EEL4744

Record this lecture!

Menu

- References:
  - > Serial Communication
- Asynchronous Data Communication
  - > Communication Terminology
  - > ASCII Communication Codes
  - > Serial Communication Principles
  - > XMEGA SCI Serial I/O Hardware
  - > XMEGA SCI Programming Example



sci_polling.asm

See docs/examples on web-site: **doc8331, ATxmega128A1Udef.inc, for GCPU++: sci_inchar_outchar.asm. Demo video (NOT on website): SCI_Polling.asm, SCI_polling_Red.asm**

1

EEL4744

Serial Communication

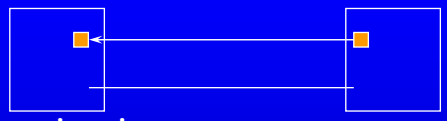
- **Serial Communication** is the process of sending one **bit at a time** over a **communication channel**
  - >As opposed to **parallel communication** (sending several bits at the same time)
- Some of the many types of serial communication and serial communication channels will be discussed

2

EEL4744

Communication Terminology

- Simplex Communication (one way)



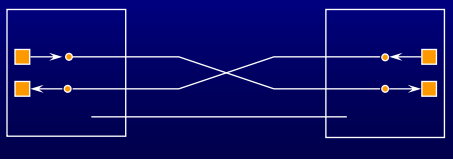


- Duplex Communication
  - > Half-duplex





  - > Full-duplex





University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo
3

3

EEL4744

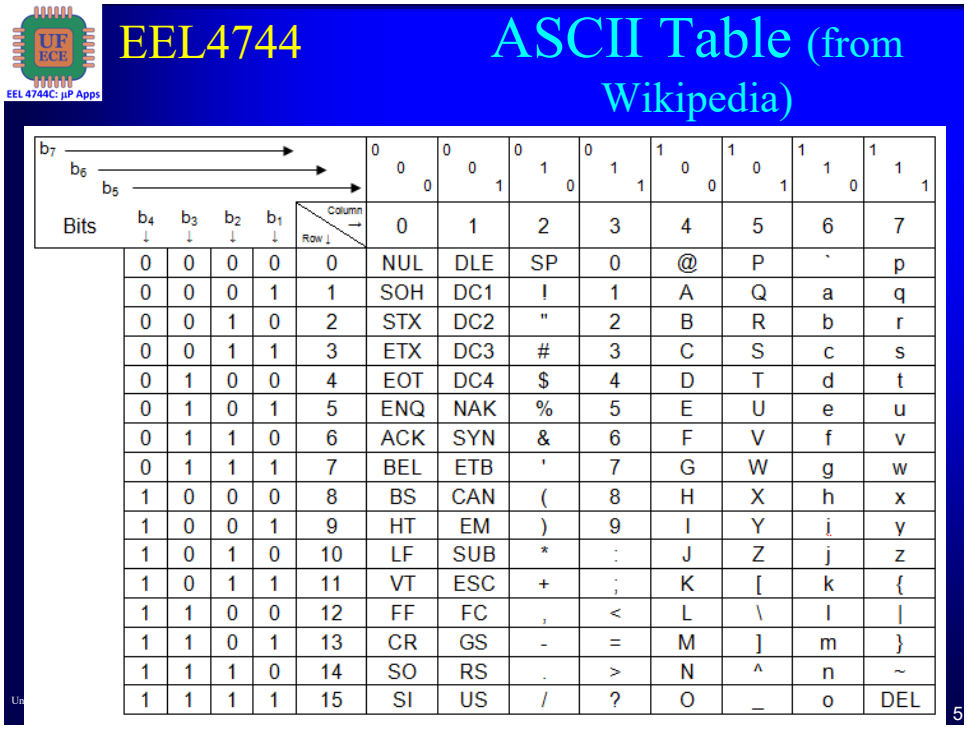
ASCII Communication Codes

- ASCII (**A**merican **S**tandard **C**ommittee for **I**nformation **I**nterchange) codes (see asciitable.com)
- Error Detection
  - >Parity bits
  - >Parity algorithms
    - » Even parity
    - » Odd parity
    - » Mark parity
    - » Space parity

[Example]	A	C
7-bit ASCII	100 0001	100 0011
» Even parity	0100 0001	1100 0011
» Odd parity	1100 0001	0100 0011
» Mark parity	1100 0001	1100 0011
» Space parity	0100 0001	0100 0011

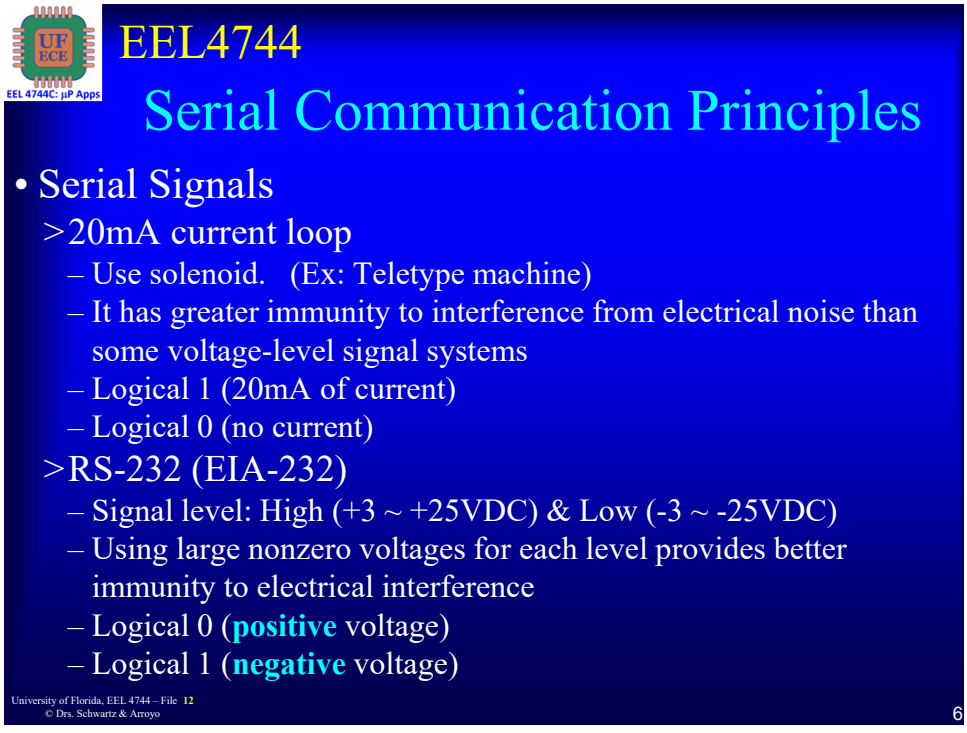
University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo
4

4

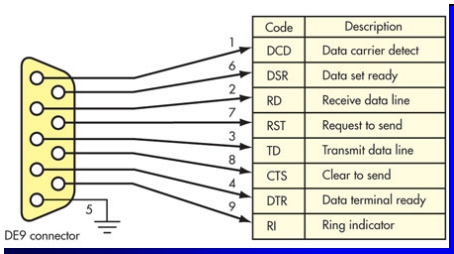


b_7	b_6	b_5	Bits b_4	b_3	b_2	b_1	Row ↓ / Column →	0 (000)	1 (001)	2 (010)	3 (011)	4 (100)	5 (101)	6 (110)	7 (111)
0	0	0	0	0	0	0	0	NUL	DLE	SP	0	@	P	`	p
0	0	0	0	1	0	0	1	SOH	DC1	!	1	A	Q	a	q
0	0	0	1	0	0	1	2	STX	DC2	"	2	B	R	b	r
0	0	0	1	1	0	1	3	ETX	DC3	#	3	C	S	c	s
0	0	1	0	0	1	0	4	EOT	DC4	$	4	D	T	d	t
0	0	1	0	1	1	0	5	ENQ	NAK	%	5	E	U	e	u
0	0	1	1	0	1	1	6	ACK	SYN	&	6	F	V	f	v
0	0	1	1	1	1	1	7	BEL	ETB	'	7	G	W	g	w
0	1	0	0	0	1	0	8	BS	CAN	(	8	H	X	h	x
0	1	0	0	1	1	0	9	HT	EM	)	9	I	Y	i	y
0	1	0	1	0	1	1	10	LF	SUB	*	:	J	Z	j	z
0	1	0	1	1	1	1	11	VT	ESC	+	;	K	[	k	{
0	1	1	0	0	1	0	12	FF	FC	,	<	L	\	l	|
0	1	1	0	1	1	0	13	CR	GS	-	=	M	]	m	}
0	1	1	1	0	1	1	14	SO	RS	.	>	N	^	n	~
0	1	1	1	1	1	1	15	SI	US	/	?	O	_	o	DEL

5



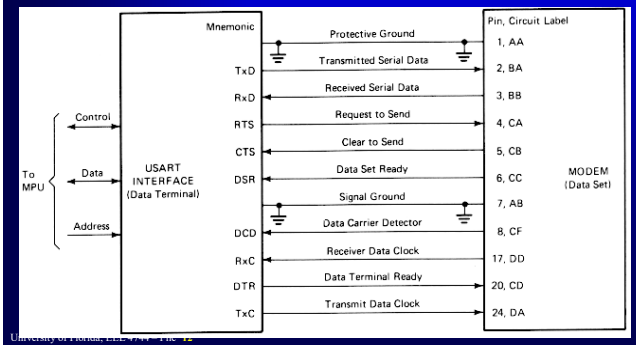
6



Code	Description
DCD	Data carrier detect
DSR	Data set ready
RD	Receive data line
RST	Request to send
TD	Transmit data line
CTS	Clear to send
DTR	Data terminal ready
RI	Ring indicator

RS232C Interface

- DTE: Data Terminal Equipment
- DCE: Data Communication Equipment



Pin	Name	Description
1	GND	Shield Ground
2	TXD	Transmit Data
3	RXD	Receive Data
4	RTS	Request to Send
5	CTS	Clear to Send
6	DSR	Data Set Ready
7	GND	System Ground
8	CD	Carrier Detect
9	-	Test pin
10	-	Test pin
11	STF	Select Transmit Channel
12	S.CD	Secondary Carrier Detect
13	S.CTS	Secondary Clear to Send
14	S.TXD	Secondary Transmit Data
15	TCK	Transmit Clock
16	S.RXD	Secondary Receive Data
17	RCK	Receive Clock
18	LL	Local Loopback
19	S.RTS	Secondary Request to Send
20	DTR	Data Terminal Ready
21	RL	Remote Loopback
22	RI	Ring Indicator
23	DSR	Signal Rate Selector
24	XCK	Auxiliary Clock
25	TI	Test Indicator

© Drs. Schwartz & Arroyo

7

7

EEL 4744C: µP Apps

EEL4744 Serial Communication Principles

- Serial Signal Format
  - > Start bit: Space
  - > 8 data bits, LSB first
  - > Stop bit: Mark
  - > Idle: Mark
- Baud Rate: ~Hz = bits/sec = f
  - > Bit time, T = 1/f
    - ex: 300Hz => 1/300 sec/bit



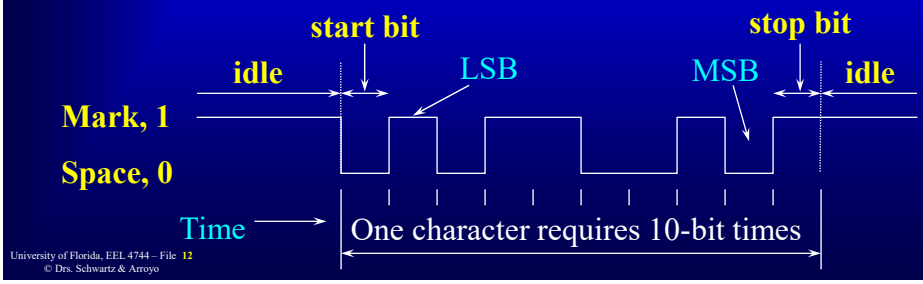
Telegraph Key



Player Piano

Telegraph Key:
https://youtu.be/BbTo0na6NPw

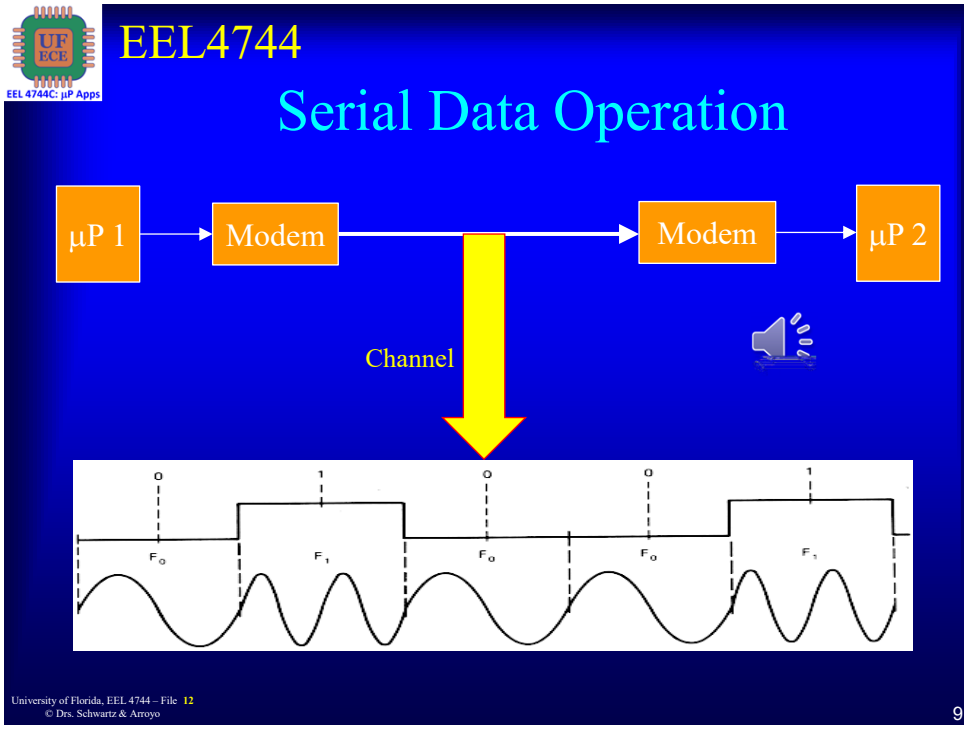
Player Piano:
- https://youtu.be/aseMAEctM1s
- https://youtu.be/hpOiSMX1ybw
- https://youtu.be/ZrNQlJZkKTg



