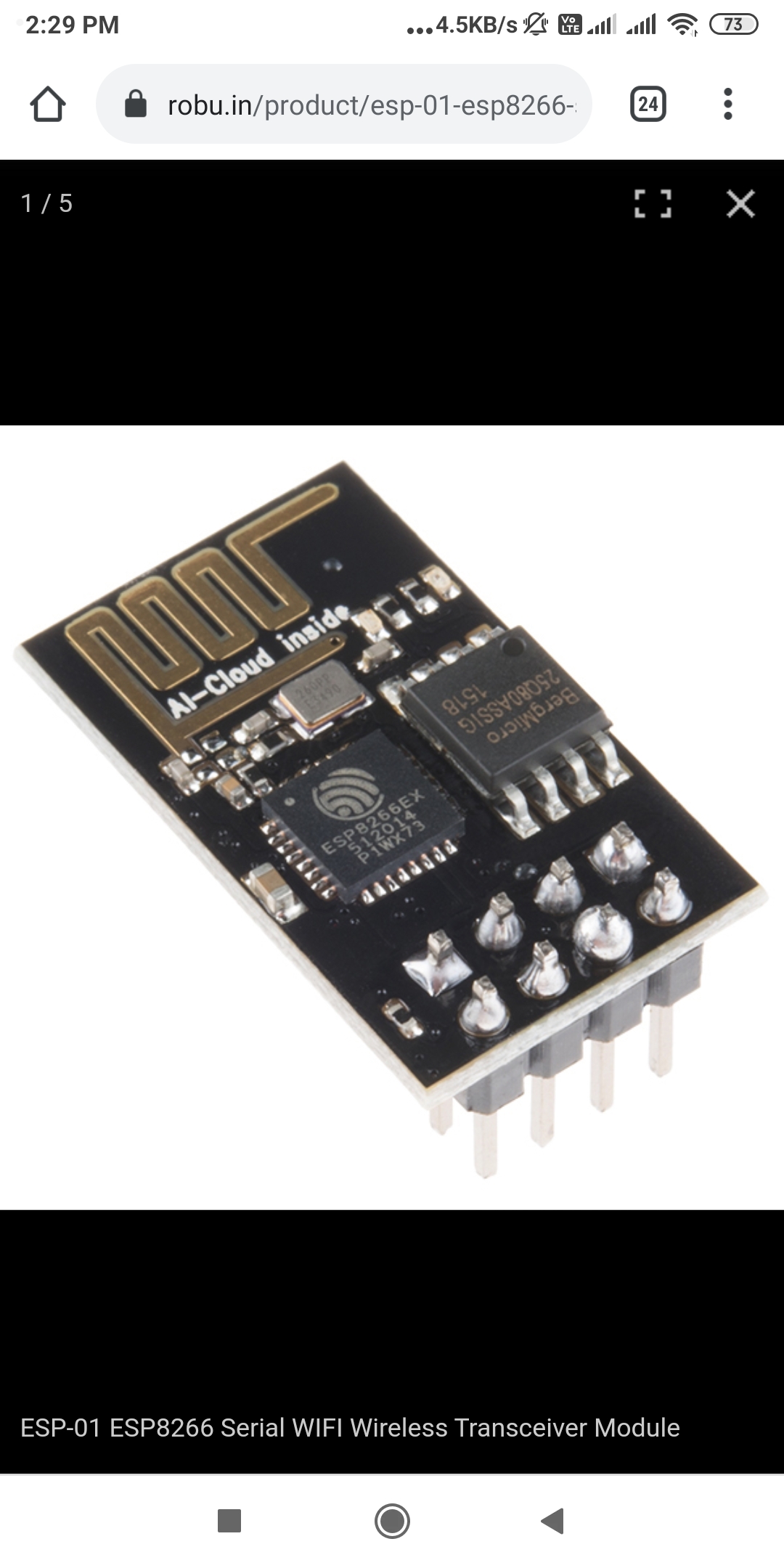
### ESP8266 ESP-01 Serial WiFi Wireless Transceiver Module



The ESP 01 ESP8266 Serial WIFI Wireless Transceiver Module is a self-contained SOC with integrated TCP/IP protocol stack that can give any microcontroller access to your WiFi network. The ESP8266 is capable of either hosting an application or offloading all Wi-Fi networking functions from another application processor.

Each ESP8266 module comes pre-programmed with an AT command set firmware, meaning, you can simply hook this up to your Arduino device and get about as much WiFi-ability as a WiFi Shield offers (and that’s just out of the box)! The ESP8266 module is an extremely cost-effective board with a huge, and ever-growing, community.

