ATmega64 T-Bird Summary

GPIOs

LEDs

Buttons

Seven-segment display

Matrix keyboard

Timers, IT, PWM

RGB LED

LCD

UART

ADC

RS485

A szakkollégium 2023-24 évi működését támogatja a Nemzeti Tehetség Program és a Kulturális és Innovációs Minisztérium, az Emberi Erőforrás Támogatáskezelő által kiírt „Szakkollégiumok tehetséggondozó programjainak támogatása” című pályázata (NTP-SZKOLL-23-0056).

KULTURÁLIS ÉS INNOVÁCIÓS MINISZTÉRIUM

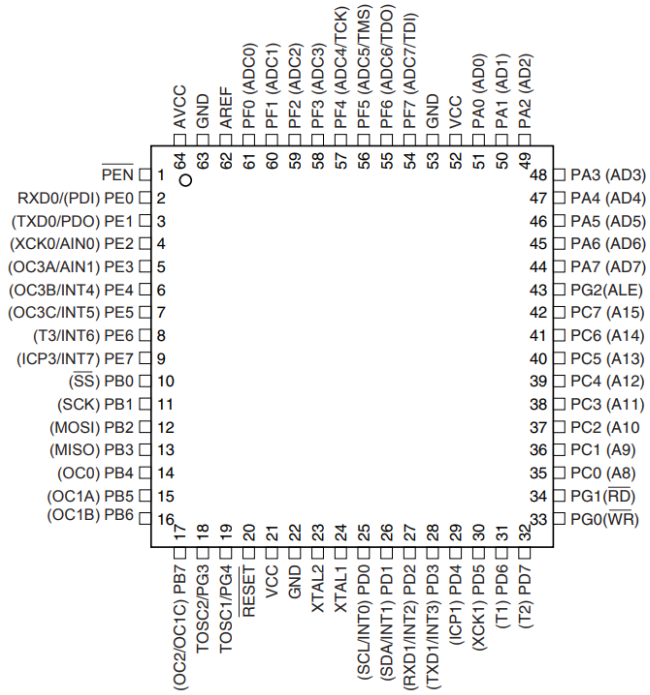
Nemzeti Tehetség Program

ATmega64

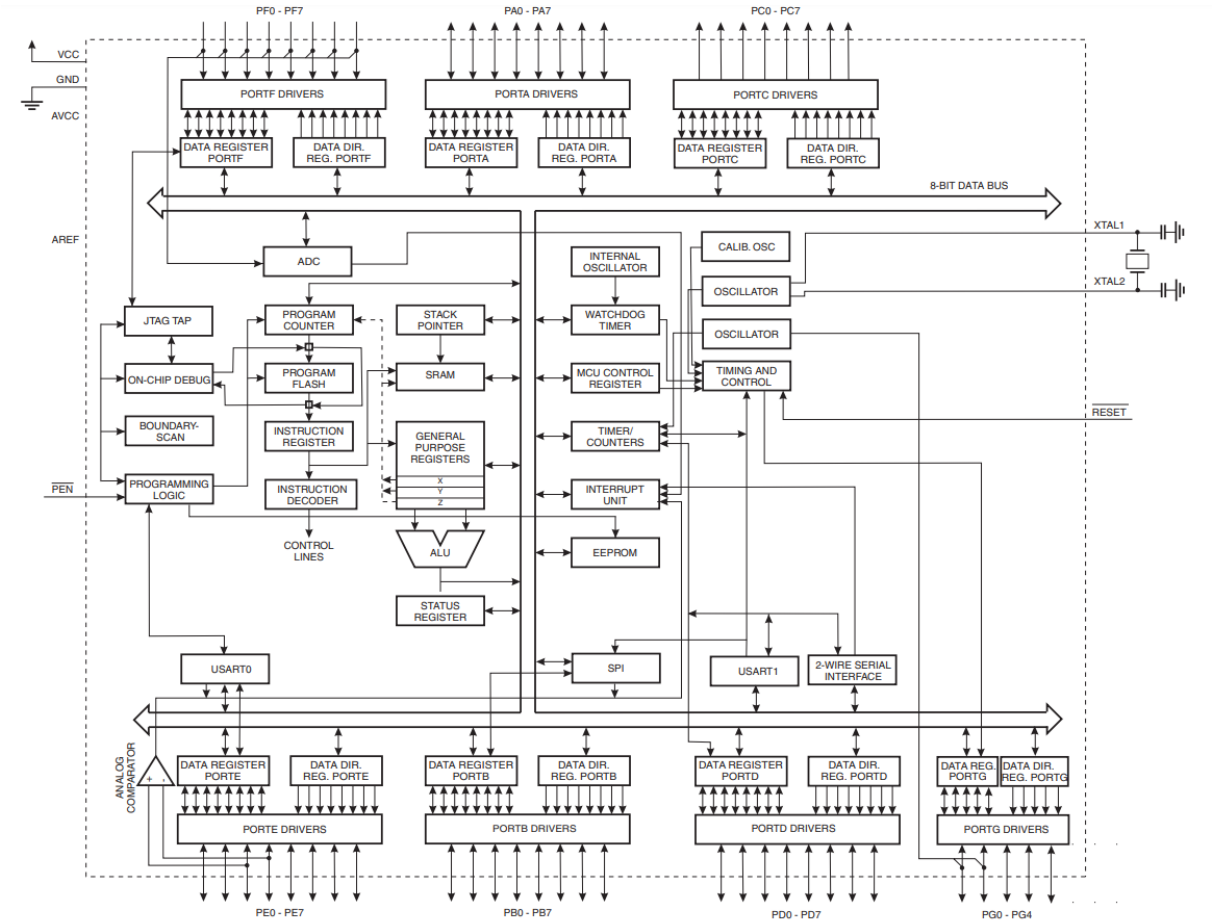
Features

- High-performance, Low-power Atmel AVR® 8-bit Microcontroller
- Advanced RISC Architecture
  - 130 Powerful Instructions – Most Single Clock Cycle Execution
  - 32 x 8 General Purpose Working Registers + Peripheral Control Registers
  - Fully Static Operation
  - Up to 16 MIPS Throughput at 16 MHz
  - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory segments
  - 64 Kbytes of In-System Reprogrammable Flash program memory
  - 2 Kbytes EEPROM
  - 4 Kbytes Internal SRAM
  - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
  - Data retention: 20 years at 85°C/100 years at 25°C(1)
  - Optional Boot Code Section with Independent Lock Bits
    In-System Programming by On-chip Boot Program
    True Read-While-Write Operation
  - Up to 64 Kbytes Optional External Memory Space
  - Programming Lock for Software Security
  - SPI Interface for In-System Programming
- JTAG (IEEE std. 1149.1 Compliant) Interface
  - Boundary-scan Capabilities According to the JTAG Standard
  - Extensive On-chip Debug Support
  - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface
- Peripheral Features
  - Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes
  - Two Expanded 16-bit Timer/Counters with Separate Prescaler, Compare Mode, and Capture Mode
  - Real Time Counter with Separate Oscillator
  - Two 8-bit PWM Channels
  - 6 PWM Channels with Programmable Resolution from 1 to 16 Bits
  - 8-channel, 10-bit ADC
    8 Single-ended Channels
    7 Differential Channels
    2 Differential Channels with Programmable Gain (1x, 10x, 200x)
  - Byte-oriented Two-wire Serial Interface
  - Dual Programmable Serial USARTs
  - Master/Slave SPI Serial Interface
  - Programmable Watchdog Timer with On-chip Oscillator
  - On-chip Analog Comparator
- Special Microcontroller Features
  - Power-on Reset and Programmable Brown-out Detection
  - Internal Calibrated RC Oscillator
  - External and Internal Interrupt Sources
  - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby
  - Software Selectable Clock Frequency
  - ATmega103 Compatibility Mode Selected by a Fuse
  - Global Pull-up Disable
- I/O and Packages
  - 53 Programmable I/O Lines
  - 64-lead TQFP and 64-pad QFN/MLF
- Operating Voltages
  - 2.7V - 5.5V for Atmel ATmega64L
  - 4.5V - 5.5V for Atmel ATmega64
- Speed Grades
  - 0 - 8 MHz for ATmega64L
  - 0 - 16 MHz for ATmega64

https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-2490-8-bit-AVR-Microcontroller-ATmega64-L_datasheet.pdf



1. Figure



2. Figure

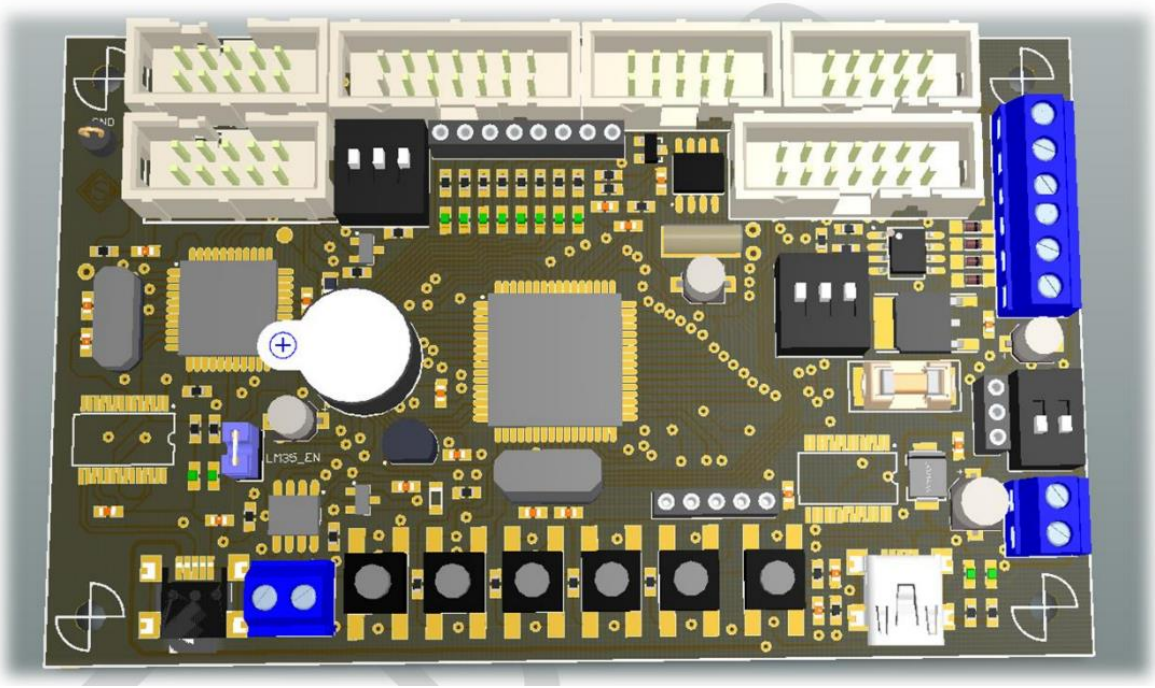
T-Bird 3

Development Board with ATmega128/64

3. Figure

Base T-Bird board

- ATmega128 or ATmega64 MCU
- 8 LEDs
- 5 buttons
- 16MHz or 8MHz CLK
- RTC
- USB-serial FTDI
- Piezo buzzer
- LM35 temperature sensor
- Bulit in JTAG debugger/programmer
- DAC
- Power over USB and external power option
- Fuse
- GPIO connectors



4. Figure

Extension Board

- LCD display
- Seven segment display
- Matrix keyboard
- RGB LED
- LM35 temperature sensor

5. Figure

PORTs

bit	7	6	5	4	3	2	1	0	
PORT									
A	**ENABLE**	**SEL2**	**SEL1**	**SEL0**	**DATA3**	**DATA2**	**DATA1**	**DATA0**	**7 SEGMENT DISPLAY**
I/O	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	
B	**Led3**	**Led2**	**Led1**	**Led0**					**LED/lo 4bit**
I/O	OUT	OUT	OUT	OUT					
C	**RED**	**KBD4row**	**KBD3row**	**KBD2row**	**KBD1row**	**KBD_right**	**KBD_centr**	**KBD_left**	**Keyboard**
I/O	OUT	OUT	OUT	OUT	OUT	IN	IN	IN	
D	**Led7**	**Led6**	**Led5**	**Led4**					**LED/hi 4bit**
I/O	OUT	OUT	OUT	OUT					
E	**LCD_DATA7**	**LCD_DATA6**	**LCD_DATA5**	**LCD_DATA4**	**GREEN**	**BLUE**			**LCD data**
I/O	OUT	OUT	OUT	OUT	OUT	OUT			
F					**LCD_E**	**LCD_R/W**	**LCD_RS**	**LM35**	**LCD Control**
I/O					OUT	OUT	OUT	IN	**Analog**
G	NC	NC	NC	**K4**	**K3**	**K2**	**K1**	**K0**	**Pushbutton**
I/O	X	X	X	IN	IN	IN	IN	IN	
bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	

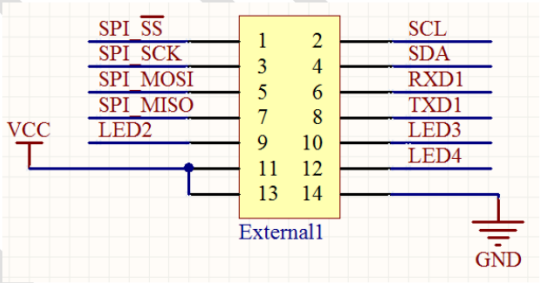
7 Segment display: E 1:Enable, 0: Disable; SEL 2-1-0: 000:10exp0, 001: 10exp1, 010:10exp2, 011:10exp3, 100:Double Leds

Led0…7: 0: Disable, 1:Enable; RED,GREEN,BLUE: 0 Disable, 1: Enable; KBDrow: 1: Select; Column: 0: Pressed

LCD_E: 0 Select, LCD_R/W: 1:Read, 0: Write; LCD_RS: 1: Data, 0: Control

6. Figure

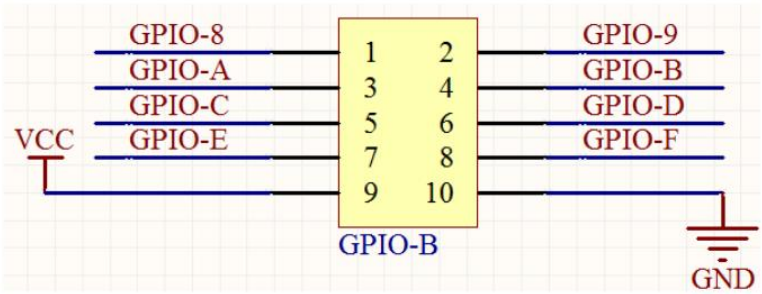
GPIO-S:



SPI_SS	- PB0	SCL	- PD0
SPI_SCK	- PB1	SDA	- PD1
SPI_MOSI	- PB2	RXD1	- PD2
SPI_MISO	- PB3	TXD1	- PD3
LED2	- PB6	LED3	- PB7
		LED4	- PD4

7. Figure

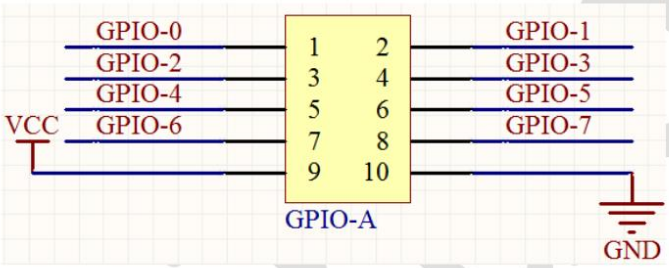
GPIO-B:



GPIO-8	- PC0	GPIO-C	- PC4
GPIO-9	- PC1	GPIO-D	- PC5
GPIO-A	- PC2	GPIO-E	- PC6
GPIO-B	- PC3	GPIO-F	- PC7

8. Figure

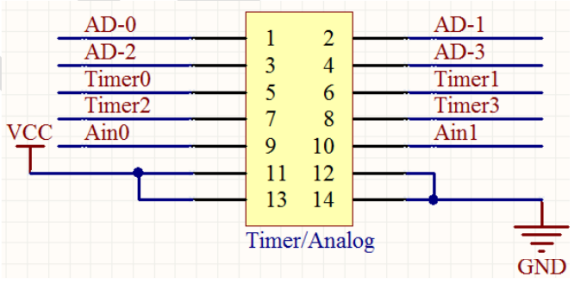
GPIO-A:



GPIO-0	- PA0	GPIO-4	- PA4
GPIO-1	- PA1	GPIO-5	- PA5
GPIO-2	- PA2	GPIO-6	- PA6
GPIO-3	- PA3	GPIO-7	- PA7

9. Figure

Timer/Analog:



AD-0	- PF0	AD-1	- PF1
AD-2	- PF2	AD-3	- PF3
Timer0	- PE4	Timer1	- PE5
Timer2	- PE6	Timer3	- PE7
Ain0	- PE2	Ain1	- PE3

10. Figure

ATmega64

https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-2490-8-bit-AVR-Microcontroller-ATmega64-L_datasheet.pdf

PORTs

bit	7	6	5	4	3	2	1	0	
PORT									
A	**ENABLE**	**SEL2**	**SEL1**	**SEL0**	**DATA3**	**DATA2**	**DATA1**	**DATA0**	**7 SEGMENT DISPLAY**
I/O	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	
B	**Led3**	**Led2**	**Led1**	**Led0**					**LED/lo 4bit**
I/O	OUT	OUT	OUT	OUT					
C	**RED**	**KBD4row**	**KBD3row**	**KBD2row**	**KBD1row**	**KBD_right**	**KBD_centr**	**KBD_left**	**Keyboard**
I/O	OUT	OUT	OUT	OUT	OUT	IN	IN	IN	
D	**Led7**	**Led6**	**Led5**	**Led4**					**LED/hi 4bit**
I/O	OUT	OUT	OUT	OUT					
E	**LCD_DATA7**	**LCD_DATA6**	**LCD_DATA5**	**LCD_DATA4**	**GREEN**	**BLUE**			**LCD data**
I/O	OUT	OUT	OUT	OUT	OUT	OUT			
F					**LCD_E**	**LCD_R/W**	**LCD_RS**	**LM35**	**LCD Control**
I/O					OUT	OUT	OUT	IN	**Analog**
G	NC	NC	NC	**K4**	**K3**	**K2**	**K1**	**K0**	**Pushbutton**
I/O	X	X	X	IN	IN	IN	IN	IN	
bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	