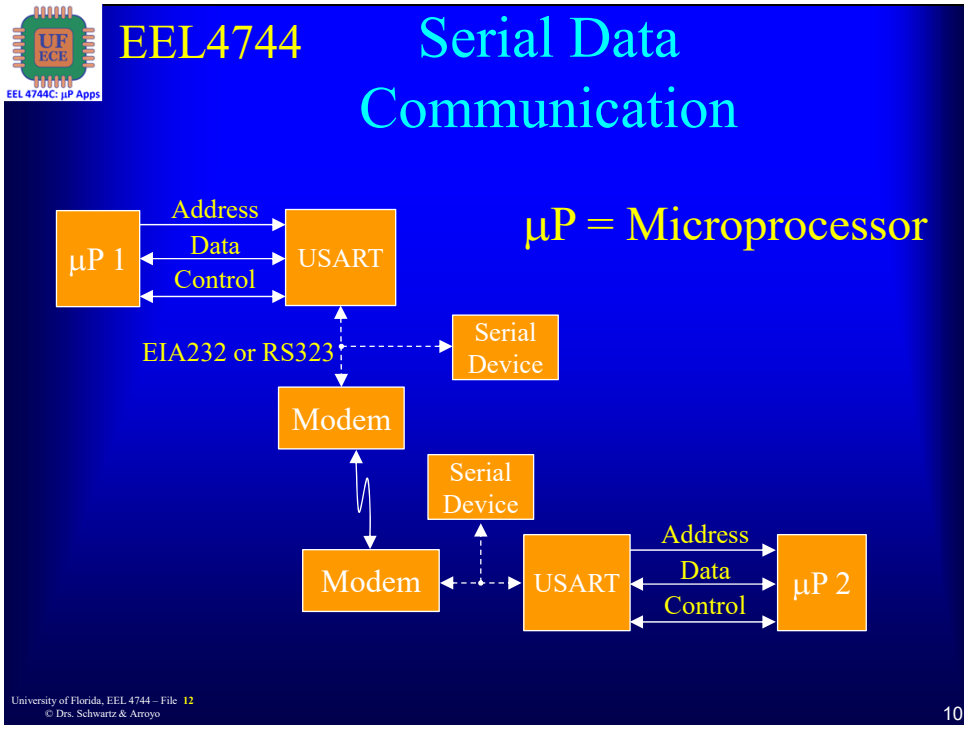
University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo

8

8


EEL4744
Serial Data Operation
µP 1
Modem
Modem
µP 2
Channel
University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo
9

9


EEL4744
Serial Data Communication
µP = Microprocessor
µP 1
Address
Data
Control
USART
EIA232 or RS323
Serial Device
Modem
Serial Device
Modem
USART
Address
Data
Control
µP 2
University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo
10

10

EEL4744

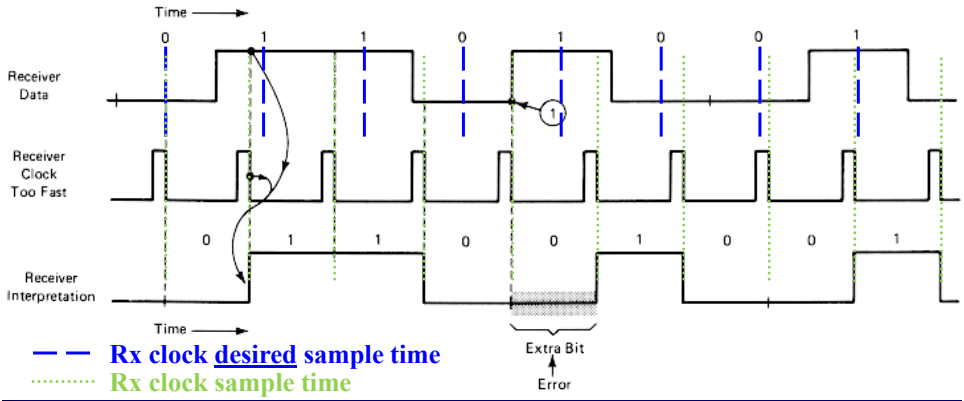
Serial Communication Principles

- UART (**U**niversal **A**synchronous **R**eceiver **T**ransmitter) hardware
  - > **USART** = **U**niversal **S**ynchronous/**A**synchronous **R**eceiver **T**ransmitter
- Serial Errors
  - >Timing errors
  - >Framing errors
  - >Overrun errors

11

EEL4744

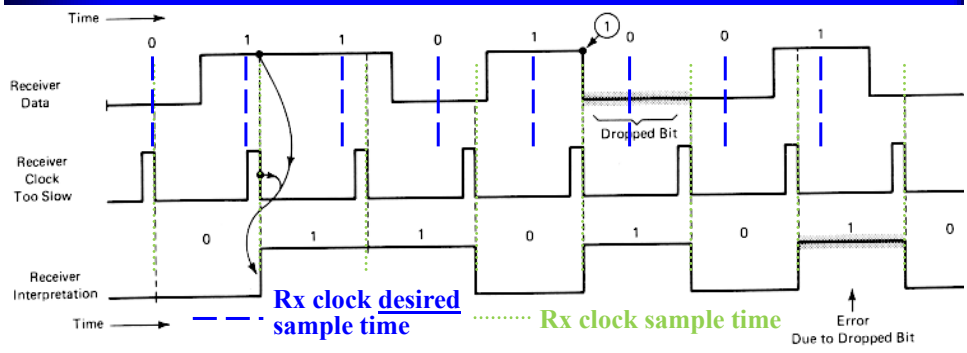
Synchronous Data Underflow [Doty: 300]



12

EEL4744

Synchronous Data Overflow [Doty: 301]

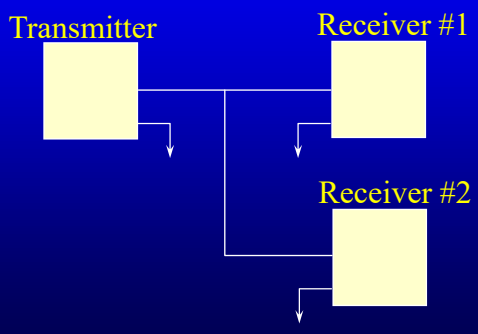


13

EEL4744

Serial Communication Principles

- Serial Transmission Rate
  - >Bit rate (or Baud rate) = rate in bits per second
  - >Waveform distortion
- Receiver Wake-up
  - >Messages
  - >Purpose of wake-up
  - >Wake-up triggering



14

EEL4744

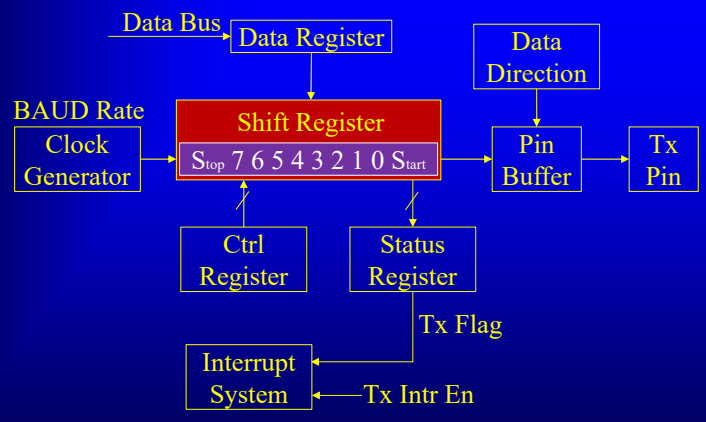
SCI: (Asynchronous) Serial Communications Interface

- Most microprocessors have at least one SCI
  - >Full-duplex two-wire asynchronous serial port (UART)
  - >Receiver and transmitter are independent, i.e., can run simultaneously

15

EEL4744

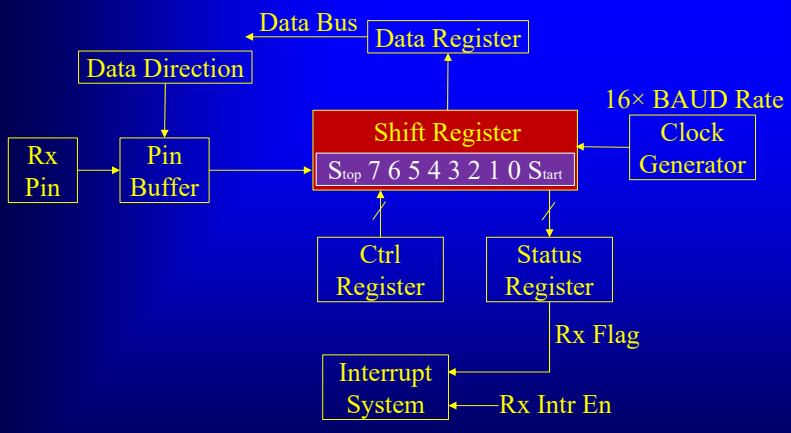
SCI Transmitter Block Diagram



16

EEL4744

SCI Receiver Block Diagram



University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo
17

17

EEL4744

Your XMEGA UARTs (SCI)

- Each of 4 ports (C-F) has two UARTs for a total of **8 asynchronous serial ports**
  - >Search for **USART** in the include file
    - – Synchronous serial communication (the "S" in USART) in most microcontrollers (including the XMEGA) is called **SPI** (synchronous peripheral interface)
      - We will talk about SPI later in the semester
      - Our XMEGA has **4 SPI ports**, one on each of Ports C-F

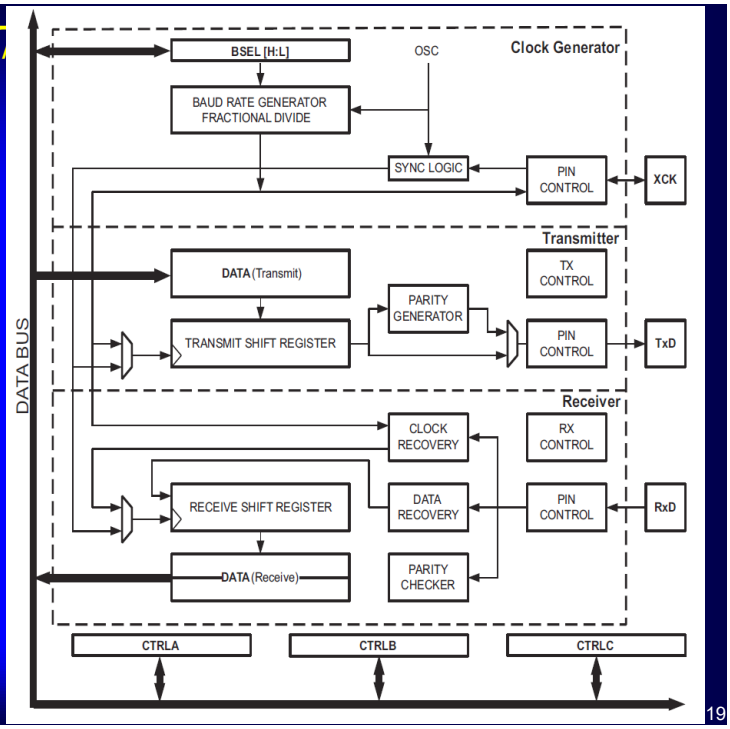
University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo
18

18

EEL47

See doc8331, Figure 23-1

XMEGA USART Block Diagram



University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo

19

19

EEL4744

XMEGA Baud Rate Generation

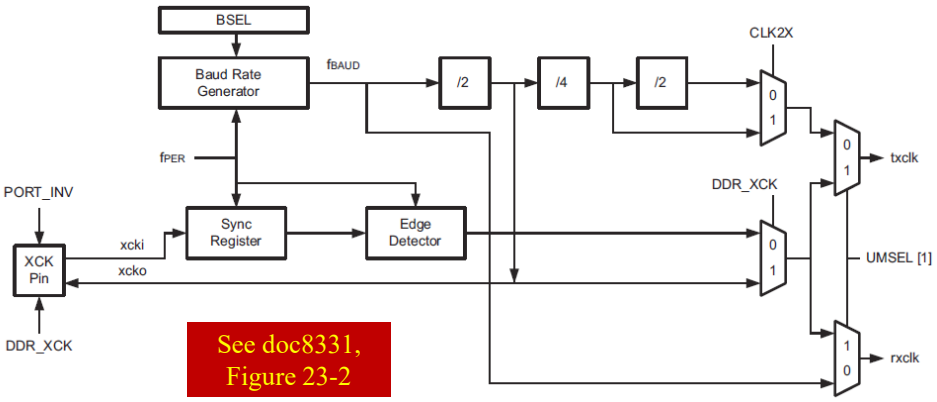
- The clock generator includes a **fractional** baud rate generator that is able to generate a wide range of USART baud rates from any system clock frequencies
- Can use baud rate generator **OR** an external transfer clock pin
  - >External transfer clock is **ONLY** used in synchronous mode

University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo

20

20

EEL4744 XMEGA Baud Rate Generation



See doc8331, Figure 23-2

- Notice that the receiver clock is **16 times faster** than the transmitter clock
- Baud rate is transfer rate in bits per second (bps)

21

EEL4744 XMEGA Baud Rate Generation

See doc8331, Section 23.3.1

- f_{BAUD} is determined by the period setting (BSEL), a scale setting (BSCALE), and the peripheral clock frequency (f_{PER})
  - > By default, f_{PER} = **2MHz** for the uPAD's XMEGA
  - > BSEL set between 0 and 4095
  - > BSCALE (optional scaling) set between -7 & +7
    - BSCALE ↑ or ↓ baud rate slightly for fractional baud rate scaling
  - > Asynchronous normal speed mode (CLK2X = 0) below

$f_{PER} = 2\text{MHz}$

Baud Rate Calculator:
https://mil.ufl.edu/4744/software/xmega_usart_baud_calc.xlsx

$$f_{BAUD} \le \frac{f_{PER}}{16}$$

$BSCALE \ge 0$

$$f_{BAUD} = \frac{f_{PER}}{2^{BSCALE} \bullet 16 \bullet (BSEL + 1)}$$

$$BSEL = \frac{f_{PER}}{2^{BSCALE} \bullet 16 \bullet f_{BAUD}} - 1$$

$BSCALE < 0$

$$f_{BAUD} = \frac{f_{PER}}{16 \bullet \left[\left(2^{BSCALE} \bullet BSEL\right) + 1\right]}$$

$$BSEL = \frac{1}{2^{BSCALE}}\left(\frac{f_{PER}}{16 \bullet f_{BAUD}} - 1\right)$$

22

EEL4744

XMEGA Baud Rate Calculations

See doc8331, Table 23-1

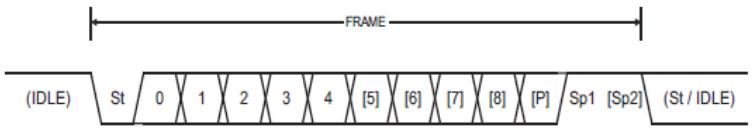
Operating mode	Conditions	Baud rate[1] calculation	BSEL value calculation
Asynchronous normal speed mode (CLK2X = 0)	BSCALE ≥ 0 $f_{BAUD} \le \frac{f_{PER}}{16}$	$f_{BAUD} = \frac{f_{PER}}{2^{BSCALE} \cdot 16(BSEL+1)}$	$BSEL = \frac{f_{PER}}{2^{BSCALE} \cdot 16 f_{BAUD}} - 1$
	BSCALE < 0 $f_{BAUD} \le \frac{f_{PER}}{16}$	$f_{BAUD} = \frac{f_{PER}}{16((2^{BSCALE} \cdot BSEL)+1)}$	$BSEL = \frac{1}{2^{BSCALE}}\left(\frac{f_{PER}}{16 f_{BAUD}} - 1\right)$
Asynchronous double speed mode (CLK2X = 1)	BSCALE ≥ 0 $f_{BAUD} \le \frac{f_{PER}}{8}$	$f_{BAUD} = \frac{f_{PER}}{2^{BSCALE} \cdot 8 \cdot (BSEL+1)}$	$BSEL = \frac{f_{PER}}{2^{BSCALE} \cdot 8 f_{BAUD}} - 1$
	BSCALE < 0 $f_{BAUD} \le \frac{f_{PER}}{8}$	$f_{BAUD} = \frac{f_{PER}}{8((2^{BSCALE} \cdot BSEL)+1)}$	$BSEL = \frac{1}{2^{BSCALE}}\left(\frac{f_{PER}}{8 f_{BAUD}} - 1\right)$
Synchronous and master SPI mode	$f_{BAUD} < \frac{f_{PER}}{2}$	$f_{BAUD} = \frac{f_{PER}}{2 \cdot (BSEL+1)}$	$BSEL = \frac{f_{PER}}{2 f_{BAUD}} - 1$