This module has a powerful enough on-board processing and storage capability that allows it to be integrated with the sensors and other application-specific devices through its GPIOs with minimal development up-front and minimal loading during runtime.

Its high degree of on-chip integration allows for minimal external circuitry, including the front-end module, which is designed to occupy minimal PCB area. The ESP8266 supports APSD for VoIP applications and Bluetooth co-existence interfaces, it contains a self-calibrated RF allowing it to work under all operating conditions and requires no external RF parts.

Note:

The ESP8266 Module is not capable of 5-3V logic shifting and will require an external Logic Level Converter. Please do not power it directly from your 5V dev board.

This new version of the ESP8266 WiFi Module has increased the flash disk size from 512k to 1MB.

**FEATURES:**

* Wi-Fi Direct (P2P), soft-AP
* Integrated TCP/IP protocol stack
* It features an integrated TR switch, balun, LNA, power amplifier and matching network
* Equips integrated PLL, regulators, DCXO and power management units
* Integrated low power 32-bit CPU could be used as an application processor
* SDIO 1.1 / 2.0, SPI, UART
* STBC, 1×1 MIMO, 2×1 MIMO
* A-MPDU & A-MSDU aggregation & 0.4ms guard interval
* Wake up and transmit packets in < 2ms
* Standby power consumption of < 1.0mW (DTIM3)

**SPECIFICATIONS:**

* Standard: 802.11 b/g/n
* Output Power(dBm): 19.5
* Length (mm): 25
* Width (mm): 15
* Height (mm): 1.33

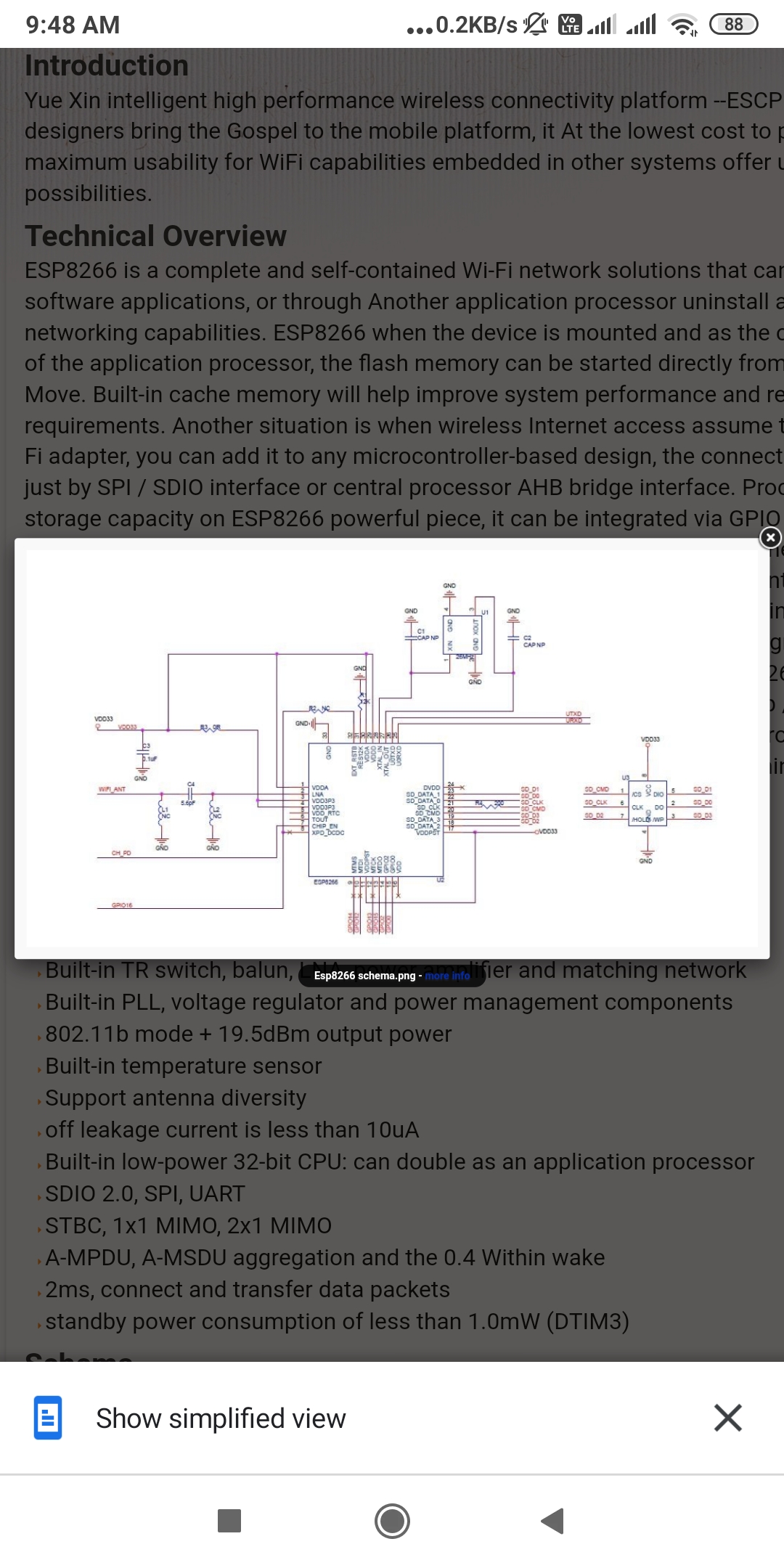
**POWER:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Mode** | **Min** | **Typ** | **Max** | **Unit** |
| 802.11b, CCK 1Mbps, POUT=+19.5dBm |  | 215 |  | mA |
| 802.11b, CCK 11Mbps, POUT=+18.5dBm |  | 197 |  | mA |
| 802.11g, OFDM 54Mbps, POUT=+16dBm |  | 145 |  | mA |
| 802.11n, MCS7, POUT =+14dBm |  | 135 |  | mA |
| 802.11b, packet size of 1024 bytes, -80dBm |  | 60 |  | mA |
| 802.11b, packet size of 1024 bytes, -70dBm |  | 60 |  | mA |
| 802.11b, packet size of 1024 bytes, -65dBm |  | 62 |  | mA |
| Standby |  | 0.9 |  | uA |
| Deep sleep |  | 10 |  | mA |
| Saving mode DTIM 1 |  | 1.2 |  | mA |
| Saving mode DTIM 3 |  | 0.86 |  | mA |
| Shutdown |  | 0.5 |  | uA |

