INTRODUCTION TO IOT

- Conducted by **Al Arabi**
BSc. EEE, BUET
IoT Developer, Oceaniot Inc. Canada
Lecturer, NUB, Bangladesh
Excom member IEEE RAS BD 2018
Excom Member IEEE BDS 2019
Team and Project Lead, Team Attendant

- With the help of **Akil Hamid Chowdhury**
EM-Lead, Team Attendant

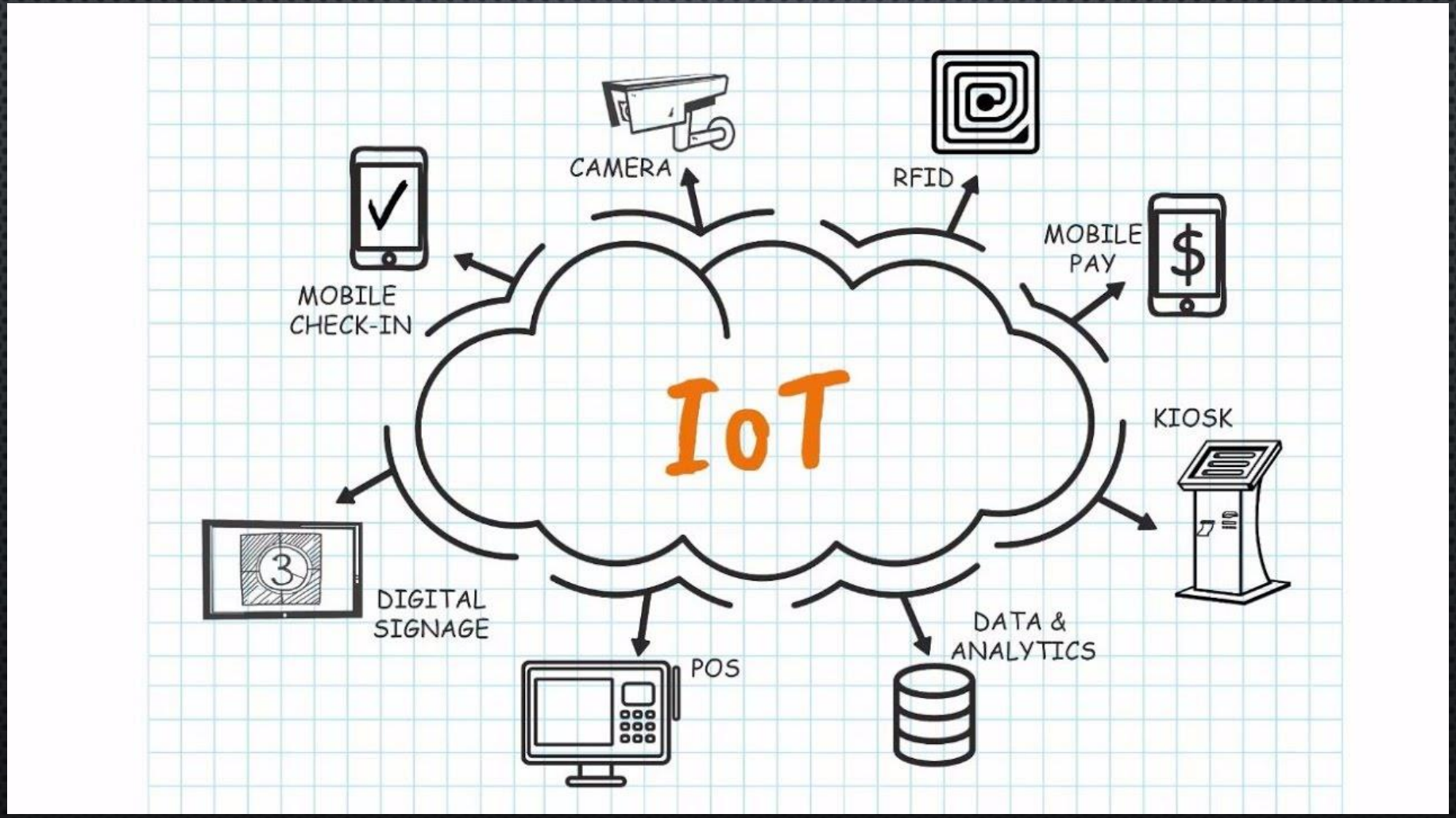
Powered by:

IEEE Robotics & Automation Society
Bangladesh Chapter

WHAT IS IOT

- THE INTERNET OF THINGS (IOT) IS THE NETWORK OF PHYSICAL DEVICES, HOME APPLIANCES, AND OTHER ITEMS EMBEDDED WITH ELECTRONICS, SOFTWARE, SENSORS, ACTUATORS…… (ACTUALLY EVERYTHING)

