

L'objet SerialPort

M242
Atelier microcontrôleur

Les constructeurs

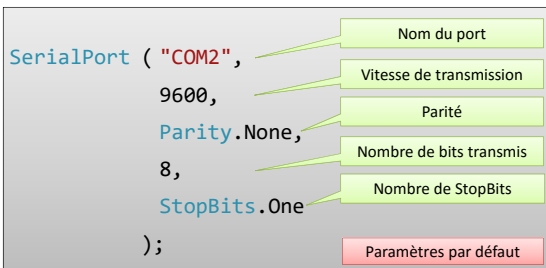
- 1) `SerialPort(string portName);`
- 2) `SerialPort(string portName, int baudRate);`
- 3) `SerialPort(string portName, int baudRate, Parity parity);`
- 4) `SerialPort(string portName, int baudRate, Parity parity, int dataBits);`
- 5) `SerialPort(string portName, int baudRate, Parity parity, int dataBits, StopBits stopBits);`

M242 & AtuC - Input Port

2

Les paramètres du constructeur

Paramètres du constructeur n° 5:



M242 & AtuC - Input Port

3

Les méthodes

```
void Open();  
int Read(byte[] buffer, int offset, int count);  
void Write(byte[] buffer, int offset, int count);  
void Close();  
void DiscardInBuffer();  
void DiscardOutBuffer();  
void Flush();  
long Seek(long offset, SeekOrigin origin);  
void SetLength(long value);
```

M242 & AtuC - Input Port

4

Les propriétés

```
Stream BaseStream { get; }  
int BaudRate { get; set; }  
int BytesToRead { get; }  
int BytesToWrite { get; }  
bool CanRead { get; }  
bool CanSeek { get; }  
bool CanTimeout { get; }  
bool CanWrite { get; }  
int DataBits { get; set; }  
Handshake Handshake { get; set; }  
bool IsOpen { get; }  
long Length { get; }  
Parity Parity { get; set; }  
String PortName { get; }  
long Position { get; set; }  
int ReadTimeout { get; set; }  
StopBits StopBits { get; set; }  
int WriteTimeout { get; set; }
```

M242 & AtuC - Input Port

5

Les propriétés des signaux de transmission

```
BaudRate // Vitesse de transmission  
DataBits // Longueur des bits de données par octet  
Handshake // définit le protocole de négociation  
Parity // le protocole de contrôle de parité  
PortName // nom du port de communications  
StopBits // nombre standard de bits d'arrêt par octet  
  
ReadTimeout // délai d'attente d'une écriture  
WriteTimeout // délai d'attente en écriture
```

M242 & AtuC - Input Port

6

Les propriétés

```
BaseStream // Obtient l'objet Stream sous-jacent
BytesToRead // Obtient le nombre d'octets présents dans le
             tampon de réception
BytesToWrite // Obtient le nombre d'octets présents dans le
              tampon d'envoi
CanRead      //
CanSeek      //
CanTimeout   //
CanWrite     //
IsOpen       // indiquant si le port est ouvert ou Length //
Position     //
```

M242 & AtuC - Input Port

7

La méthode Write()

Ecrit un nombre spécifié d'octets en utilisant les données à partir d'un buffer.

Syntaxes

```
void Write ( byte[] buffer, int offset,
             int count);
```

M242 & AtuC - Input Port

8

La méthode Read()

Lit un nombre d'octets du tampon d'entrée et écrit ces octets dans un tableau d'octets à l'emplacement spécifié.

```
int Read( byte[] buffer,
           int offset,
           int count);
```

M242 & AtuC - Input Port

9

Les évènements

```
event
SerialDataReceivedEventHandler
DataReceived;
```

```
event
SerialErrorReceivedEventHandler
ErrorReceived;
```

M242 & AtuC - Input Port

10