MUX / DMUX

Current Measurment

Last R

Counter / Timer /RTC

USB-Hub (10x, je 900mA)

Audio-Modul (In&Out)

Motordriver (PWM/STEP/BCDL)

Frequenzgeni

DSO/Logic Analyzer

DMM

HMI-Modul

Com-Modul (galv) (UART/RS232/RS485/CAN/ETH/I2C/SPI..)

Templogger

High / Low Side Swith-Modul (galv, getrennt

ADC/DAC-Modul

GPIO-Modul

CPU-PCB

Speisungsaufbereitung (12V ,+-15, 5, 3V3, Ref

Grundspeisung +24V

PC / Framework

SPI / I2C

USB / ETH / WLAN

Module

CPU-Card

Protokoll PC -> CPU-Card Plaintext, einfache Komandi, logisch!

CONFIG;ADC2;PORTD;IN3;RES8;SAMPLE10K;

Konfiguriere;ADC-Modul 2 (20Port); Port D; Input 3; Auflösung 8-Bit; Samplerate 10Ks/s;

GET;ADC2;PORTD;IN3;SINGLE/CONT/DUR;

Lesen; ADC-Modul 2;Port D; Input 3; Einmal / Kontinuierlich (max…); Dauer (s/samples)

SET;GPIO1;PORTA;PIN3;IN/OUT;HIGH/LOW/PWM;DUTY25;FREQ50K;

Setzten; GPIO Karte1; Port A; PIN 3; Input/Output/PWM; Dutycycle in %; Frequenz in Hz;

INFO;GPIO1;-> Config ganze Karte

INFO;GPIO;PORTA; -> Config Port A

INFO; GPIO;PORTA; PinX; -> Config von Pin

Config als XML pro Karte auf bspw. SD-Card auf CPU-Modul

Programme als XML

Resultate als XML einsehbar auf Webserver, senden an QDM oder Framework (verschiedene Protokolle, Vorgabe Struktur, Variablen)

Raspberry Computing Module als PC-Ersatz? HMI einfach zu gestalten, Ressourcen sparen, Headless betrieb, Camera OpenCV, Display, USB, SPI/I2C vorhanden, ETH erschlagen,