WHAT IS IOT


CAMERA
RFID
MOBILE CHECK-IN
MOBILE PAY
IoT
KIOSK
DIGITAL SIGNAGE
POS
DATA & ANALYTICS

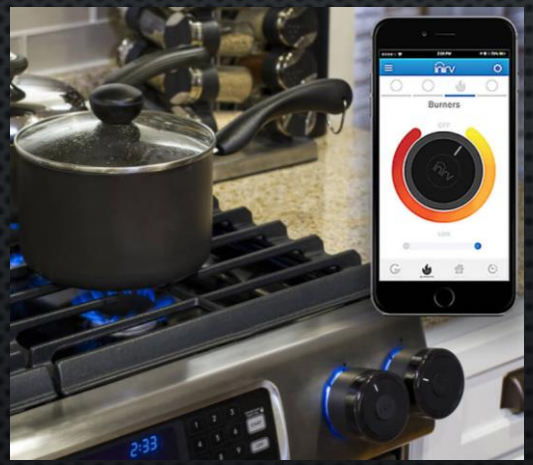
WHAT IS IOT

LIFE WITH IOT

LIFE WITH IOT

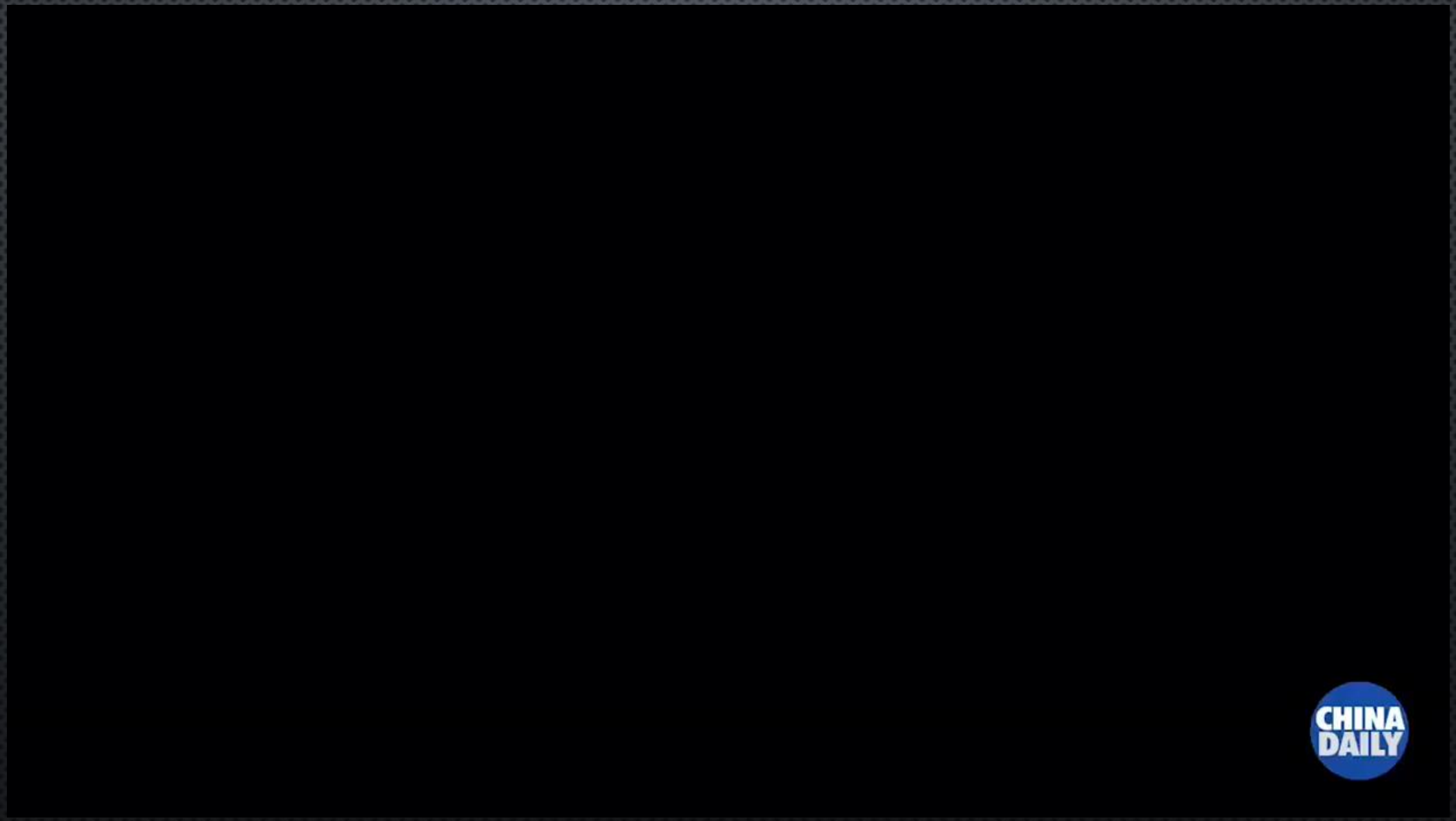
2015: A concept

2018: Almost a reality



LIFE WITHOUT IOT VS WITH IOT

IOT IN ROBOTICS



IOT IN ROBOTICS

Warehouse

One million square feet

23 acres

Football field is 1.32 acres

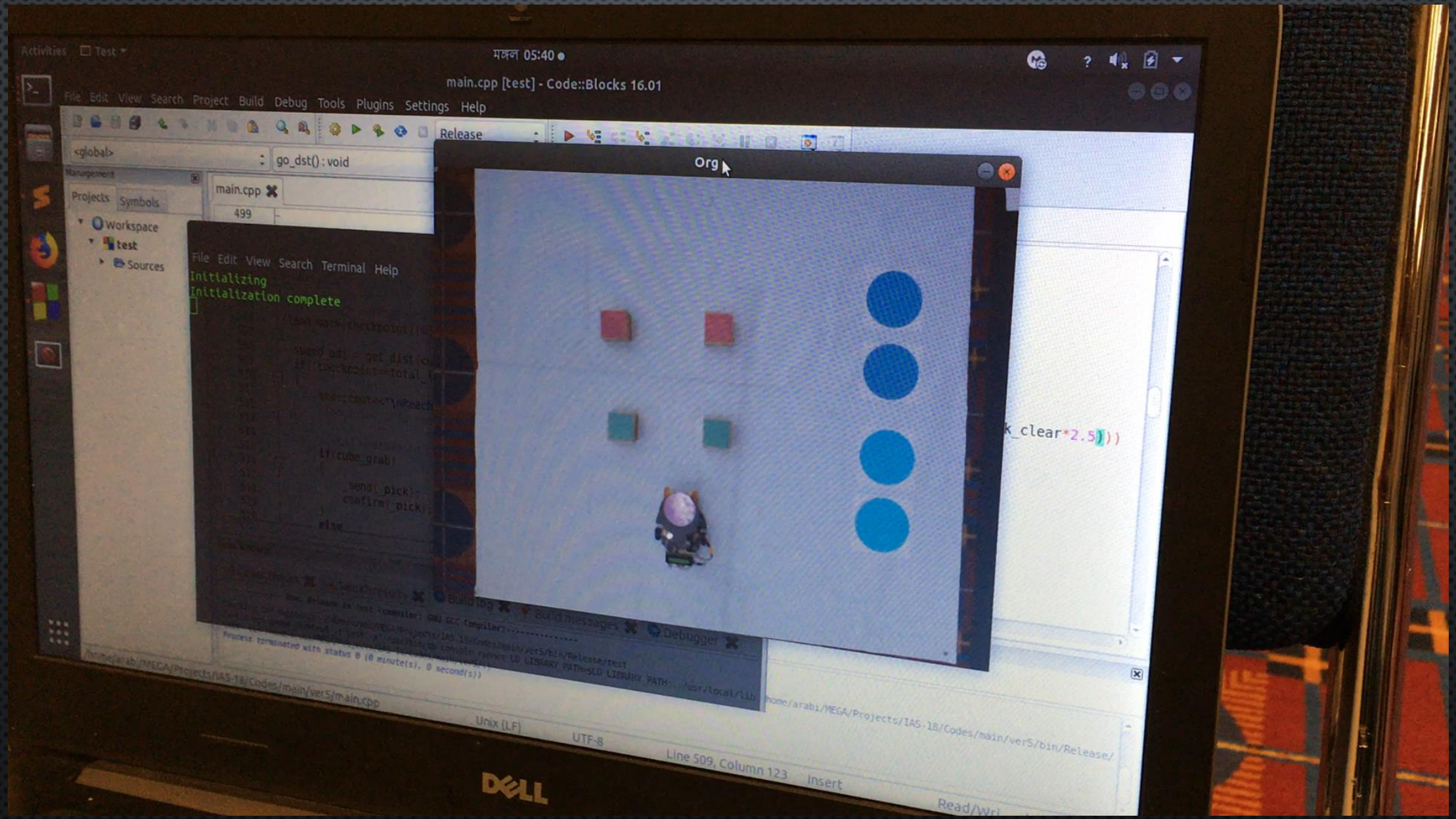
So around 23/1.32 = 17….

