WEB SDK

Table of Contents

1. Class Description

2. Enumeration

3. Function

- Class Description

Class Name	Description
DotPadScanner	• Class for scanning **Bluetooth/USB** devices
DotPadSDK	• Process class that centrally manages **Bluetooth/USB**connections • Manages multiple devices simultaneously (maintains internal list)
DotDevice	•**Device model class** managing**connection**with a single DotPad device, **board information, and output status**

- Enumeration

export	Description	value
const DisplayMode	Device Output Mode	• GraphicMode: Output sequentially from the top-left of the cell downward • TextMode: Output pins 7 and 8 as a cursor
const DataCodes	Indicates**the type of event/response**occurring during communication with the device.	• Connected: Device information setup complete, **ready for use**. • ConnectedFail: Connection attempt failed. • Disconnected: Connection terminated. • BoardInfo: Received board layout/support information (graphic/text support, number of rows/columns, etc.). • BleMacAddress: Received BLE

		MAC address information. • DeviceName: Received device name. • DeviceFWVersion: Received firmware version. • DeviceHWVersion: Received hardware version. • *ResponseDisplayLineAck*: Line received. • *ResponseDisplayLineNonAck*: Line not received. • *ResponseDisplayLineComplete*: Line output complete. • *CommandError*: Command error. • *CommandNone*: Undefined command.
const KeyCodes	Indicates the type of key event sent from the device.	• KeyFunction1: Function 1 key input • KeyFunction2: Function 2 key input • KeyFunction3: Function 3 key input • KeyFunction4: Function 4 key input • KeyFunction12 : Function 1 and 2 key input • KeyFunction13 : Function 1 and 3 key input • KeyFunction23 : Function 2 and 3 key input • KeyFunction24 : Function 2 and 4 key input • KeyFunction34 : Function 3 and 4 key input • KeyElse: Input of a combination key not defined in KeyCodes • PanningAll: Input of Panning Left and Right keys • PanningLeft: Input of Panning Left key • PanningRight: Input of Panning Right key • LPF1: Input of Panning Left key and Function 1 key

		• RPF4: Input of Panning Right key and Function 4 key

- Function

1. **DotPadScanner**

Function Name	Description
async startBleScan()	Requests the user to select a Bluetooth device to connect via the browser
async startUsbScan()	Requests the user to select a Serial device to connect via the browser

2. **DotPadSDK**

2.1 Device Connection and Disconnection

Function Name	Description
getConnectedDevices(): DotDevice[];	• Retrieves the list of connected devices
connectBleDevice(device: BluetoothDevice): Promise<DotDevice | null | undefined>;	• Connects to a BLE device • Returns a new DotDevice if successful, null otherwise
connectUsbDevice(device: SerialPort): Promise<DotDevice | null | undefined>;	• Connect USB • Returns a new DotDevice internally registered upon success, null upon failure
disconnect(device?: DotDevice | null | undefined): void;	• Disconnect • If dotDevice == null,**disconnect all devices**and reset the internal list • If a specific dotDevice is specified, disconnect only that device and remove it from the list **Disconnect all:** disconnect() **Disconnect specific device:** disconnect(targetDevice)

2.2 Device Protocol Transmission

Function Name	Description
displayGraphicData(hexData: string, device?: DotDevice | null | undefined, displayMode?: DisplayMode): void;	• Transmit **graphic data** •**Display graphic data (hex string)** • **Compare differences between existing output and new data to**update • If dotDevice == null, transmit**to all**

	connected devices • Default displayMode is GraphicMode Display to all devices: displayGraphicData("FFFFFFFF") Display to specific device: displayGraphicData("FFFFFFFF", targetDevice)
displayTextData(hexData: string, device?: DotDevice | null | undefined, displayMode?: DisplayMode): void;	• Transmit text data • **Display text data (hex string)** • Default displayMode is TextMode Display on all devices: displayTextData("FFFFFF") Display on specific device: displayTextData("FFFFFF", targetDevice)
displayLineData(lineId: number, startCellIndex: number, hexData: string, displayMode: DisplayMode, device?: DotDevice | null | undefined): void;	• Send specific line data • Display specific **line data (hex string)** • If dotDevice == null, send**to all connected devices** • **lineId = specific line**number • **startCellIndex = starting cell index**

3.3 Listeners

Function Name	Description
setCallBack(messageCallBack: ((device: DotDevice, dataCode: DataCodes, msg: string) ⇒ void) | null, keyCallBack: ((device: DotDevice, keyCode: KeyCodes, msg: string) ⇒ void) | null): void;	• Register callback • messageCallBack: Responds with DataCodes • keyCallBack: Responds with KeyCodes