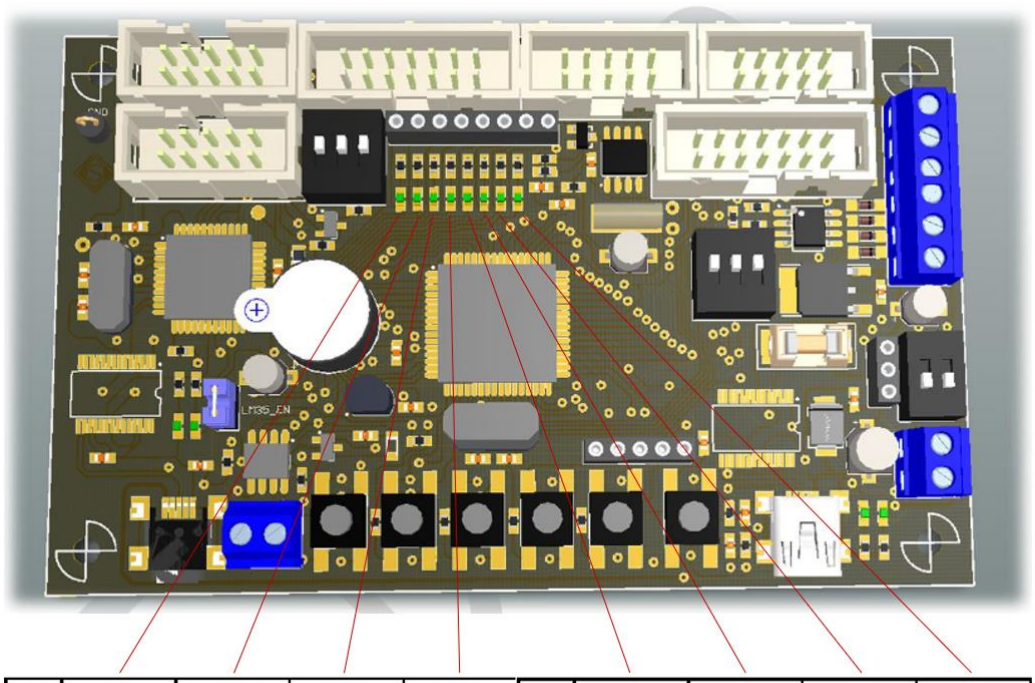
7 Segment display: E 1:Enable, 0: Disable; SEL 2-1-0: 000:10exp0, 001: 10exp1, 010:10exp2, 011:10exp3, 100:Double Leds

Led0…7: 0: Disable, 1:Enable; RED,GREEN,BLUE: 0 Disable, 1: Enable; KBDrow: 1: Select; Column: 0: Pressed

LCD_E: 0 Select, LCD_R/W: 1:Read, 0: Write; LCD_RS: 1: Data, 0: Control

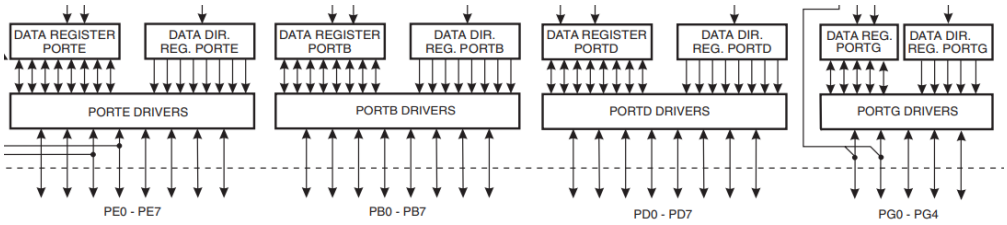
11. Figure

GPIO - LEDs



D	Led7	Led6	Led5	Led4	B	Led3	Led2	Led1	Led0
I/O	OUT	OUT	OUT	OUT	I/O	OUT	OUT	OUT	OUT
bit	7	6	5	4	bit	7	6	5	4

12. Figure



13. Figure

Port B Data Register – PORTB

Bit	7	6	5	4	3	2	1	0	
	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	PORTB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Port B Data Direction Register – DDRB

Bit	7	6	5	4	3	2	1	0	
	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	DDRB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

14. Figure

Port D Data Direction Register – DDRD

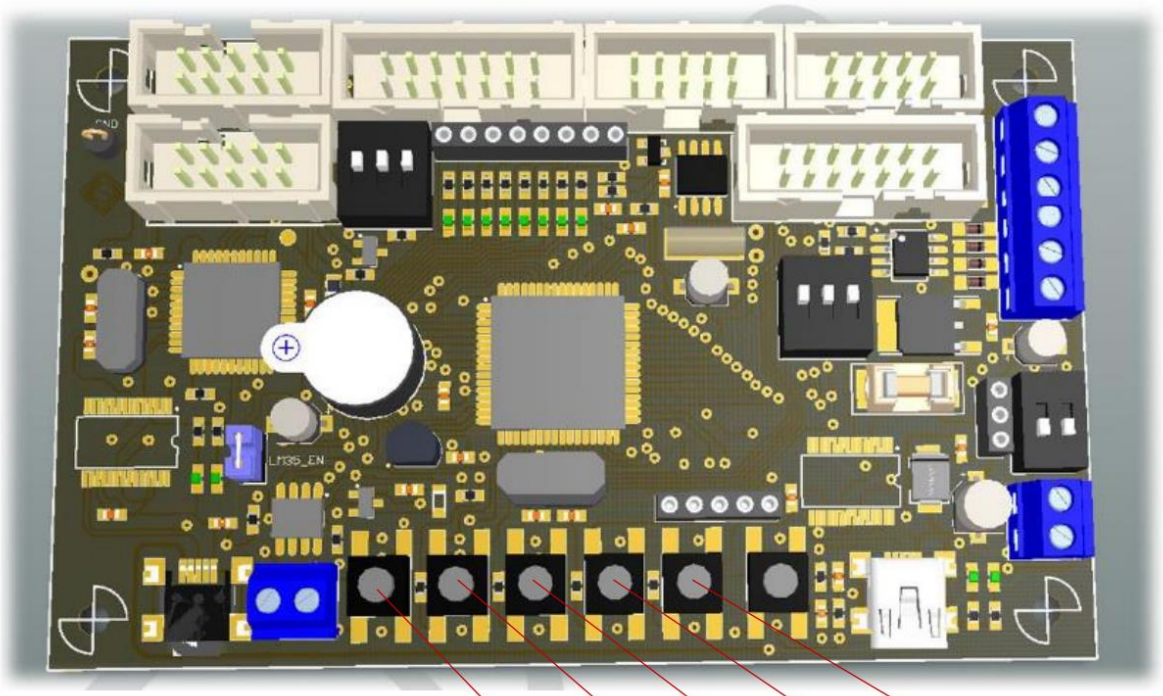
Bit	7	6	5	4	3	2	1	0	
	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	DDRD
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Port D Input Pins Address – PIND

Bit	7	6	5	4	3	2	1	0	
	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	PIND
Read/Write	R	R	R	R	R	R	R	R	
Initial Value	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	

15. Figure

GPIO – PUSH BUTTONs



G	NC	NC	NC	K4	K3	K2	K1	K0	Pushbutton
I/O	X	X	X	IN	IN	IN	IN	IN	
bit	7	6	5	4	3	2	1	0	

16. Figure

DDRn – Data Direction Register

Set a PIN to INPUT (0) or OUTPUT (1)

Readable and writable.

PORTn – PORT Write Register

Set a PIN to LOW (0) or HIGH (1)

Readable and writable.

PINn – PORT Read Register

Read a PIN: LOW (0) or HIGH (1)

Readable (only).

GPIO – SEVEN-SEGMENT DISPLAYs

DIGIT3	DIGIT2	:	DIGIT1	DIGIT0

bit	7	6	5	4	3	2	1	0	
PORT									
A	ENABLE	SEL2	SEL1	SEL0	DATA3	DATA2	DATA1	DATA0	7 SEGMENT DISPLAY
I/O	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	

ENABLE: Enable the seven-segment display (0x80)

SEL2-0: Select DIGIT (0-4)

Operation:

Digit0:	*SEL3-0*	*000*
Digit1:	*SEL3-0*	*001*
Digit2:	*SEL3-0*	*010*
Digit3:	*SEL3-0*	*011*
: :	*SEL3-0*	*100*

DATA3-0: BCD number for display (0-9: 0000-1001)

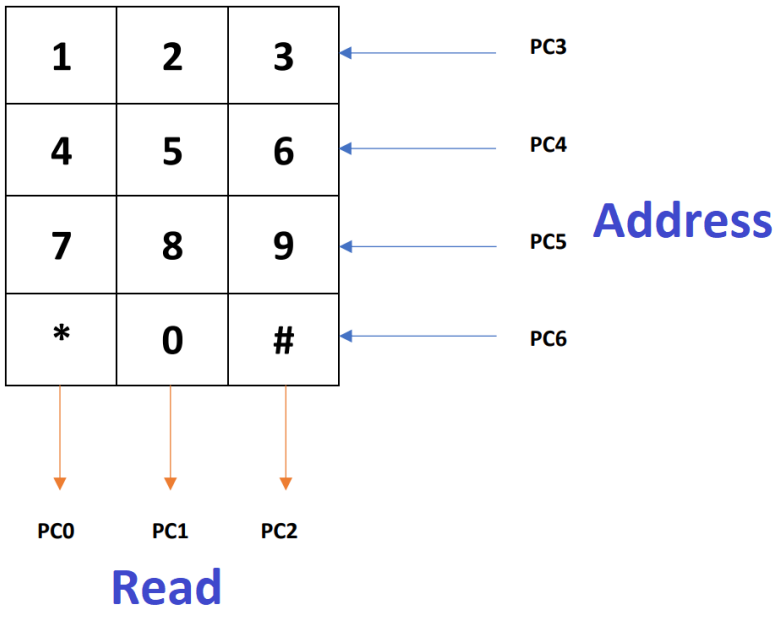
Note: Only one digit can be active at a time!

GPIO – Matrix keyboard

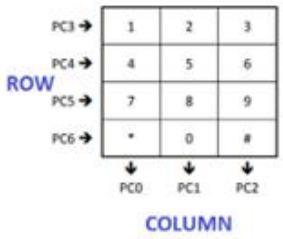
C	RED	KBD4row	KBD3row	KBD2row	KBD1row	KBD_right	KBD_centr	KBD_left	Keyboard
I/O	OUT	OUT	OUT	OUT	OUT	IN	IN	IN	

RGB RED

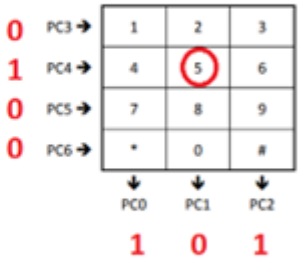
KBDrow: 1: Select; Column: 0: Pressed



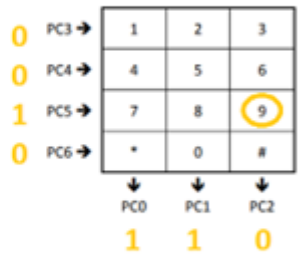
Key	Address				Read							
	PC3	PC4	PC5	PC6	PC6	PC5	PC4	PC3	PC2	PC1	PC0	
1									1	1	0	14
2	1	0	0	0	0	0	0	1	1	0	1	13
3									0	1	1	11
4									1	1	0	22
5	0	1	0	0	0	0	1	0	1	0	1	21
6									0	1	1	19
7									1	1	0	38
8	0	0	1	0	0	1	0	0	1	0	1	37
9									0	1	1	35
*									1	1	0	70
0	0	0	0	1	1	0	0	0	1	0	1	69
#									0	1	1	67



Key	Address				Read							
	PC3	PC4	PC5	PC6	PC6	PC5	PC4	PC3	PC2	PC1	PC0	
1									1	1	0	14
2	1	0	0	0	0	0	0	1	1	0	1	13
3									0	1	1	11
4									1	1	0	22
5	0	1	0	0	0	0	1	0	1	0	1	21
6									0	1	1	19
7									1	1	0	38
8	0	0	1	0	0	1	0	0	1	0	1	37
9									0	1	1	35
*									1	1	0	70
0	0	0	0	1	1	0	0	0	1	0	1	69
#									0	1	1	67



Key	Address				Read							
	PC3	PC4	PC5	PC6	PC6	PC5	PC4	PC3	PC2	PC1	PC0	
1									1	1	0	14
2	1	0	0	0	0	0	0	1	1	0	1	13
3									0	1	1	11
4									1	1	0	22
5	0	1	0	0	0	0	1	0	1	0	1	21
6									0	1	1	19
7									1	1	0	38
8	0	0	1	0	0	1	0	0	1	0	1	37
9									0	1	1	35
*									1	1	0	70
0	0	0	0	1	1	0	0	0	1	0	1	69
#									0	1	1	67



Constant array for matrix keyboard

```
const unsigned char keys_array[12] = { 69, 14, 13, 11, 22, 21, 19, 38, 37, 35, 70, 67 };
//                                      0   1   2   3   4   5   6   7   8   9   *   #
```

8-bit Timer/Counter0 with PWM and Asynchronous Operation

http://ww1.microchip.com/downloads/en/DeviceDoc/atmel-2490-8-bit-avr-microcontroller-atmega64-l_datasheet.pdf (pp. 93-111)

Timer/Counter0 is a general purpose, single-channel, 8-bit Timer/Counter module:

- Single Channel Counter
- Clear Timer on Compare Match (Auto Reload)
- Glitch-free, Phase Correct Pulse Width Modulator (PWM)
- Frequency Generator
- 10-bit Clock Prescaler
- Overflow and Compare Match Interrupt Sources (TOV0 and OCF0)
- Allows Clocking from External 32 kHz Watch Crystal Independent of the I/O Clock

Figure 34. 8-bit Timer/Counter Block Diagram

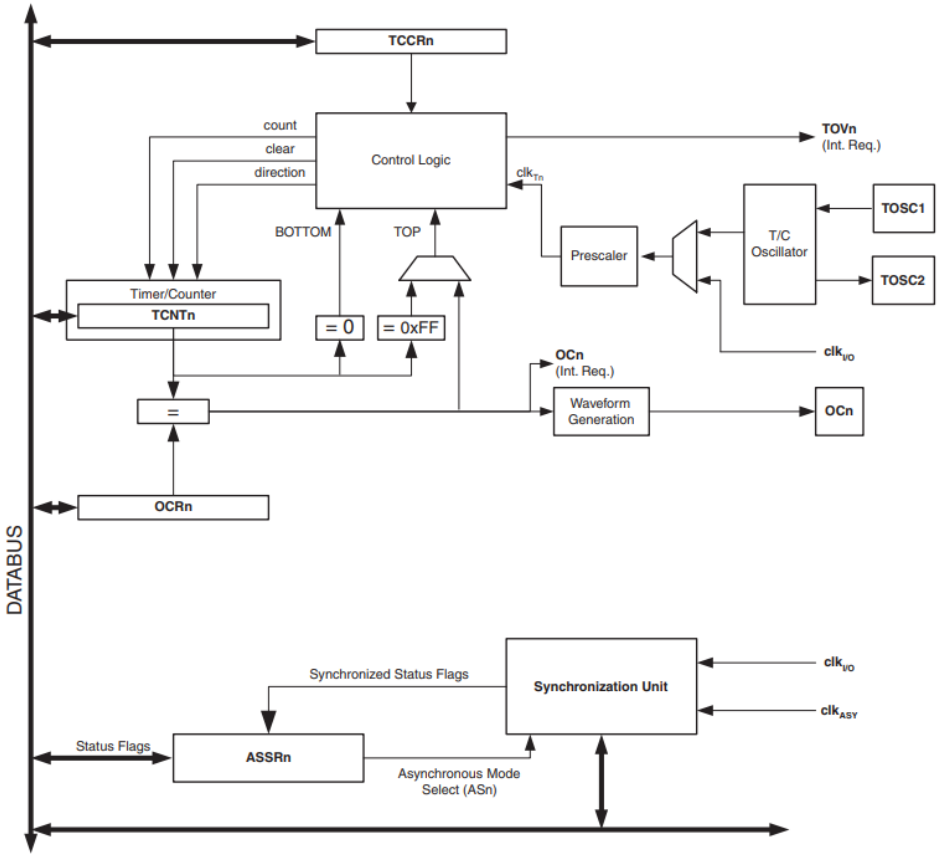
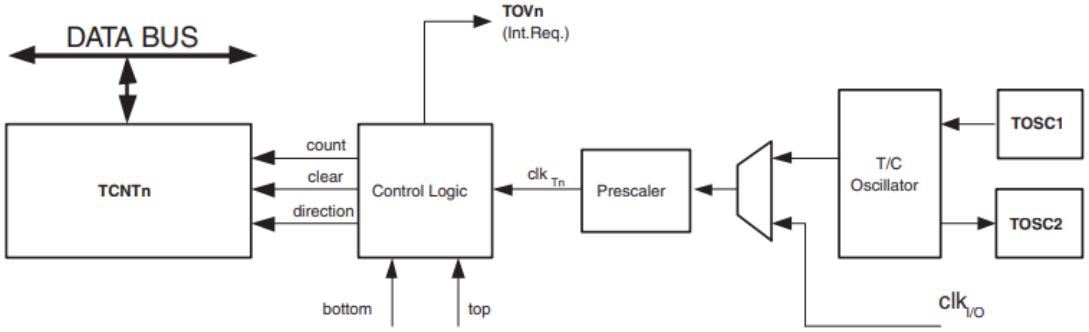


Table 51. Definitions

BOTTOM	The counter reaches the BOTTOM when it becomes zero (0x00).
MAX	The counter reaches its MAXimum when it becomes 0xFF (decimal 255).
TOP	The counter reaches the TOP when it becomes equal to the highest value in the count sequence. The TOP value can be assigned to be the fixed value 0xFF (MAX) or the value stored in the OCR0 Register. The assignment is dependent on the mode of operation.