23

EEL4744

XMEGA Frame Formats

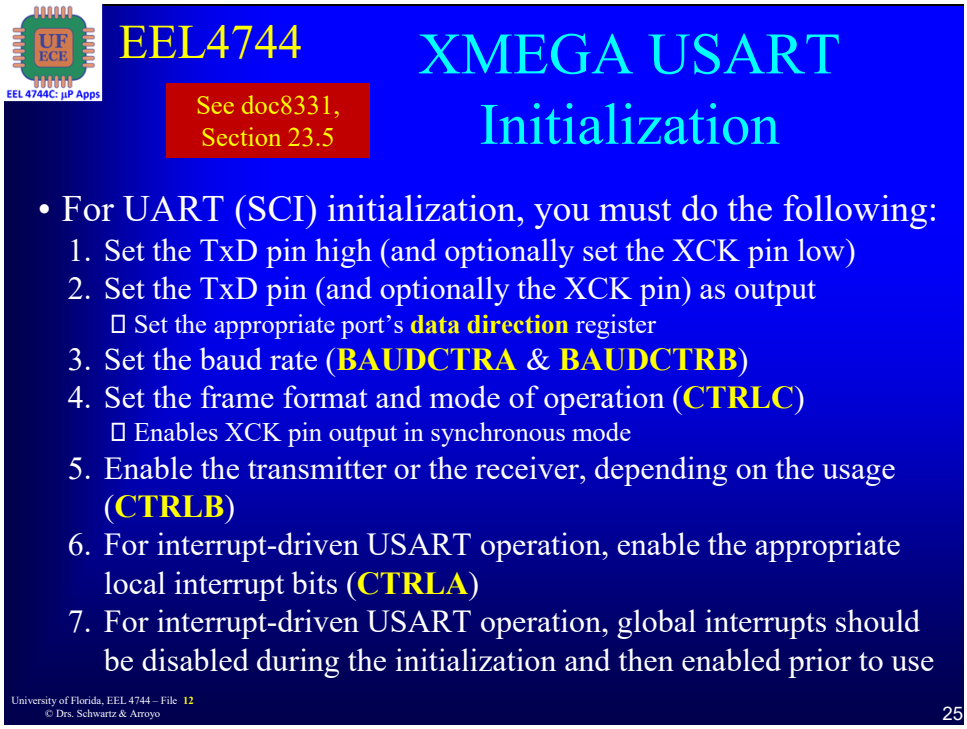
- A serial frame consists of one character of data bits with synchronization bits (start and stop bits) and an optional parity bit for error checking
  - >1 start bit
  - >None, even, or odd parity bit
  - > 5, 6, 7, 8, or 9 data bits
  - > 1 or 2 stop bits



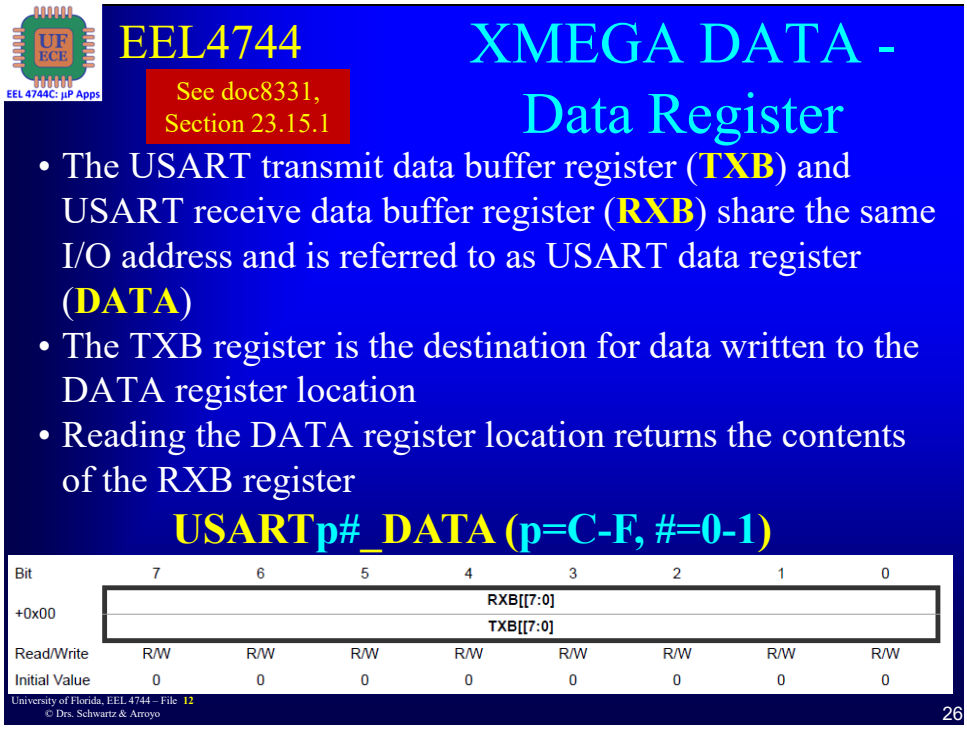
See doc8331, Figure 23-5

St	:	Start bit, always low.
(n)	:	Data bits (0 to 8).
P	:	Parity bit, may be odd or even.
Sp	:	Stop bit, always high.
IDLE	:	No transfers on the communication line (RxD or TxD). The IDLE state is always high.

24


EEL4744
See doc8331, Section 23.5
XMEGA USART Initialization
• For UART (SCI) initialization, you must do the following:
1. Set the TxD pin high (and optionally set the XCK pin low)
2. Set the TxD pin (and optionally the XCK pin) as output
Set the appropriate port's data direction register
3. Set the baud rate (BAUDCTRA & BAUDCTRB)
4. Set the frame format and mode of operation (CTRLC)
Enables XCK pin output in synchronous mode
5. Enable the transmitter or the receiver, depending on the usage (CTRLB)
6. For interrupt-driven USART operation, enable the appropriate local interrupt bits (CTRLA)
7. For interrupt-driven USART operation, global interrupts should be disabled during the initialization and then enabled prior to use
University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo
25

25

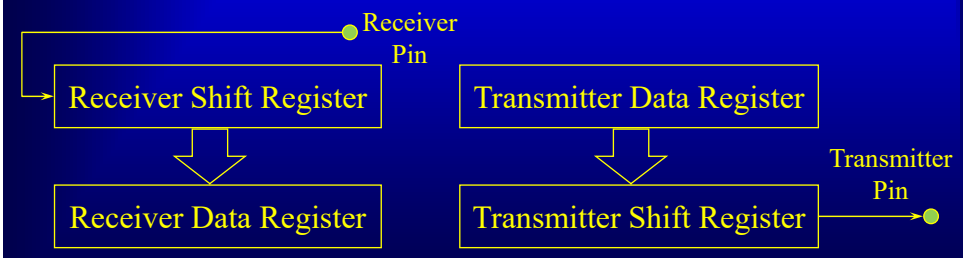

EEL4744
See doc8331, Section 23.15.1
XMEGA DATA - Data Register
• The USART transmit data buffer register (TXB) and USART receive data buffer register (RXB) share the same I/O address and is referred to as USART data register (DATA)
• The TXB register is the destination for data written to the DATA register location
• Reading the DATA register location returns the contents of the RXB register
USARTp#_DATA (p=C-F, #=0-1)
Bit 7 6 5 4 3 2 1 0
+0x00 RXB[[7:0]] TXB[[7:0]]
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
University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo
26

26

EEL4744

XMEGA SCI Double-buffering

- SCI Data Register, **USARTp#_DATA (p=C-F, #=0-1)**
  - >Double buffered SCI receiver and transmitter hardware



27

EEL4744

See doc8331, Section 23.15.2

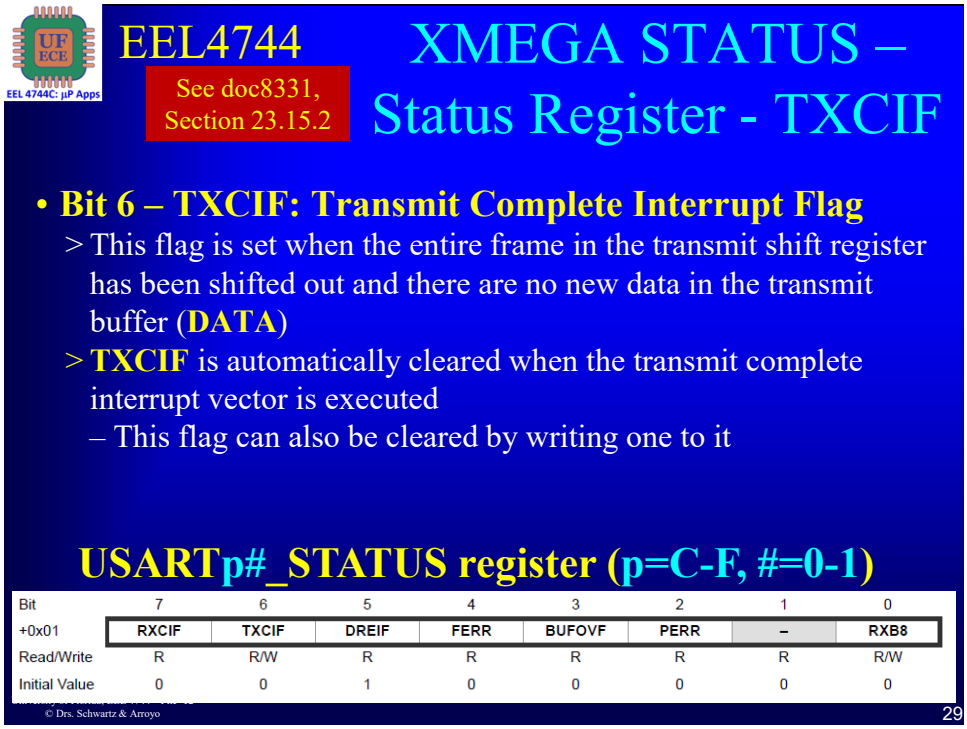
XMEGA STATUS – Status Register - RXCIF

- **Bit 7 – RXCIF: Receive Complete Interrupt Flag**
  - > Set when there are unread data in the receive buffer
  - > Cleared when the receive buffer is empty
  - > When the receiver is disabled, the receive buffer will be flushed, and consequently RXCIF will become zero
  - > When receiver interrupts are used, the receive complete interrupt service routine must read the received data from **DATA** in order to clear **RXCIF**
    - – If not, a new interrupt will occur directly after the return from the current interrupt
    - – This flag can also be cleared by writing one to it

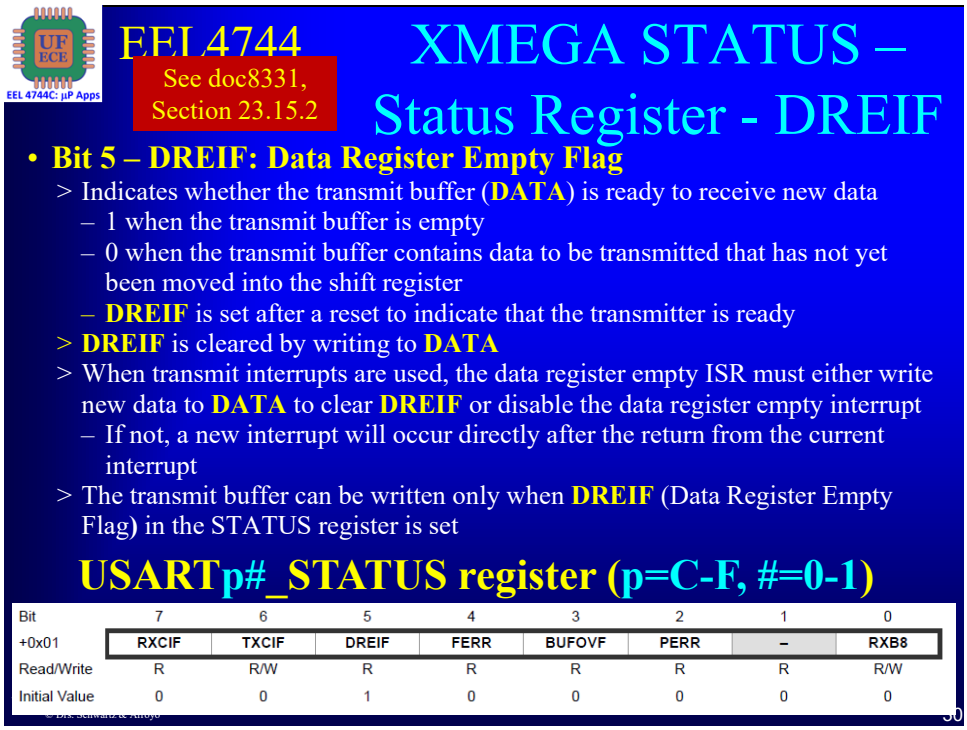
USARTp#_STATUS register (p=C-F, #=0-1)

Bit	7	6	5	4	3	2	1	0
+0x01	RXCIF	TXCIF	DREIF	FERR	BUFOVF	PERR	–	RXB8
Read/Write	R	R/W	R	R	R	R	R	R/W
Initial Value	0	0	1	0	0	0	0	0