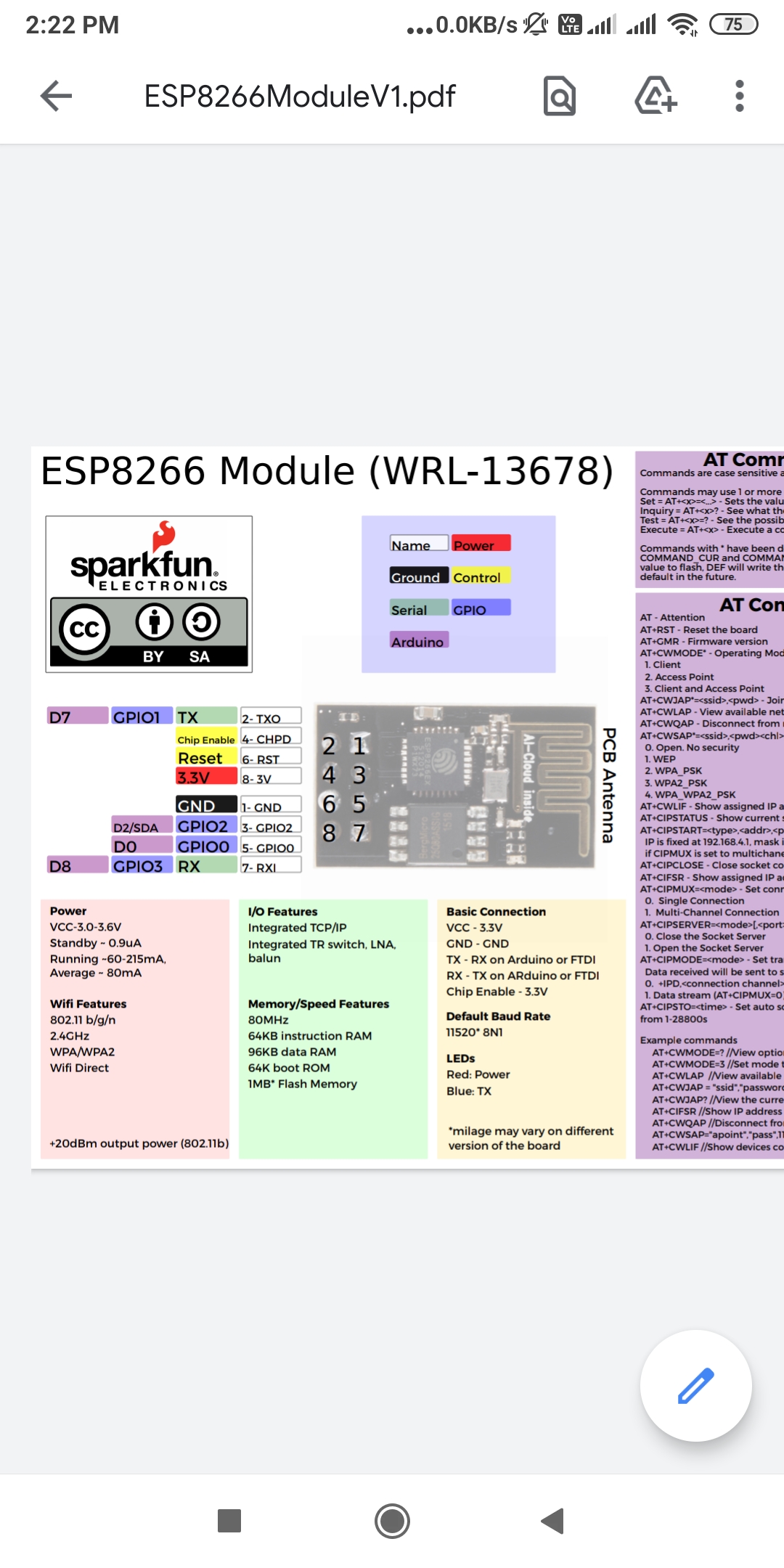
**RF SPECIFICATIONS:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Description** | **Min** | **Typ** | **Max** | **Unit** |
| Input Frequency | 2412 |  | 2484 | Mhz |
| Input resistance |  | 50 |  | Ω |
| Input reflection |  |  | -10 | db |
| At 72.2Mbps, PA output power | 14 | 15 | 16 | dbm |
| 11b mode, PA output power | 17.5 | 18.5 | 19.5 | dbm |
| **Sensitivity** | | | | |
| CCK, 1Mbps ￼ |  | -98 |  | dbm |
| CCK, 11Mbps ￼ |  | -91 |  | dbm |
| 6Mbps (1/2 BPSK) ￼￼ |  | -93 |  | dbm |
| 54Mbps (3/4 64-QAM) ￼￼ |  | -75 |  | dbm |
| HT20, MCS7 (65Mbps, 72.2Mbps) ￼ ￼￼ |  | -71 |  | dbm |
| **Adjustment suppression** | | | | |
| OFDM, 6Mbps |  | 37 |  | db |
| OFDM, 54Mbps |  | 21 |  | db |
| HT20, MCS0 |  | 37 |  | db |
| HT20, MCS7 |  | 20 |  | db |