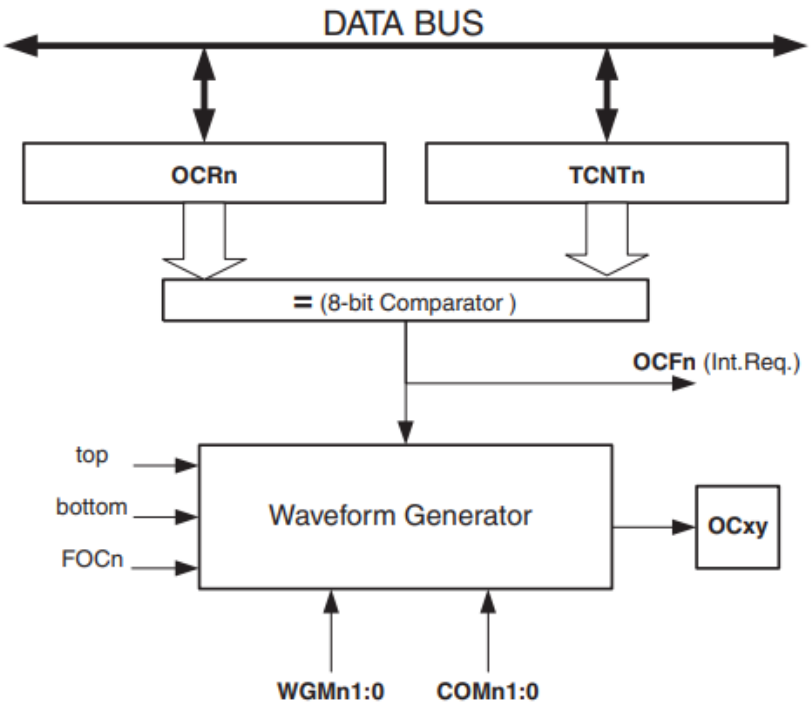
Figure 35. Counter Unit Block Diagram



Signal description (internal signals):

count	Increment or decrement TCNT0 by 1.
direction	Selects between increment and decrement.
clear	Clear TCNT0 (set all bits to zero).
clk_{T0}	Timer/Counter clock.
top	Signalizes that TCNT0 has reached maximum value.
bottom	Signalizes that TCNT0 has reached minimum value (zero).

Figure 36. Output Compare Unit, Block Diagram



Modes of Operation

- Normal Mode
- Clear Timer on Compare Match (CTC) Mode
- Fast PWM Mode
- Phase Correct PWM Mode

Normal Mode

Figure 41. Timer/Counter Timing Diagram, no Prescaling

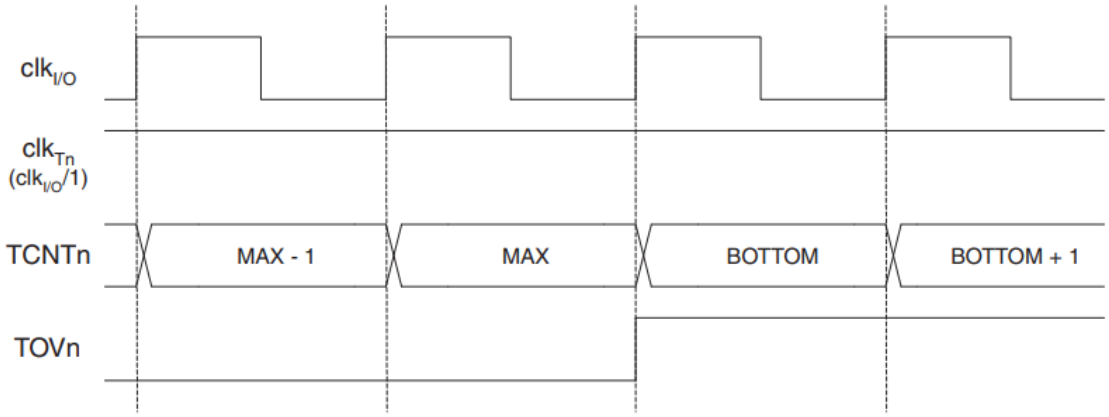
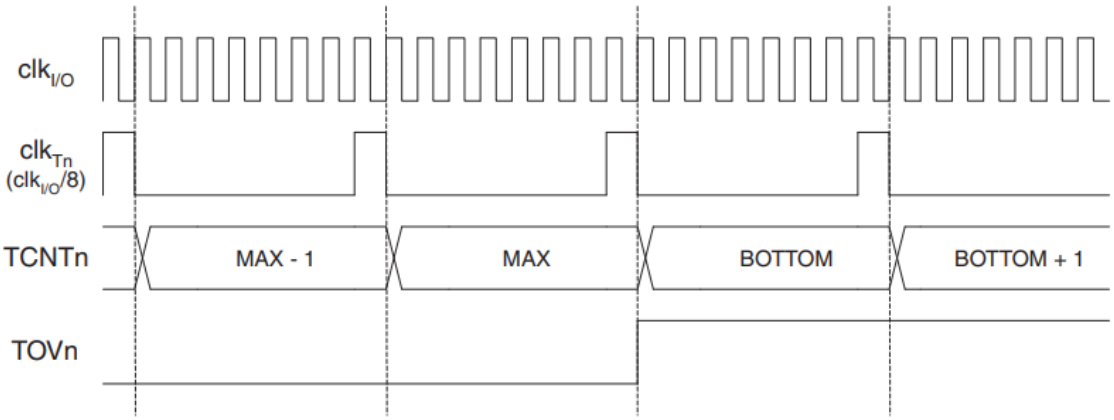


Figure 42. Timer/Counter Timing Diagram, with Prescaler ($f_{clk\_I/O}/8$)



Clear Timer on Compare Match (CTC) Mode

Figure 44 shows the setting of OCF0 and the clearing of TCNT0 in CTC mode.

Figure 44. Timer/Counter Timing Diagram, Clear Timer on Compare Match Mode, with Prescaler ($f_{clk\_I/O}/8$)

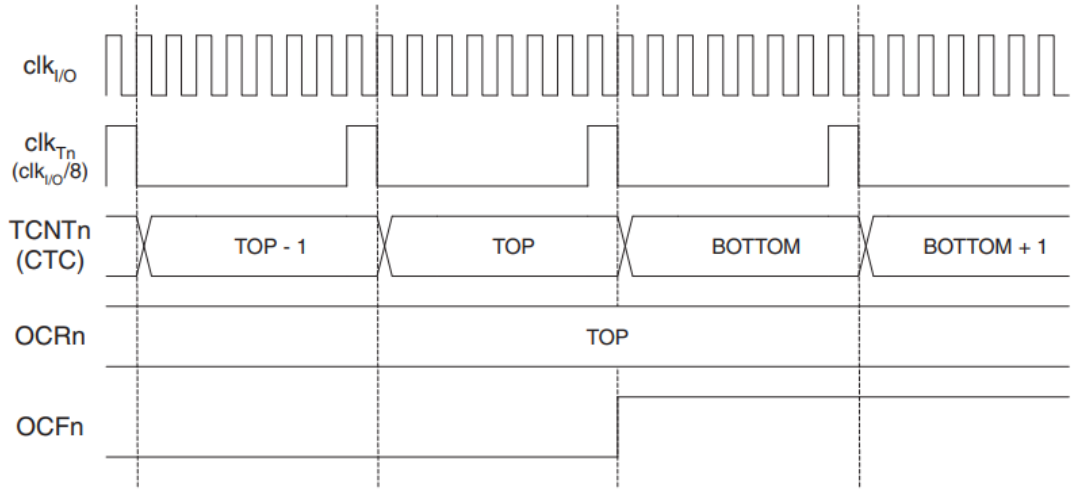
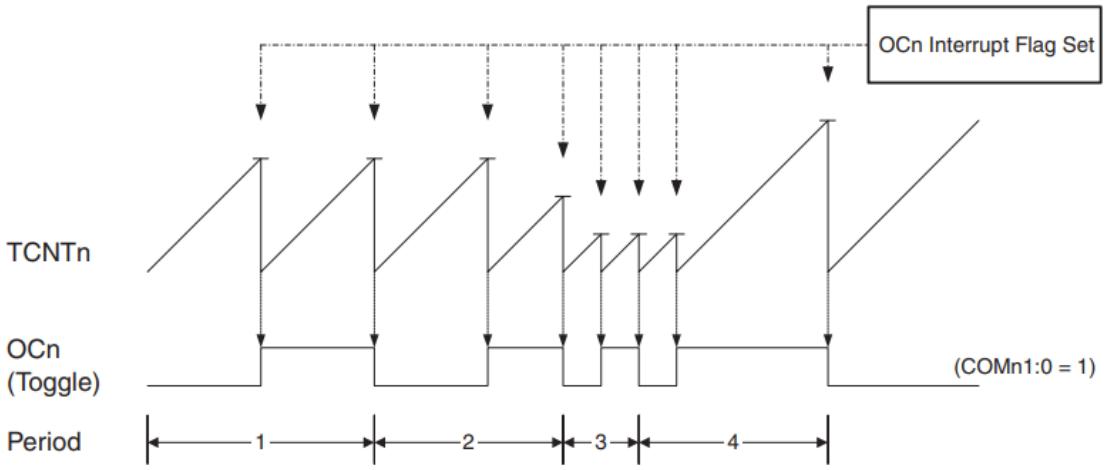
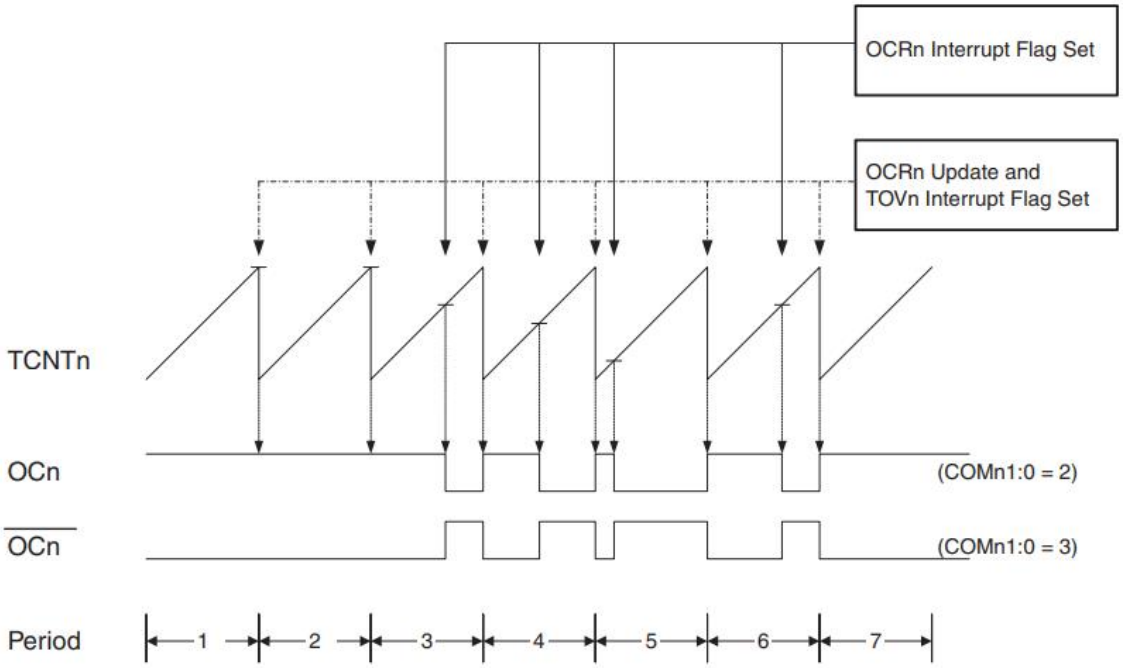


Figure 38. CTC Mode, Timing Diagram



Fast PWM Mode

Figure 39. Fast PWM Mode, Timing Diagram



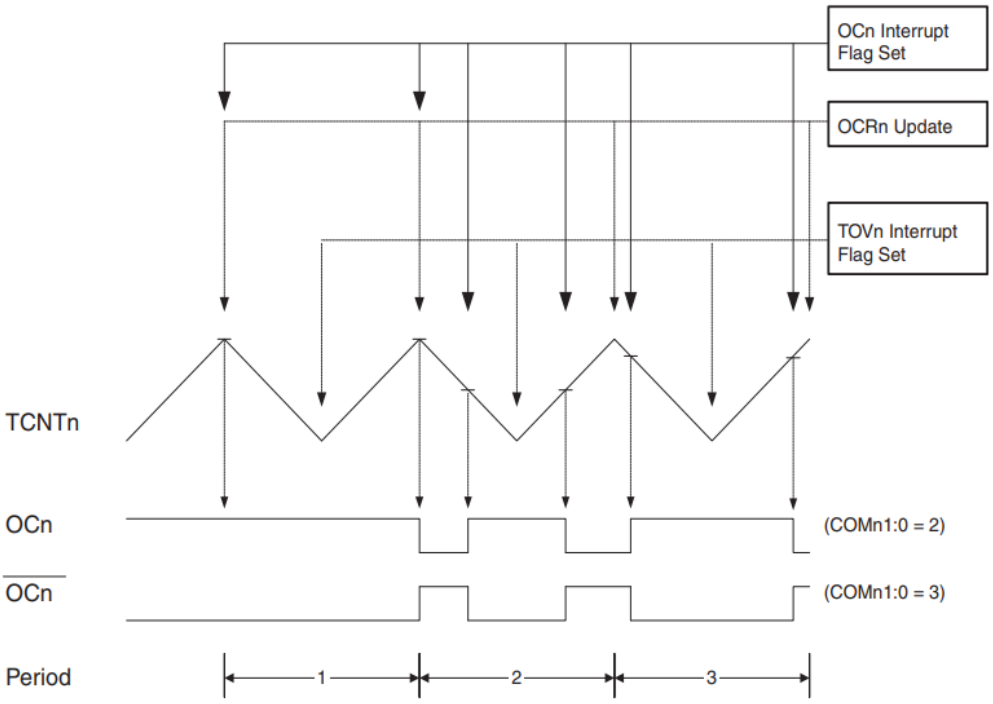
The PWM frequency for the output can be calculated by the following equation:

$$f_{OCnPWM} = \frac{f_{clk\_I/O}}{N \cdot 256}$$

The N variable represents the prescale factor (1, 8, 32, 64, 128, 256, or 1024).

Phase Correct PWM Mode

Figure 40. Phase Correct PWM Mode, Timing Diagram



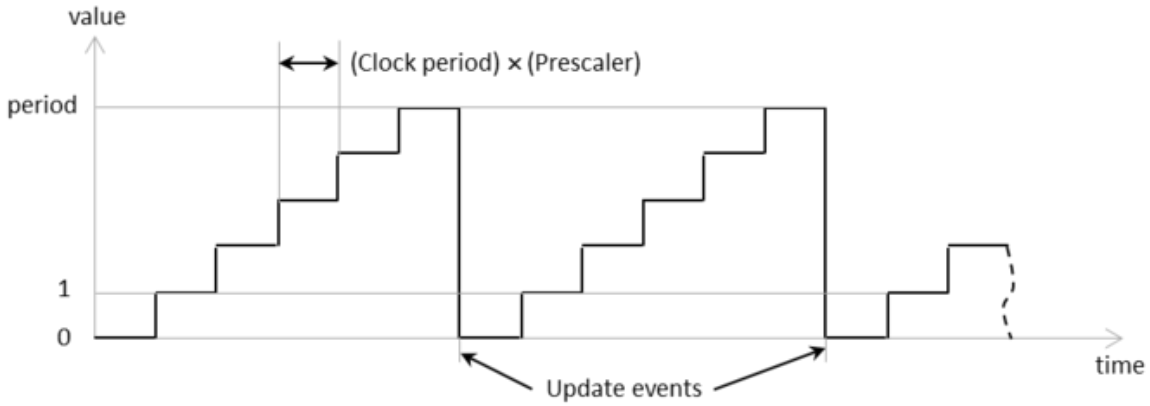
The PWM frequency for the output when using phase correct PWM can be calculated by the following equation:

$$f_{OCnPCPWM} = \frac{f_{clk\_I/O}}{N \cdot 510}$$

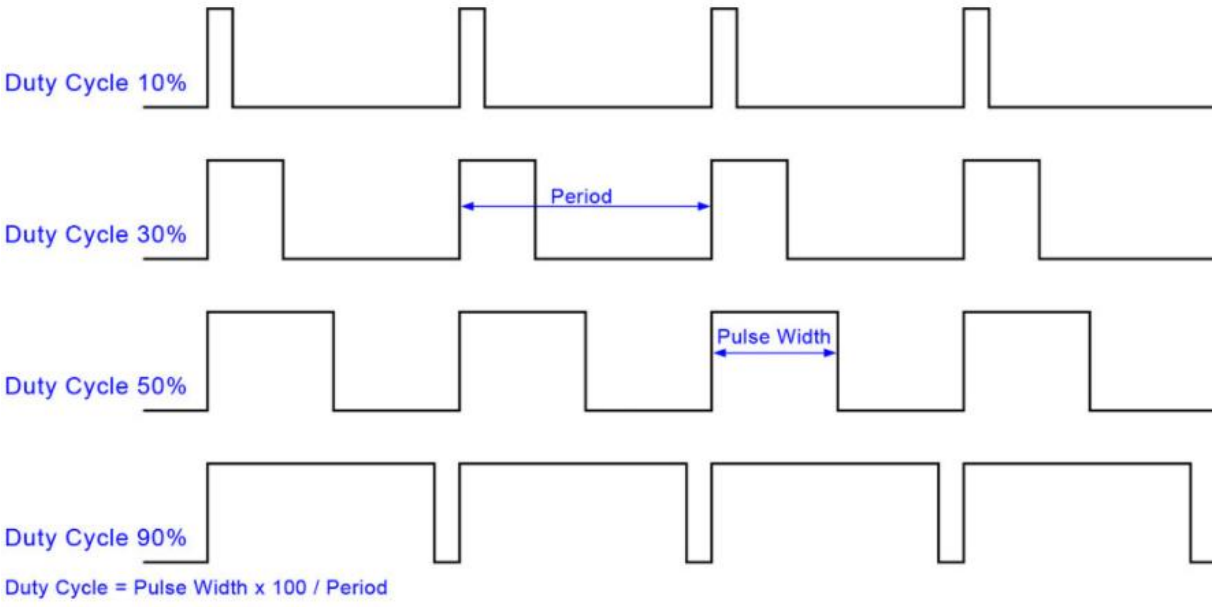
The N variable represents the prescale factor (1, 8, 32, 64, 128, 256, or 1024).