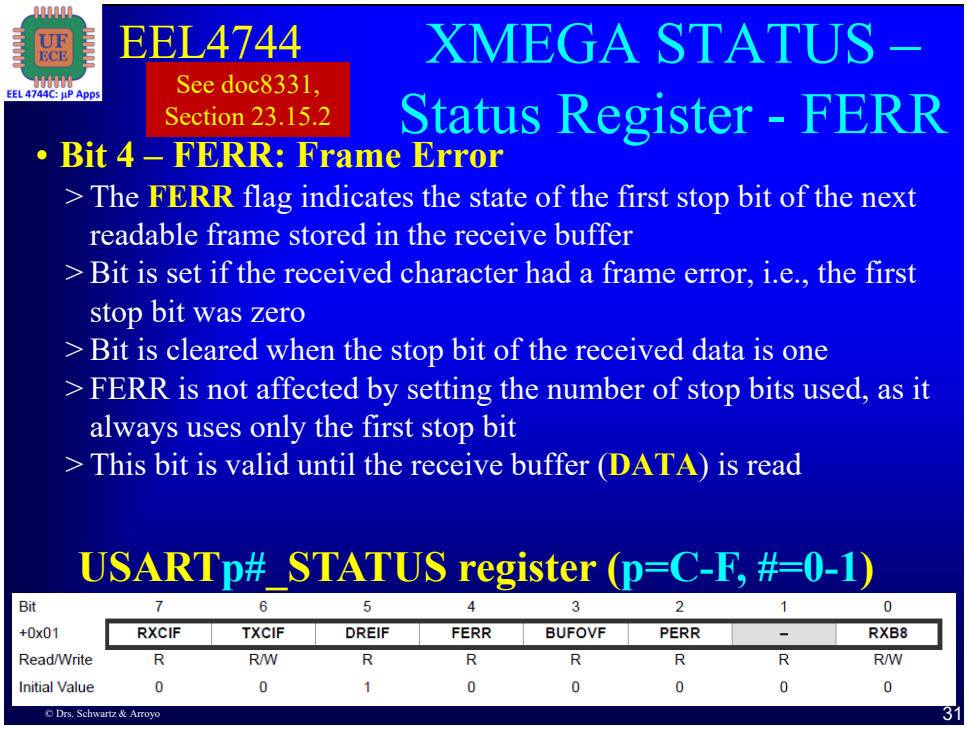
28


EEL4744
See doc8331, Section 23.15.2
XMEGA STATUS – Status Register - TXCIF
• Bit 6 – TXCIF: Transmit Complete Interrupt Flag
> This flag is set when the entire frame in the transmit shift register has been shifted out and there are no new data in the transmit buffer (DATA)
> TXCIF is automatically cleared when the transmit complete interrupt vector is executed
– This flag can also be cleared by writing one to it
USARTp#_STATUS register (p=C-F, #=0-1)
Bit 7 6 5 4 3 2 1 0
+0x01 RXCIF TXCIF DREIF FERR BUFOVF PERR – RXB8
Read/Write R R/W R R R R R R/W
Initial Value 0 0 1 0 0 0 0 0
© Drs. Schwartz & Arroyo
29

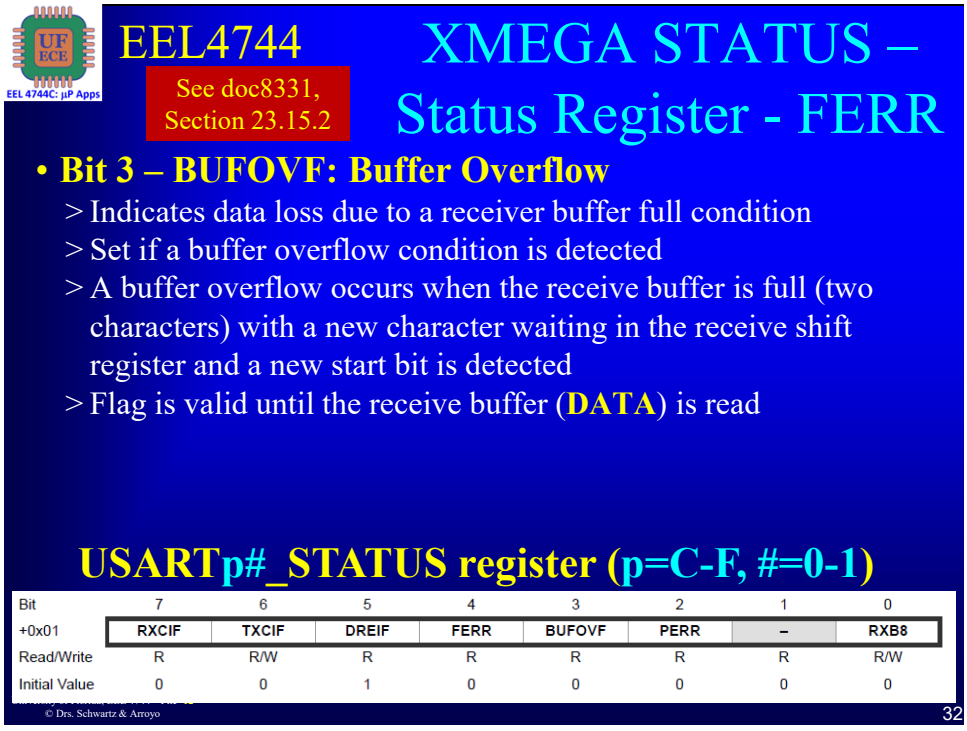
29


EEL4744
See doc8331, Section 23.15.2
XMEGA STATUS – Status Register - DREIF
• Bit 5 – DREIF: Data Register Empty Flag
> Indicates whether the transmit buffer (DATA) is ready to receive new data
– 1 when the transmit buffer is empty
– 0 when the transmit buffer contains data to be transmitted that has not yet been moved into the shift register
– DREIF is set after a reset to indicate that the transmitter is ready
> DREIF is cleared by writing to DATA
> When transmit interrupts are used, the data register empty ISR must either write new data to DATA to clear DREIF or disable the data register empty interrupt
– If not, a new interrupt will occur directly after the return from the current interrupt
> The transmit buffer can be written only when DREIF (Data Register Empty Flag) in the STATUS register is set
USARTp#_STATUS register (p=C-F, #=0-1)
Bit 7 6 5 4 3 2 1 0
+0x01 RXCIF TXCIF DREIF FERR BUFOVF PERR – RXB8
Read/Write R R/W R R R R R R/W
Initial Value 0 0 1 0 0 0 0 0
30

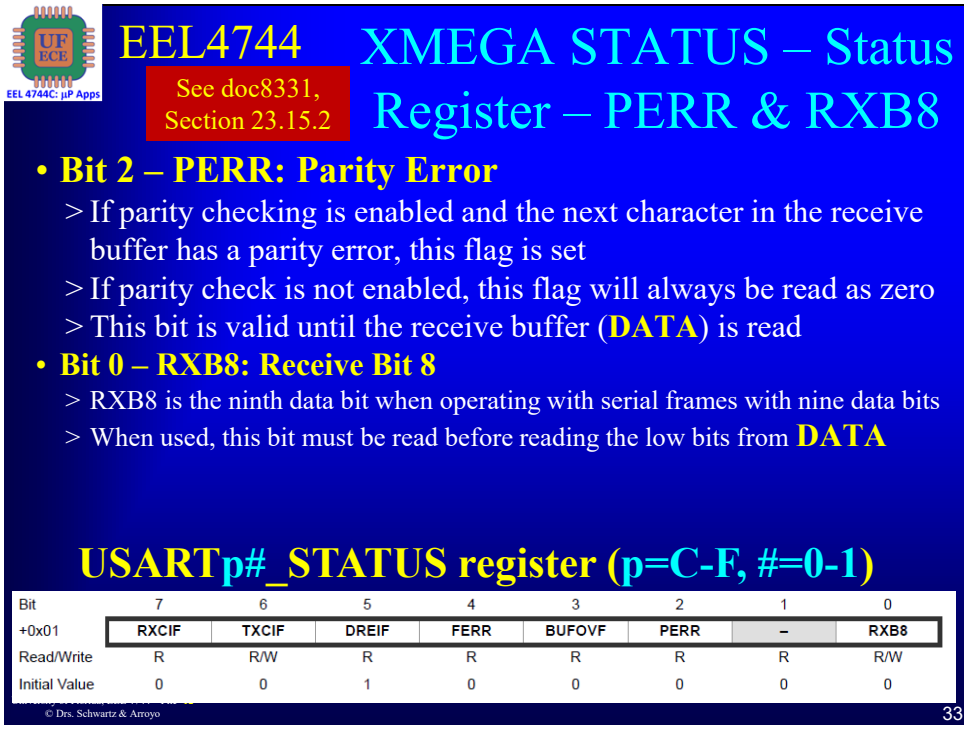
30


EEL4744
See doc8331, Section 23.15.2
XMEGA STATUS – Status Register - FERR
• Bit 4 – FERR: Frame Error
> The FERR flag indicates the state of the first stop bit of the next readable frame stored in the receive buffer
> Bit is set if the received character had a frame error, i.e., the first stop bit was zero
> Bit is cleared when the stop bit of the received data is one
> FERR is not affected by setting the number of stop bits used, as it always uses only the first stop bit
> This bit is valid until the receive buffer (DATA) is read
USARTp#_STATUS register (p=C-F, #=0-1)
Bit 7 6 5 4 3 2 1 0
+0x01 RXCIF TXCIF DREIF FERR BUFOVF PERR – RXB8
Read/Write R R/W R R R R R R/W
Initial Value 0 0 1 0 0 0 0 0
© Drs. Schwartz & Arroyo
31

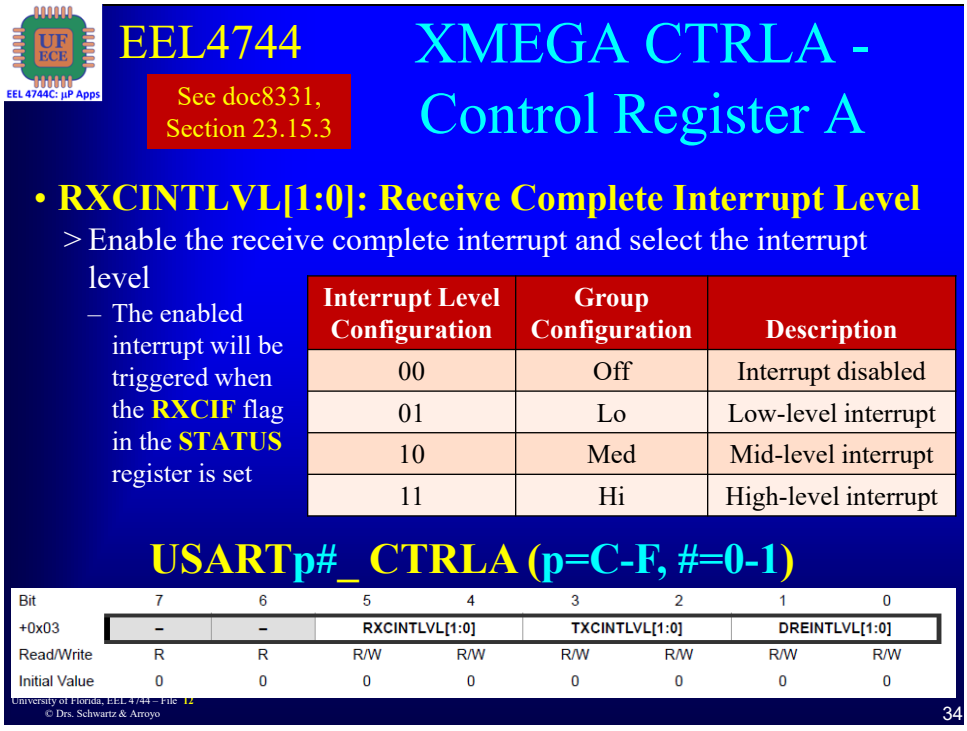
31


EEL4744
See doc8331, Section 23.15.2
XMEGA STATUS – Status Register - FERR
• Bit 3 – BUFOVF: Buffer Overflow
> Indicates data loss due to a receiver buffer full condition
> Set if a buffer overflow condition is detected
> A buffer overflow occurs when the receive buffer is full (two characters) with a new character waiting in the receive shift register and a new start bit is detected
> Flag is valid until the receive buffer (DATA) is read
USARTp#_STATUS register (p=C-F, #=0-1)
Bit 7 6 5 4 3 2 1 0
+0x01 RXCIF TXCIF DREIF FERR BUFOVF PERR – RXB8
Read/Write R R/W R R R R R R/W
Initial Value 0 0 1 0 0 0 0 0
© Drs. Schwartz & Arroyo
32

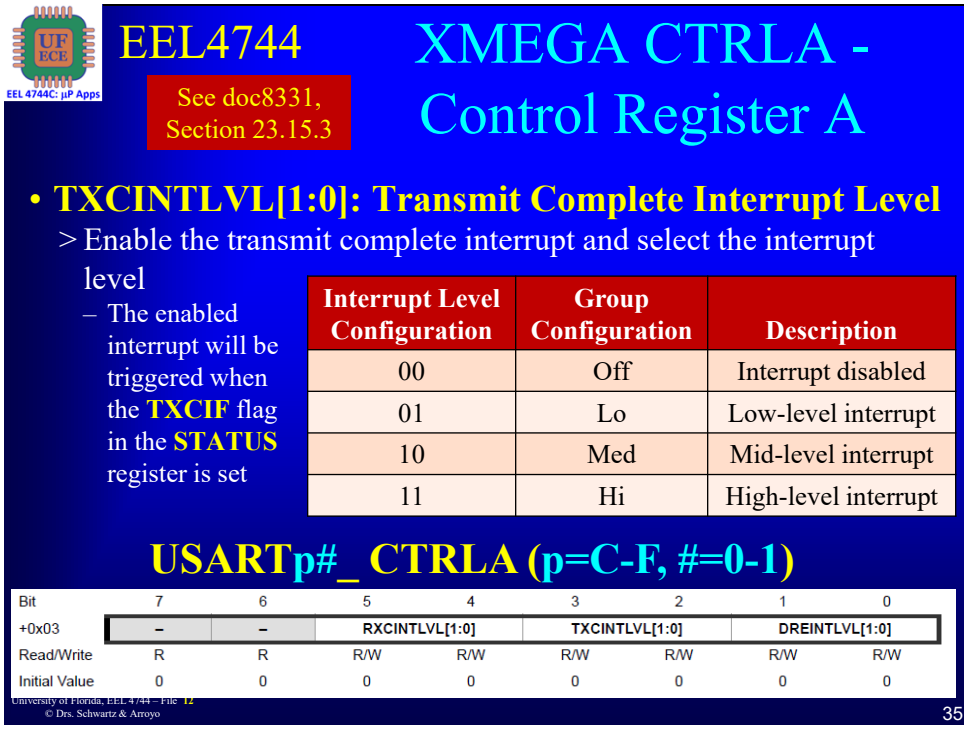
32


EEL4744
See doc8331, Section 23.15.2
XMEGA STATUS – Status Register – PERR & RXB8
• Bit 2 – PERR: Parity Error
> If parity checking is enabled and the next character in the receive buffer has a parity error, this flag is set
> If parity check is not enabled, this flag will always be read as zero
> This bit is valid until the receive buffer (DATA) is read
• Bit 0 – RXB8: Receive Bit 8
> RXB8 is the ninth data bit when operating with serial frames with nine data bits
> When used, this bit must be read before reading the low bits from DATA
USARTp#_STATUS register (p=C-F, #=0-1)
Bit 7 6 5 4 3 2 1 0
+0x01 RXCIF TXCIF DREIF FERR BUFOVF PERR – RXB8
Read/Write R R/W R R R R R R/W
Initial Value 0 0 1 0 0 0 0 0
© Drs. Schwartz & Arroyo
33