IOT IN ROBOTICS

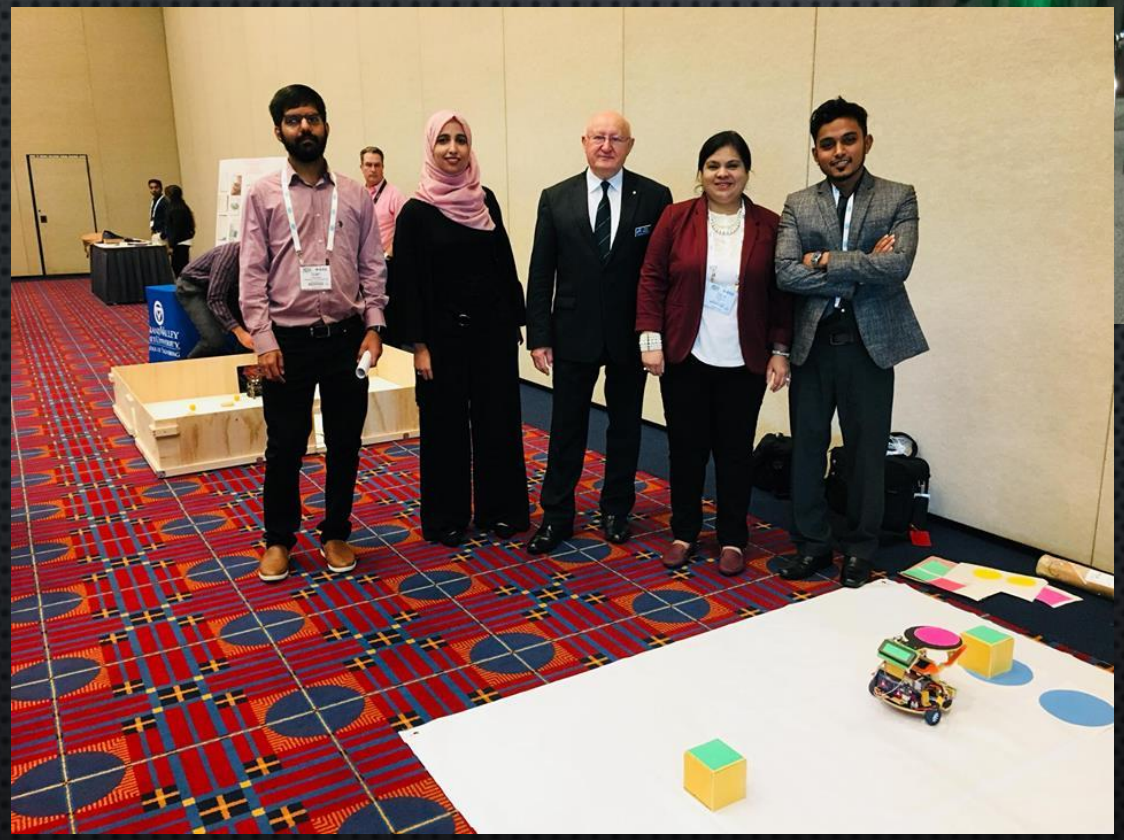
amazon

IOT IN ROBOTICS

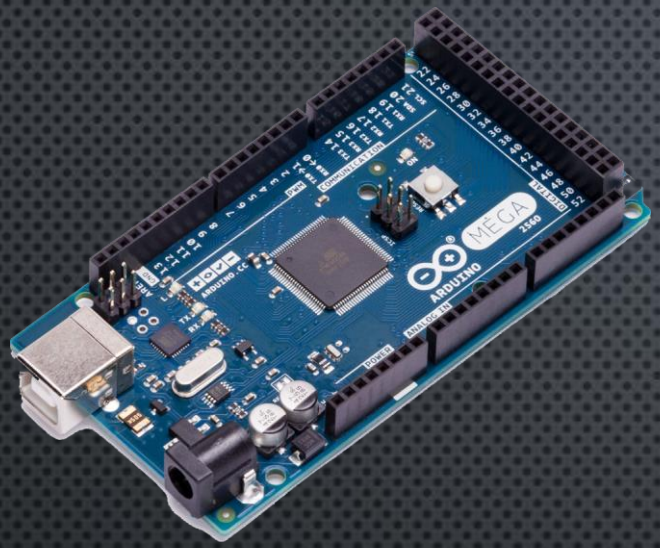
OUR ACHIEVEMENT: IEEE IAS CMD ROBOTICS CONTEST

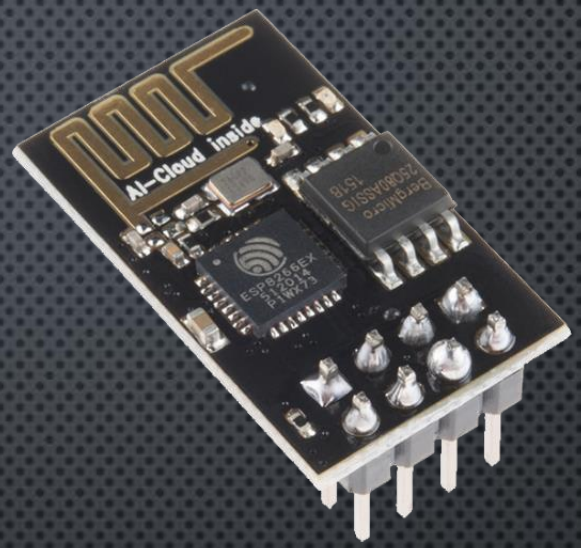


2ND POSITION ACROSS THE GLOBE

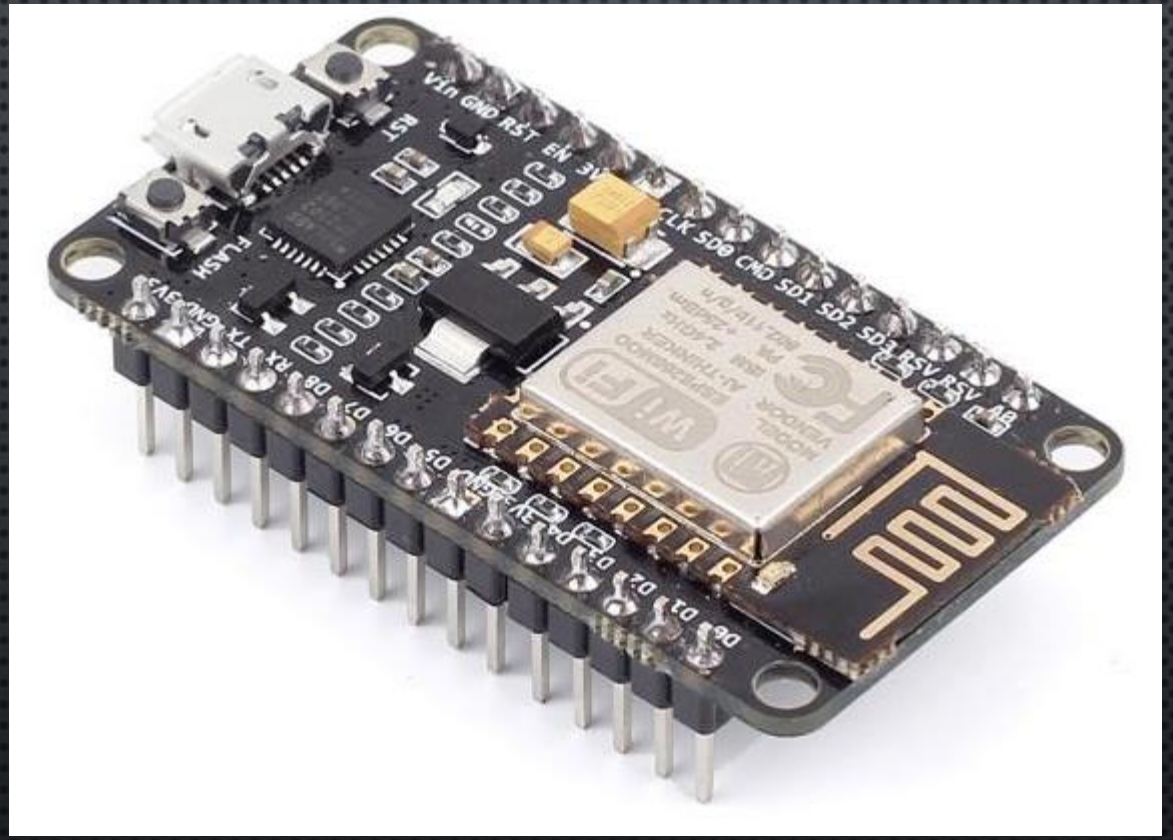


COMPONENTS INTRO