Interrupts (Basic)

$$UpdateEvent = \frac{Timer_{clock}}{(Prescaler\quad)(Period+1)}$$



PWM



8-bit Timer/Counter Register Description

TCCR0 – Timer/Counter Control Register

Bit	7	6	5	4	3	2	1	0	
0x33 (0x53)	FOC0	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00	TCCR0
Read/Write	W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Table 52. Waveform Generation Mode Bit Description(1)

Mode	WGM01 (CTC0)	WGM00 (PWM0)	Timer/Counter Mode of Operation	TOP	Update of OCR0 at	TOV0 Flag Set on
0	0	0	Normal	0xFF	Immediate	MAX
1	0	1	PWM, Phase Correct	0xFF	TOP	BOTTOM
2	1	0	CTC	OCR0	Immediate	MAX
3	1	1	Fast PWM	0xFF	BOTTOM	MAX

Table 56. Clock Select Bit Description

CS02	CS01	CS00	Description
0	0	0	No clock source (Timer/counter stopped)
0	0	1	clk_{T0S}/(No prescaling)
0	1	0	clk_{T0S}/8 (From prescaler)
0	1	1	clk_{T0S}/32 (From prescaler)
1	0	0	clk_{T0S}/64 (From prescaler)
1	0	1	clk_{T0S}/128 (From prescaler)
1	1	0	clk_{T0S}/256 (From prescaler)
1	1	1	clk_{T0S}/1024 (From prescaler)

Figure 45. Prescaler for Timer/Counter0

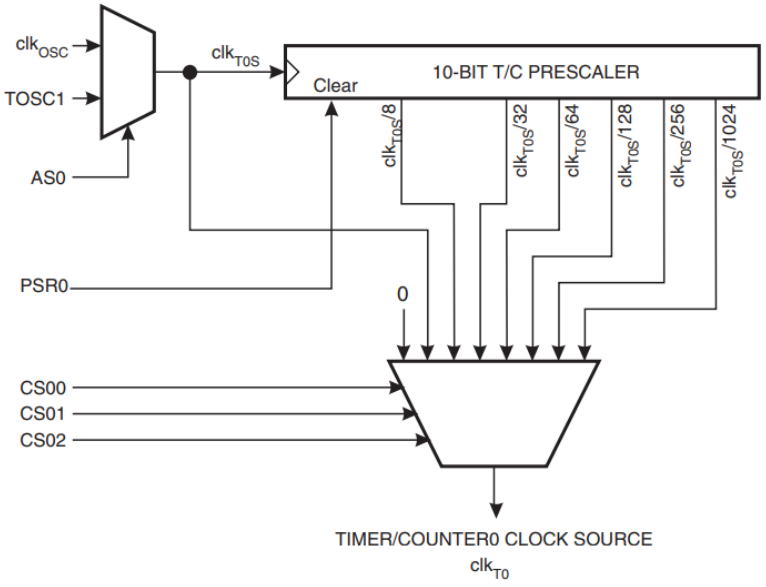


Table 53. Compare Output Mode, non-PWM Mode

COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected.
0	1	Toggle OC0 on Compare Match.
1	0	Clear OC0 on Compare Match.
1	1	Set OC0 on Compare Match.

Table 54. Compare Output Mode, Fast PWM Mode(1)

COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected.
0	1	Reserved
1	0	Clear OC0 on Compare Match, set OC0 at BOTTOM, (non-inverting mode).
1	1	Set OC0 on Compare Match, clear OC0 at BOTTOM, (inverting mode).

Table 55. Compare Output Mode, Phase Correct PWM Mode(1)

COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected.
0	1	Reserved.
1	0	Clear OC0 on Compare Match when up-counting. Set OC0 on Compare Match when downcounting.
1	1	Set OC0 on Compare Match when up-counting. Clear OC0 on Compare Match when downcounting.

TCNT0 – Timer/Counter Register

Bit	7	6	5	4	3	2	1	0	
0x32 (0x52)	TCNT0[7:0]								TCNT0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

OCR0 – Output Compare Register

Bit	7	6	5	4	3	2	1	0	
0x31 (0x51)	OCR0[7:0]								OCR0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

TIMSK – Timer/Counter Interrupt Mask Register

Bit	7	6	5	4	3	2	1	0	
0x37 (0x57)	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	OCIE0	TOIE0	TIMSK
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 1 – OCIE0: Timer/Counter0 Output Compare Match Interrupt Enable**

When the OCIE0 bit is written to one, and the I-bit in the Status Register is set (one), the Timer/Counter0 Compare Match interrupt is enabled. The corresponding interrupt is executed if a Compare Match in Timer/Counter0 occurs, that is, when the OCF0 bit is set in the Timer/Counter Interrupt Flag Register – TIFR.

- **Bit 0 – TOIE0: Timer/Counter0 Overflow Interrupt Enable**

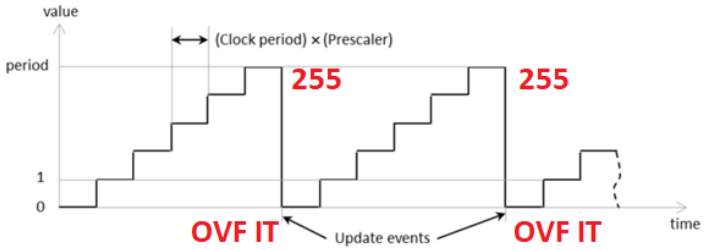
When the TOIE0 bit is written to one, and the I-bit in the Status Register is set (one), the Timer/Counter0 Overflow interrupt is enabled. The corresponding interrupt is executed if an overflow in Timer/Counter0 occurs, that is, when the TOV0 bit is set in the Timer/Counter Interrupt Flag Register – TIFR.

Example 1 – TIM0, Normal Mode, 1024 prescaler, OVF IT

$$UpdateEvent = \frac{Timer_{clock}}{(Prescaler)(Period + 1)}$$

8000000Hz
1024
255 + 1

= 8000000Hz/((1024)*(255+1))
= 8000000Hz/((1024)*(256))
= 30,517578125Hz



0,032768s
32,768ms - UpdateEvent

Table 52. Waveform Generation Mode Bit Description(1)

Mode	WGM01 (CTC0)	WGM00 (PWM0)	Timer/Counter Mode of Operation	TOP	Update of OCR0 at	TOV0 Flag Set on
0	0	0	Normal	0xFF	Immediate	MAX
1	0	1	PWM, Phase Correct	0xFF	TOP	BOTTOM
2	1	0	CTC	OCR0	Immediate	MAX
3	1	1	Fast PWM	0xFF	BOTTOM	MAX

Table 56. Clock Select Bit Description

CS02	CS01	CS00	Description
0	0	0	No clock source (Timer/counter stopped)
0	0	1	clk_{T0S}/(No prescaling)
0	1	0	clk_{T0S}/8 (From prescaler)
0	1	1	clk_{T0S}/32 (From prescaler)
1	0	0	clk_{T0S}/64 (From prescaler)
1	0	1	clk_{T0S}/128 (From prescaler)
1	1	0	clk_{T0S}/256 (From prescaler)
1	1	1	clk_{T0S}/1024 (From prescaler)

TIMSK – Timer/Counter Interrupt Mask Register

Bit	7	6	5	4	3	2	1	0	
0x37 (0x57)	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	OCIE0	TOIE0	TIMSK
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 1 – OCIE0: Timer/Counter0 Output Compare Match Interrupt Enable**

When the OCIE0 bit is written to one, and the I-bit in the Status Register is set (one), the Timer/Counter0 Compare Match interrupt is enabled. The corresponding interrupt is executed if a Compare Match in Timer/Counter0 occurs, that is, when the OCF0 bit is set in the Timer/Counter Interrupt Flag Register – TIFR.

- **Bit 0 – TOIE0: Timer/Counter0 Overflow Interrupt Enable**

When the TOIE0 bit is written to one, and the I-bit in the Status Register is set (one), the Timer/Counter0 Overflow interrupt is enabled. The corresponding interrupt is executed if an overflow in Timer/Counter0 occurs, that is, when the TOV0 bit is set in the Timer/Counter Interrupt Flag Register – TIFR.

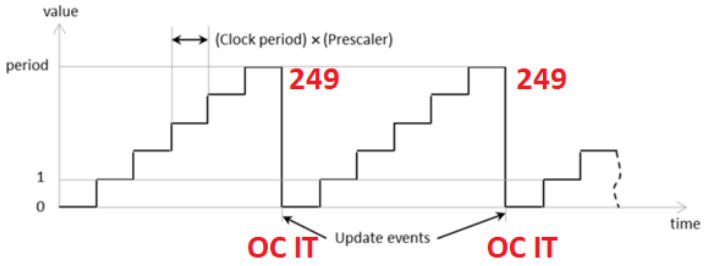
Example 2 – TIM0, CTC Mode, 32 prescaler, OC IT

$$UpdateEvent = \frac{Timer_{clock}}{(Prescaler)(Period + 1)}$$

8000000Hz

32 249 + 1

= 8000000Hz/((32)*(249+1))
= 8000000Hz/((32)*(250))
= 1000Hz



0,001s

1ms - UpdateEvent

Table 52. Waveform Generation Mode Bit Description[1]

Mode	WGM01 (CTC0)	WGM00 (PWM0)	Timer/Counter Mode of Operation	TOP	Update of OCR0 at	TOV0 Flag Set on
0	0	0	Normal	0xFF	Immediate	MAX
1	0	1	PWM, Phase Correct	0xFF	TOP	BOTTOM
2	1	0	CTC	OCR0	Immediate	MAX
3	1	1	Fast PWM	0xFF	BOTTOM	MAX

Table 56. Clock Select Bit Description

CS02	CS01	CS00	Description
0	0	0	No clock source (Timer/counter stopped)
0	0	1	clk_{T0S}/(No prescaling)
0	1	0	clk_{T0S}/8 (From prescaler)
0	1	1	clk_{T0S}/32 (From prescaler)
1	0	0	clk_{T0S}/64 (From prescaler)
1	0	1	clk_{T0S}/128 (From prescaler)
1	1	0	clk_{T0S}/256 (From prescaler)
1	1	1	clk_{T0S}/1024 (From prescaler)

TIMSK – Timer/Counter Interrupt Mask Register

Bit	7	6	5	4	3	2	1	0	
0x37 (0x57)	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	OCIE0	TOIE0	TIMSK
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 1 – OCIE0: Timer/Counter0 Output Compare Match Interrupt Enable**

When the OCIE0 bit is written to one, and the I-bit in the Status Register is set (one), the Timer/Counter0 Compare Match interrupt is enabled. The corresponding interrupt is executed if a Compare Match in Timer/Counter0 occurs, that is, when the OCF0 bit is set in the Timer/Counter Interrupt Flag Register – TIFR.

- **Bit 0 – TOIE0: Timer/Counter0 Overflow Interrupt Enable**

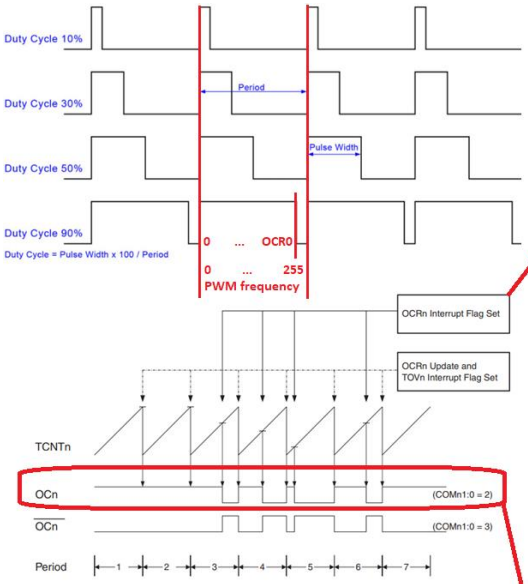
When the TOIE0 bit is written to one, and the I-bit in the Status Register is set (one), the Timer/Counter0 Overflow interrupt is enabled. The corresponding interrupt is executed if an overflow in Timer/Counter0 occurs, that is, when the TOV0 bit is set in the Timer/Counter Interrupt Flag Register – TIFR.

OCR0 – Output Compare Register

Bit	7	6	5	4	3	2	1	0	
0x31 (0x51)	OCR0[7:0]								OCR0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

249

Example 3 – TIM0, Fast PWM Mode, 8 prescaler



The PWM frequency for the output can be calculated by the following equation:

$$f_{OCnPWM} = \frac{f_{clk\_I/O}}{N \cdot 256}$$

The N variable represents the prescale factor (1, 8, 32, 64, 128, 256, or 1024).

CLK=8000000Hz
OCR0: 0-255
10%: (256/100)*10=25.6
30%: (256/100)*30=76.8
50%: (256/100)*50=128
90%: (256/100)*90=230,4

PWM frequency

8000000/(8*256)=3906.25Hz

0.256ms

Table 52. Waveform Generation Mode Bit Description[1]

Mode	WGM01 (CTC0)	WGM00 (PWM0)	Timer/Counter Mode of Operation	TOP	Update of OCR0 at	TOV0 Flag Set on
0	0	0	Normal	0xFF	Immediate	MAX
1	0	1	PWM, Phase Correct	0xFF	TOP	BOTTOM
2	1	0	CTC	OCR0	Immediate	MAX
3	1	1	Fast PWM	0xFF	BOTTOM	MAX

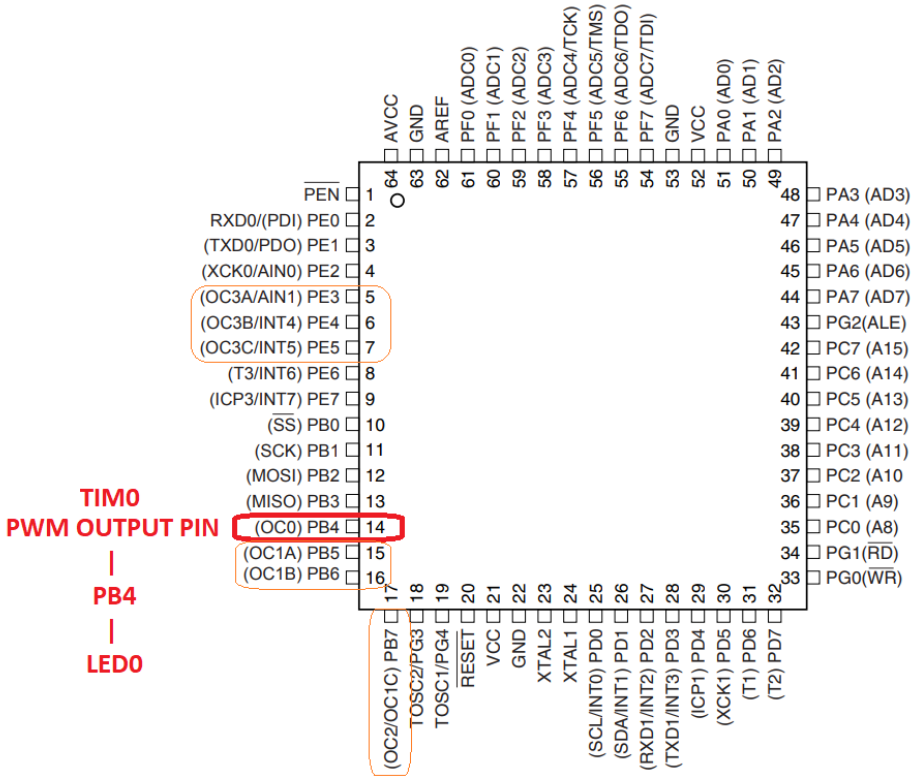
Table 56. Clock Select Bit Description

CS02	CS01	CS00	Description
0	0	0	No clock source (Timer/counter stopped)
0	0	1	clk_{T0S}/(No prescaling)
0	1	0	clk_{T0S}/8 (From prescaler)
0	1	1	clk_{T0S}/32 (From prescaler)
1	0	0	clk_{T0S}/64 (From prescaler)
1	0	1	clk_{T0S}/128 (From prescaler)
1	1	0	clk_{T0S}/256 (From prescaler)
1	1	1	clk_{T0S}/1024 (From prescaler)

fast enough

Table 54. Compare Output Mode, Fast PWM Mode[1]

COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected.
0	1	Reserved
1	0	Clear OC0 on Compare Match, set OC0 at BOTTOM, (non-inverting mode).
1	1	Set OC0 on Compare Match, clear OC0 at BOTTOM, (inverting mode).