**SCHEMATIC DIAGRAM:**



**PIN FUNCION:**



* VCC - 3.3V
* GND - GND
* TX - RX on Arduino or FTDI
* RX - TX on ARduino or FTDI
* Chip Enable - 3.3V

**AT COMMAND MODE:**

* AT - Attention
* AT+RST - Reset the board
* AT+GMR - Firmware version
* AT+CWMODE\* - Operating Mode 1. Client 2. Access Point 3. Client and Access Point
* AT+CWJAP\*=<ssid>,<pwd> - Join network
* AT+CWLAP - View available networks
* AT+CWQAP - Disconnect from network
* AT+CWSAP\*=<ssid>,<pwd><chl><ecn> - Set up access point 0. Open. No security 1. WEP 2. WPA\_PSK 3. WPA2\_PSK 4. WPA\_WPA2\_PSK
* AT+CWLIF - Show assigned IP addresses as access point
* AT+CIPSTATUS - Show current status as socket client or server
* AT+CIPSTART=<type>,<addr>,<port> - Connect to socket server IP is fixed at 192.168.4.1, mask is fixed at 255.255.255.0 if CIPMUX is set to multichanel add <id> to beginning of string
* AT+CIPCLOSE - Close socket connection
* AT+CIFSR - Show assigned IP address when connected to network
* AT+CIPMUX=<mode> - Set connection 0. Single Connection 1. Multi-Channel Connection
* AT+CIPSERVER=<mode>[,<port>](AT+CIPMUX=1) - Default port is 333 0. Close the Socket Server 1. Open the Socket Server
* AT+CIPMODE=<mode> - Set transpartent mode Data received will be sent to serial port as 0. +IPD,<connection channel>,<length>format (AT+CIPMUX=[0,1]) 1. Data stream (AT+CIPMUX=0)
* AT+CIPSTO=<time> - Set auto socket client disconnect timout from 1-28800s

**PACKAGE INCLUDES:**

1 x ESP-01 ESP8266 Serial WIFI Wireless Transceiver Module