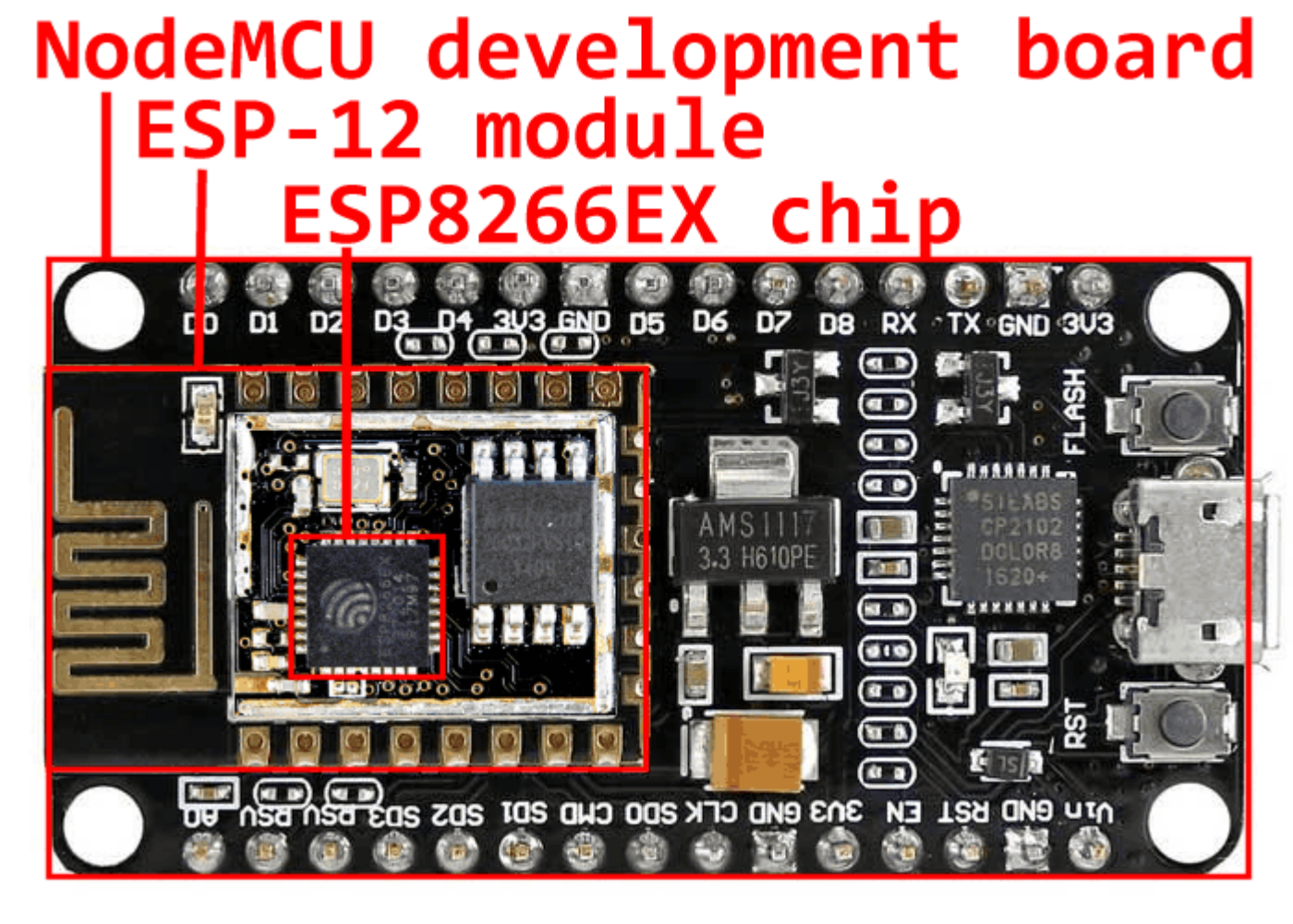




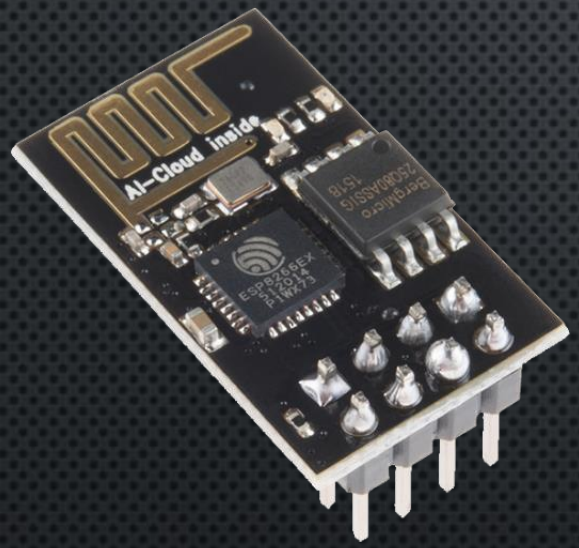
NODE MCU

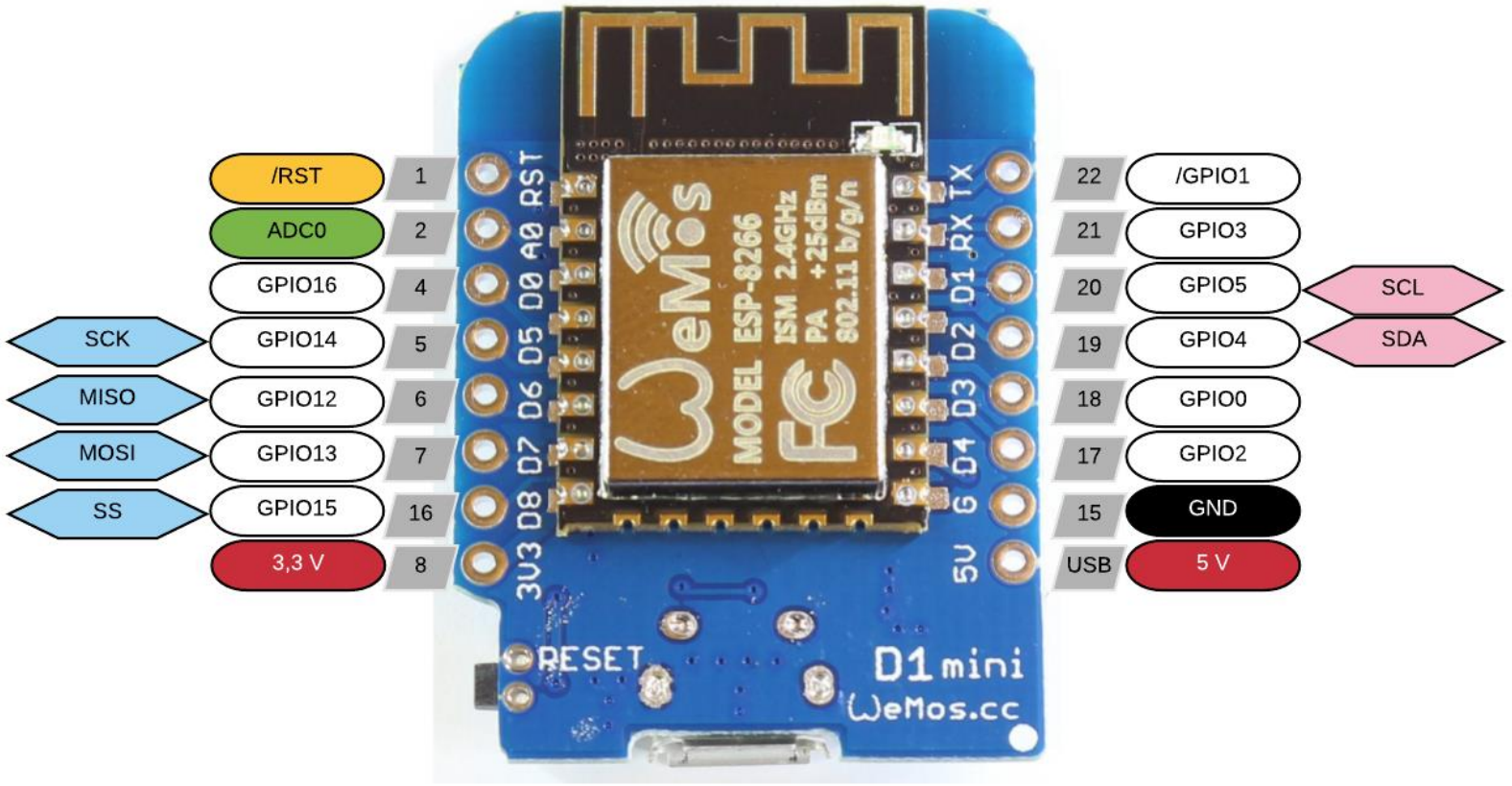

Vin GND RST EN
RST
FLASH
CLK SD0 CMD SD1 SD2 SD3 RSV RSV RSV
3V3 GND TX RX D8 D7 D6 D5 3V3 GND D4 D3 D2 D1 D0

NODE MCU

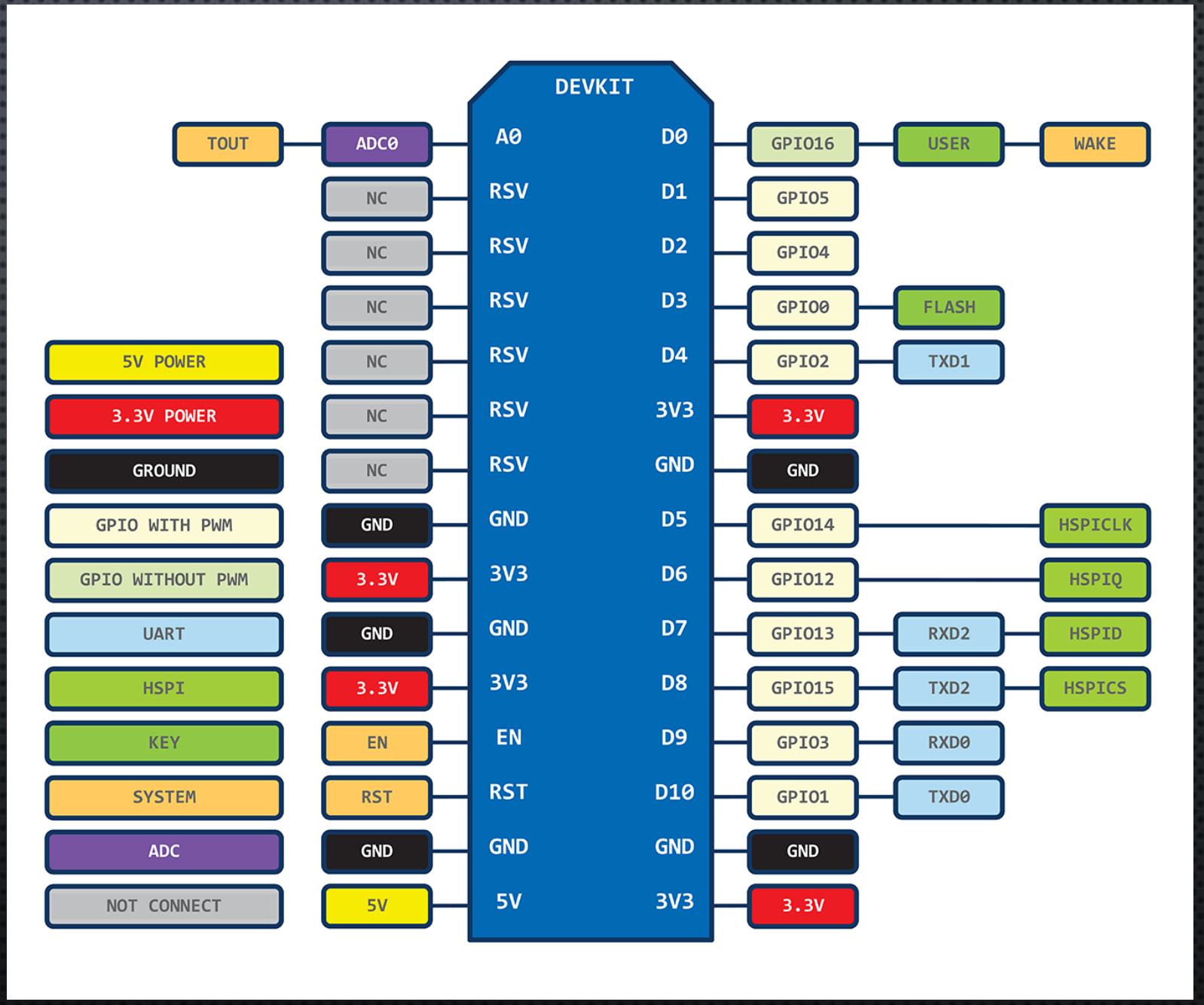

NodeMCU development board