TIM0 PWM OUTPUT PIN

PB4

LED0

RGB LED

bit	7	6	5	4	3	2	1	0	
PORT									
A	ENABLE	SEL2	SEL1	SEL0	DATA3	DATA2	DATA1	DATA0	7 SEGMENT DISPLAY
I/O	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	
B	Led3	Led2	Led1	Led0					LED/lo 4bit
I/O	OUT	OUT	OUT	OUT					
C	RED	KBD4row	KBD3row	KBD2row	KBD1row	KBD_right	KBD_centr	KBD_left	Keyboard
I/O	OUT	OUT	OUT	OUT	OUT	IN	IN	IN	
D	Led7	Led6	Led5	Led4					LED/hi 4bit
I/O	OUT	OUT	OUT	OUT					
E	LCD_DATA7	LCD_DATA6	LCD_DATA5	LCD_DATA4	GREEN	BLUE			LCD data
I/O	OUT	OUT	OUT	OUT	OUT	OUT			
F					LCD_E	LCD_R/W	LCD_RS	LM35	LCD Control
I/O					OUT	OUT	OUT	IN	Analog
G	NC	NC	NC	K4	K3	K2	K1	K0	Pushbutton
I/O	X	X	X	IN	IN	IN	IN	IN	
bit	7	6	5	4	3	2	1	0	

7 Segment display: E 1:Enable, 0: Disable; SEL 2-1-0: 000:10exp0, 001: 10exp1, 010:10exp2, 011:10exp3, 100:Double Leds

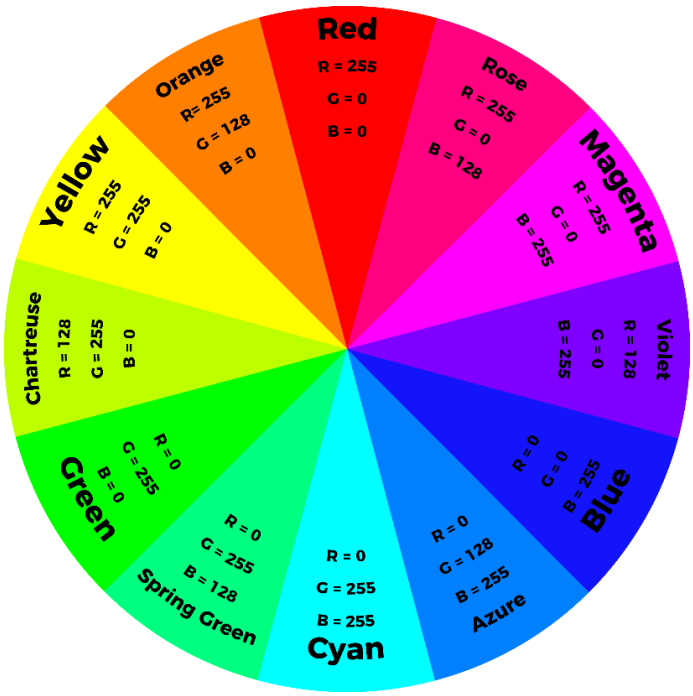
Led0…7: 0: Disable, 1:Enable; RED,GREEN,BLUE: 0 Disable, 1: Enable; KBDrow: 1: Select; Column: 0: Pressed

LCD_E: 0 Select, LCD_R/W: 1:Read, 0: Write; LCD_RS: 1: Data, 0: Control

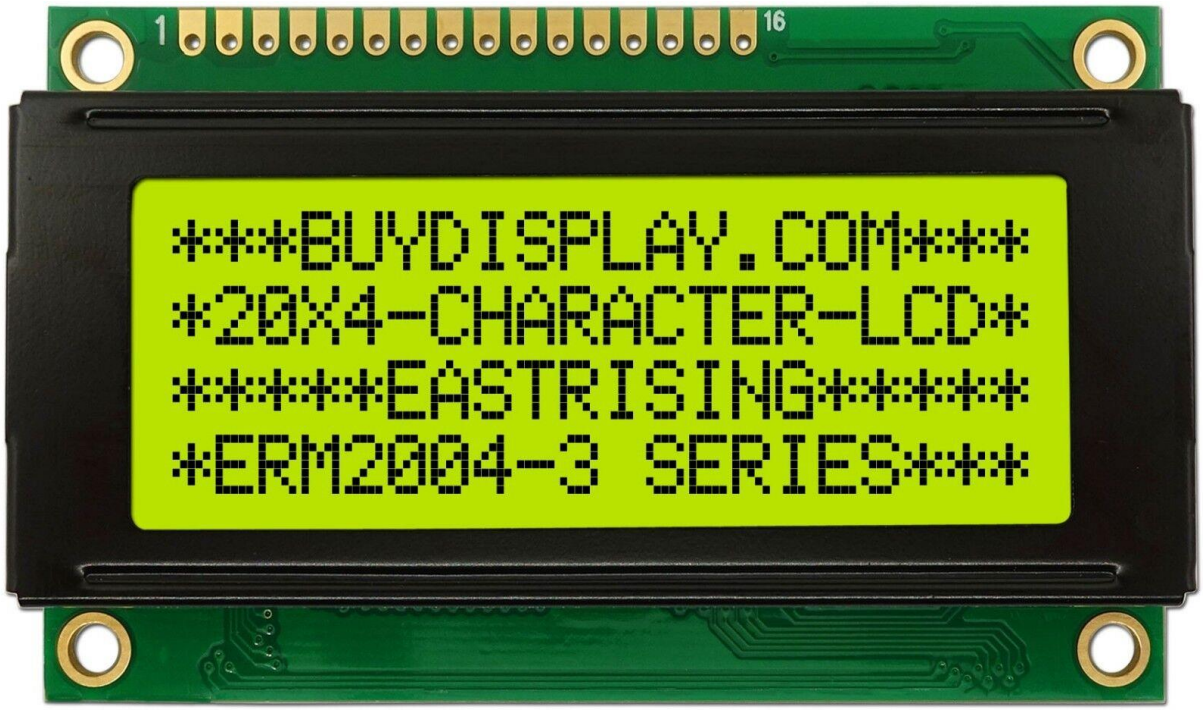
RGB LED – RED **PC7**

RGB LED – GREEN **PE3**

RGB LED – BLUE **PE2**



GPIO – LCD



http://lcd-linux.sourceforge.net/pdfdocs/hd44780.pdf

T-Bird GPIO-A/T			
2	PF1	LCD_RS	OUT
3	PF2	LCD_R/W	OUT
4	PF3	LCD_E	OUT
5	PE4	LCD_DATA4	OUT
6	PE5	LCD_DATA5	OUT
7	PE6	LCD_DATA6	OUT
8	PE7	LCD_DATA7	OUT
13	VCC	VCC	VCC
14	GND	GND	GND

Pin Functions

Signal	No. of Lines	I/O	Device Interfaced with	Function
RS	1	I	MPU	Selects registers. 0: Instruction register (for write) Busy flag: address counter (for read) 1: Data register (for write and read)
R/$\overline{W}$	1	I	MPU	Selects read or write. 0: Write 1: Read
E	1	I	MPU	Starts data read/write.
DB4 to DB7	4	I/O	MPU	Four high order bidirectional tristate data bus pins. Used for data transfer and receive between the MPU and the HD44780U. DB7 can be used as a busy flag.
DB0 to DB3	4	I/O	MPU	Four low order bidirectional tristate data bus pins. Used for data transfer and receive between the MPU and the HD44780U. These pins are not used during 4-bit operation.

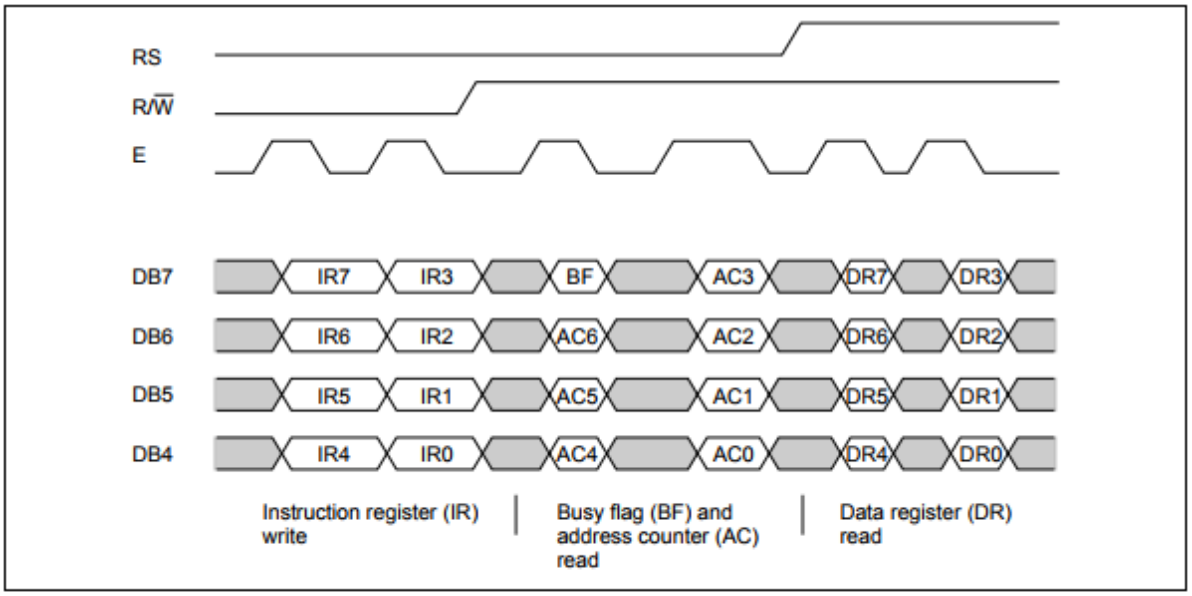


Figure 9 4-Bit Transfer Example

RS	0	Instruction register write	0	Status read	1	Data register write	1	Data register read
R/W	0		1		0		1	
E	0-1-0		0-1-0		0-1-0		0-1-0	

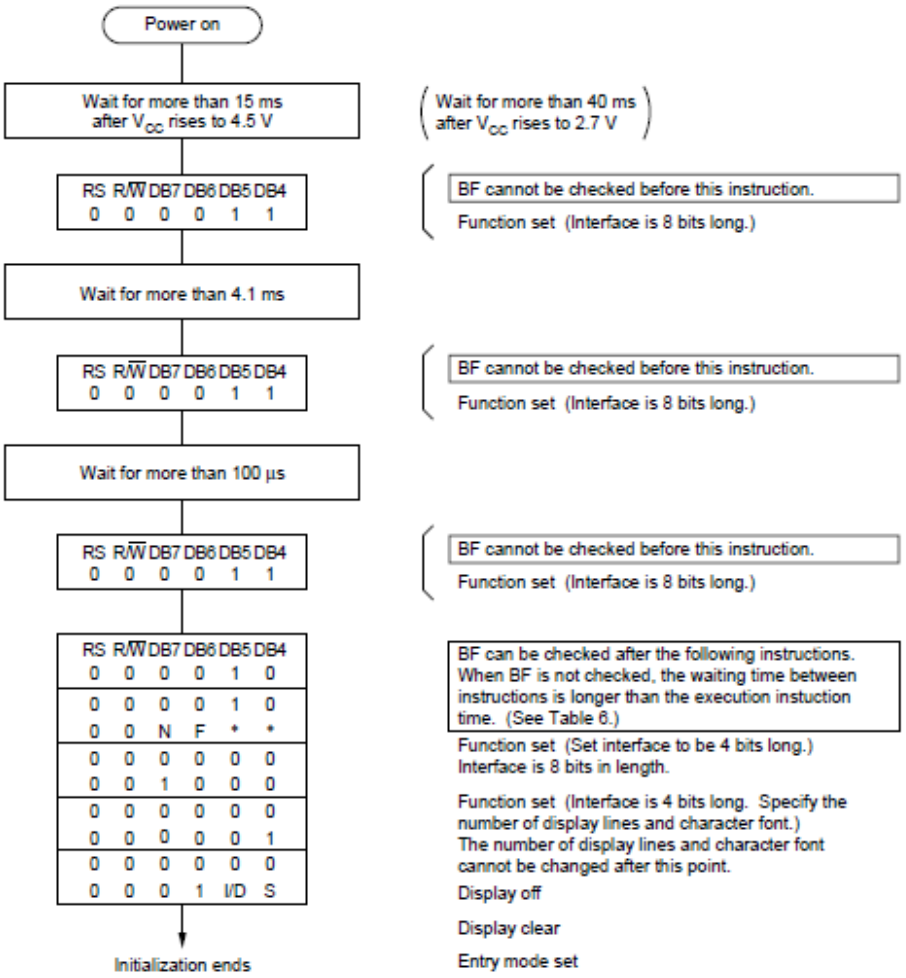
Instruction	Code RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Description	Execution Time (max) (when f_{cp} or f_{OSC} is 270 kHz)
Clear display	0	0	0	0	0	0	0	0	0	1	Clears entire display and sets DDRAM address 0 in address counter.	
Return home	0	0	0	0	0	0	0	0	1	—	Sets DDRAM address 0 in address counter. Also returns display from being shifted to original position. DDRAM contents remain unchanged.	1.52 ms
Entry mode set	0	0	0	0	0	0	0	1	I/D	S	Sets cursor move direction and specifies display shift. These operations are performed during data write and read.	37 µs
Display on/off control	0	0	0	0	0	0	1	D	C	B	Sets entire display (D) on/off, cursor on/off (C), and blinking of cursor position character (B).	37 µs
Cursor or display shift	0	0	0	0	0	1	S/C	R/L	—	—	Moves cursor and shifts display without changing DDRAM contents.	37 µs
Function set	0	0	0	0	1	DL	N	F	—	—	Sets interface data length (DL), number of display lines (N), and character font (F).	37 µs
Set CGRAM address	0	0	0	1	ACG	ACG	ACG	ACG	ACG	ACG	Sets CGRAM address. CGRAM data is sent and received after this setting.	37 µs
Set DDRAM address	0	0	1	ADD	ADD	ADD	ADD	ADD	ADD	ADD	Sets DDRAM address. DDRAM data is sent and received after this setting.	37 µs
Read busy flag & address	0	1	BF	AC	AC	AC	AC	AC	AC	AC	Reads busy flag (BF) indicating internal operation is being performed and reads address counter contents.	0 µs

Table 6 Instructions (cont)

Instruction	Code RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Description	Execution Time (max) (when f_{cp} or f_{OSC} is 270 kHz)
Write data to CG or DDRAM	1	0	Write data								Writes data into DDRAM or CGRAM.	37 µs t_{ADD} = 4 µs*
Read data from CG or DDRAM	1	1	Read data								Reads data from DDRAM or CGRAM.	37 µs t_{ADD} = 4 µs*
	I/D = 1: Increment I/D = 0: Decrement S = 1: Accompanies display shift S/C = 1: Display shift S/C = 0: Cursor move R/L = 1: Shift to the right R/L = 0: Shift to the left DL = 1: 8 bits, DL = 0: 4 bits N = 1: 2 lines, N = 0: 1 line F = 1: 5 × 10 dots, F = 0: 5 × 8 dots BF = 1: Internally operating BF = 0: Instructions acceptable										DDRAM: Display data RAM CGRAM: Character generator RAM ACG: CGRAM address ADD: DDRAM address (corresponds to cursor address) AC: Address counter used for both DD and CGRAM addresses	Execution time changes when frequency changes Example: When f_{cp} or f_{OSC} is 250 kHz, $37\ \mu s \times \frac{270}{250} = 40\ \mu s$

Note: — indicates no effect.

* After execution of the CGRAM/DDRAM data write or read instruction, the RAM address counter is incremented or decremented by 1. The RAM address counter is updated after the busy flag turns off. In Figure 10, t_{ADD} is the time elapsed after the busy flag turns off until the address counter is updated.



Instruction	Code RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Description	Execution Time (max) (when f_{cp} or f_{OSC} is 270 kHz)
Clear display	0	0	0	0	0	0	0	0	0	1	Clears entire display and sets DDRAM address 0 in address counter.	
Return home	0	0	0	0	0	0	0	0	1	—	Sets DDRAM address 0 in address counter. Also returns display from being shifted to original position. DDRAM contents remain unchanged.	1.52 ms
Entry mode set	0	0	0	0	0	0	0	1	I/D	S	Sets cursor move direction and specifies display shift. These operations are performed during data write and read.	37 µs
Display on/off control	0	0	0	0	0	0	1	D	C	B	Sets entire display (D) on/off, cursor on/off (C), and blinking of cursor position character (B).	37 µs
Cursor or display shift	0	0	0	0	0	1	S/C	R/L	—	—	Moves cursor and shifts display without changing DDRAM contents.	37 µs
Function set	0	0	0	0	1	DL	N	F	—	—	Sets interface data length (DL), number of display lines (N), and character font (F).	37 µs
Set CGRAM address	0	0	0	1	ACG	ACG	ACG	ACG	ACG	ACG	Sets CGRAM address. CGRAM data is sent and received after this setting.	37 µs
Set DDRAM address	0	0	1	ADD	ADD	ADD	ADD	ADD	ADD	ADD	Sets DDRAM address. DDRAM data is sent and received after this setting.	37 µs
Read busy flag & address	0	1	BF	AC	AC	AC	AC	AC	AC	AC	Reads busy flag (BF) indicating internal operation is being performed and reads address counter contents.	0 µs

Instruction	Code RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Description	Execution Time (max) (when f_{cp} or f_{OSC} is 270 kHz)
Write data to CG or DDRAM	1	0	Write data								Writes data into DDRAM or CGRAM.	37 µs t_{ADD} = 4 µs*
Read data from CG or DDRAM	1	1	Read data								Reads data from DDRAM or CGRAM.	37 µs t_{ADD} = 4 µs*

I/D = 1: Increment
I/D = 0: Decrement
S = 1: Accompanies display shift
S/C = 1: Display shift
S/C = 0: Cursor move
R/L = 1: Shift to the right
R/L = 0: Shift to the left
DL = 1: 8 bits, DL = 0: 4 bits
N = 1: 2 lines, N = 0: 1 line
F = 1: 5 × 10 dots, F = 0: 5 × 8 dots
BF = 1: Internally operating
BF = 0: Instructions acceptable

DDRAM: Display data RAM
CGRAM: Character generator RAM
ACG: CGRAM address
ADD: DDRAM address (corresponds to cursor address)
AC: Address counter used for both DD and CGRAM addresses

Execution time changes when frequency changes
Example: When f_{cp} or f_{OSC} is 250 kHz,
$37\ \mu s \times \frac{270}{250} = 40\ \mu s$

Command: 0x28-> 4bits, 2 lines, 5x8 dots

- Display on/off control
  - 00001DCB
    - D: dislay on/off
    - C: Cursor on/off
    - B: Cursor blinking on/off

Display on/off control	0	0	0	0	0	0	1	D	C	B	Sets entire display (D) on/off, cursor on/off (C), and blinking of cursor position character (B). 37 µs

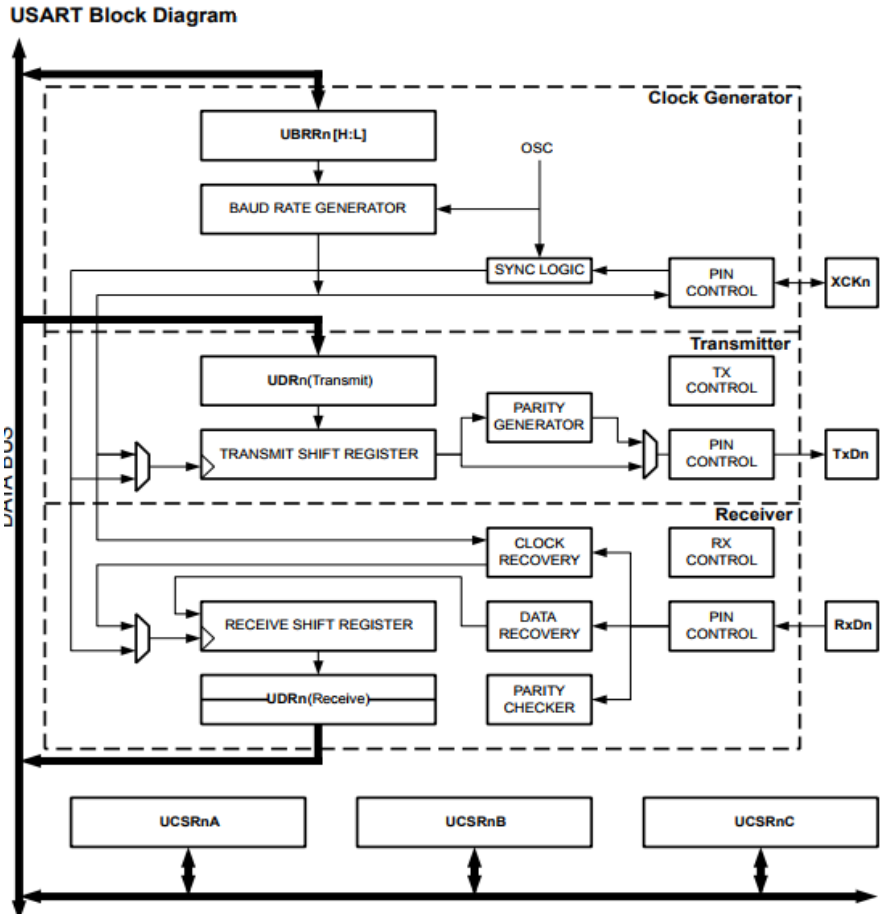
- ***When the busy flag is 1, the HD44780U is in the internal operation mode, and the next instruction will not be accepted. When RS = 0 and R/W = 1 the busy flag is output to DB7. The next instruction must be written after ensuring that the busy flag is 0.***

		RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Read busy flag and address	Code	0	1	BF	A	A	A	A	A	A	A
					← Higher order bit				Lower order bit →		

No	HEX Value	COMMAND TO LCD
1	0x01	Clear Display Screen
2	0x30	Function Set: 8-bit, 1 Line, 5x7 Dots
3	0x38	Function Set: 8-bit, 2 Line, 5x7 Dots
4	0x20	Function Set: 4-bit, 1 Line, 5x7 Dots
5	0x28	Function Set: 4-bit, 2 Line, 5x7 Dots
6	0x06	Entry Mode
7	0x08	Display off, Cursor off
8	0x0E	Display on, Cursor on
9	0x0C	Display on, Cursor off
10	0x0F	Display on, Cursor blinking
11	0x18	Shift entire display left
12	0x1C	Shift entire display right
13	0x10	Move cursor left by one character
14	0x14	Move cursor right by one character
15	0x80	Force cursor to beginning of 1st row
16	0xC0	Force cursor to beginning of 2nd row

UART

- ***Full Duplex Operation (Independent Serial Receive and Transmit Registers)***
- ***Asynchronous or Synchronous Operation***
- ***Master or Slave Clocked Synchronous Operation***
- ***High Resolution Baud Rate Generator***
- ***Supports Serial Frames with 5, 6, 7, 8, or 9 data bits and 1 or 2 stop bits***
- ***Odd or Even Parity Generation and Parity Check Supported by Hardware***
- ***Data OverRun Detection***
- ***Framing Error Detection***
- ***Noise Filtering Includes False Start Bit Detection and Digital Low Pass Filter***
- ***Three Separate Interrupts on TX Complete, TX Data Register Empty and RX Complete***
- ***Multi-processor Communication Mode***
- ***Double Speed Asynchronous Communication Mode***
- ***USART0, USART1***



Operating Mode	Equation for Calculating Baud Rate[1]	Equation for Calculating UBRR Value
Asynchronous Normal mode (U2X = 0)	$\text{BAUD} = \frac{f_{\text{OSC}}}{16(\mathbf{UBRR}+1)}$	$\mathbf{UBRR} = \frac{f_{\text{OSC}}}{16\text{BAUD}} - 1$
Asynchronous Double Speed mode (U2X = 1)	$\text{BAUD} = \frac{f_{\text{OSC}}}{8(\mathbf{UBRR}+1)}$	$\mathbf{UBRR} = \frac{f_{\text{OSC}}}{8\text{BAUD}} - 1$
Synchronous Master mode	$\text{BAUD} = \frac{f_{\text{OSC}}}{2(\mathbf{UBRR}+1)}$	$\mathbf{UBRR} = \frac{f_{\text{OSC}}}{2\text{BAUD}} - 1$

Note: 1. The baud rate is defined to be the transfer rate in bit per second (bps).

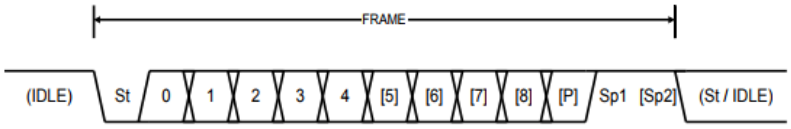
BAUD	Baud rate (in bits per second, bps).
f_{osc}	System oscillator clock frequency.
UBRR	Contents of the UBRRH and UBRRL Registers, (0-4095).

Baud Rate (bps)	f_{osc} = 8.0000MHz			
	U2X = 0		U2X = 1	
	UBRR	Error	UBRR	Error
2400	207	0.2%	416	-0.1%
4800	103	0.2%	207	0.2%
9600	51	0.2%	103	0.2%
14.4k	34	-0.8%	68	0.6%
19.2k	25	0.2%	51	0.2%
28.8k	16	2.1%	34	-0.8%
38.4k	12	0.2%	25	0.2%
57.6k	8	-3.5%	16	2.1%
76.8k	6	-7.0%	12	0.2%
115.2k	3	8.5%	8	-3.5%
230.4k	1	8.5%	3	8.5%
250k	1	0.0%	3	0.0%
0.5M	0	0.0%	1	0.0%
1M	–	–	0	0.0%
Max (1)	0.5Mbps		1Mbps	

Baud Rate (bps)	f_{osc} = 16.0000MHz			
	U2X = 0		U2X = 1	
	UBRR	Error	UBRR	Error
2400	416	-0.1%	832	0.0%
4800	207	0.2%	416	-0.1%
9600	103	0.2%	207	0.2%
14.4k	68	0.6%	138	-0.1%
19.2k	51	0.2%	103	0.2%
28.8k	34	-0.8%	68	0.6%
38.4k	25	0.2%	51	0.2%
57.6k	16	2.1%	34	-0.8%
76.8k	12	0.2%	25	0.2%
115.2k	8	-3.5%	16	2.1%
230.4k	3	8.5%	8	-3.5%
250k	3	0.0%	7	0.0%
0.5M	1	0.0%	3	0.0%
1M	0	0.0%	1	0.0%
Max (1)	1Mbps		2Mbps	

Baud Rate (bps)	f_{osc} = 8.0000MHz				f_{osc} = 11.0592MHz				f_{osc} = 14.7456MHz			
	U2X = 0		U2X = 1		U2X = 0		U2X = 1		U2X = 0		U2X = 1	
	UBRR	Error	UBRR	Error	UBRR	Error	UBRR	Error	UBRR	Error	UBRR	Error
2400	207	0.2%	416	-0.1%	287	0.0%	575	0.0%	383	0.0%	767	0.0%
4800	103	0.2%	207	0.2%	143	0.0%	287	0.0%	191	0.0%	383	0.0%
9600	51	0.2%	103	0.2%	71	0.0%	143	0.0%	95	0.0%	191	0.0%
14.4k	34	-0.8%	68	0.6%	47	0.0%	95	0.0%	63	0.0%	127	0.0%
19.2k	25	0.2%	51	0.2%	35	0.0%	71	0.0%	47	0.0%	95	0.0%
28.8k	16	2.1%	34	-0.8%	23	0.0%	47	0.0%	31	0.0%	63	0.0%
38.4k	12	0.2%	25	0.2%	17	0.0%	35	0.0%	23	0.0%	47	0.0%
57.6k	8	-3.5%	16	2.1%	11	0.0%	23	0.0%	15	0.0%	31	0.0%
76.8k	6	-7.0%	12	0.2%	8	0.0%	17	0.0%	11	0.0%	23	0.0%
115.2k	3	8.5%	8	-3.5%	5	0.0%	11	0.0%	7	0.0%	15	0.0%
230.4k	1	8.5%	3	8.5%	2	0.0%	5	0.0%	3	0.0%	7	0.0%
250k	1	0.0%	3	0.0%	2	-7.8%	5	-7.8%	3	-7.8%	6	5.3%
0.5M	0	0.0%	1	0.0%	–	–	2	-7.8%	1	-7.8%	3	-7.8%
1M	–	–	0	0.0%	–	–	–	–	0	-7.8%	1	-7.8%

Frame Formats



St Start bit, always low.

(n) Data bits (0 to 8).

P Parity bit. Can be odd or even.

Sp Stop bit, always high.

IDLE No transfers on the communication line (RxD or TxD). An IDLE line must be high.

- 1 start bit
- 5, 6, 7, 8, or 9 data bits
- no, even or odd parity bit
- 1 or 2 stop bits

USARTn I/O Data Register – UDRn

Bit	7	6	5	4	3	2	1	0	
	RXBn[7:0]								UDRn (Read)
	TXBn[7:0]								UDRn (Write)
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

USART Control and Status Register A – UCSRnA

Bit	7	6	5	4	3	2	1	0	
	RXCn	TXCn	UDREn	FEn	DORn	UPEn	U2Xn	MPCMn	UCSRnA
Read/Write	R	R/W	R	R	R	R	R/W	R/W	
Initial Value	0	0	1	0	0	0	0	0	

- **Bit 7 – RXCn: USART Receive Complete**
- **Bit 6 – TXCn: USART Transmit Complete**
- **Bit 5 – UDREn: USART Data Register Empty**
- **Bit 4 – FEn: Frame Error**
- **Bit 3 – DORn: Data OverRun**
- **Bit 2 – UPEn: Parity Error**
- **Bit 1 – U2Xn: Double the USART Transmission Speed**
- **Bit 0 – MPCMn: Multi-Processor Communication Mode**

USARTn Control and Status Register B – UCSRnB

Bit	7	6	5	4	3	2	1	0	
	RXCIEn	TXCIEn	UDRIEn	RXENn	TXENn	UCSZn2	RXB8n	TXB8n	UCSRnB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7 – RXCIEn: RX Complete Interrupt Enable**
- **Bit 6 – TXCIEn: TX Complete Interrupt Enable**
- **Bit 5 – UDRIEn: USART Data Register Empty Interrupt Enable**
- **Bit 4 – RXENn: Receiver Enable**
- **Bit 3 – TXENn: Transmitter Enable**
- **Bit 2 – UCSZn2: Character Size**
- **Bit 1 – RXB8n: Receive Data Bit 8**
- **Bit 0 – TXB8n: Transmit Data Bit 8**

USART Baud Rate Registers – UBRRnL and UBRRnH

Bit	15	14	13	12	11	10	9	8	
	–	–	–	–	UBRRn[11:8]				UBRRnH
	UBRRn[7:0]								UBRRnL
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

USART Control and Status Register C – UCSRnC

Bit	7	6	5	4	3	2	1	0	
	–	UMSELn	UPMn1	UPMn0	USBSn	UCSZn1	UCSZn0	UCPOLn	UCSRnC
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	1	1	0	

- **Bit 6 – UMSELn: USART Mode Select**

This bit selects between Asynchronous and Synchronous mode of operation.

Table 77. UMSELn Bit Settings

UMSELn	Mode
0	Asynchronous Operation
1	Synchronous Operation

- **Bit 5:4 – UPMn1:0: Parity Mode**

Table 78. UPMn Bits Settings

UPMn1	UPMn0	Parity Mode
0	0	Disabled
0	1	(Reserved)
1	0	Enabled, Even Parity
1	1	Enabled, Odd Parity

- **Bit 3 – USBSn: Stop Bit Select**

Table 79. USBSn Bit Settings

USBSn	Stop Bit(s)
0	1-bit
1	2-bits

- **Bit 2:1 – UCSZn1:0: Character Size**

Table 80. UCSZn Bits Settings

UCSZn2	UCSZn1	UCSZn0	Character Size
0	0	0	5-bit
0	0	1	6-bit
0	1	0	7-bit
0	1	1	8-bit
1	0	0	Reserved
1	0	1	Reserved
1	1	0	Reserved
1	1	1	9-bit

- **Bit 0 – UCPOLn: Clock Polarity**

C Code Example[1]

```
#define FOSC 1843200// Clock Speed
#define BAUD 9600
#define MYUBRR FOSC/16/BAUD-1
void main( void )
{
...
USART_Init ( MYUBRR );
...
}
void USART_Init( unsigned int ubrr )
{
  /* Set baud rate */
  UBRRH = (unsigned char)(ubrr>>8);
  UBRRL = (unsigned char)ubrr;
  /* Enable receiver and transmitter */
  UCSRB = (1<<RXEN)|(1<<TXEN);
  /* Set frame format: 8data, 2stop bit */
  UCSRC = (1<<USBS)|(3<<UCSZ0);
}
```

C Code Example[1]

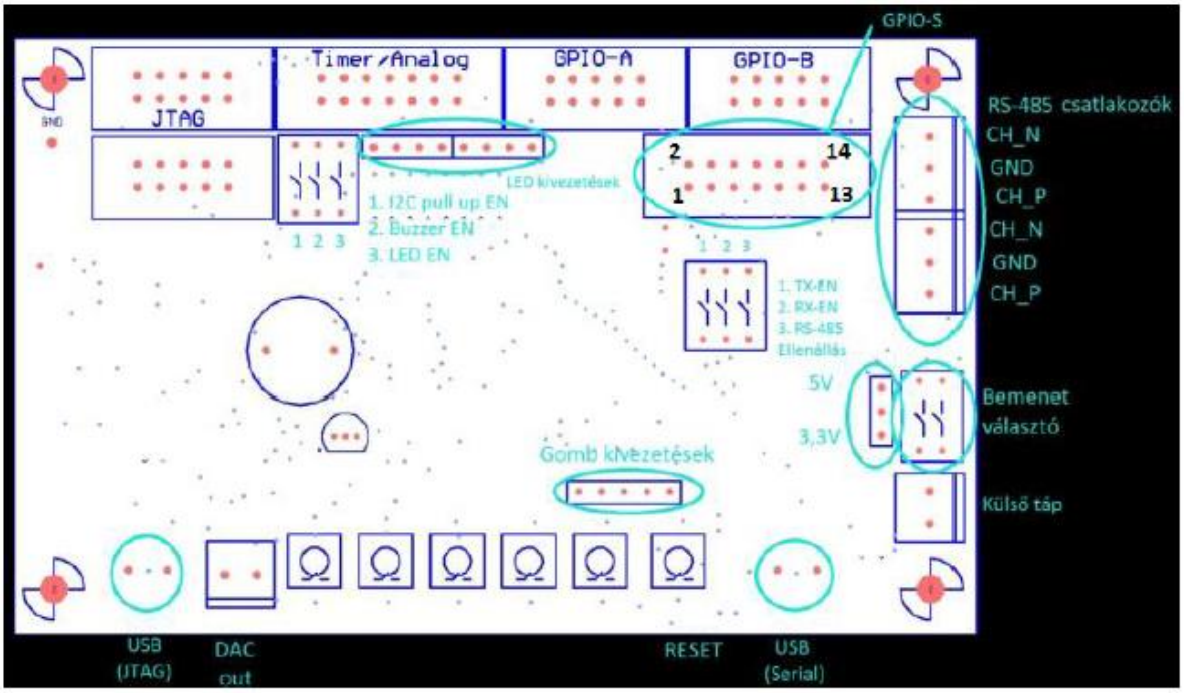
```
void USART_Transmit( unsigned char data )
{
  /* Wait for empty transmit buffer */
  while ( !( UCSRA & (1<<UDRE)) )
        ;
  /* Put data into buffer, sends the data */
  UDR = data;
}
```

C Code Example[1]

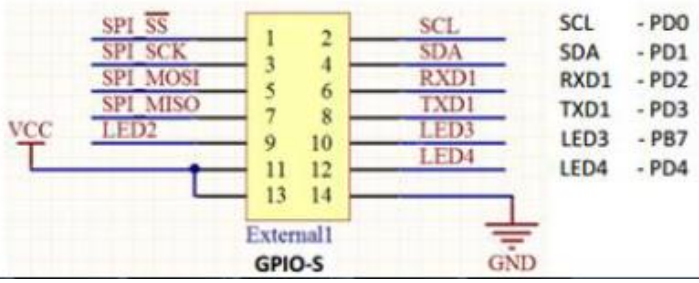
```
unsigned char USART_Receive( void )
{
  /* Wait for data to be received */
  while ( !(UCSRA & (1<<RXC)) )
        ;
  /* Get and return received data from buffer */
  return UDR;
}
```

19	$0024	USART0, RX	USART0, Rx Complete
20	$0026	USART0, UDRE	USART0 Data Register Empty
21	$0028	USART0, TX	USART0, Tx Complete

31	$003C(3)	USART1, RX	USART1, Rx Complete
32	$003E(3)	USART1, UDRE	USART1 Data Register Empty
33	$0040(3)	USART1, TX	USART1, Tx Complete