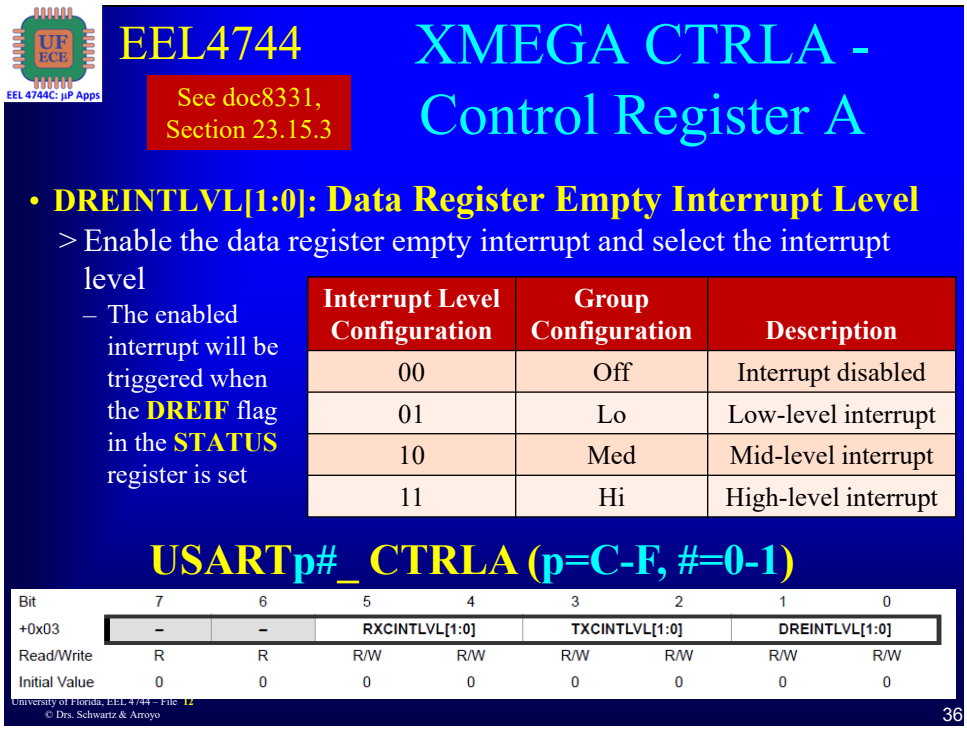
33


EEL4744
See doc8331, Section 23.15.3
XMEGA CTRLA - Control Register A
• RXCINTLVL[1:0]: Receive Complete Interrupt Level
> Enable the receive complete interrupt and select the interrupt level
– The enabled interrupt will be triggered when the RXCIF flag in the STATUS register is set
Interrupt Level Configuration | Group Configuration | Description
00 | Off | Interrupt disabled
01 | Lo | Low-level interrupt
10 | Med | Mid-level interrupt
11 | Hi | High-level interrupt
USARTp#_ CTRLA (p=C-F, #=0-1)
Bit 7 6 5 4 3 2 1 0
+0x03 – – RXCINTLVL[1:0] TXCINTLVL[1:0] DREINTLVL[1:0]
Read/Write R R R/W R/W R/W R/W R/W R/W
Initial Value 0 0 0 0 0 0 0 0
© Drs. Schwartz & Arroyo
34

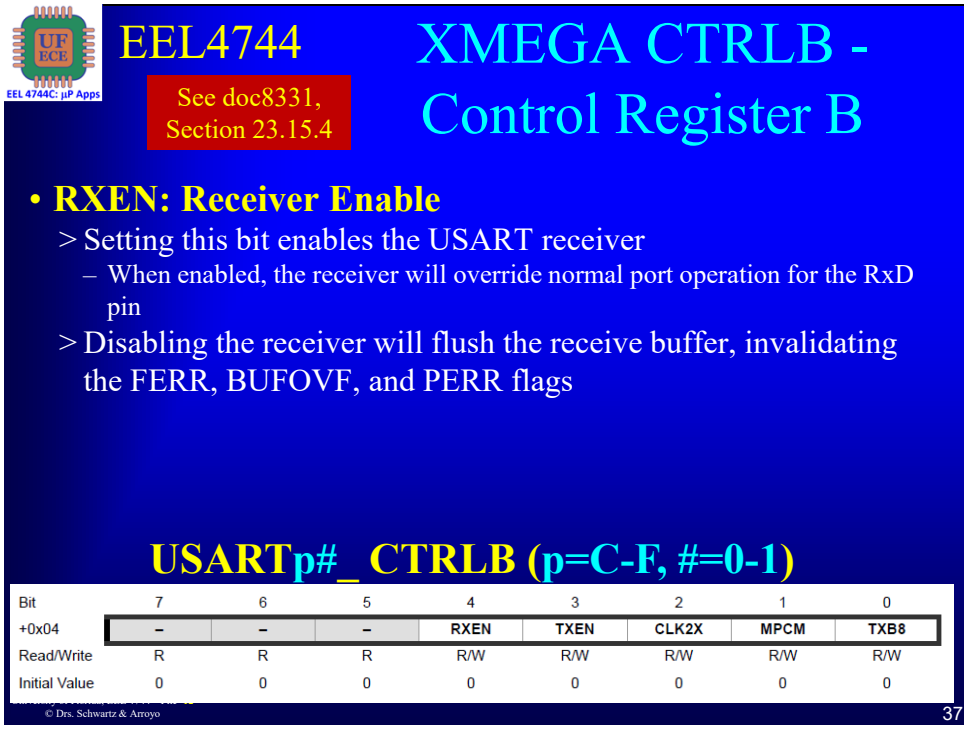
34


EEL4744
See doc8331, Section 23.15.3
XMEGA CTRLA - Control Register A
• TXCINTLVL[1:0]: Transmit Complete Interrupt Level
> Enable the transmit complete interrupt and select the interrupt level
– The enabled interrupt will be triggered when the TXCIF flag in the STATUS register is set
Interrupt Level Configuration | Group Configuration | Description
00 | Off | Interrupt disabled
01 | Lo | Low-level interrupt
10 | Med | Mid-level interrupt
11 | Hi | High-level interrupt
USARTp#_ CTRLA (p=C-F, #=0-1)
Bit 7 6 5 4 3 2 1 0
+0x03 – – RXCINTLVL[1:0] TXCINTLVL[1:0] DREINTLVL[1:0]
Read/Write R R R/W R/W R/W R/W R/W R/W
Initial Value 0 0 0 0 0 0 0 0
© Drs. Schwartz & Arroyo
35

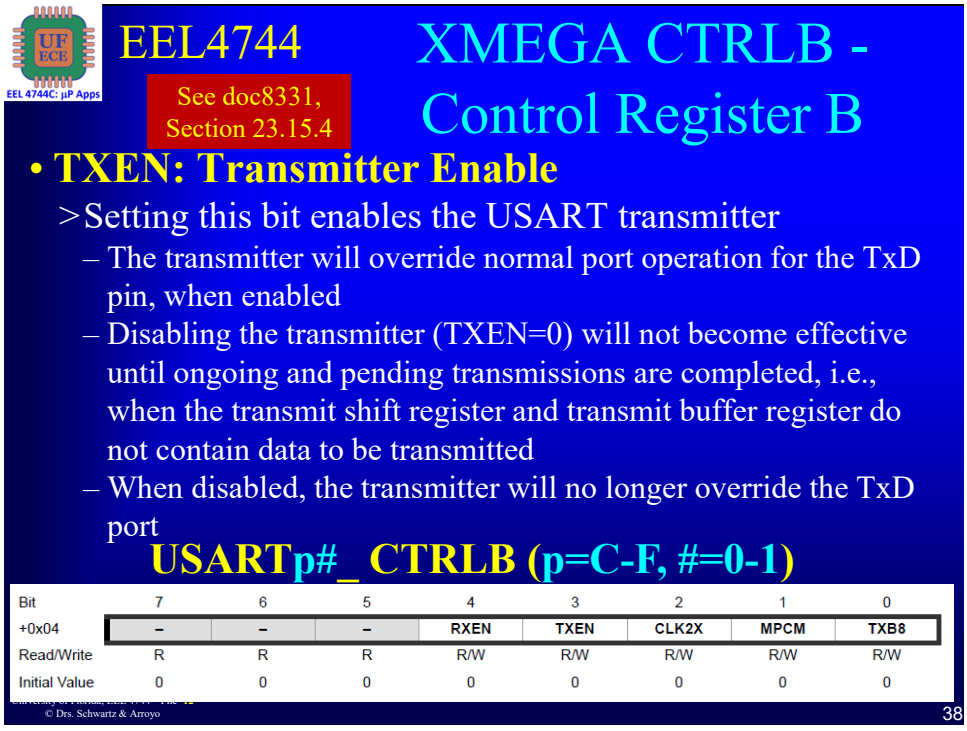
35


EEL4744
See doc8331, Section 23.15.3
XMEGA CTRLA - Control Register A
• DREINTLVL[1:0]: Data Register Empty Interrupt Level
> Enable the data register empty interrupt and select the interrupt level
– The enabled interrupt will be triggered when the DREIF flag in the STATUS register is set
Interrupt Level Configuration | Group Configuration | Description
00 | Off | Interrupt disabled
01 | Lo | Low-level interrupt
10 | Med | Mid-level interrupt
11 | Hi | High-level interrupt
USARTp#_ CTRLA (p=C-F, #=0-1)
Bit 7 6 5 4 3 2 1 0
+0x03 – – RXCINTLVL[1:0] TXCINTLVL[1:0] DREINTLVL[1:0]
Read/Write R R R/W R/W R/W R/W R/W R/W
Initial Value 0 0 0 0 0 0 0 0
© Drs. Schwartz & Arroyo
36

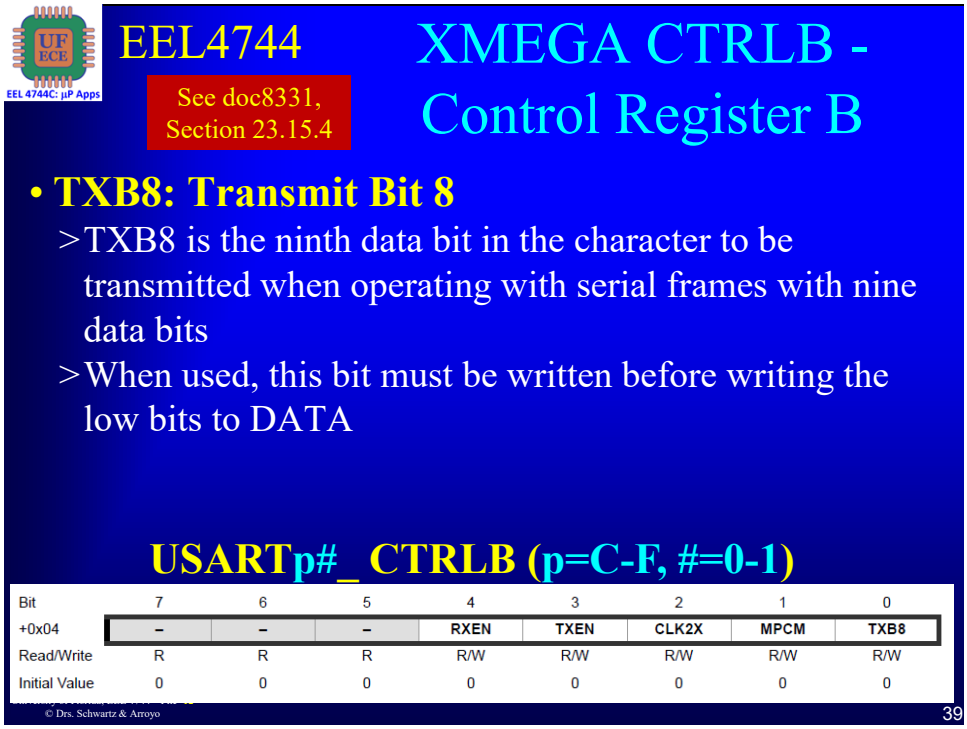
36


EEL4744
See doc8331, Section 23.15.4
XMEGA CTRLB - Control Register B
• RXEN: Receiver Enable
> Setting this bit enables the USART receiver
– When enabled, the receiver will override normal port operation for the RxD pin
> Disabling the receiver will flush the receive buffer, invalidating the FERR, BUFOVF, and PERR flags
USARTp#_ CTRLB (p=C-F, #=0-1)
Bit 7 6 5 4 3 2 1 0
+0x04 – – – RXEN TXEN CLK2X MPCM TXB8
Read/Write R R R R/W R/W R/W R/W R/W
Initial Value 0 0 0 0 0 0 0 0
© Drs. Schwartz & Arroyo
37

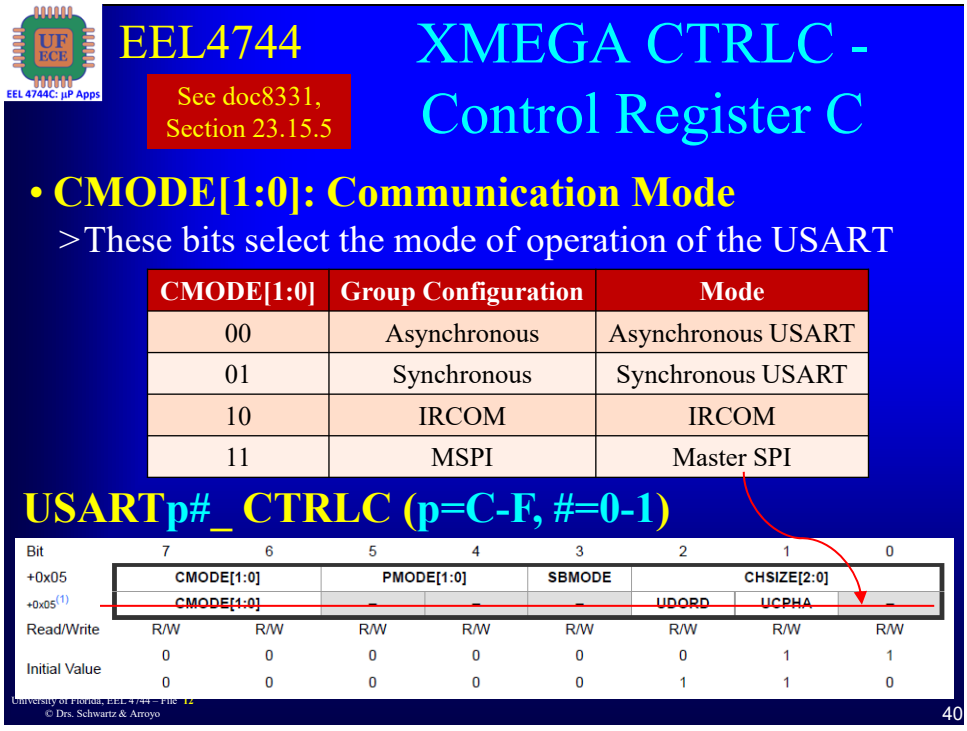
37


EEL4744
See doc8331, Section 23.15.4
XMEGA CTRLB - Control Register B
• TXEN: Transmitter Enable
> Setting this bit enables the USART transmitter
– The transmitter will override normal port operation for the TxD pin, when enabled
– Disabling the transmitter (TXEN=0) will not become effective until ongoing and pending transmissions are completed, i.e., when the transmit shift register and transmit buffer register do not contain data to be transmitted
– When disabled, the transmitter will no longer override the TxD port
USARTp#_ CTRLB (p=C-F, #=0-1)
Bit 7 6 5 4 3 2 1 0
+0x04 – – – RXEN TXEN CLK2X MPCM TXB8
Read/Write R R R R/W R/W R/W R/W R/W
Initial Value 0 0 0 0 0 0 0 0
© Drs. Schwartz & Arroyo
38

