verzögerung entstehen, sich nicht auf den Verbrauch bzw. Überlauf auswirken.

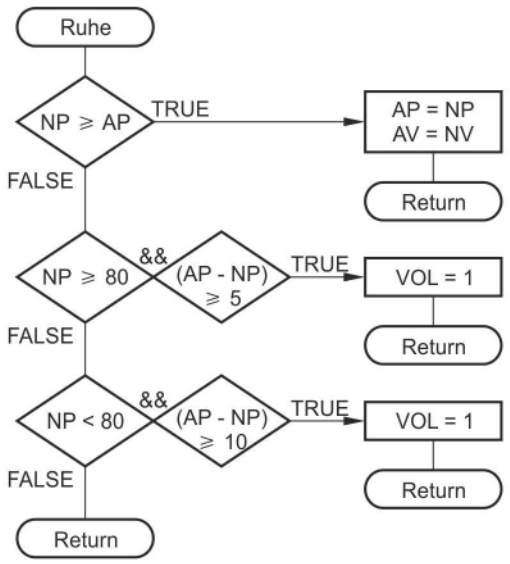


Bild 3.3.4.4-01: Ruhe

Als erstes wird der vorherige Anzeige-Pegelwert **AP** mit dem neuen Pegelwert **NP** verglichen. Wenn **NP** größer oder gleich **AP** ist, dann ist entweder Wasser ergänzt worden oder der Pegel ist gleich geblieben. Der jeweilige Stand wird registriert.

Andernfalls ist Wasser entnommen worden:

Bei Pegelwerten größer 80 cm treten nur geringe Schwankungen auf, so dass bereits ab einer Pegel-Absenkung von 5 cm der Verbrauch registriert wird (**VOL = 1**).

Bei Pegelwerten unter 80 cm treten bereits erhebliche Schwankungen auf, so dass erst ab einer Pegel-Absenkung von 10 cm der Verbrauch registriert wird (**VOL = 1**).

Andernfalls ist das aus der Zisterne entnommene Wasser noch nicht ausreichend, um registriert zu werden: **VOL** bleibt **FALSE** (**= 0**).

3.3.4.5 Ermitteln der Verbrauche

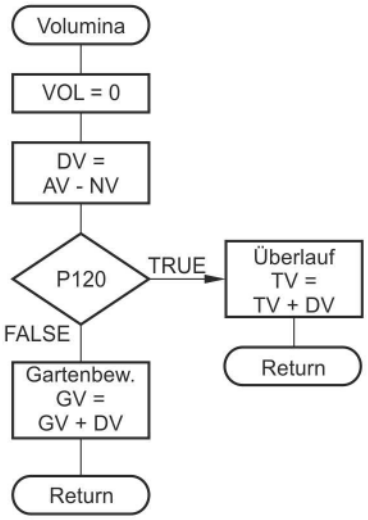


Bild 3.3.4.5-01: Volumina

Logischen Schalter für Volumenminderung **VOL** auf **FALSE** setzen (**VOL = 0**).

Bevor geprüft wird, ob die **Volumina** für den Gartenwasser-**Verbrauch** oder für den Teich-**Überlauf** erhöht werden müssen, wird zunächst die Differenz

DV = AV - NV (Volumenminderung)

zwischen dem alten Volumen (**AV**) und dem neuen Volumen (**NV**) festgestellt. Je nach Zustand des logischen Schalters **P120** wird der **Verbrauch** dem Gartenwasser oder dem **Überlauf** zugerechnet.

Anmerkung: Trotz vieler Experimente mit der "Beruhigung" sind die angezeigten Werte für den **Verbrauch** als Gartenwasser oder dem **Überlauf** in den Gartenteich sehr ungenau. Sie können lediglich nur als Näherungswerte betrachtet werden.

3.3.4.6 Steuerung des Überlaufs und der Leerlauf-Überwachung

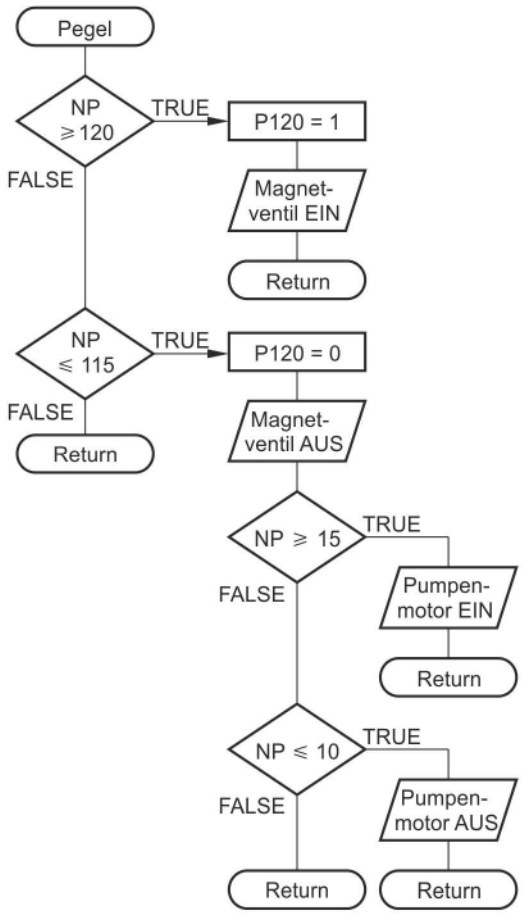


Bild 3.3.4.6-01: Pegel

Mit **NP** wird zunächst der Fall des Überlaufes geprüft. Wenn der Pegel **NP** größer oder gleich **120 cm** beträgt, dann wird der logische Schalter **P120** auf **TRUE** und das Magnetventil auf **EIN** gesetzt, d.h. das Magnetventil öffnet und es wird Wasser in den Gartenteich gepumpt.

Wenn dann der Pegel **NP** wieder auf **115 cm** gesunken ist, dann muss der logische Schalter wieder auf **FALSE** gesetzt und das Magnetventil wieder geschlossen werden (Magnetventil **AUS**).

(Erst) wenn der Pegel **NP** größer oder gleich **15 cm** ist, dann wird das Pumpenaggregat (wieder) auf **EIN** gestellt.

(Erst) wenn der Pegel **NP** nach dieser negierten Abfrage tatsächlich kleiner oder gleich **10 cm** erreicht, wird das Pumpenaggregat zum Leerlaufschutz auf **AUS** gestellt.

Die Differenz erklärt sich aus den größeren Schwankungen im unteren Messbereich. Ohne diese "Hysterese" würde die Motorpumpe im untersten Pegel-Bereich häufig ein- und ausgeschaltet werden (Überhitzungsgefahr!).

Der Programm-Quell-Code (C- und Header-Dateien) befindet sich im Abschnitt

603_Projekt_IRDMS.

Neue Messmethode zur Steuerung und Kontrolle der RWNA mittels Pegelmesser siehe neuestes Projekt

604_Projekt_Pegelsonde