ICS Humla CTF

BY ARUN MANE

SR. SECURITY RESEARCHER

About

- Sr. Security Researcher at Payatu Software Labs
- Focused in
  - IoT, ICS , Vehicle Security
  - Co-Trainer for Practical IoT Hacking and also speaker at various conferences.
- About Payatu
  - A boutique security testing company specializing in IoT, Mobile, Cloud – https://payatu.com
  - Products
    - Expliot – IoT Security Testing framework - https://bitbucket.org/aseemjakhar/expliot_framework
    - Cloudfuzz – Countinous Fuzzing framework
    - Hacksys Extreme Vulnerable Driver - http://www.payatu.com/hacksys-extreme-vulnerable-driver/
    - Damn Insecure and Vulnerable App for Android - http://www.payatu.com/damn-insecure-and-vulnerable-app/
  - In-house Fuzz testing Infrastructure
  - Mobile/Windows kernel/IoT exploitation training – Blackhat, Brucon, Hack In Paris, HITB and Corporate trainings

Agenda

- Introduction to ICS
  - ICS jargons
  - Typical Network
  - Process communication protocols
  - AST
- Introduction to Hardware Analysis
  - I2C
  - SPI
- Tools for hardware analysis
- Prizes
- CTF url

Payatu

Introduction

Industrial control system (ICS) is a general term that encompasses several types of control systems used in industrial production, including supervisory control and data acquisition (SCADA) systems, distributed control systems (DCS), and other smaller control system configurations such as programmable logic controllers (PLC) often found in the industrial sectors and critical infrastructures

Payatu

Introduction

Payatu

ICS Jargon

- SCADA
- HMI
- PLC/RTU

Payatu

ICS Jargons - SCADA

- **SCADA** (**supervisory control and data acquisition**) is a type of industrial control system (ICS)
- A system for remote monitoring and control that operates with coded signals over communication channels (using typically one communication channel per remote station)
- Industrial control systems are computer-controlled systems that monitor and control industrial processes that exist in the physical world.
- Used in water treatment and distribution, wastewater collection and treatment, oil and gas pipelines, electrical power transmission, wind farms, civil defence, siren systems, and large communication systems.

Payatu

ICS Jargons - SCADA

Payatu

ICS Jargons - HMI

- **Human–machine interface (HMI)** is the apparatus or device which presents processed data to a human operator.
- Human operator can monitors and interacts with the process.
- HMI is a client that requests data from a data acquisition server.

Payatu

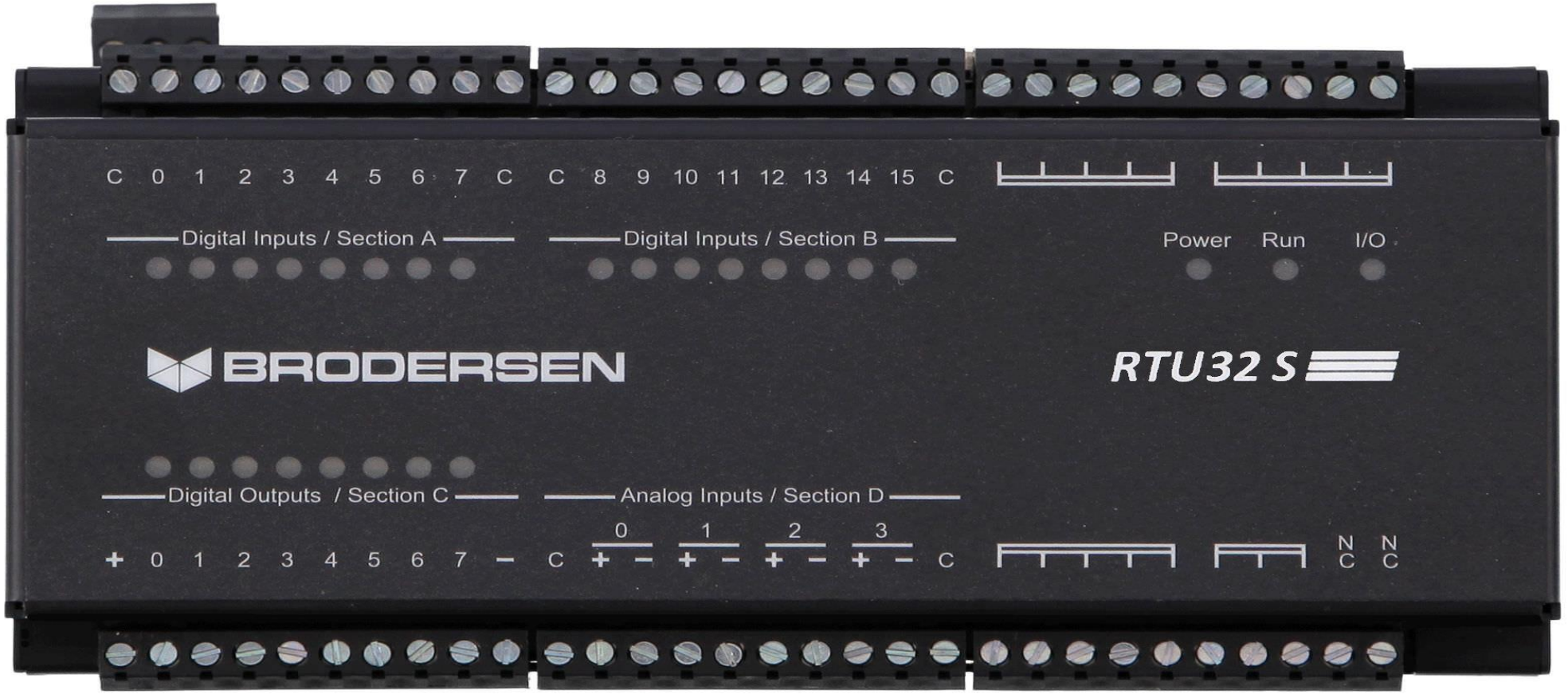
ICS Jargons - HMI

ICS Jargons - RTU

- **Remote terminal units (RTUs)** connect to sensors in the process and convert sensor signals to digital data.
- Have telemetry hardware capable of sending digital data to the supervisory system, as well as receiving digital commands from the supervisory system.