15. ábra – T-bird3 csatlakozó kiosztása

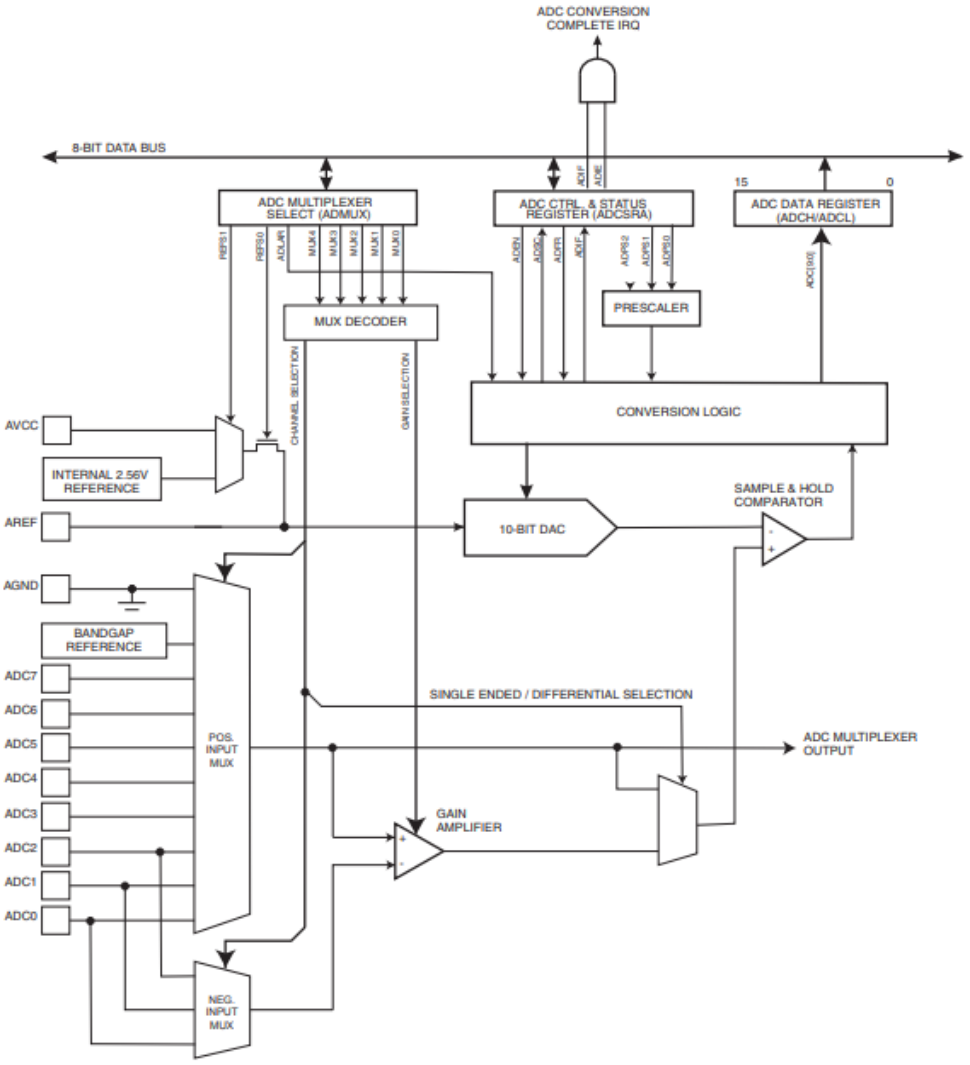


ADC

Analog to Digital Converter

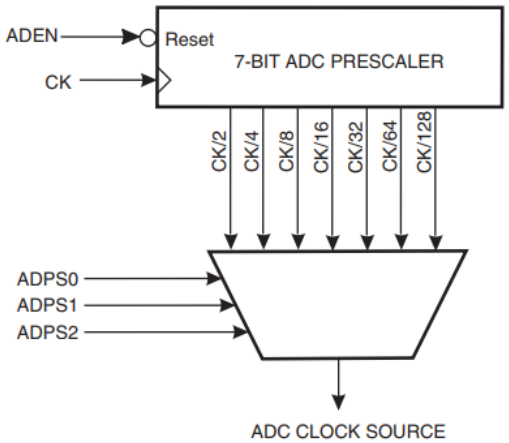
Features

- 10-bit Resolution
- 0.75 LSB Integral Non-linearity
- ±1.5 LSB Absolute Accuracy
- 13 µs - 260 µs Conversion Time
- Up to 15 kSPS at Maximum Resolution
- Eight Multiplexed Single Ended Input Channels
- Seven Differential Input Channels
- Two Differential Input Channels with Optional Gain of 10x and 200x
- Optional Left Adjustment for ADC Result Readout
- 0V - V_{CC} ADC Input Voltage Range
- 2.7V - V_{CC} Differential ADC Voltage Range
- Selectable 2.56V ADC Reference Voltage
- Free Running or Single Conversion Mode
- ADC Start Conversion by Auto Triggering on Interrupt Sources
- Interrupt on ADC Conversion Complete
- Sleep Mode Noise Canceler



Port Pin	Alternate Function
PF7	ADC7/TDI (ADC input channel 7 or JTAG Test Data Input)
PF6	ADC6/TDO (ADC input channel 6 or JTAG Test Data Output)
PF5	ADC5/TMS (ADC input channel 5 or JTAG Test Mode Select)
PF4	ADC4/TCK (ADC input channel 4 or JTAG Test ClocK)
PF3	ADC3 (ADC input channel 3)
PF2	ADC2 (ADC input channel 2)
PF1	ADC1 (ADC input channel 1)
PF0	ADC0 (ADC input channel 0)

ADC Prescaler



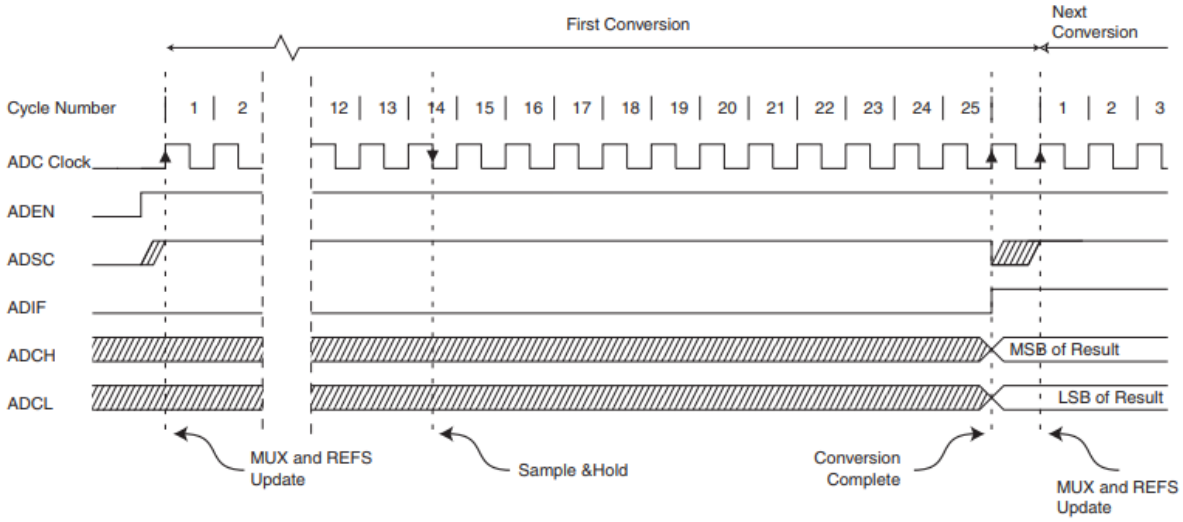
50-200kHz – 10bit

200kHz - <10bit

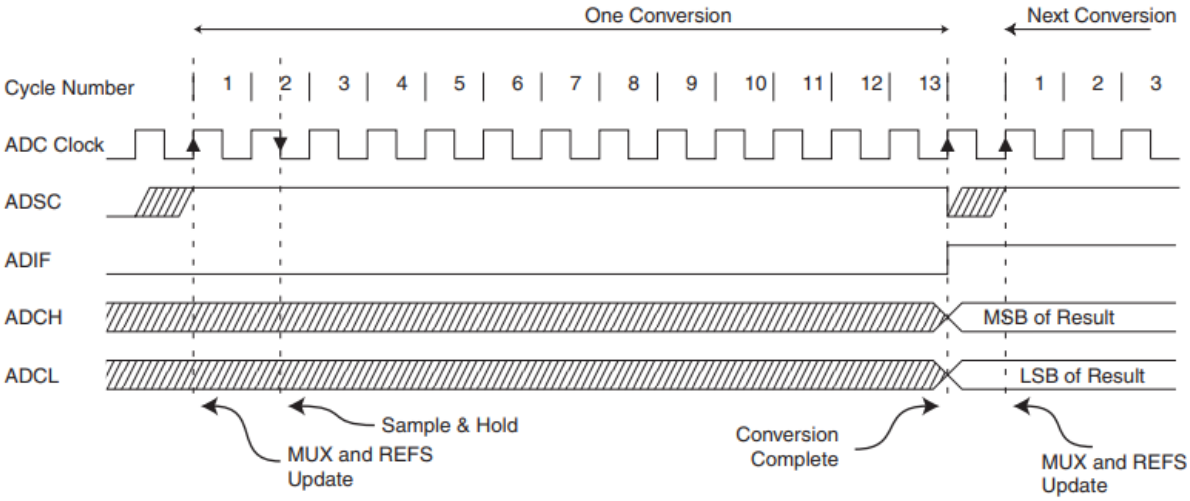
ADC Conversion Time

Condition	Sample & Hold (Cycles from Start of Conversion)	Conversion Time (Cycles)
First conversion	13.5	25
Normal conversions, single ended	1.5	13

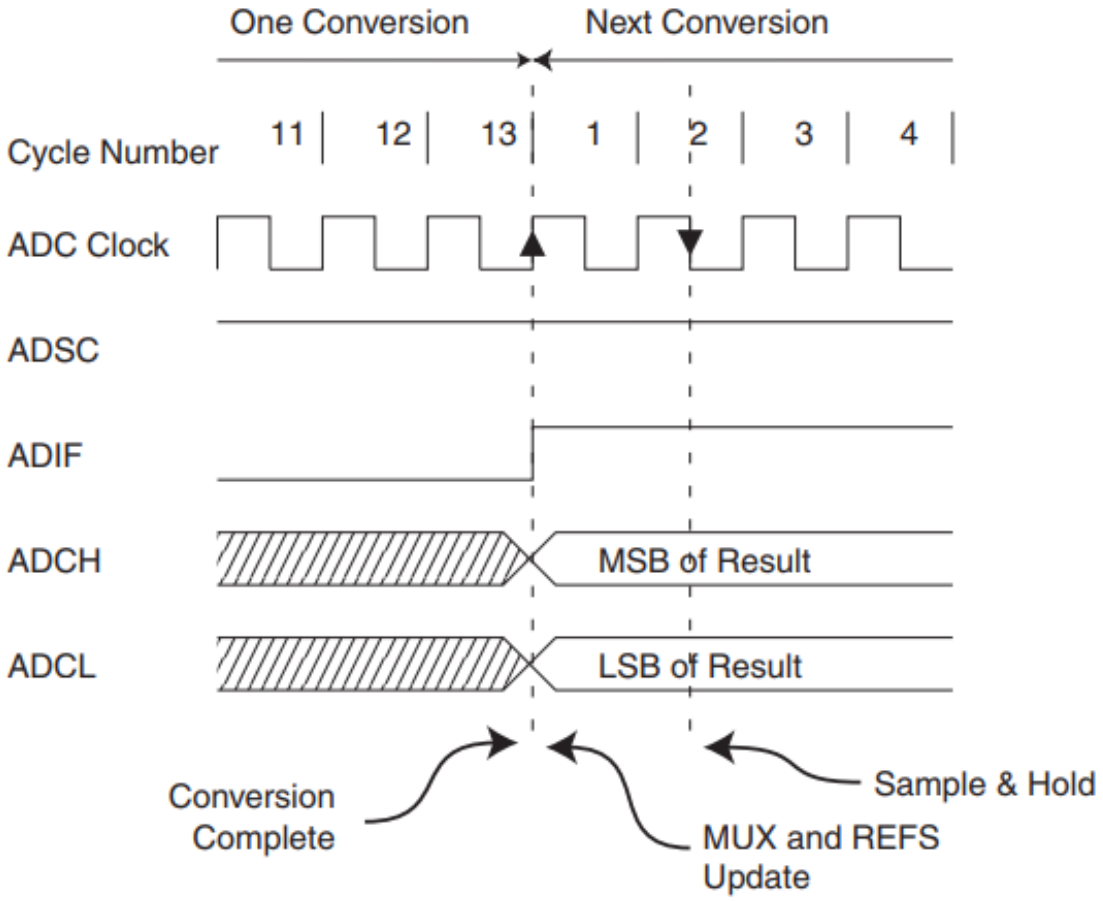
ADC Timing Diagram, First Conversion (Single Conversion Mode)



ADC Timing Diagram, Single Conversion



ADC Timing Diagram, Free Running Conversion

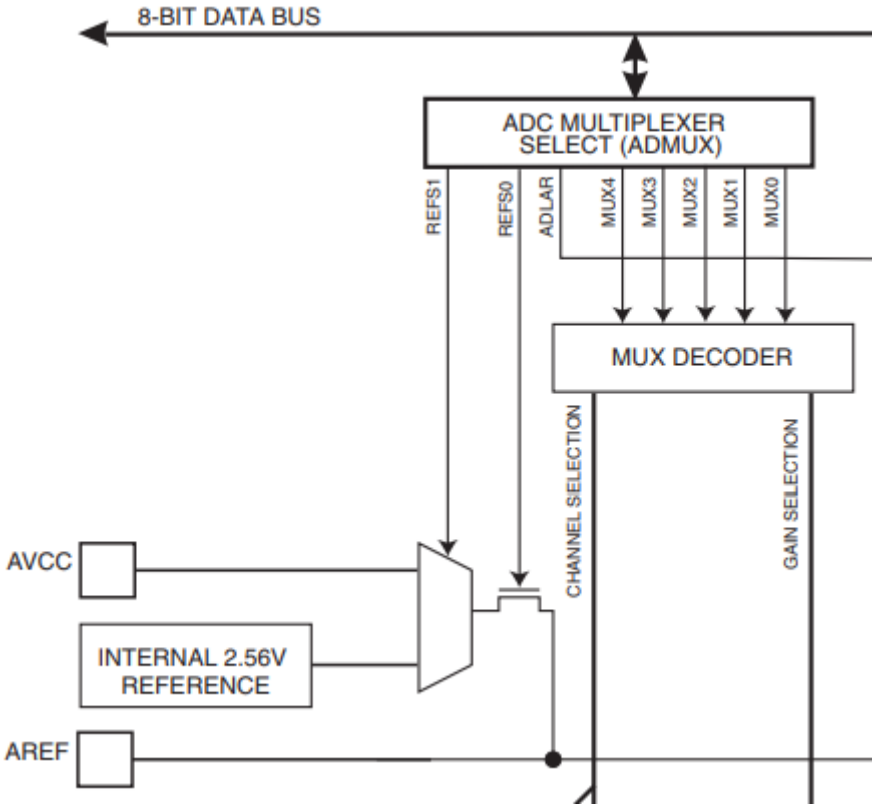


$$ADC = \frac{V_{IN} \cdot 1024}{V_{REF}}$$

$$V_{REF} = 4096mV$$

ADC Multiplexer Selection Register – ADMUX

Bit	7	6	5	4	3	2	1	0	
	REFS1	REFS0	ADLAR	MUX4	MUX3	MUX2	MUX1	MUX0	ADMUX
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	



- **Bit 7:6 – REFS1:0: Reference Selection Bits**

Table 97. Voltage Reference Selections for ADC

REFS1	REFS0	Voltage Reference Selection
0	0	AREF, Internal Vref turned off
0	1	AVCC with external capacitor at AREF pin
1	0	Reserved
1	1	Internal 2.56V Voltage Reference with external capacitor at AREF pin

- **Bit 5 – ADLAR: ADC Left Adjust Result**

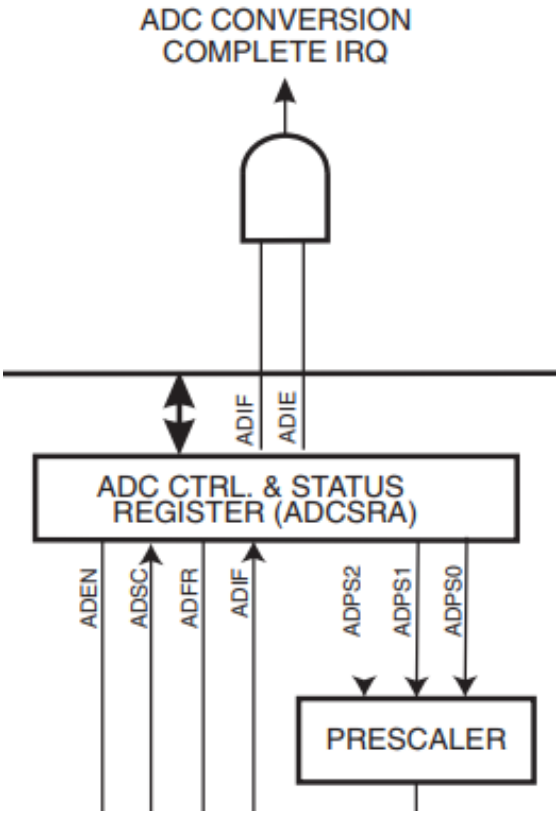
- **Bits 4:0 – MUX4:0: Analog Channel and Gain Selection Bits**

Input Channel and Gain Selections

MUX4..0	Single Ended Input	Positive Differential Input	Negative Differential Input	Gain
00000	ADC0	N/A		
00001	ADC1			
00010	ADC2			
00011	ADC3			
00100	ADC4			
00101	ADC5			
00110	ADC6			
00111	ADC7			

ADC Control and Status Register A – ADCSRA

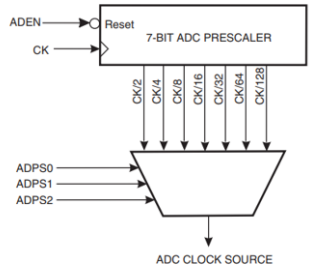
Bit	7	6	5	4	3	2	1	0	
	ADEN	ADSC	ADFR	ADIF	ADIE	ADPS2	ADPS1	ADPS0	ADCSRA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	



22	$002A	ADC	ADC Conversion Complete

- **Bit 7 – ADEN: ADC Enable**
- **Bit 6 – ADSC: ADC Start Conversion**
- **Bit 5 – ADFR: ADC Free Running Select**
- **Bit 4 – ADIF: ADC Interrupt Flag**
- **Bit 3 – ADIE: ADC Interrupt Enable**
- **Bits 2:0 – ADPS2:0: ADC Prescaler Select Bits**

ADC Prescaler



ADC Prescaler Selections

ADPS2	ADPS1	ADPS0	Division Factor
0	0	0	2
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

The ADC Data Register – ADCL and ADCH

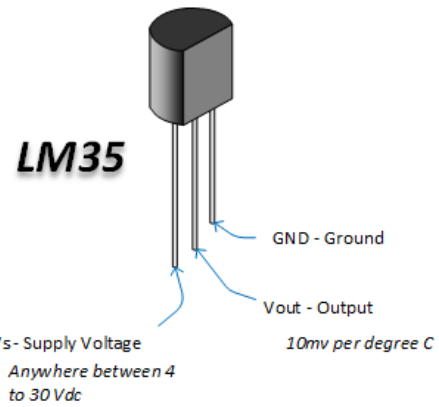
ADLAR = 0:

Bit	15	14	13	12	11	10	9	8	
	–	–	–	–	–	–	**ADC9**	**ADC8**	**ADCH**
	ADC7	**ADC6**	**ADC5**	**ADC4**	**ADC3**	**ADC2**	**ADC1**	**ADC0**	**ADCL**
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	R	R	R	R	
	R	R	R	R	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

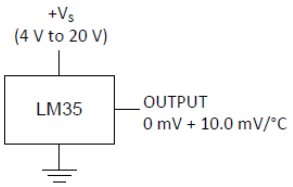
ADLAR = 1:

Bit	15	14	13	12	11	10	9	8	
	ADC9	**ADC8**	**ADC7**	**ADC6**	**ADC5**	**ADC4**	**ADC3**	**ADC2**	**ADCH**
	ADC1	**ADC0**	–	–	–	–	–	–	**ADCL**
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	R	R	R	R	
	R	R	R	R	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	

