

望遠鏡・機器制御の 容易なUSB化について



大島 修

(岡山県立鴨方高等学校)

発表の概要

- 簡単な自己紹介
- 望遠鏡の周辺は多様なレガシーインターフェースだらけ
- 大きな望遠鏡にUSBなんか使えない？
- USBのデバイスドライバ開発は大変？
- 容易なUSB化の例として
個人プロジェクト「バルマー逓減率輝線サーベイ」用望遠鏡の制御系の改修としてUSB化を行った

2005/11/9

第25回天文学に関する技術シンポジウムin 倉敷



プロフィール(天文技術面)

- 1974年からOAO91cm望遠鏡でフレア星の測光を卒論に。
- 1980年ころから教職の傍ら、光電測光装置の自作を試みる。田中済さんの雑誌連載記事を参考にし、フォトダイオード測光器試作。(ハンダごてを初めて購入)
- 1983年フォトマルを使った光子計数型測光器製作
- 1985年天文学会内地留学で同回路の改良
ダイナミックレンジを浜ホトの製品より2桁近く広げることに成功
(DC～数100kHzを～数10MHzへ)
- 1989年(株)AESより光電測光システムを製品化する
大学・博物館・公共天文台・アマチュア等へ20台程度納入される

プロフィール(その2)

- 1990年から美星天文台(BAO)計画策定委員
天文台全般の計画、シーイング対策対応建築設計、101cm望遠鏡を清水実さんと共に設計し、制御系の設計を担当する。ハイデンハインABSエンコーダ読取ボード・GPS 1pps信号に同期した高精度時計ボードの設計製作。光電測光装置設計。
- BAO101cm望遠鏡の試験機を兼ねて、国内初のフリクシヨンドライブ赤道儀(20cm望遠鏡ツイン用)を設計し法月技研で製作。
- 1993年から4年間 美星天文台研究員・主幹として、清水さんと共に望遠鏡と観測装置の立ち上げ、ソフト開発も担当。
- 1997年に高校現場に復帰、2003年から個人プロジェクト「バルマー逓減率測光サーベイ」用に望遠鏡制御系を改修開始