Payatu

ICS Jargons - RTU



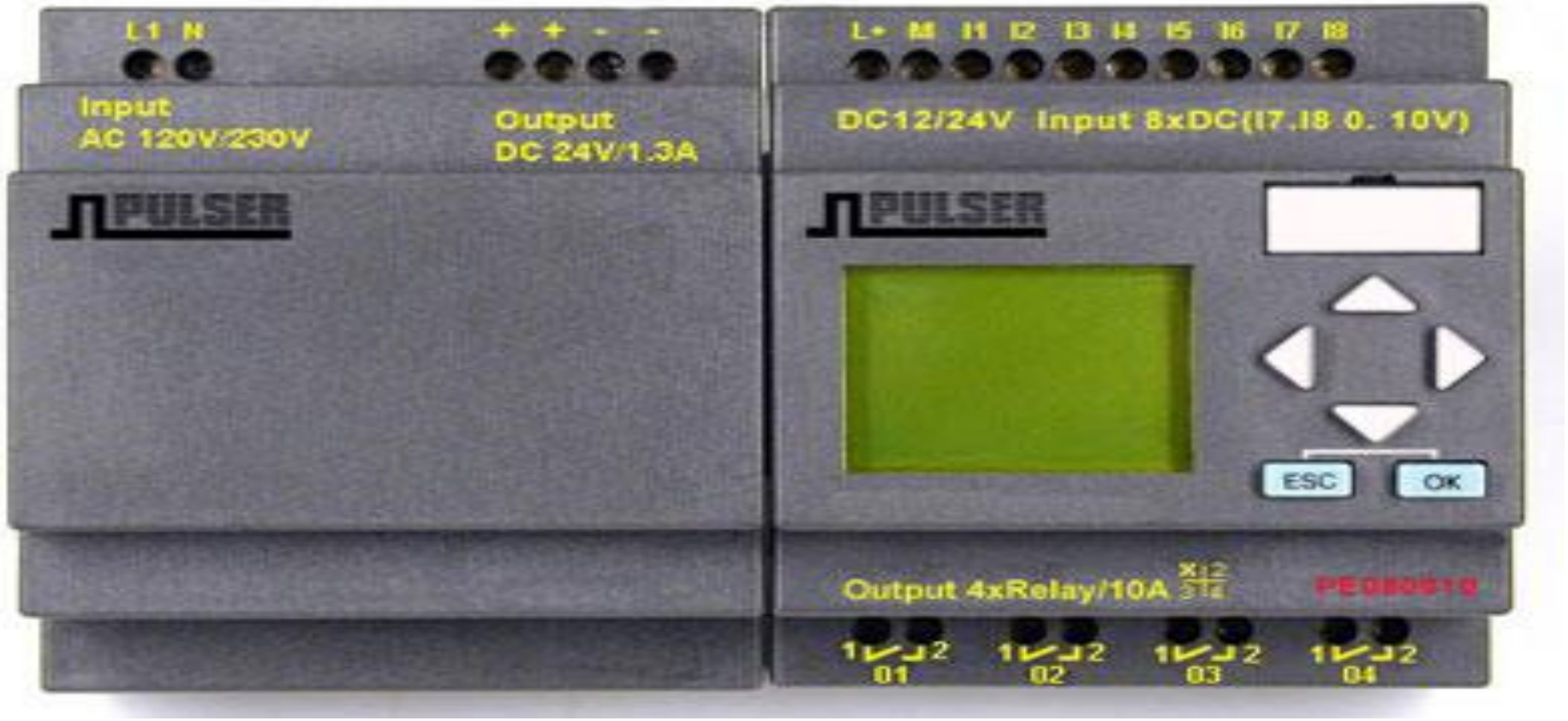
Payatu

ICS Jargons - PLC

- **Programmable logic controller (PLCs)** connect to sensors in the process and converting sensor signals to digital data.
- PLCs have more sophisticated embedded control capabilities, typically one or more IEC 61131-3 programming languages, than RTUs
- PLCs are sometimes used in place of RTUs as field devices because they are more economical, versatile, flexible, and configurable.