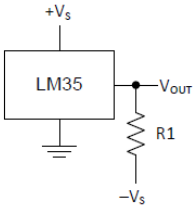
LM35



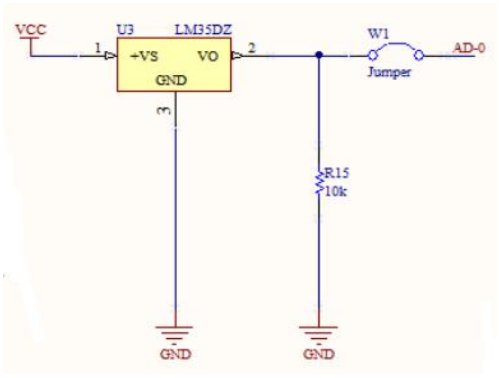
Basic Centigrade Temperature Sensor (2°C to 150°C)

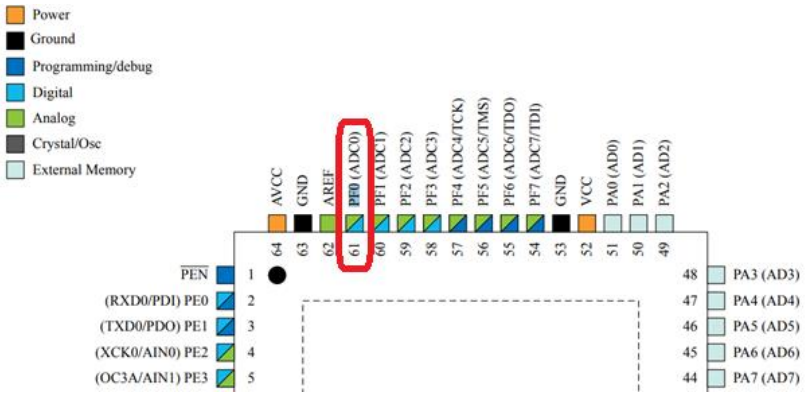


Full-Range Centigrade Temperature Sensor



Choose $R_1 = -V_S / 50\ \mu A$
V_{OUT} = 1500 mV at 150°C
V_{OUT} = 250 mV at 25°C
V_{OUT} = –550 mV at –55°C

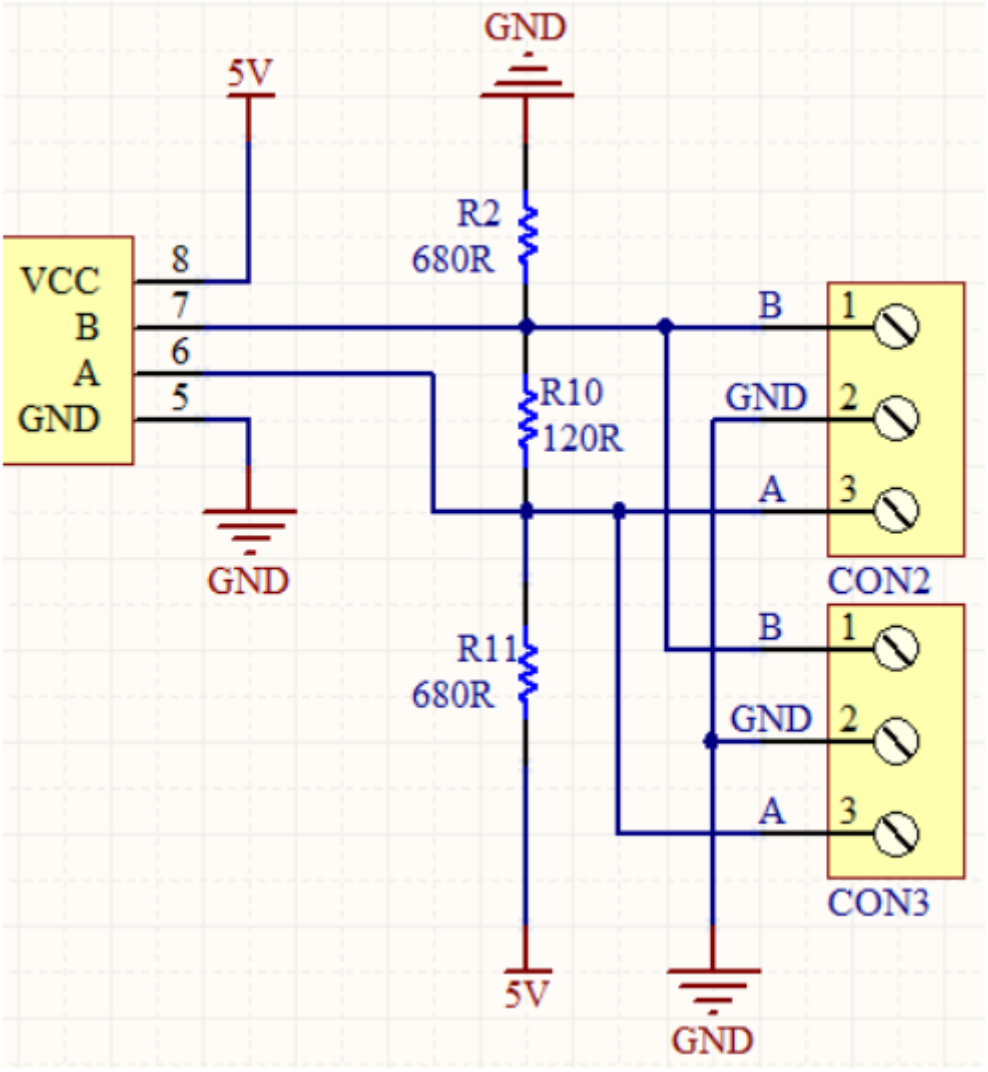




RS-485

RS-485 bus communication is based on UART and a SN75176 line driver IC. The system provides a bidirectional half-duplex transmission.

The maximum number of transmitters and receivers that can be connected to the bus is 32, which can be extended to 128 participants using signal repeaters. A big advantage is that the devices are connected in parallel to the system, where the first and last device are terminated with a 120Ω resistor each (a terminating resistor is inserted between the RX and TX lines). The communication signal lines have a "totem pole" output, which means that the lines are pulled to ground and to supply with a resistor each.



RS-485 has a maximum data rate of 35Mbit/sec (10Mbit/sec) up to 10m, but only 100kbit/sec at 1200m.

SN75176A Differential Bus Transceiver

1 Features

- Bidirectional Transceiver
- Meets or Exceeds the Requirements of ANSI Standards EIA/TIA-422-B and ITU Recommendations V.11
- Designed for Multipoint Transmission on Long Bus Lines in Noisy Environments
- 3-State Driver and Receiver Outputs
- Individual Driver and Receiver Enables
- Wide Positive and Negative Input/Output Bus Voltage Ranges
- Driver Output Capability ±60 mA Max
- Thermal-Shutdown Protection
- Driver Positive-Current Limiting and Negative-Current Limiting
- Receiver Input Impedance 12 kΩ Min
- Receiver Input Sensitivity ±200 mV
- Receiver Input Hysteresis 50 mV Typ
- Operates From Single 5-V Supply
- Lower Power Requirements

2 Applications

- Low Speed RS485 communication (5 Mbps or less)
- For 10 Mbps, use SN75176B

3 Description

The SN75176A differential bus transceiver is a monolithic integrated circuit designed for bidirectional data communication on multipoint bus-transmission lines. It is designed for balanced transmission lines and meets ANSI Standard EIA/TIA-422-B and ITU Recommendation V.11.

The SN75176A combines a 3-state differential line driver and a differential input line receiver, both of which operate from a single 5-V power supply. The driver and receiver have active-high and active-low enables, respectively, that can be externally connected together to function as a direction control. The driver differential outputs and the receiver differential inputs are connected internally to form differential input/output (I/O) bus ports that are designed to offer minimum loading to the bus whenever the driver is disabled or VCC = 0. These ports feature wide positive and negative common-mode voltage ranges making the device suitable for party-line applications.

The driver is designed to handle loads up to 60 mA of sink or source current. The driver features positive- and negative-current limiting and thermal shutdown for protection from line fault conditions. Thermal shutdown is designed to occur at a junction temperature of approximately 150°C. The receiver features a minimum input impedance of 12 kΩ, an input sensitivity of ±200 mV, and a typical input hysteresis of 50 mV.

The SN75176A can be used in transmission-line applications employing the SN75172 and SN75174 quadruple differential line drivers and SN75173 and SN75175 quadruple differential line receivers.

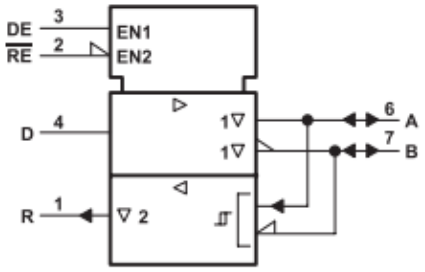
The SN75176A is characterized for operation from 0°C to 70°C.

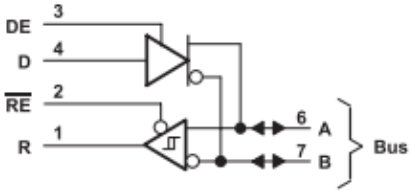
Device Information(1)

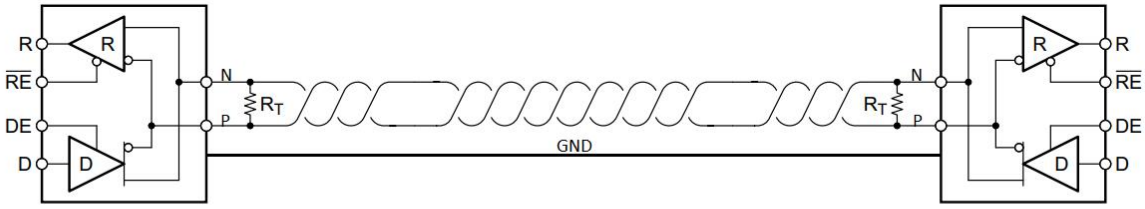
PART NUMBER	PACKAGE (PIN)	BODY SIZE (NOM)
SN75176A	SOIC (8)	4.90 mm × 3.91 mm
	PDIP (8)	9.81 mm × 6.35 mm

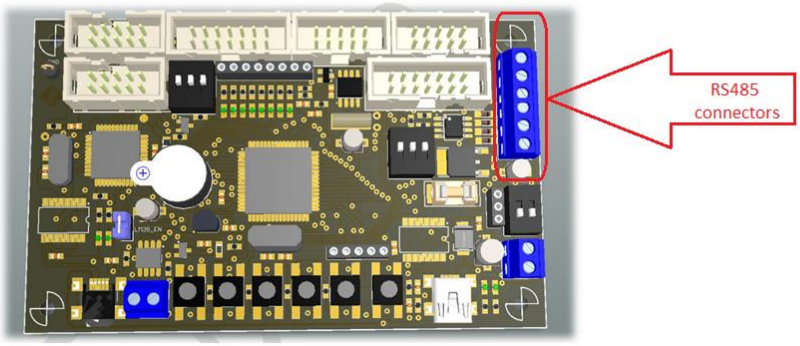
(1) For all available packages, see the orderable addendum at the end of the datasheet.

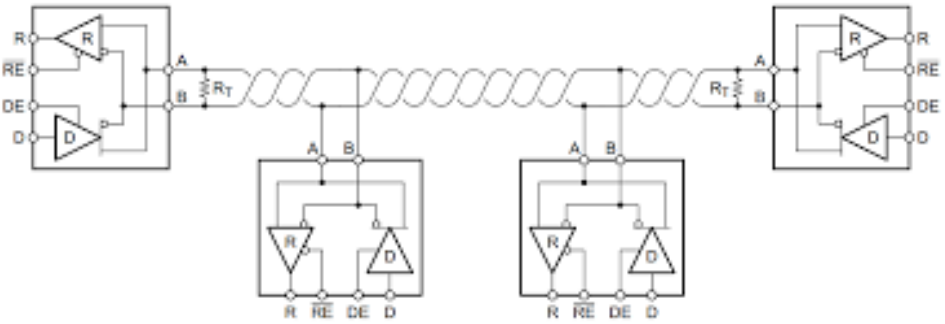
Simplified Schematics

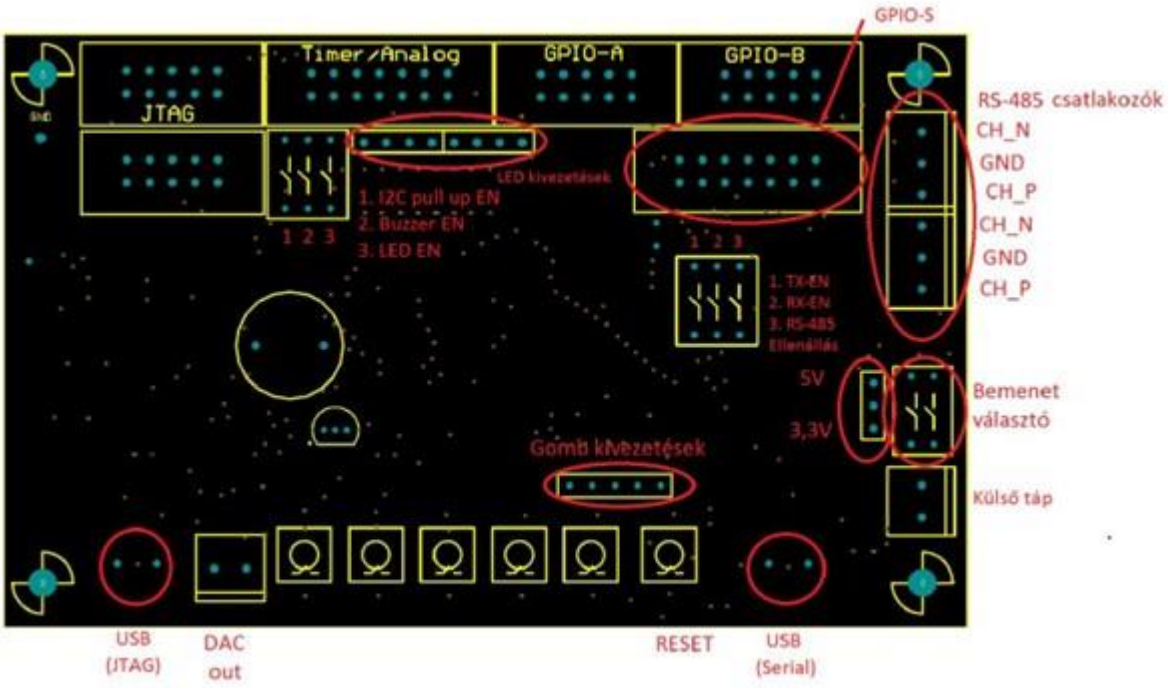




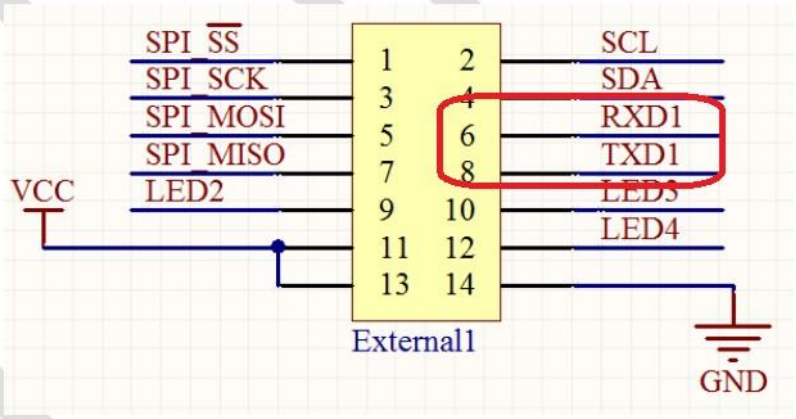

R
RE
DE
D
N
P
RT
GND


RS485
connectors


R
RE
DE
D
A
B
RT


GPIO-S
Timer/Analog
GPIO-A
GPIO-B
JTAG
LED kivezetések
1. I2C pull up EN
2. Buzzer EN
3. LED EN
1. TX-EN
2. RX-EN
3. RS-485 Ellenállás
RS-485 csatlakozók
CH_N
GND
CH_P
CH_N
GND
CH_P
5V
3,3V
Bemenet választó
Gomb kivezetések
Külső táp
USB (JTAG)
DAC out
RESET
USB (Serial)

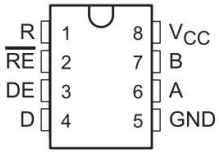
GPIO-S: Általános célú IO kivezetés. (bővebben a kapcsolási rajzon)



SPI_SS	- PB0	SCL	- PD0
SPI_SCK	- PB1	SDA	- PD1
SPI_MOSI	- PB2	RXD1	- PD2
SPI_MISO	- PB3	TXD1	- PD3
LED2	- PB6	LED3	- PB7
		LED4	- PD4

5 Pin Configuration and Functions

D OR P PACKAGE
(TOP VIEW)



Pin Functions

PIN		TYPE	DESCRIPTION	
NAME	NO.			
R	1	O	Logic Data Output from RS-485 Receiver	**PD2 - RXD1**
$\overline{RE}$	2	I	Receive Enable (active low)	**PC7 - OUT**
DE	3	I	Driver Enable (active high)	**PE2 - OUT**
D	4	I	Logic Data Input to RS-485 Driver	**PD3 - TXD1**
GND	5	—	Device Ground Pin	
A	6	I/O	RS-422 or RS-485 Data Line	
B	7	I/O	RS-422 or RS-485 Data Line	
V_{CC}	8	—	Power Input. Connect to 5-V Power Source.	

R	RxD1	PD2	UART1 Rx
D	TxD1	PD3	UART1 Tx
/RE	GPIO-F	PC7	OUT
DE	Ain0	PE2	OUT

	/RE	DE
Receive	H-1	H-1
Transmit	L-0	L-0

```
#define RS_RE_DDR   DDRC
#define RS_RE_PORT  PORTC
#define RS_RE_PIN   PC7

#define RS_DE_DDR   DDRE
#define RS_DE_PORT  PORTE
#define RS_DE_PIN   PE2
```

```
void RS_drivebus(void)
{
  RS_RE_PORT |= (1<<RS_RE_PIN);
  RS_DE_PORT |= (1<<RS_DE_PIN);
}

void RS_releasebus(void)
{
  RS_DE_PORT  &= ~(1<<RS_DE_PIN);
  RS_RE_PORT  &= ~(1<<RS_RE_PIN);
}
```