10数年前の設計

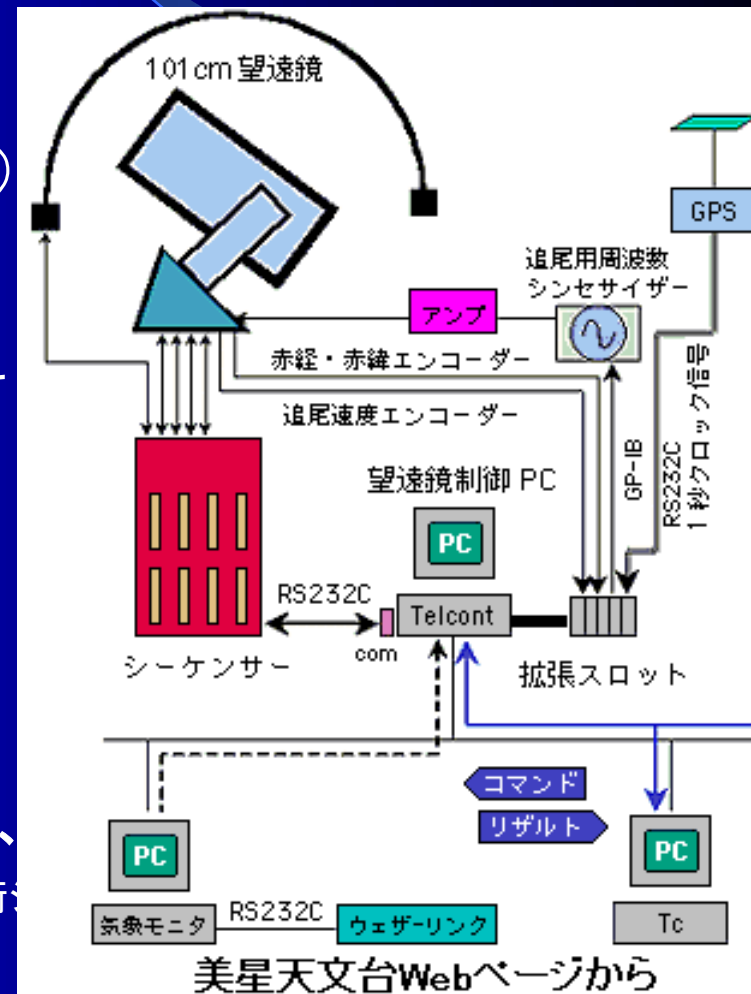
BAO101cm望遠鏡制御系

- **PC9801用Cバス**に挿したボード
GPS1PPS信号同期精密クロックボード、エンコーダ読取ボード(以上自作、以下市販)
GP-IBボード、カウンターボード
- 数年後にPCはすべてIBM互換機に
→**ISA-Cバス変換BOX**を使い、ボードはそのまま使用
- **RS-232C**(GPS、望遠鏡用シーケンサー制御)
- **GP-IB** 周波数シンセサイザ(恒星時追尾シンクロナスマータ用)

当時の技術で精一杯がんばったつもりだが、
今から見ると雑多なインターフェース

2005/11/9

第25回天文学に関する技術
ジウムin 倉敷



開発機器の インターフェースとしてのUSB

- ◆活栓挿抜PnP、電源供給、高速通信(480Mbpsまで)、256台まで接続可能
- ◆ホストPC側が主に通信制御を担当するので、ターゲット(機器)側の機能は軽量化できる(周辺機器のコストにメリット)

時代の流れはUSBが標準インターフェースに

- PCの周辺機器は何でもUSB化へ
- PCでレガシーインターフェースが使えなくなる時代
パラレルもシリアルポートもないPCが出始めている。
PC内バススロットも高速化複雑化する。(CバスやISAバスなら工作は簡単だった)

しかし、天文台の現場感覚としては

- ◆ケーブル長が最大5m。大きな望遠鏡の装置との通信には使えない？
- ◆PC側のデバイスドライバの開発が大変では？

ケーブルが短いから使えない？

1. ケーブルの届く範囲にPCを置き、LANで

→PCの廃熱が嬉しい

2. USB延長器はたくさんある

- UTPケーブルや光ファイバで延ばす
- USB2.0 High Speed 480Mbpsに対応した製品はまだ少ない

3. USB over IP機器・・・LANに乗せる

IBSジャパン : 「USB anywhere」 12Mbps

サイレックス・テクノロジー :

- SX1000U (Full Speed 12Mbps)
- SX2000U2 (High Speed 480Mbps)
- SX2000WG (IEEE802.11g 無線LAN 54Mbps)

問題は、対応OSが今のところWindowsのみ

第25回天文学に関する技術シンポ
ジウムin 倉敷



USBはデバイスドライバの 開発が大変なのは

- 1品ものである天文用機器毎に専用ドライバを書くとなると確かに大変のようです
- しかし、シリアルto USB変換チップ(下記)を使えば、シリアル信号をそのまま利用でき、既成のドライバがWindowsにもLinuxにも対応している

Linuxでは、対応しているkernelを使えば挿すだけですぐ認識される
(ドライバのインストールさえ不要)

- RS-232c to USBケーブル変換器もある
- FTシリーズ、CP210Xシリーズなら基板上の狭いスペースに載る

BDSプロジェクトに向けて 望遠鏡制御系の改造を行った

	旧システム	新システム
SPモータ制御	2軸制御 Cバス/ISAバスボード	4軸制御 USB対応ボード
エンコーダ読取	インクリメンタルエンコーダ 600P/Rを15倍増幅4遜倍＝ 15ビット相当 SPモータ制御ボード	アブソリュートエンコーダ (多摩川製17bit)PIC＋ USB
恒星時クロック・ ハンドセット入力	ハードロジック＋PLD(GAL)	PICマイコン＋USB
PCとの接続	D-sub 25pin太いケーブル	1本のUSBケーブル