ESP-12 module
ESP8266EX chip
D0 D1 D2 D3 D4 3V3 GND D5 D6 D7 D8 RX TX GND 3V3
FLASH
RST
AMS1117
3.3 H610PE
CP2102
Vin GND RST EN 3V3 GND CLK SDO CMD SD1 SD2 SD3 RSV RSV A0

ESP 8266


AI-Cloud inside


/RST 1
ADC0 2
GPIO16 4
SCK GPIO14 5
MISO GPIO12 6
MOSI GPIO13 7
SS GPIO15 16
3,3 V 8
22 /GPIO1
21 GPIO3
20 GPIO5 SCL
19 GPIO4 SDA
18 GPIO0
17 GPIO2
15 GND
USB 5 V
RST A0 D0 D5 D6 D7 D8 3V3
TX RX D1 D2 D3 D4 G 5V
MODEL ESP-8266
ISM 2.4GHz
PA +25dBm
802.11 b/g/n
RESET
D1 mini
WeMos.cc
ESP pin
Analog
SPI
Power
Control
I2C

NODE MCU PINOUT



SPECS

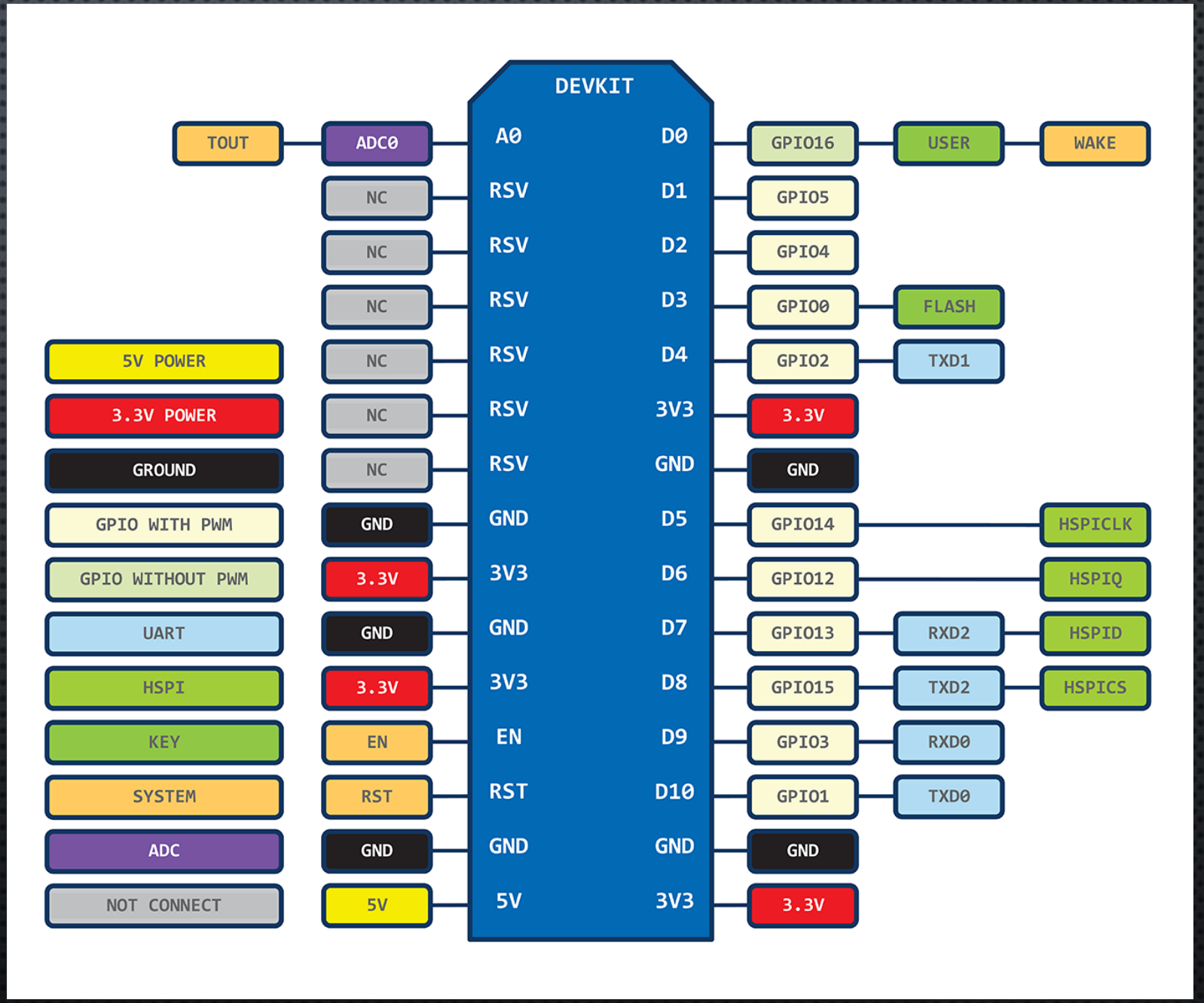
- The term “NodeMCU” by default refers to the firmware rather than the dev kits
- ESP8266 Was built with AT-Commands
- NodeMCU firmware was developed so that AT commands can be replaced with Lua scripting
- NodeMCU is an open source Lua based firmware for the ESP8266
- NodeMCU is implemented in C
- NodeMCU programming model is similar to that of Node.js, only in Lua