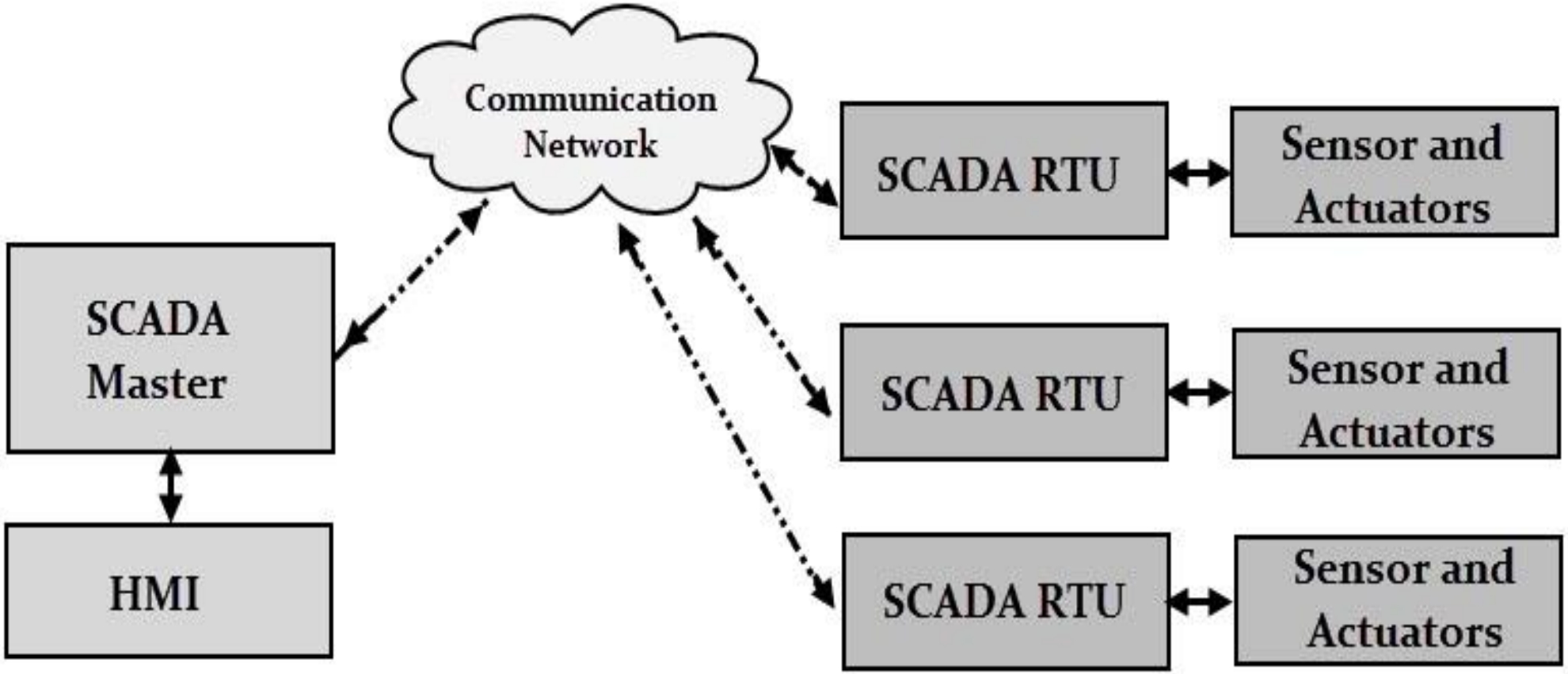
Payatu

ICS Jargons - PLC



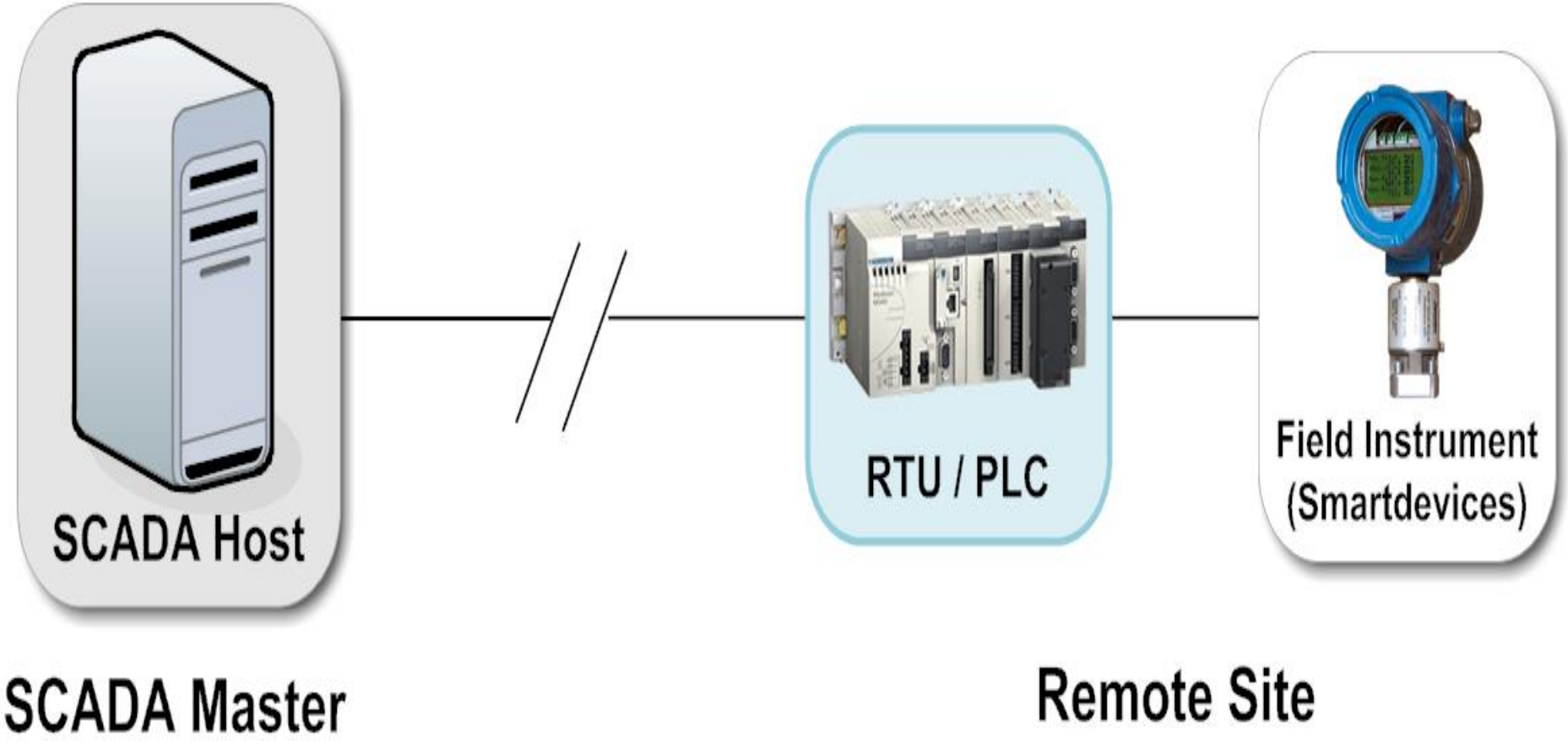
Payatu

Typical ICS network



Payatu

Typical ICS network



Payatu

Communication Protocols

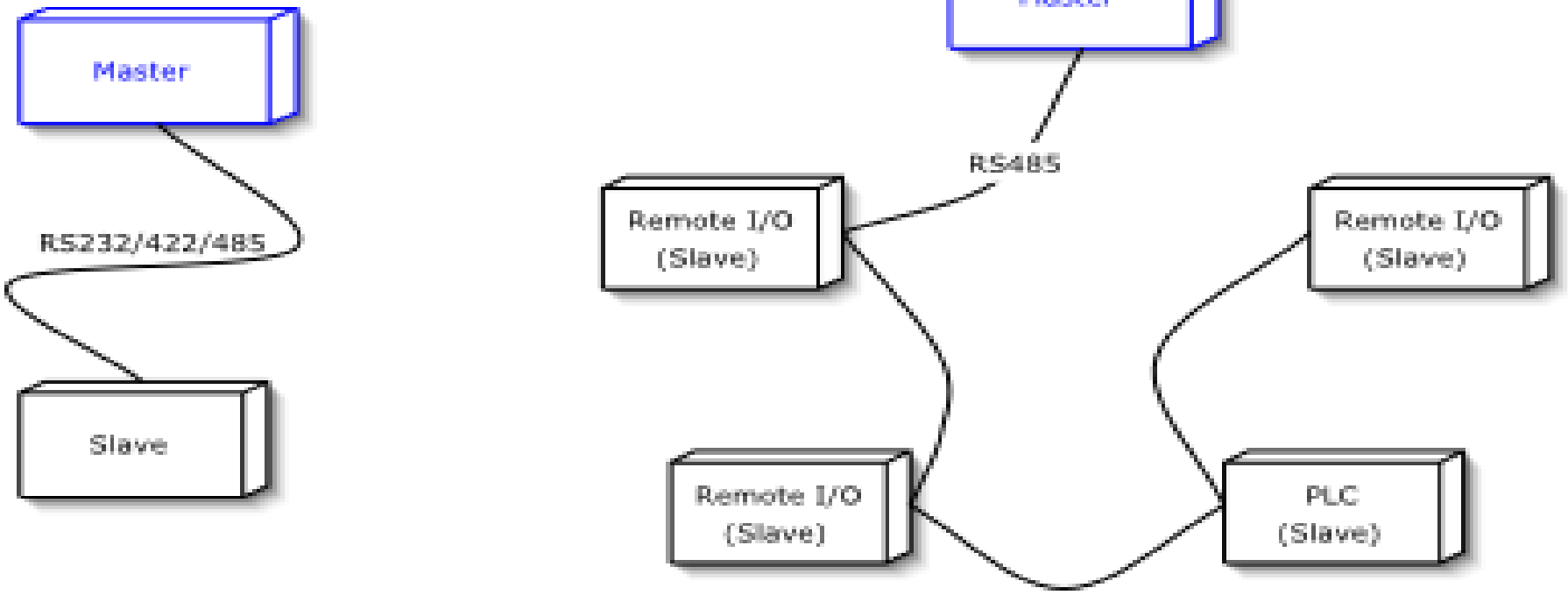
- Modbus serials/TCP - process and meter reading
- DNP3 - power system automation
- Profibus - prosess
- BACnet - building automation…..etc

Payatu

Communication Protocols - Modbus

- **Modbus** is a serial communications protocol originally published by Modicon (now Schneider Electric) in 1979 for use with its programmable logic controllers (PLCs).
- Developed for industrial application
- Royalty-Free
- One of the standards for industrial communication

Communication Protocols - Modbus



Payatu

Communication Protocols - Modbus

- **- Working**
- Master / Slave Protocol
- Master must regularly poll slaves to get information
- Modbus addresses are 8 bit long, so only 247 slaves per master
- There is no object description: a request returns a value, without any context or unit

- Security Issues

- Clear-Text
- No authentication

Payatu

Communication Protocols - Modbus

- Modbus was originally made for serial communications
- However it is now often used over TCP