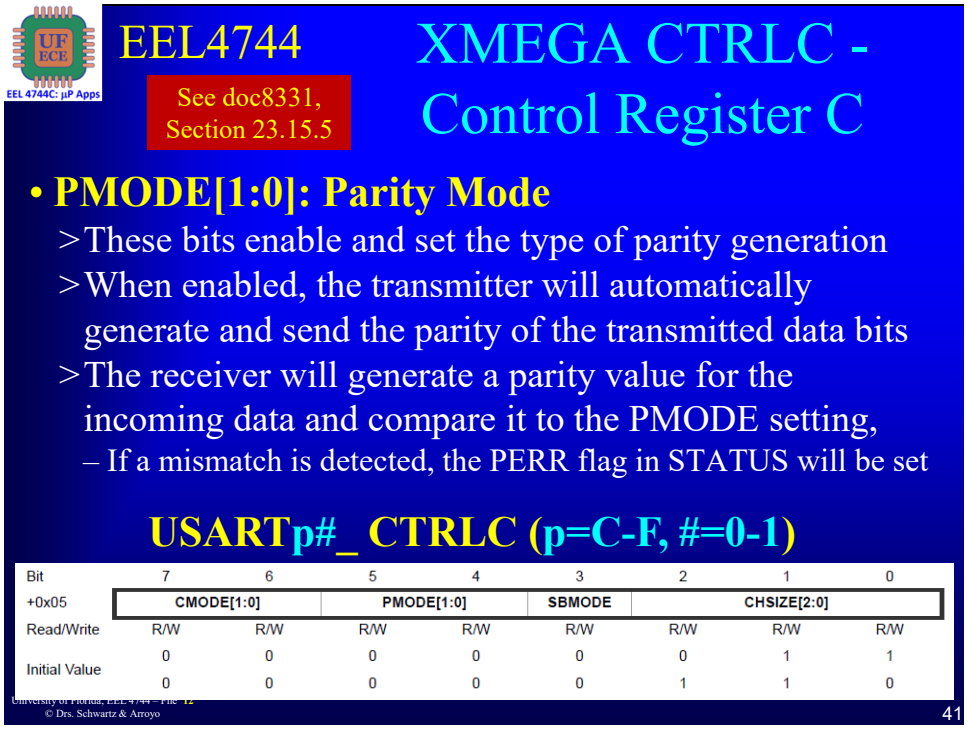
38


EEL4744
See doc8331, Section 23.15.4
XMEGA CTRLB - Control Register B
• TXB8: Transmit Bit 8
>TXB8 is the ninth data bit in the character to be transmitted when operating with serial frames with nine data bits
>When used, this bit must be written before writing the low bits to DATA
USARTp#_ CTRLB (p=C-F, #=0-1)
Bit 7 6 5 4 3 2 1 0
+0x04 – – – RXEN TXEN CLK2X MPCM TXB8
Read/Write R R R R/W R/W R/W R/W R/W
Initial Value 0 0 0 0 0 0 0 0
© Drs. Schwartz & Arroyo
39

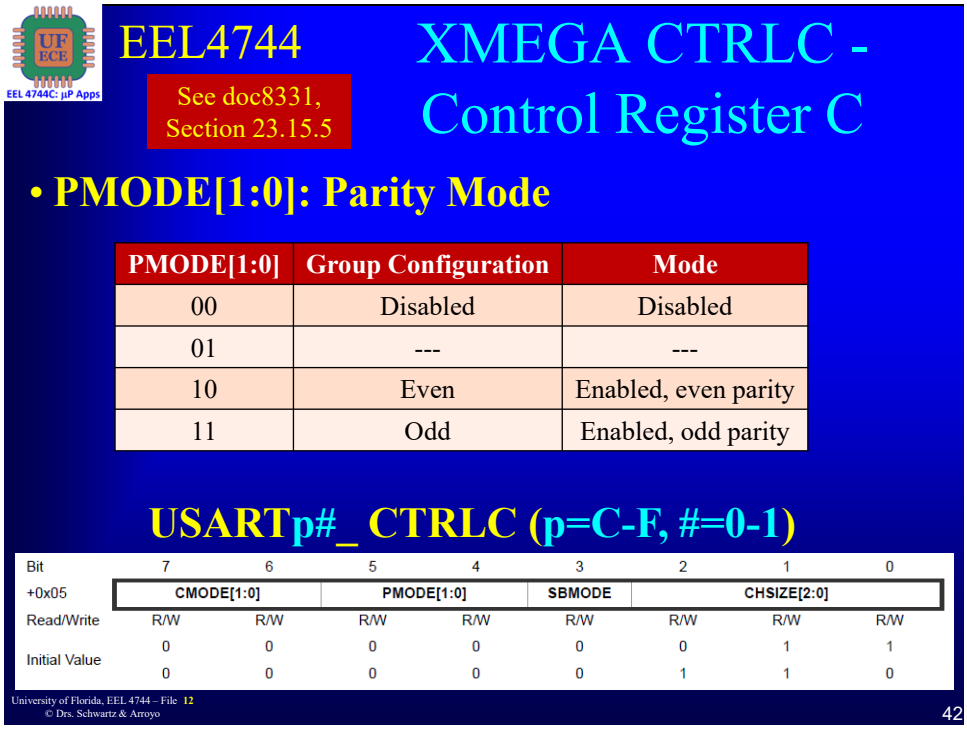
39


EEL4744
See doc8331, Section 23.15.5
XMEGA CTRLC - Control Register C
• CMODE[1:0]: Communication Mode
>These bits select the mode of operation of the USART
CMODE[1:0] Group Configuration Mode
00 Asynchronous Asynchronous USART
01 Synchronous Synchronous USART
10 IRCOM IRCOM
11 MSPI Master SPI
USARTp#_ CTRLC (p=C-F, #=0-1)
Bit 7 6 5 4 3 2 1 0
+0x05 CMODE[1:0] PMODE[1:0] SBMODE CHSIZE[2:0]
+0x05(1) CMODE[1:0] – – – UDORD UCPHA –
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 1 1
0 0 0 0 0 1 1 0
© Drs. Schwartz & Arroyo
40

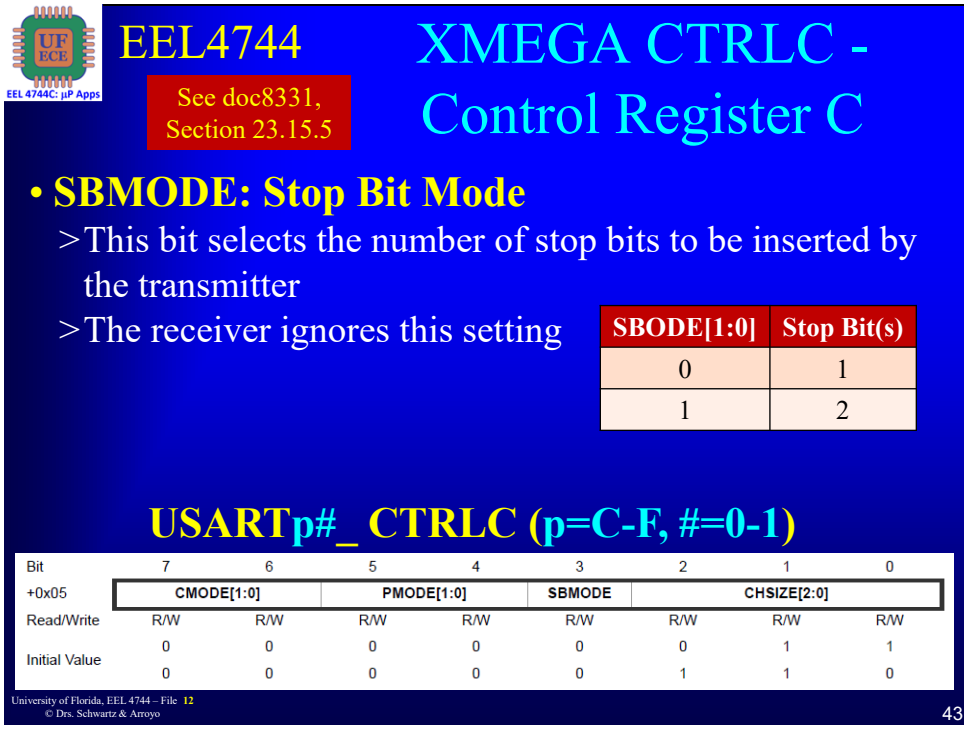
40


EEL4744
See doc8331, Section 23.15.5
XMEGA CTRLC - Control Register C
• PMODE[1:0]: Parity Mode
>These bits enable and set the type of parity generation
>When enabled, the transmitter will automatically generate and send the parity of the transmitted data bits
>The receiver will generate a parity value for the incoming data and compare it to the PMODE setting,
– If a mismatch is detected, the PERR flag in STATUS will be set
USARTp#_ CTRLC (p=C-F, #=0-1)
Bit 7 6 5 4 3 2 1 0
+0x05 CMODE[1:0] PMODE[1:0] SBMODE CHSIZE[2:0]
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 1 1
0 0 0 0 0 1 1 0
© Drs. Schwartz & Arroyo
41

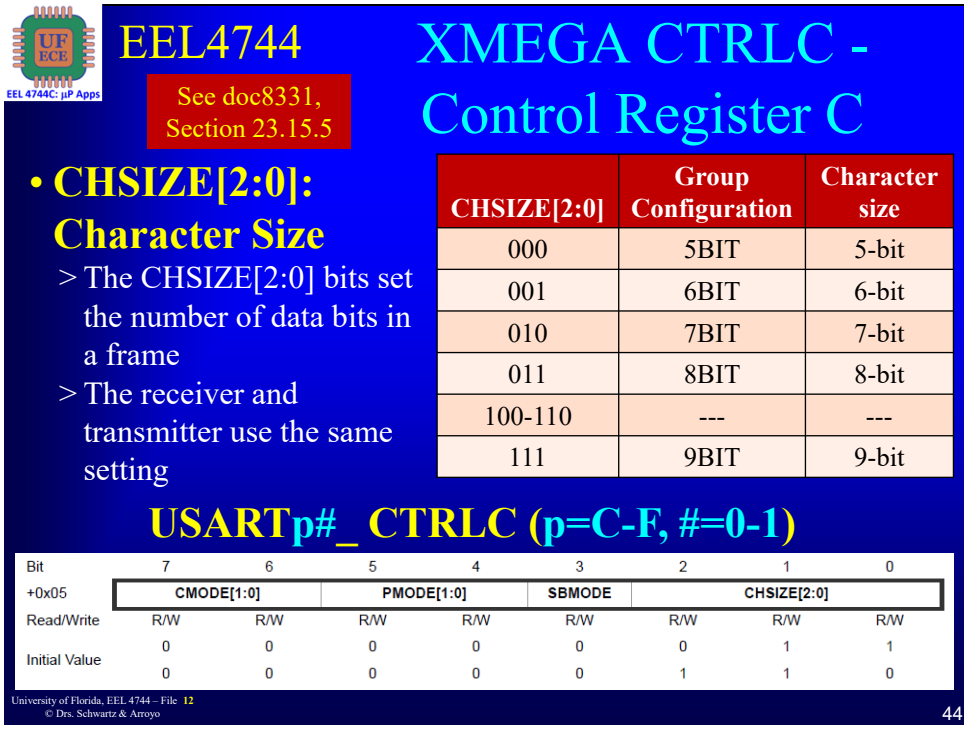
41


EEL4744
See doc8331, Section 23.15.5
XMEGA CTRLC - Control Register C
• PMODE[1:0]: Parity Mode
PMODE[1:0] | Group Configuration | Mode
00 | Disabled | Disabled
01 | --- | ---
10 | Even | Enabled, even parity
11 | Odd | Enabled, odd parity
USARTp#_ CTRLC (p=C-F, #=0-1)
Bit 7 6 5 4 3 2 1 0
+0x05 CMODE[1:0] PMODE[1:0] SBMODE CHSIZE[2:0]
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 1 1
0 0 0 0 0 1 1 0
© Drs. Schwartz & Arroyo
42

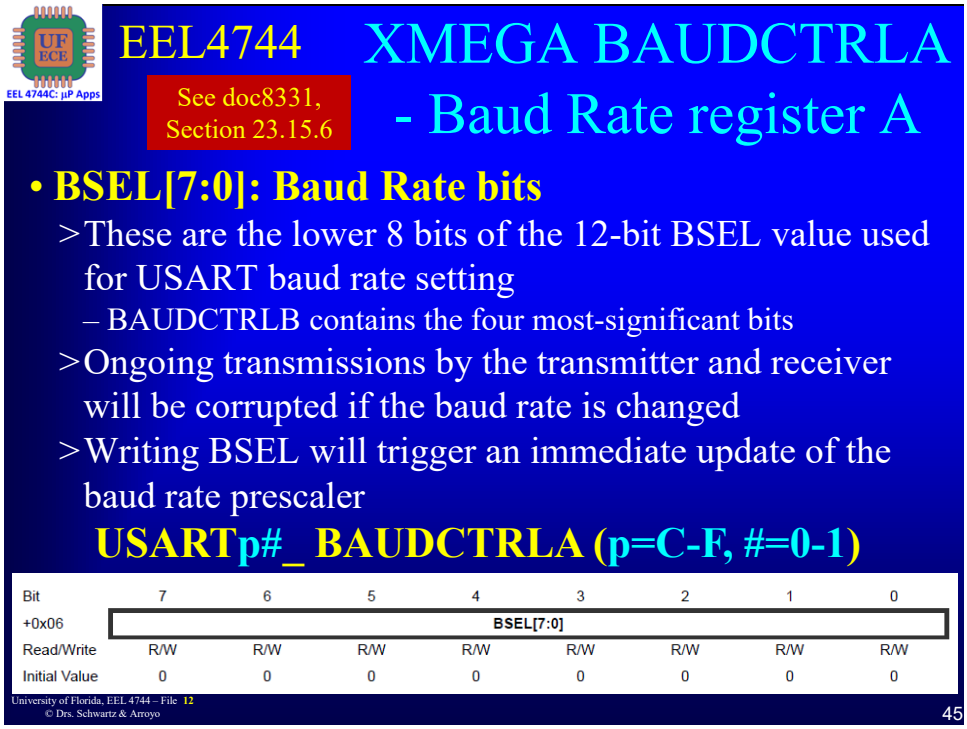
42


EEL4744
See doc8331, Section 23.15.5
XMEGA CTRLC - Control Register C
• SBMODE: Stop Bit Mode
> This bit selects the number of stop bits to be inserted by the transmitter
> The receiver ignores this setting
SBODE[1:0] | Stop Bit(s)
0 | 1
1 | 2
USARTp#_ CTRLC (p=C-F, #=0-1)
Bit 7 6 5 4 3 2 1 0
+0x05 CMODE[1:0] PMODE[1:0] SBMODE CHSIZE[2:0]
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 1 1
0 0 0 0 0 1 1 0
University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo
43