SPECS

- Type : Single-board microcontroller
- Operating system : XTOS
- CPU : ESP8266
- Memory : 128kBytes
- Storage : 4MBytes
- Power By : USB
- Power Voltage : 3v ,5v (used with 3.3v Regulator)
- Code : Arduino Cpp
- IDE Used : Arduino IDE
- GPIO : 10

FUNCTIONALITY

pinMode(),
digitalRead(),
digitalWrite(),
analogWrite()



Analog ADC

ESP8266EX also integrates a generic purpose 10-bit analog ADC.

The ADC range is from 0V to 1.0V.

The ADC cannot be used when the chip is transmitting.

PWM

analogWrite(pin, value) enables software PWM

PWM may be used on pins 0 to 15.

Call analogWrite(pin, 0) to disable PWM on the pin.

Value may be in range from 0 to 1023. 0 to 255 is normal on an Arduino board, as it's an 8 bit register, but ESP8266 uses software PWM so 1023 is full duty cycle.

Pin Functions

The most usable pin functions are mapped to the macro SPECIAL, so calling pinMode(pin, SPECIAL) will switch that pin to UART RX/TX on pins 1 - 3, HSPI for pins 12-15 and CLK functions for pins 0, 4 and 5.

SPECIAL maps to:

0. CLK_OUT
1. TX0
2. TX1
3. RX0
4. CLK_XTAL
5. CLK_RTC
12. SPI_MISO
13. SPI_MOSI
14. SPI_CLK
15. SPI_SS

Pin Functions

GPIO	Inst Name	Function 0	Function 1	Function 2	Function 3	Function 4	At Reset	After Reset	Sleep
0	GPIO0_U	GPIO0	SPICS2			CLK_OUT	oe=0, wpu	wpu	oe=0
1	U0TXD_U	U0TXD	SPICS1		GPIO1	CLK_RTC	oe=0, wpu	wpu	oe=0
2	GPIO2_U	GPIO2	I2SO_WS	U1TXD		U0TXD	oe=0, wpu	wpu	oe=0
3	U0RXD_U	U0RXD	I2SO_DATA		GPIO3	CLK_XTAL	oe=0, wpu	wpu	oe=0
4	GPIO4_U	GPIO4	CLK_XTAL				oe=0		oe=0
5	GPIO5_U	GPIO5	CLK_RTC				oe=0		oe=0
6	*SD_CLK_U*	*SD_CLK*	*SPICLK*		*GPIO6*	*U1CTS*	*oe=0*		*oe=0*
7	*SD_DATA0_U*	*SD_DATA0*	*SPIQ*		*GPIO7*	*U1TXD*	*oe=0*		*oe=0*
8	*SD_DATA1_U*	*SD_DATA1*	*SPID*		*GPIO8*	*U1RXD*	*oe=0*		*oe=0*
9	*SD_DATA2_U*	*SD_DATA2*	*SPIHD*		*GPIO9*	*HSPIHD*	*oe=0*		*oe=0*
10	*SD_DATA3_U*	*SD_DATA3*	*SPIWP*		*GPIO10*	*HSPIWP*	*oe=0*		*oe=0*
11	*SD_CMD_U*	*SD_CMD*	*SPICS0*		*GPIO11*	*U1RTS*	*oe=0*		*oe=0*
12	MTDI_U	MTDI	I2SI_DATA	HSPIQ MISO	GPIO12	U0DTR	oe=0, wpu	wpu	oe=0
13	MTCK_U	MTCK	I2SI_BCK	HSPID MOSI	GPIO13	U0CTS	oe=0, wpu	wpu	oe=0
14	MTMS_U	MTMS	I2SI_WS	HSPICLK	GPIO14	U0DSR	oe=0, wpu	wpu	oe=0
15	MTDO_U	MTDO	I2SO_BCK	HSPICS	GPIO15	U0RTS	oe=0, wpu	wpu	oe=0
16	XPD_DCDC	XPD_DCDC	RTC_GPIO0	EXT_WAKEUP	DEEPSLEEP	BT_XTAL_EN	oe=1,wpd	oe=1,wpd	oe=1

Mode of Operation

0 Low
1 High
x floating

MODE	GPIO15	GPIO0	GPIO2
SDIO (BootSDCard)	1	x	x
UART (UploadCode)	0	0	x or 1
FLASH (NormalRunning)	0	1	x or 1

Mode of Operation

FLASH mode is when running the program. Take GPIO0 high or float or it will stall on first reset, intentional or not.

UART mode is how the code is uploaded to the chip and GPIO0 and GPIO15 must be low on boot to enter this mode.

SDIO mode is where the chip boots from an SD card. I don't think this is available to us yet.

Mode of Operation

Notes :- GPIO2 It is considered safer to pull it high with a resistor on boot rather than leave it floating to avoid possible chip damage

Maximum Current

When using a GPIO as output (i.e. to drive something such as an LED) it is important to note that the maximum output current is 12mA. (IMAX=12mA per Espressif datasheet)

R=(Vout-Vled)/I
so (3.3V-1.8Vred) at 12mA = 125Ω min for a red LED.

Begin With Arduino IDE