Name	Length (bytes)	Function
Transaction identifier	2	For synchronization between messages of server & client
Protocol identifier	2	Zero for Modbus/TCP
Length field	2	Number of remaining bytes in this frame
Unit identifier	1	Slave address (255 if not used)
Function code	1	Function codes as in other variants
Data bytes	n	Data as response or commands

Payatu

Communication Protocols - Modbus

- The most common Modbus functions allow to read and write data from/to a PLC
- Other functions, such as file read and diagnostic function also exist

Function Name	Function Code
Read Coils	1
Write single coil	5
Read Holding register	3
Write single register	6
Write multiple register	16
Read/Write multiple registers	23

Introduction – AST – Above Surface storage Tank

- Gas Station Tanks
- Backup Generator Tanks
- Terminal Station Tanks
- Other Tanks?

10001
Automated
Tank Gauge

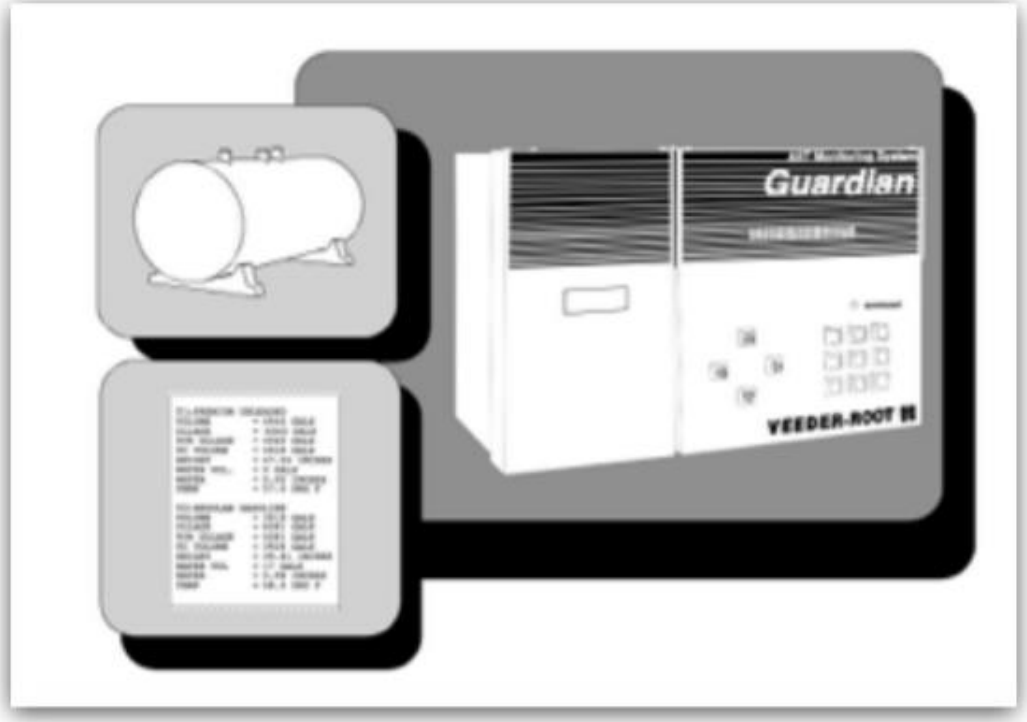
I20100
JUL 6, 2015 10:27 PM

LINCOLN ENERGY SOLUT
4227 CROMWELL RD
CHATTANOOGA,TN 37421

IN-TANK INVENTORY

TANK	PRODUCT	VOLUME	TC VOLUME	ULLAGE	HEIGHT	WATER	TEMP
1	BIO DIESEL 1	15417	15214	14670	67.29	0.96	89.82
2	BIO DIESEL 2	15352	15152	14735	67.06	4.84	89.58
3	BIO DIESEL 3	14847	14653	15240	65.32	4.52	89.54