2005/11/9

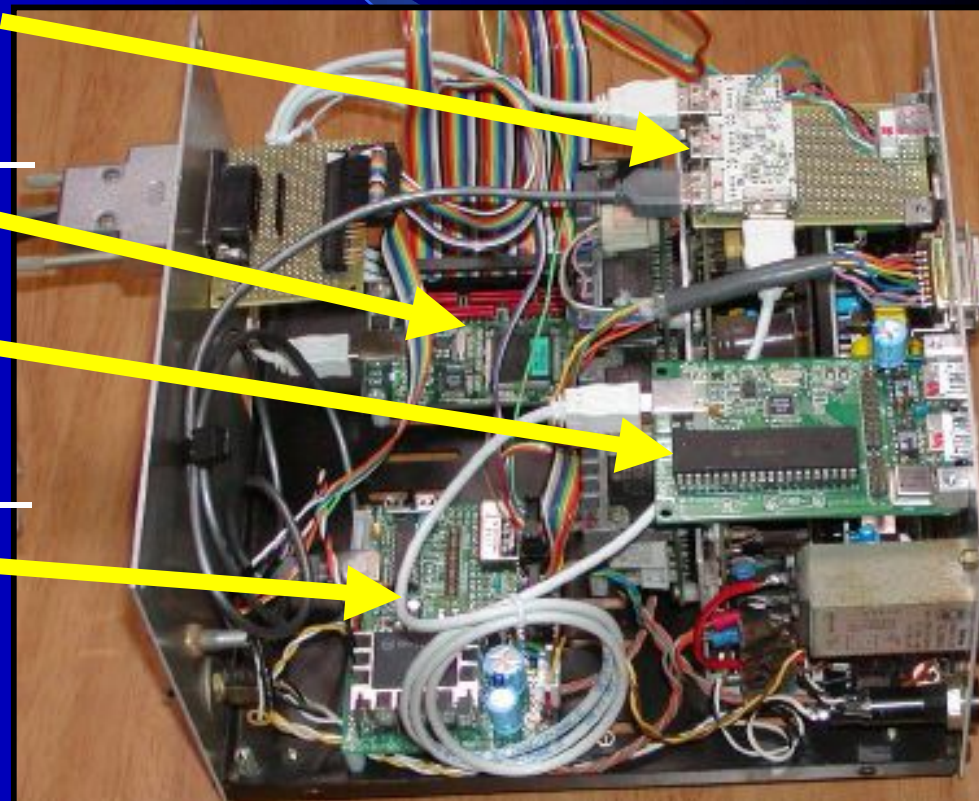
第25回天文学に関する技術シンポジウムin 倉敷

BDS望遠鏡制御回路のUSB化

裸にしたUSB-HUBに3枚の
ボードを接続

- ◆4軸パルスモータ信号発生ボード(市販品ボード使用)
- ◆アブソリュートエンコーダ読取ボード(市販評価用ボード利用)
- ◆恒星時駆動用クロック発生ボード(市販試作用ボード使用)

3枚とも市販の名刺サイズの小型ボード
制御PCへは1本のUSBケーブルだけで
接続。



アブソリュートエンコーダ読取ボード

- 多摩川精機17ビットエンコーダ TS5667N320 ×2
- 40PIN PICとUSB変換部 (FT245BM) を持った既成のUSB245/877ボード(IPI製)を改造
 - PICを18F452に交換
 - 水晶発振器を交換し10MHzに
 - エンコーダとのインターフェースは2.5MHz RS485ドライバー2組とコネクタを載せた
 - ファームウェアソフトはアセンブラで木下一男氏が製作