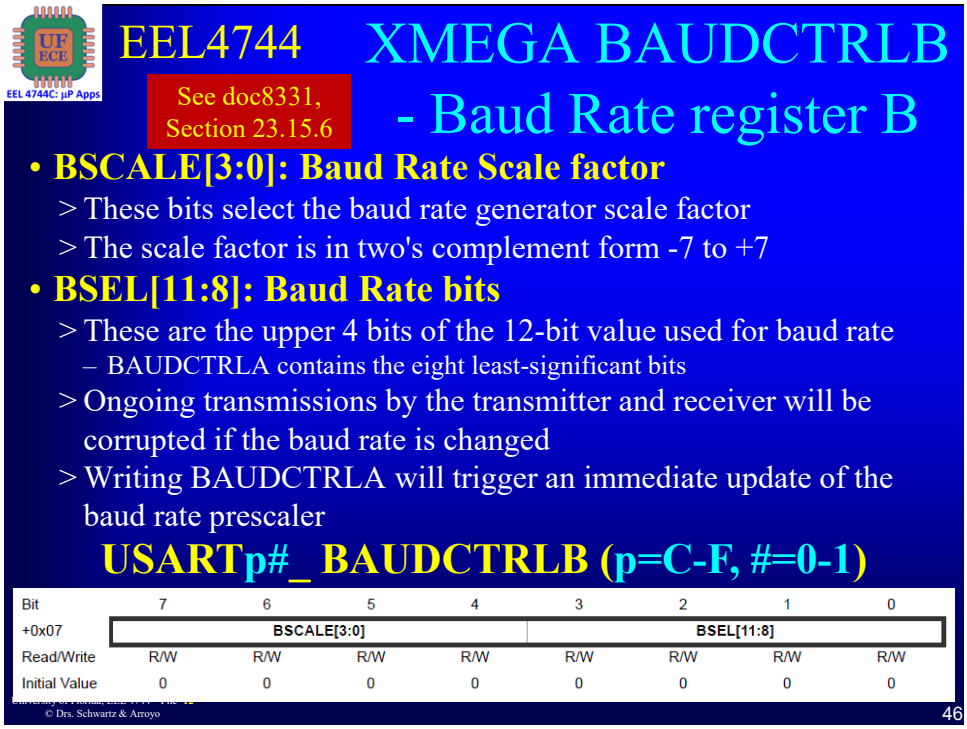
43


EEL4744
See doc8331, Section 23.15.5
XMEGA CTRLC - Control Register C
• CHSIZE[2:0]: Character Size
> The CHSIZE[2:0] bits set the number of data bits in a frame
> The receiver and transmitter use the same setting
CHSIZE[2:0] | Group Configuration | Character size
000 | 5BIT | 5-bit
001 | 6BIT | 6-bit
010 | 7BIT | 7-bit
011 | 8BIT | 8-bit
100-110 | --- | ---
111 | 9BIT | 9-bit
USARTp#_ CTRLC (p=C-F, #=0-1)
Bit 7 6 5 4 3 2 1 0
+0x05 CMODE[1:0] PMODE[1:0] SBMODE CHSIZE[2:0]
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 1 1
0 0 0 0 0 1 1 0
University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo
44

44


EEL4744
XMEGA BAUDCTRLA
- Baud Rate register A
See doc8331, Section 23.15.6
• BSEL[7:0]: Baud Rate bits
>These are the lower 8 bits of the 12-bit BSEL value used for USART baud rate setting
– BAUDCTRLB contains the four most-significant bits
>Ongoing transmissions by the transmitter and receiver will be corrupted if the baud rate is changed
>Writing BSEL will trigger an immediate update of the baud rate prescaler
USARTp#_ BAUDCTRLA (p=C-F, #=0-1)
Bit 7 6 5 4 3 2 1 0
+0x06 BSEL[7:0]
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
University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo
45

45


EEL4744
XMEGA BAUDCTRLB
- Baud Rate register B
See doc8331, Section 23.15.6
• BSCALE[3:0]: Baud Rate Scale factor
> These bits select the baud rate generator scale factor
> The scale factor is in two's complement form -7 to +7
• BSEL[11:8]: Baud Rate bits
> These are the upper 4 bits of the 12-bit value used for baud rate
– BAUDCTRLA contains the eight least-significant bits
> Ongoing transmissions by the transmitter and receiver will be corrupted if the baud rate is changed
> Writing BAUDCTRLA will trigger an immediate update of the baud rate prescaler
USARTp#_ BAUDCTRLB (p=C-F, #=0-1)
Bit 7 6 5 4 3 2 1 0
+0x07 BSCALE[3:0] BSEL[11:8]
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
© Drs. Schwartz & Arroyo
46

46

EEL4744

See doc8331, Section 23.16, 23.17

XMEGA USART Register Summary & Interrupt Vectors

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DATA	DATA[7:0]							
STATUS	RXCIF	TXCIF	DREIF	FERR	BUFOVF	PERR	–	RXB8
Reserved	–	–	–	–	–	–	–	–
CTRLA	–	–	RXCINTLVL[1:0]		TXCINTLVL[1:0]		DREINTLVL[1:0]	
CTRLB	–	–	–	RXEN	TXEN	CLK2X	MPCM	TXB8
CTRLC	CMODE[1:0]		PMODE[1:0]		SBMODE	CHSIZE[2:0]		
BAUDCTRLA	BSEL[7:0]							
BAUDCTRLB	BSCALE[3:0]				BSEL[11:8]			

Offset	Source	Interrupt description
0x00	RXC_vect	USART receive complete interrupt vector
0x02	DRE_vect	USART data register empty interrupt vector
0x04	TXC_vect	USART transmit complete interrupt vector

© Drs. Schwartz & Arroyo
47

47

EEL4744

XMEGA: SCI (USART) Interrupt Vectors (Ports C-F)

```
; USARTC0 interrupt vectors
.equ USARTC0_RXC_vect = 50 ; Reception Complete Interrupt
.equ USARTC0_DRE_vect = 52 ; Data Register Empty Interrupt
.equ USARTC0_TXC_vect = 54 ; Transmission Complete Interrupt

; USARTC1 interrupt vectors
.equ USARTC1_RXC_vect = 56 ; Reception Complete Interrupt
.equ USARTC1_DRE_vect = 58 ; Data Register Empty Interrupt
.equ USARTC1_TXC_vect = 60 ; Transmission Complete Interrupt

; USARTF0 interrupt vectors
.equ USARTF0_RXC_vect = 238 ; Reception Complete Interrupt
.equ USARTF0_DRE_vect = 240 ; Data Register Empty Interrupt
.equ USARTF0_TXC_vect = 242 ; Transmission Complete Interrupt

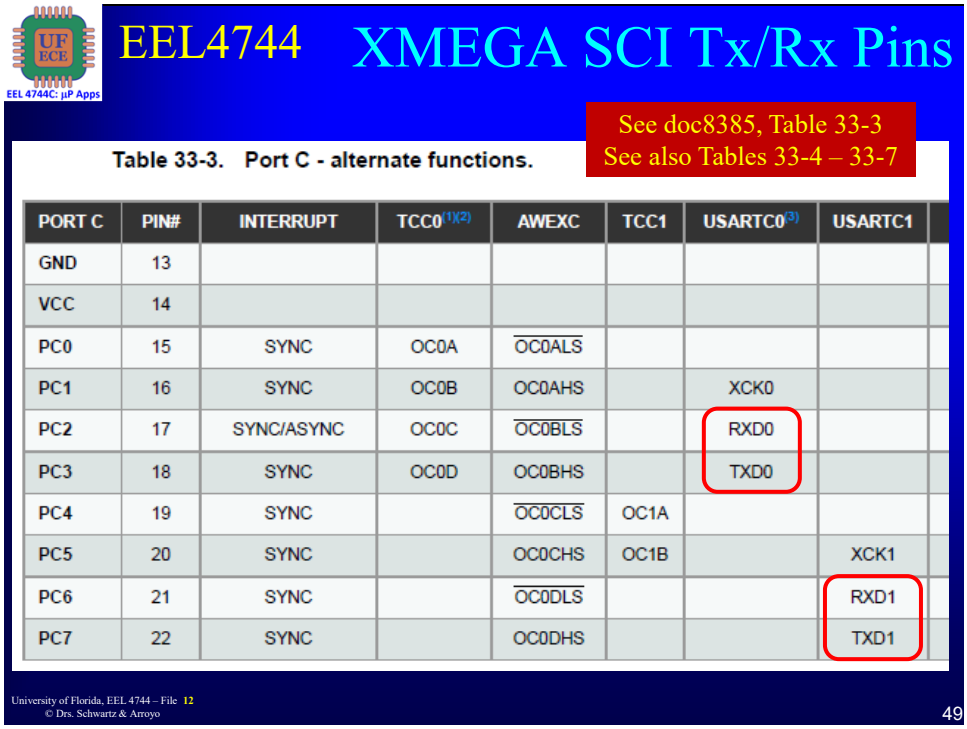
; USARTF1 interrupt vectors
.equ USARTF1_RXC_vect = 244 ; Reception Complete Interrupt
.equ USARTF1_DRE_vect = 246 ; Data Register Empty Interrupt
.equ USARTF1_TXC_vect = 248 ; Transmission Complete Interrupt
```

ATxmega128A1Udef.inc

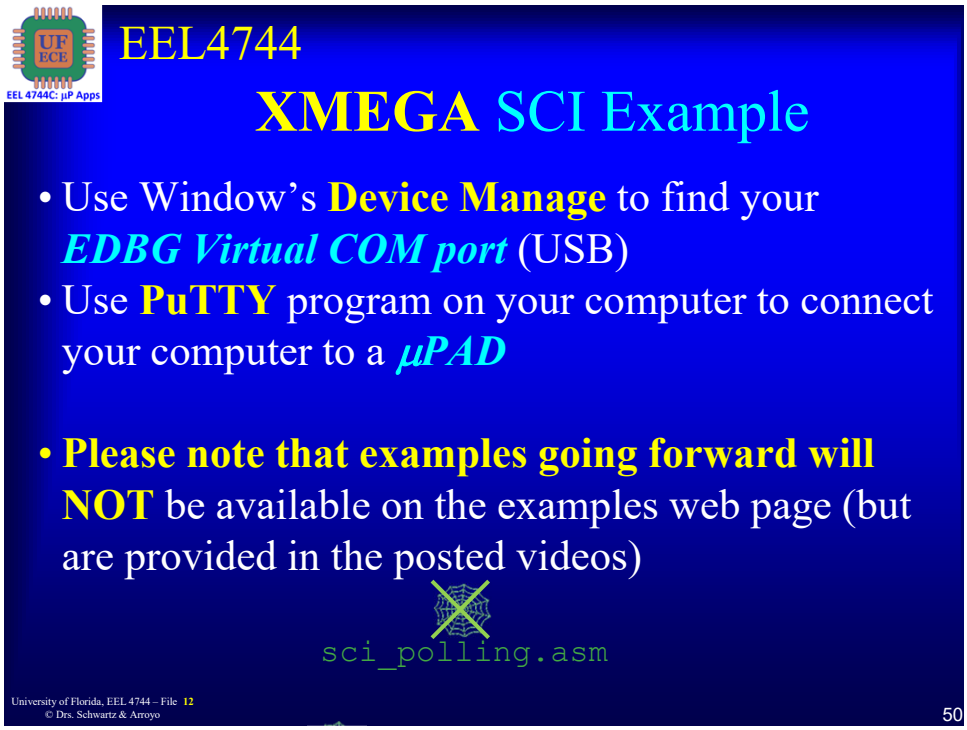
Similar for ports D & E

University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo
48

48


EEL4744 XMEGA SCI Tx/Rx Pins
See doc8385, Table 33-3
See also Tables 33-4 – 33-7
Table 33-3. Port C - alternate functions.
PORT C | PIN# | INTERRUPT | TCC0(1)(2) | AWEXC | TCC1 | USARTC0(3) | USARTC1
GND | 13
VCC | 14
PC0 | 15 | SYNC | OC0A | OC0ALS
PC1 | 16 | SYNC | OC0B | OC0AHS | | XCK0
PC2 | 17 | SYNC/ASYNC | OC0C | OC0BLS | | RXD0
PC3 | 18 | SYNC | OC0D | OC0BHS | | TXD0
PC4 | 19 | SYNC | | OC0CLS | OC1A
PC5 | 20 | SYNC | | OC0CHS | OC1B | | XCK1
PC6 | 21 | SYNC | | OC0DLS | | | RXD1
PC7 | 22 | SYNC | | OC0DHS | | | TXD1
University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo
49

49


EEL4744
XMEGA SCI Example
• Use Window's Device Manage to find your EDBG Virtual COM port (USB)
• Use PuTTY program on your computer to connect your computer to a µPAD
• Please note that examples going forward will NOT be available on the examples web page (but are provided in the posted videos)
sci_polling.asm
University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo
50

50

EEL4744

SPI: Synchronous Serial

- Lab 5: SCI in Assembly language
- Lab 6: C language going forward:
  - >Redo Lab 5 SCI in C
  - >Use SPI to talk to an IMU (inertial measurement unit)

51

EEL4744

Universal Serial Bus (USB)



USB Logo

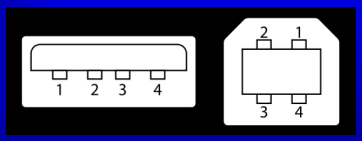
- USB is a serial bus standard to interface devices.
- See Wikipedia for more info
  - > http://en.wikipedia.org/wiki/Usb



USB Series A plug



Mini-A plug (left), Mini-B plug (right)



USB Type A (left) and B (right) connectors

Pin	Function
1	V_{BUS} (4.75-5.25V)
2	D-
3	D+
4	GND
Shell	Shield



Micro-B plug





Type A plug and receptacle

52

EEL4744

USB 3.0

- USB 3.0
  - >The USB 3.0 plug also accepts old Micro-B plugs
- The USB-C is a connector shape, not a tech standard
  - >It 24 pins and can be connected in either orientation (unlike the prior standards)
  - >Used for transmitting **both data and power** on a single cable.
  - >Connector is **flippable**!