Introduction – AST – Above Surface storage Tank



Payatu

Introduction to Hardware Analysis

- Components on a board communicate with each other
- Different methods of communication
- Some protocols used for communication
  - UART
  - I2C
  - SPI
  - JTAG
  - And more…

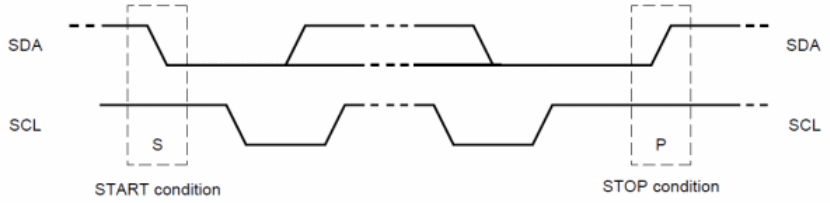
Payatu

I2C

- I2C – Inter-Integrated Circuit
- Invented by Philips(Now NXP)
- Serial protocol for 2-wire interface
- connects low speed peripherals to similar peripherals on the same board
- Master-slave architecture
  - Master – Device that generates the clock and initiates the communication
  - Slave – Device that listens on the bus and responds when addressed by the master
- Communication
  - One master – one/many slave(s)
  - Multi-master
- Bus
  - SDA – Serial Data Line
  - SCL – Serial Clock Line
  - Both lines are pulled high(up) via resistor to positive voltage
- No. of devices on a bus is almost unlimited

I2C – Serial Data Transfer

- Command from master
  - Starts with a START Condition
    - START = SDA High to Low transition
  - Ends with a STOP Condition
    - STOP – SDA Low to High transition
  - For both conditions SCL (clock) should be high
  - After START condition bus is considered busy till STOP condition



- Data transfer
  - One bit is transferred in each clock pulse
  - SDA signal can only change when SCL (clock) is low
  - Data is transferred in 8-bit (1 byte) packets
  - Each byte is followed by an ACK bit by the slave

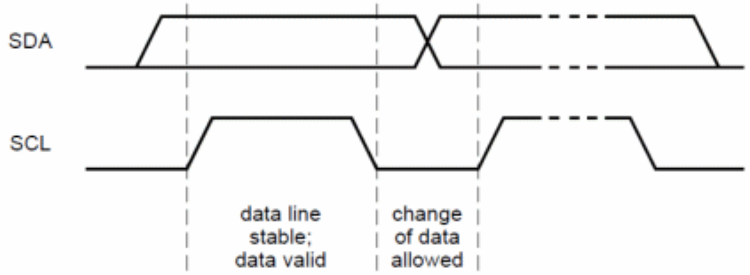
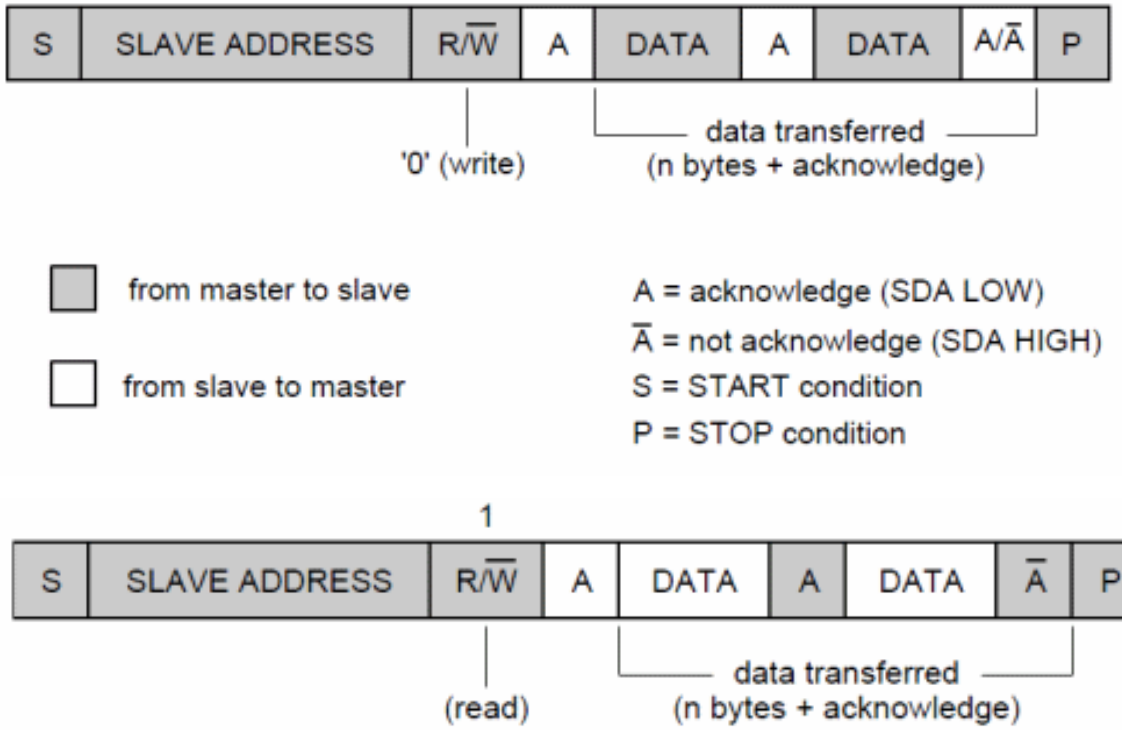


Image source: http://i2c.info/i2c-bus-specification

I2C – Serial Data Transfer

- Slave devices have a unique 7-bit address on the bus for receiving commands
- Bit 0 is for read(1) or write(0)



Payatu

SPI

- SPI – Serial Peripheral Interface
- Synchronous serial communication bus
- Developed by Motorola
- Used for short distance communication (between peripherals typically on the same board)
- Higher throughput than I2C
- Master-slave architecture
  - One master – One/many slave(s)
  - Master selects the slave, to communicate, through slave select line
- Full Duplex communication
- 4-wire Interface