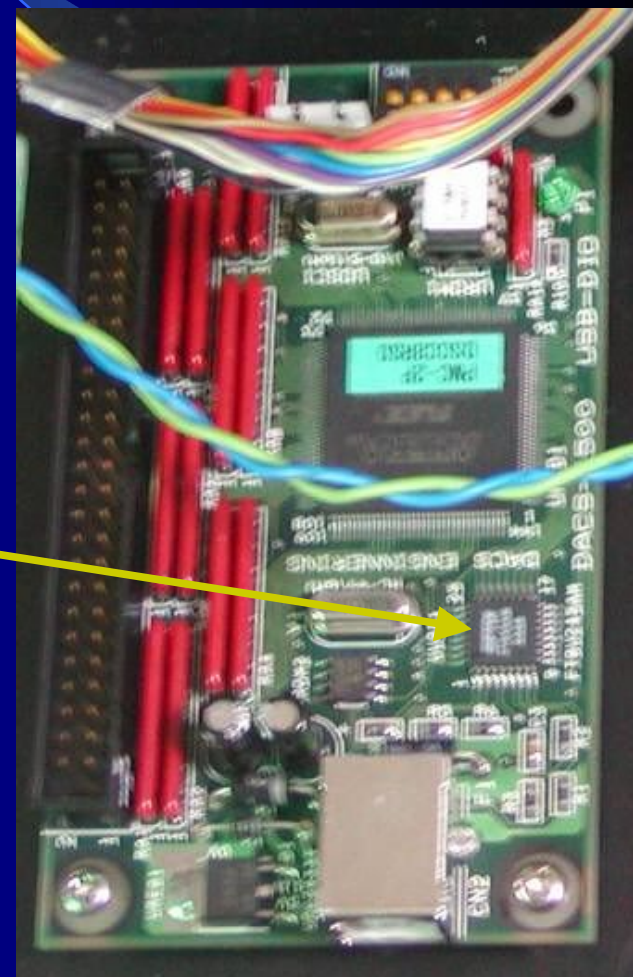
2005/11/9

第25回天文学に関する
ジウムin



4軸SPモータ用制御信号発生ボード

- DACS1500PMC-2P FGDPLD (ALTERA製)DACs技研製
- 4軸同時制御のうち2軸を使用
- 16ビットパレル入出力
ハンドセット入力・CCDガイド信号入力・リミットSW入力など
- USB変換部は、FT245AMが使っている
- 自動制御以外に、ハンドセット入力による制御もrubyで書いたソフトでPC側から行う



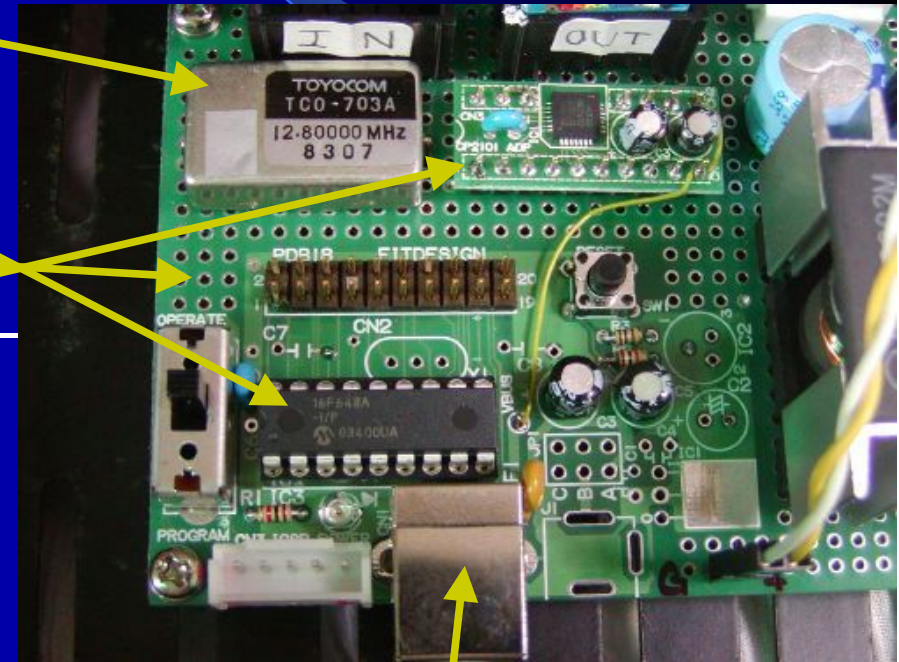
FTDI社製のFTシリーズchip

- ◆ デバドラ開発不要。PC側ソフトはシリアル接続として書くだけ。
 - WindowsにはFTDI提供のデバドラがある
 - Linuxにはkernel2.4以降は自動認識される
 - FT232BM(機器側はシリアル接続)
 - FT245BM(機器側はパラレルFIFO接続も可能)
 - FT2232C(232BM+245BM、8ビットマイコン(8051など)バスエミュレートなど)
- ◆ 「トランジスタ技術」2005年1月号(現在発売中)に特集あり