USB 3.0 Micro-B plug



USB Series C plug

USB-C used for Thunderbolt 3 and 4

53

EEL4744

USB Info

- USB signals are transmitted on a twisted pair of data cables, labeled D+ and D−.
  - >These collectively use half-duplex differential signaling to combat the effects of electromagnetic noise on longer lines.
  - >D+ and D− usually operate together; they are not separate simplex connections.
  - >Transmitted signal levels are 0 to 0.3V for low and 2.8V to 3.6V for high.
  - >**USB 1.0 speed is 12 Mb/s**
  - >**USB 2.0 speed is 480 Mb/s**
  - >**USB 3.0 speed is up to 4800 Mb/s**

Thunderbolt 3 and 4 up to 40 Gb/s

54

EEL4744 USB to UART Bridge from FTDI (**not** on µPAD)

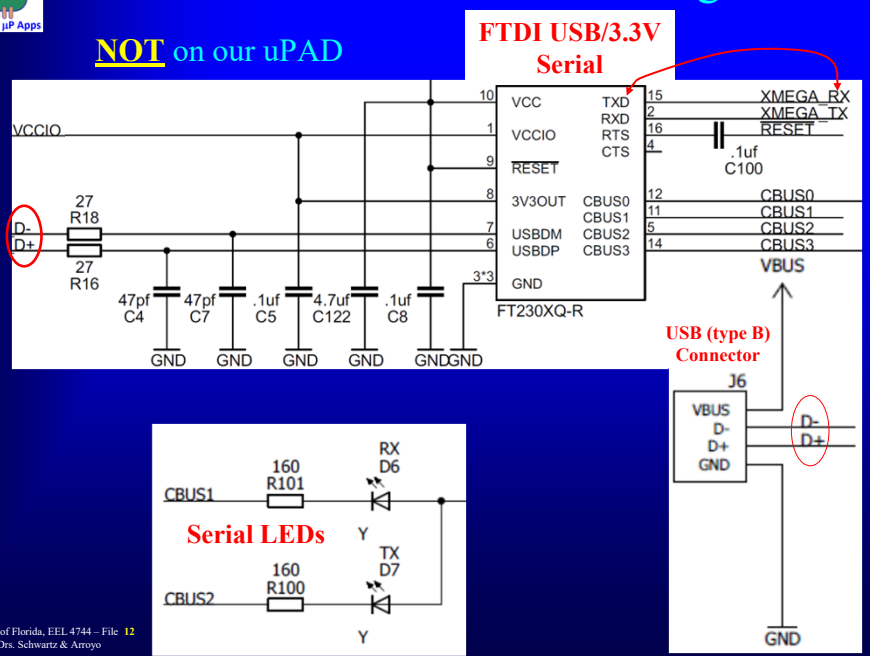
- The FT232RL is a USB 2.0 to UART Bridge from FTDI (http://www.ftdichip.com/). This chip is capable of taking full-speed data (12Mbps) from a USB cable and converting it into serial format using its internal UART.
  - >The UART has an internal clock and can provide baud rates up to 1 Mbit/s.
  - >The FTDI part's data sheet can be found at
    - – http://mil.ufl.edu/4744/docs/DS_FT232R.pdf
    - – Also see http://mil.ufl.edu/4744/docs/FTDI_USB-Serial_DS_FT230X.pdf
- Some **old versions** of µPAD used this bridge chip
  - >Our µPAD has the converter **within** the EDBG device, **not** shown on the schematics (proprietary)

University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo
55

55

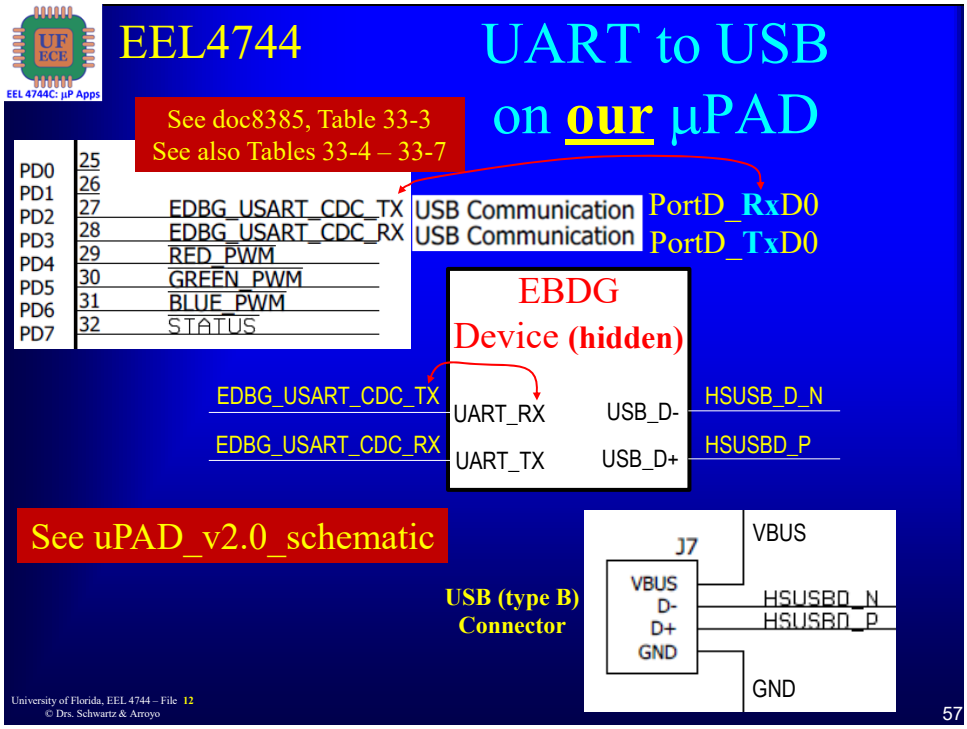
EEL4744 USB to UART Bridge: FTDI

NOT on our uPAD

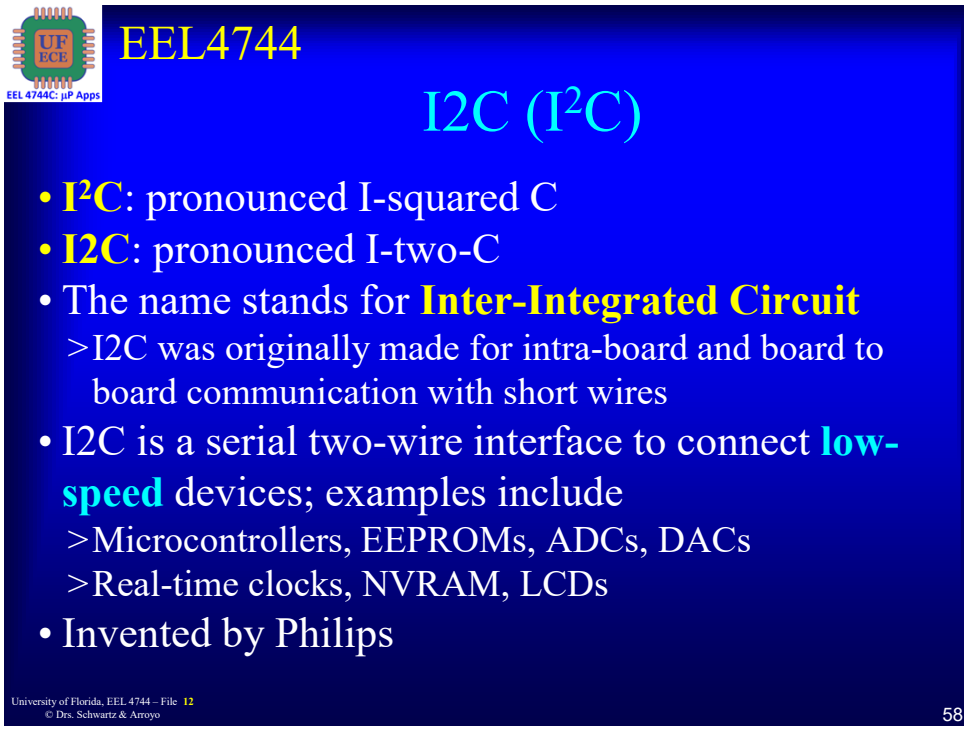


University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo
56

56


EEL4744
UART to USB on our µPAD
EEL 4744C: µP Apps
See doc8385, Table 33-3
See also Tables 33-4 – 33-7
PD0
PD1
PD2
PD3
PD4
PD5
PD6
PD7
25
26
27
28
29
30
31
32
EDBG_USART_CDC_TX
EDBG_USART_CDC_RX
RED_PWM
GREEN_PWM
BLUE_PWM
STATUS
USB Communication
USB Communication
PortD_RxD0
PortD_TxD0
EBDG
Device (hidden)
EDBG_USART_CDC_TX
EDBG_USART_CDC_RX
UART_RX
UART_TX
USB_D-
USB_D+
HSUSB_D_N
HSUSBD_P
See uPAD_v2.0_schematic
USB (type B) Connector
J7
VBUS
D-
D+
GND
VBUS
HSUSBD_N
HSUSBD_P
GND
University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo
57

57

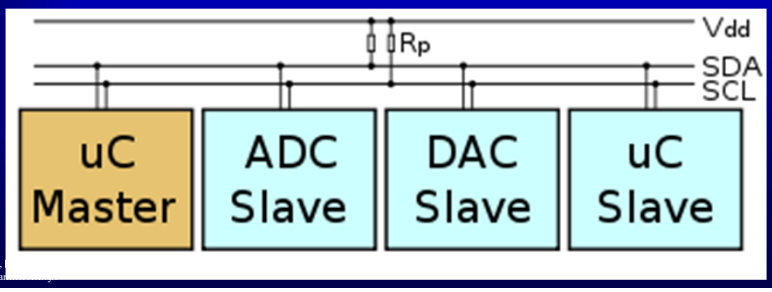

EEL4744
I2C (I²C)
EEL 4744C: µP Apps
• I²C: pronounced I-squared C
• I2C: pronounced I-two-C
• The name stands for Inter-Integrated Circuit
>I2C was originally made for intra-board and board to board communication with short wires
• I2C is a serial two-wire interface to connect low-speed devices; examples include
>Microcontrollers, EEPROMs, ADCs, DACs
>Real-time clocks, NVRAM, LCDs
• Invented by Philips
University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo
58

58

EEL4744

I2C (I^2C)

- Can have more than one master
- Slaves all need unique addresses
  - >Can have up to 127 slaves (or 1023)
- Master generates the clock and starts communication with slaves
- Slaves responds when **addressed** by the master
- I2C uses 8-bit data
- SCL = Serial Clock; SDA = Serial Data



University of Florida,
© Drs. Schwa

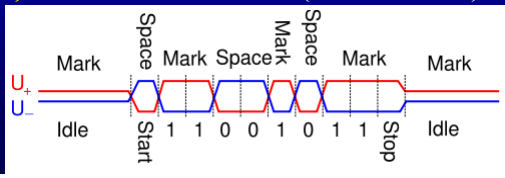
59

59

EEL4744

EIA-485 (RS-485)

- 2-wire serial connection (for half-duplex)
  - >Can use 4-wire serial for full-duplex
  - >Uses a differential balanced line over a twisted pair (like EIA-422)
  - >It can span 4000 feet @ 100 kbit/s; 30ft @ 5Mbit/s
  - >Two pins
    - **A** = '–' = TxD-/RxD- = **inverting** pin which is **negative (compared to B)** when the line is idle (i.e., data is 1).
    - **B** = '+' = TxD+/RxD+ = **non-inverting** pin which is **positive (compared to A)** when the line is idle (i.e., data is 1).



University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo

60

60

EEL 4744C: µP Apps

EE



Bluetooth



- Serial wireless standard for sending data over short distances
  - >Originally made as an alternative to RS-232 cables
  - >79 Bluetooth RF channels
  - >Master/Slave, but can change to masters
  - >Every device has a 48-bit address
    - – Approximately 281 trillion = 281×10^{12}
  - >Generally low power
  - >Packet-based







61

EEL 4744C: µP Apps

EEL4744

Bluetooth

- Range typically less than 10 m
- Bluetooth 5.0 range up to 400 m
- Harold "Bluetooth" Gormsson was a king of Denmark and Norway over 1100 years ago
  - >He united Scandinavia, like Bluetooth unites devices

Get a little history about Harold at https://youtu.be/VdmQp9M9jUo



62

EEL4744 Ethernet (IEEE 802.3)

- 10BASE-T: 10 Mbit/s
  - >Uses RJ-45 connectors with twisted-pair cables
  - >Maximum length 100m

8P8C plug used with 10BASE-T



- 100 BASE-T: 100 Mbit/s (IEEE 802.3u)
  - >100-BASE-TX: two pairs of CAT-5 twisted-pair wires
  - >100-BASE-T4: four pairs of normal twisted-pair wires (CAT-5)
  - >100-BASE-FX: fiber optic cables

Cat 5 cable used with 100 BASE-T

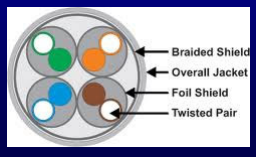


- 100BASE-T (Gigabit Ethernet): 1Gbit/s
- 10GigE: 10Gbit/s (Cat 7); 100 GigaBit: 100Gbit/s

Cat 7 cable used with 10GigE





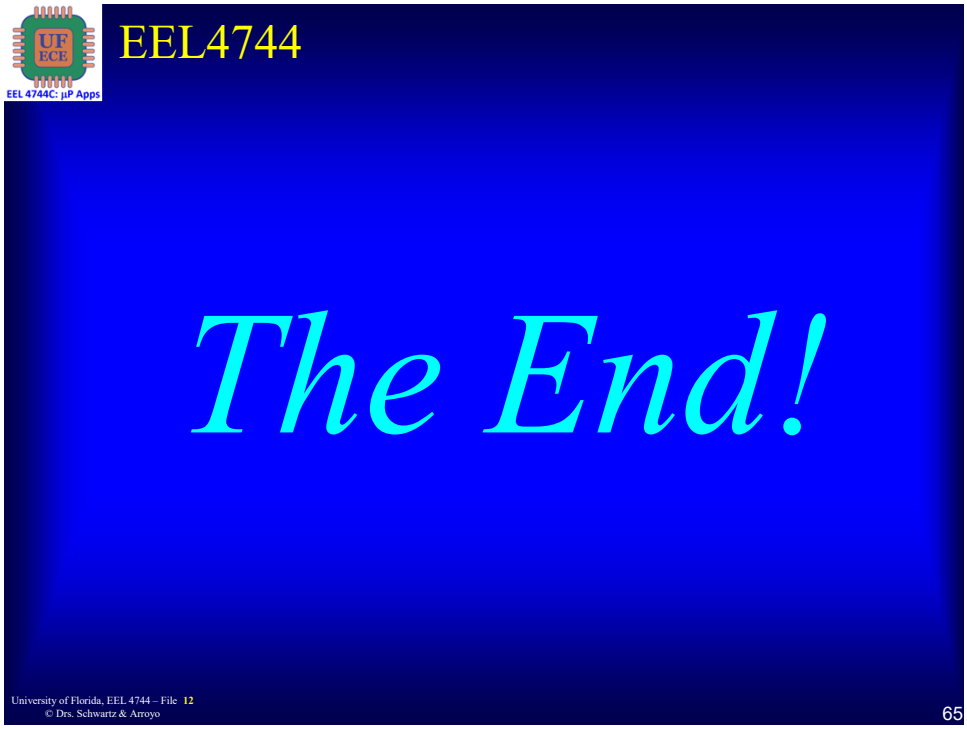


63

EEL4744 Wi-Fi (IEEE 802.11)

- Wireless serial communication
  - >Uses in the 2.4GHz band (which is also used in cordless telephones) or 5GHz band
  - >802.11b (2.4GHz) [11 Mbit/s max data rate]
    - – 120 ft indoors 300 ft outdoors (with normal antenna)
  - >802.11g (2.4GHz) [54 Mbit/s max]
    - – 120 ft indoors, 300 ft outdoors (with normal antenna)
  - >802.11n (2.4GH, 5GHz band) [248 Mbit/s max]
    - – 230 ft indoors 820 ft outdoors (with normal antenna)
  - >802.11ac (2.4GH, 5GHz band) [depends on # antennas]
    - – Up to 1300 Mbit/s on 5GHz and 450 Mbit/s on 2.4GHz
    - – The **effective** range is best by far
  - >More have come in the last few years … getting better all the time!

64


EEL4744
EEL 4744C: µP Apps
The End!
University of Florida, EEL 4744 – File 12
© Drs. Schwartz & Arroyo
65

65