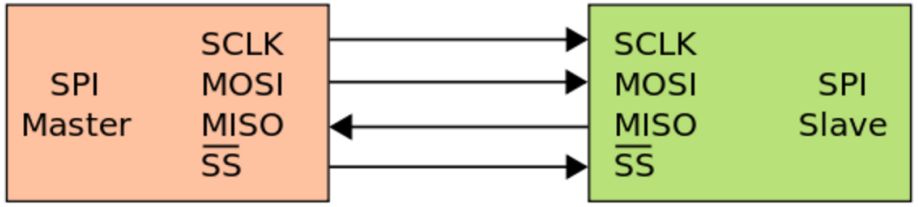
Payatu

SPI – Bus Interface

Line name	Description	Alternative naming convention
SCLK	Serial Clock	SCK, CLK
MOSI	Master output Slave input	SIMO, SDI(for slave devices), DI, DIN, SI, MTST
MISO	Master input Slave output	SOMI, SDO (for slave devices), DO, DOUT, SO, MRSR
SS	Slave Select	nCS, CS, CSB, CSN, EN, nSS, STE, SYNC, SSQ

SPI – Bus Interface

One Master – One Slave



One Master – Many Slave

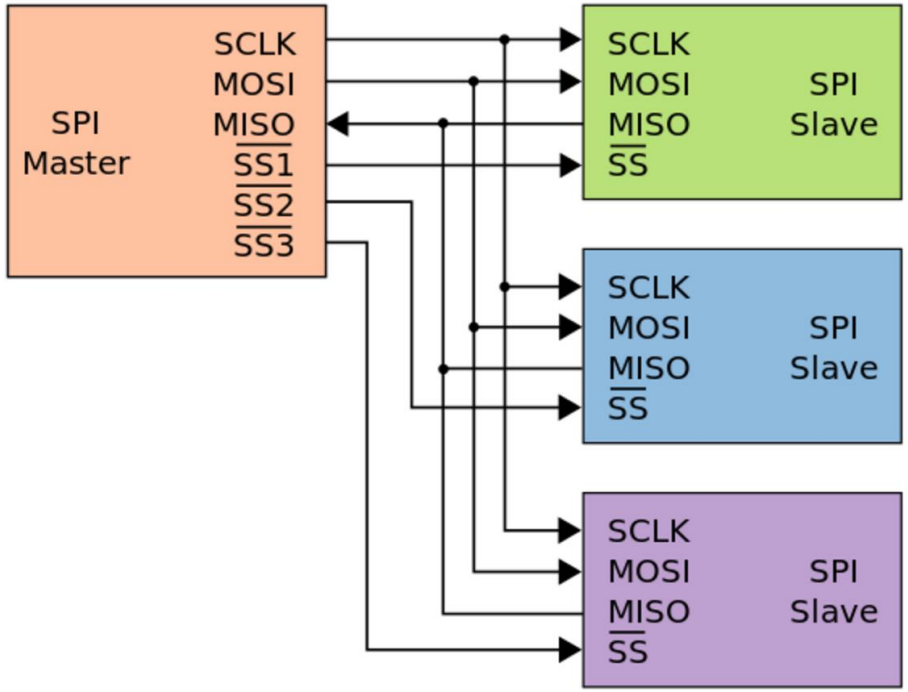


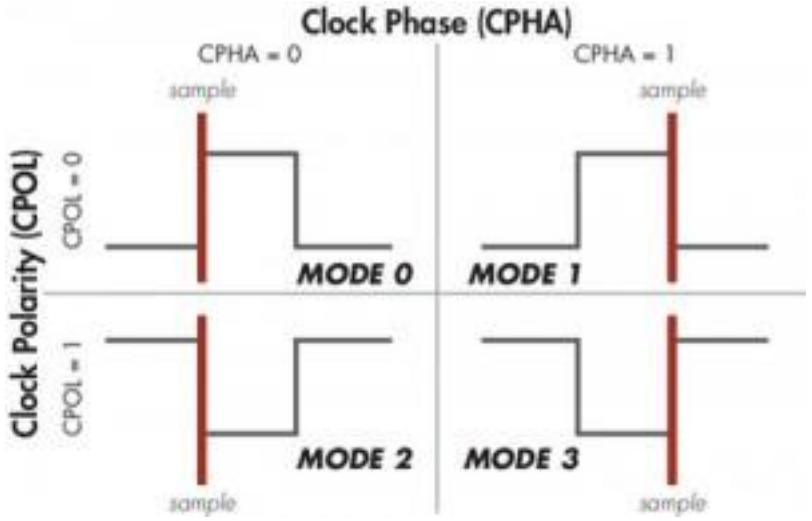
Image Source: https://en.wikipedia.org/wiki/File:SPI_single_slave.svg
https://en.wikipedia.org/wiki/File:SPI_three_slaves.svg

SPI – Clock Polarity and Phase

- CPOL = 0
  - The base value of clock is 0. Active state is 1 and idle state is 0
  - Rising/falling – Leading edge of the clock is rising and trailing edge is falling
- CPOL = 1
  - The base value of the clock is 1. Active state is 0 and idle state is 1
  - Falling/Rising – Leading edge of the clock is falling and the trailing edge is rising

	CPOL = 0	**CPOL = 1**
CPHA = 0	Data is captured/sampled on clock's rising edge (and output on falling edge)	Data is captured/Sampled on the clock's falling edge (and output on rising edge)
CPHA = 1	Data is captured/Sampled on the clock's falling edge (and output on rising edge)	Data is captured/sampled on clock's rising edge (and output on falling edge)

