

CentipedeManager.h

```
typedefs
std::array<uint16_t, N_CP_PORTS>: CP_Masks

const N_CP_PORTS = 8: uint8_t
```

<struct> CP\_Address

+port: uint8\_t  
+bit : uint8\_t

CentipedeManager

-cp\_ : Centipede  
-masks\_ : CP\_Masks  
+CentipedeManager()  
+begin()  
+clear\_masks()  
+add\_to\_masks(CP\_Address)  
+set\_masks(CP\_Masks)  
+get\_masks(): CP\_Masks  
+report\_masks(Stream &)  
+send\_masks()

constants.h

```
const NUMEL_PCS_AXIS = 15 : uint8_t
const NUMEL_LED_AXIS = 16 : uint8_t
const N_VALVES = 112 : uint8_t

const P2VALVE[][] : uint8_t
const P2LED[][] : uint8_t
const VALVE2CP_PORT[] : uint8_t
const VALVE2CP_BIT[] : uint8_t

const N_LEDS : uint16_t
const PIN_LED_MATRIX : uint8_t
const PIN_R_CLICK_1 : uint8_t
const PIN_R_CLICK_2 : uint8_t
const PIN_R_CLICK_3 : uint8_t
const PIN_R_CLICK_4 : uint8_t
const R_CLICK_1_CALIB : RT_Click_Calibration
const R_CLICK_2_CALIB : RT_Click_Calibration
const R_CLICK_3_CALIB : RT_Click_Calibration
const R_CLICK_4_CALIB : RT_Click_Calibration
const DAQ_DT : uint32_t
const DAQ_LP : float
```

<struct> Omega\_Calib

+balance\_mA : float  
+sensitivity\_mA : float  
+full\_range\_bar : float

```
const OMEGA_1_CALIB : Omega_Calib
const OMEGA_2_CALIB : Omega_Calib
const OMEGA_3_CALIB : Omega_Calib
const OMEGA_4_CALIB : Omega_Calib

inline mA2bar(float, Omega_Calib): float
```

ProtocolManager.h

```
typedefs
std::array<P, MAX_POINTS_PER_LINE> : Line
std::array<uint16_t, NUMEL_PCS_AXIS> : PackedLine
std::array<PackedTimeLine, MAX_LINES> : Program

const MAX_LINES = 5000 : uint16_t
(↑ make as large as free RAM allows)
const MAX_POINTS_PER_LINE =
NUMEL_PCS_AXIS * NUMEL_PCS_AXIS + 1 : uint16_t
const P_NULL_VAL = -128 : int8_t
```

P  
("Point in the Protocol Coordinate System")

+x: int8\_t  
+y: int8\_t  
+P(int8\_t = P\_NULL\_VAL, int8\_t = P\_NULL\_VAL)  
+isNull()  
+setNull()  
+print(Stream &)

<struct> TimeLine

+time : uint32\_t  
+line : Line

<struct> PackedTimeLine

+time : uint32\_t  
+packed : PackedLine

ProtocolManager

-program\_ : Program  
-N\_program\_lines\_ : uint16\_t  
-current\_pos\_ : uint16\_t  
+ProtocolManager()  
+clear()  
+pack\_and add(TimeLine): PackedTimeLine  
+unpack()  
  
OPTIONAL?  
+add(ProtoLine)

translations.h

```
VALVE2P[][] : int8_t
p2valve(P) : uint8_t
p2led(P) : uint8_t
valve2p(uint8_t) : P
init_valve2p()
valve2cp(uint8_t) : CP_Address
```