恒星時駆動用 高精度クロック発生器

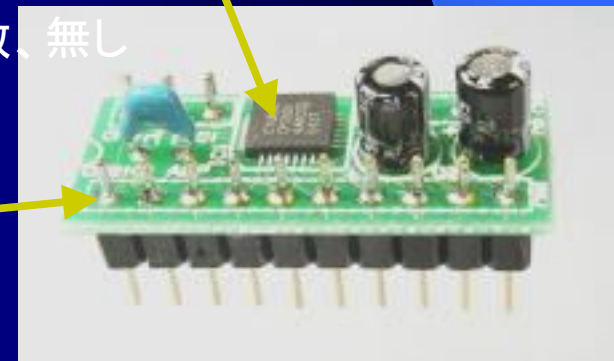
- 温度補償型水晶発振器(12.8MHz、1ppm)とPIC16F648の16bitタイマーを使用しクロックを発生
- USBインターフェース(CP2102)
- FITDESIGN製PDB18-DS ボードに
- 同社製のCP2102の20PIN DIP変換ボードCPA02Bを載せた
- ファームウェアはC言語で自作
 - 恒星時駆動クロックの周波数はPCからEEPROMへ書き込む(フリクションドライブは駆動周波数の合せ込みが必要なため)
 - RA軸駆動は、恒星時パルスとSPモータ用パルス発生器からのパルスと切り替えて使う



USB Bポート

恒星時クロック発生ボードのUSB化に使用した CP210Xシリーズの特徴

- SILICON LABORATORIES社の5mm角小さなパッケージ
- USBとURATを変換
- **水晶発振子や抵抗**の外付部品が不要
- 全てのUART制御信号をサポート（使うのはTxDとRxDの2本のみで十分）
- ベンダIDやプロダクトID書込用に512Byteの**EEPROM**内蔵
- 送受信バッファ512Byte
- 3.3Vレギュレータ内蔵、パワーオン・リセット回路内蔵
- 通信条件
データ: 8bit、ストップ: 1bit、パリティ: 偶数、奇数、無し
ボーレート: 300bps~921.6kbps
- LinuxはKernel2.6.11以降で自動認識
- DIP変換ボードCPA02Bが便利



制御プログラミング

- PICマイコンのファームウェア制作
Cコンパイラを利用(CSC社)
- PC側ソフトはRubyを使用
 - Linux Fedora Core 4 (kernel 2.6.11以降ならPnPでCP2102チップを自動認識しデバイスドライバーモジュールを自動で取り込む)
 - Cursesライブラリを使ったコンソール画面(ロボット制御による自動観測が目的なのでGUIは無用の長物)
 - 機器制御にはオブジェクト指向言語が最適
 - USBデバイスは複数の機器を抜き差しするとデバイス番号が変わるという問題点があるが、、、

Linux kernel のUSB認識

‘dmesg | grep usb’の例

```
usb 2-2: MCT U232 converter now attached to ttyUSB0
drivers/usb/serial/mct_u232.c: Magic Control Technology USB-RS232
converter driver z2.0
drivers/usb/serial/usb-serial.c: USB Serial support registered for FTDI
USB Serial Device
drivers/usb/serial/usb-serial.c: USB Serial support registered for CP2101
usb 2-1.1: FTDI USB Serial Device converter now attached to ttyUSB1
usb 2-1.4: FTDI USB Serial Device converter now attached to ttyUSB2
usb 2-1.2: CP2101 converter now attached to ttyUSB3
```

↑
USB-HUBポート番号

は不変
2005/11/9

↑
活栓挿抜するとデバイス番号は変化する<例>

/dev/ttyUSB3 → ttyUSB6へ

PC側のプログラム (Ruby)

```
class DACS1500
  require "DACS1500PMC.rb"
  BAUDRATE = B115200
  begin # get device number for multi HUB port
    str=`dmesg | grep usb`
    str.grep( "usb 2-1.4" ){ |line |
      device=line.scan( /ttyUSB./ )
    }
    @device= "/dev/"<<device[0]
  rescue => ex
    print "There are no USB FTDI device ¥n"
    print ex.message, "¥n"
  end
  def initialize
    @slctClck =0
  end
end
```

←←←←←
USBハブポートを
手掛かりにして

←←←←←
正規表現でデバイ
ス番号を得る

まとめ

- シリアル信号を簡単にUSB化できる
 - CP210XシリーズかFTシリーズチップを使う
- デバイスドライバの開発不要
- 多様な機器をUSB HUBでまとめて1本のUSBケーブル(またはLAN)で制御できる
 - USB over IPデバイスのLinuxドライバさえあれば、地球の裏からUSB機器の制御が、、、
- 既成の装置もシリアル信号があれば容易にUSB化できる

RS232c over TCP/IPもあるが

- RS-232cをethernetに接続するアダプターもあるが、通信速度はRS-232cのまま
- RS-232cをUSBに変換すると、複数のRS-232c機器をUSB-HUBでまとめ1本の配線でまとめて通信できる。さらにUSB over IPならLAN接続できる。