SPI – Clock Polarity and Phase



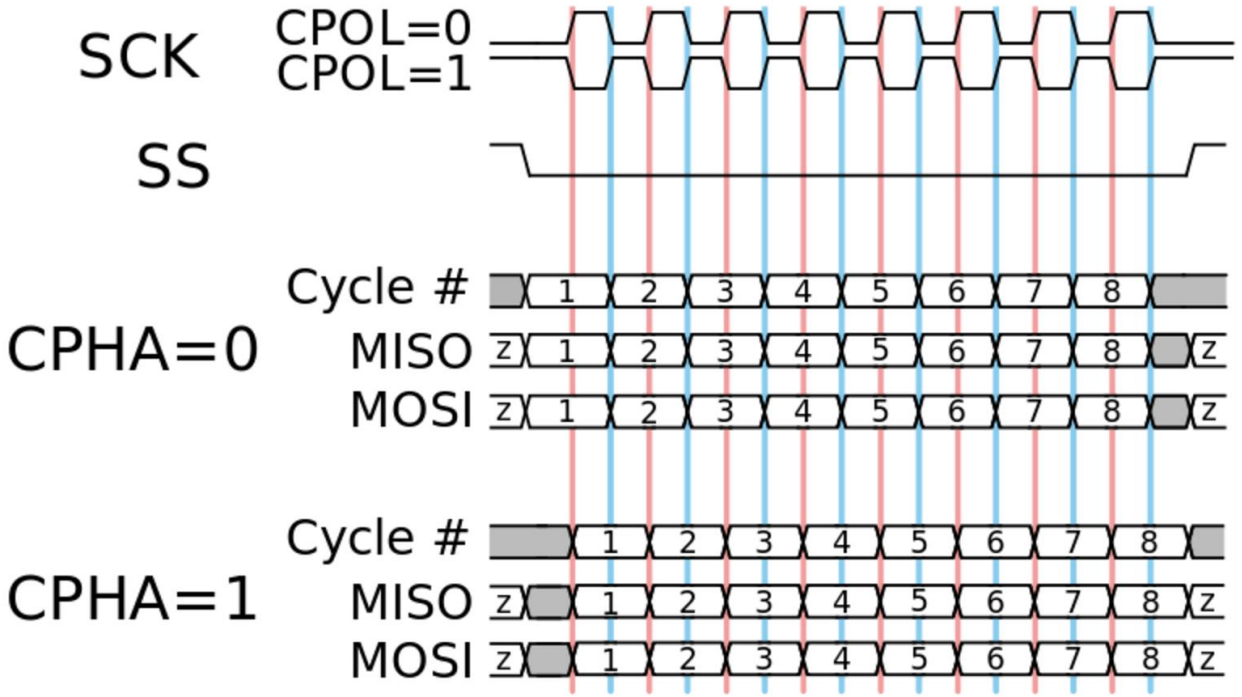


Image Source: https://upload.wikimedia.org/wikipedia/commons/thumb/6/6b/SPI_timing_diagram2.svg/800px-SPI_timing_diagram2.svg.png
http://www.totalphase.com/media/blog/2013/08/CheetahSPIClockPhases-300x193.jpg

Payatu

UART

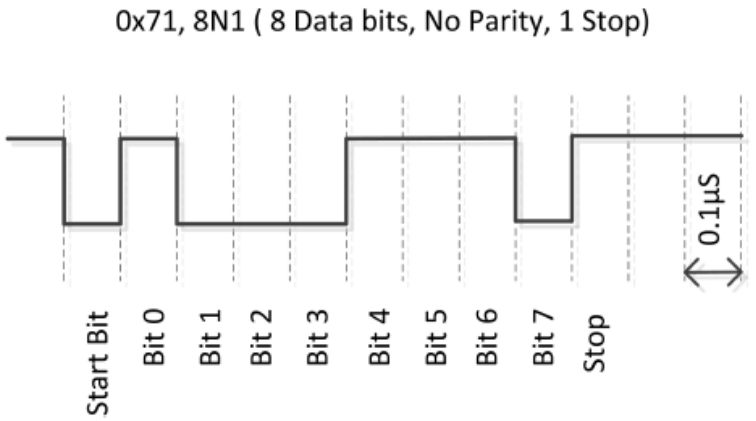
- **U**niversal **A**synchronous **R**eceiver and **T**ransmitter
- UART Protocols is a serial communication with two wire protocol.
- The data cable signal lines are labelled as **Rx** and **Tx**.
- Simple way to transfer data directly to and from microcontrollers without the need of any intermediary hardware
- Most commonly used in embedded devices
-

UART

- It has one start bit, 5 to 8-bit data and one stop bit mean the 8-bit data transfer ones signal is high to low.

Bit No.	1	2	3	4	5	6	7	8	9	10	11
	Start bit	5-8 data bits								Stop bit(s)	
	Start	Data 0	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7	Stop	

- Start Bit – Low / Space / 0 / Positive voltage
- Stop Bit – High / Mark / 1 / Negative voltage
- Parity Bit – Optional, used if no. of bits per character are not 9
- Data bits – 5-8 (or even 9, in which case no parity bit is used).
  - Least significant bit sent first



UART

- Tranmitting UART converts parallel data into serial and transmits
- Receiving UART converts received serial data into parallel
- Data flows from the Tx pin of the transmitting UART to the Rx pin of the receiving UART

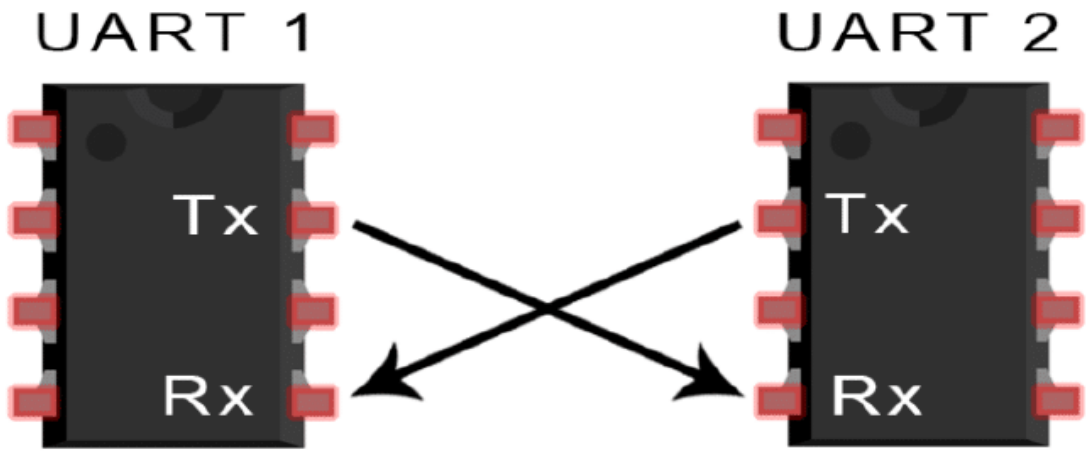


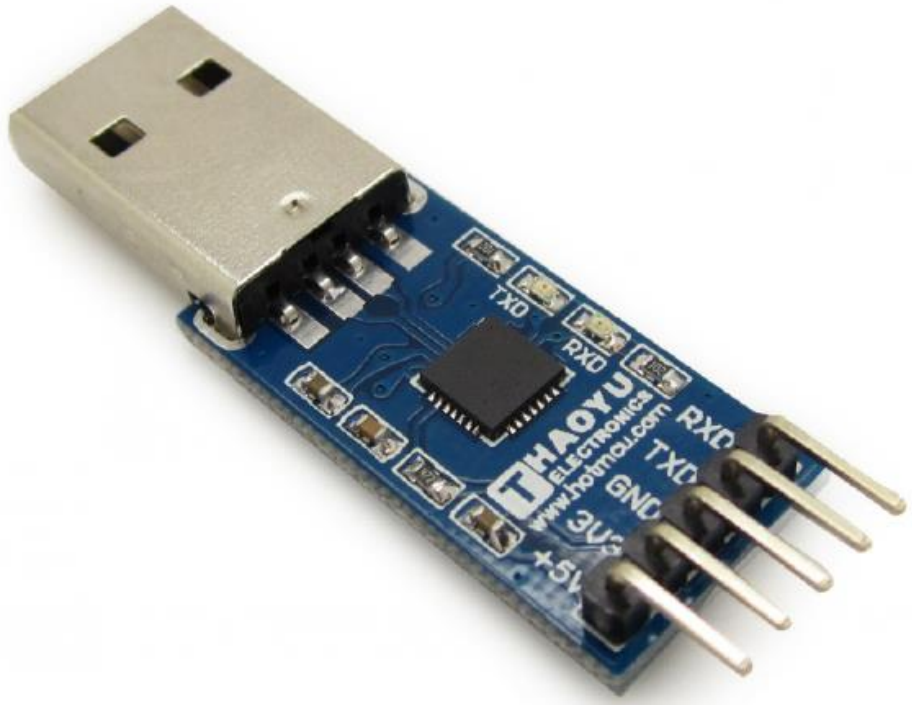
Image Source: http://www.circuitbasics.com/wp-content/uploads/2016/01/Introduction-to-UART-Basic-Connection-Diagram-768x376.png

Tools for Hardware Analysis

1. DMM (Digital MultiMeter)

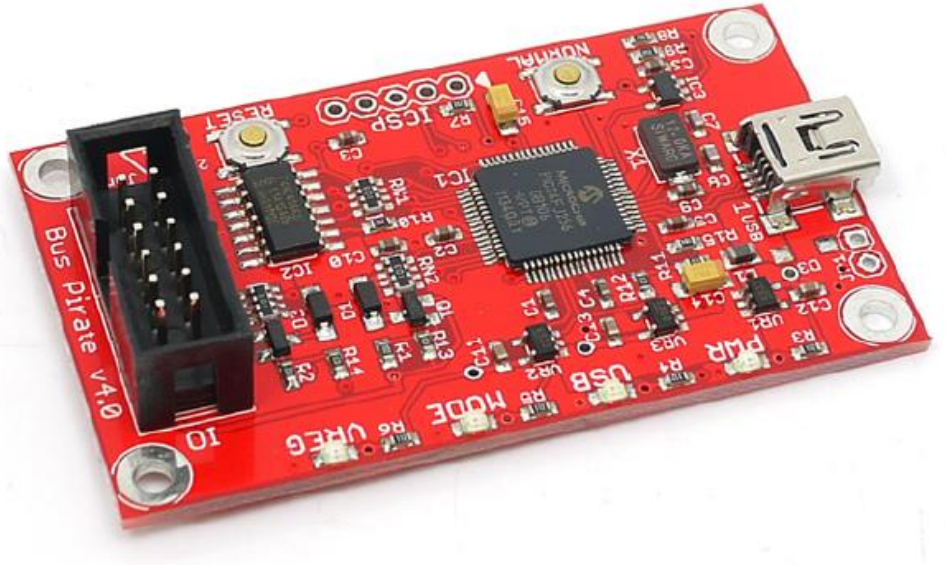
Tools for Hardware Analysis

1. DMM (Digital MultiMeter)
2. USB UART Serial Adapter (CP210x)



Tools for Hardware Analysis

1. DMM (Digital MultiMeter)
2. USB UART Serial Adapter (CP210x)
3. Bus Pirate

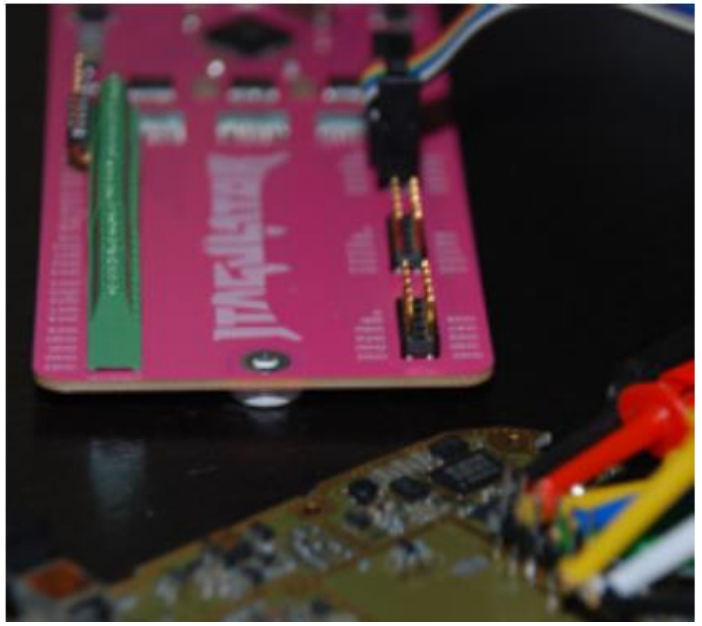


Tools for Hardware Analysis

1. DMM (Digital MultiMeter)
2. USB UART Serial Adapter (CP210x)
3. Bus Pirate
4. Saleae logic Analyzer

Tools for Hardware Analysis

1. DMM (Digital MultiMeter)
2. USB UART Serial Adapter (CP210x)
3. Bus Pirate
4. Saleae logic Analyzer
5. JTAGulator/Jtagenum



Tools for Hardware Analysis

1. DMM (Digital MultiMeter)
2. USB UART Serial Adapter (CP210x)
3. Bus Pirate
4. Saleae logic Analyzer
5. JTAGulator
6. CH341a

PRIZES

- Lan Turtle
- Geekcreit® UNO R3 Basic Starter Learning Kit No Battery Version For Arduino
- 1x Ubertooth One 2.4 GHz Duck Antenna – Black
- Payatu – DIVA – Vulnerable Board
- Payatu T-Shirts
- Nullcon Security Conference 2018 Goa, India Pass
- And some stickers

CTF URL – Register and Start ICS hacking

- https://icshumlactf.nullcon.net/

Thank You