UNIVERSITÉ Concordia UNIVERSITY

GPS Clock

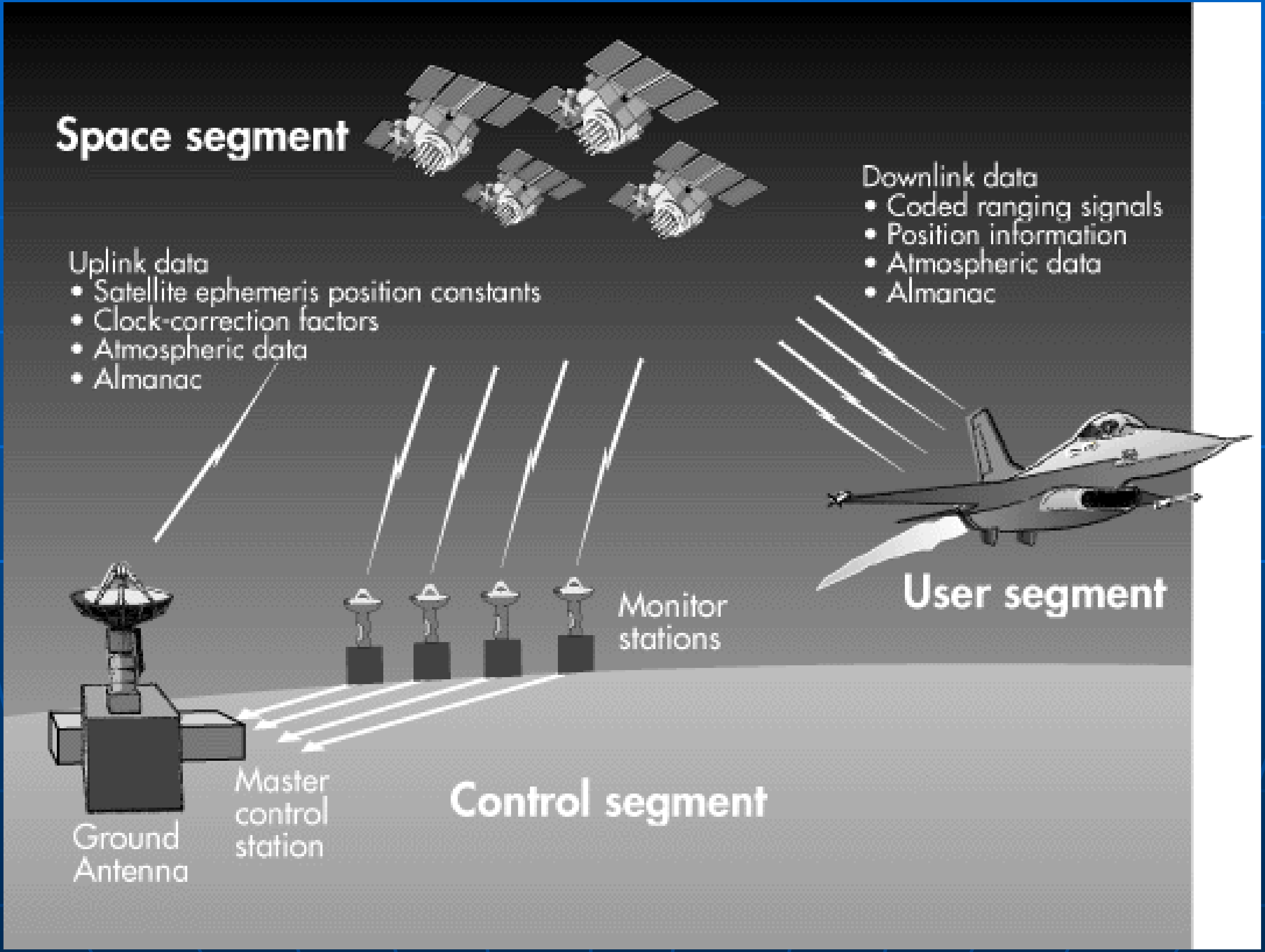
MECH 6621 Final Project

by
Siamak Arbatani
Konstantin Kalayev

April 21st, 2011

INTRODUCTION

GPS OVERVIEW

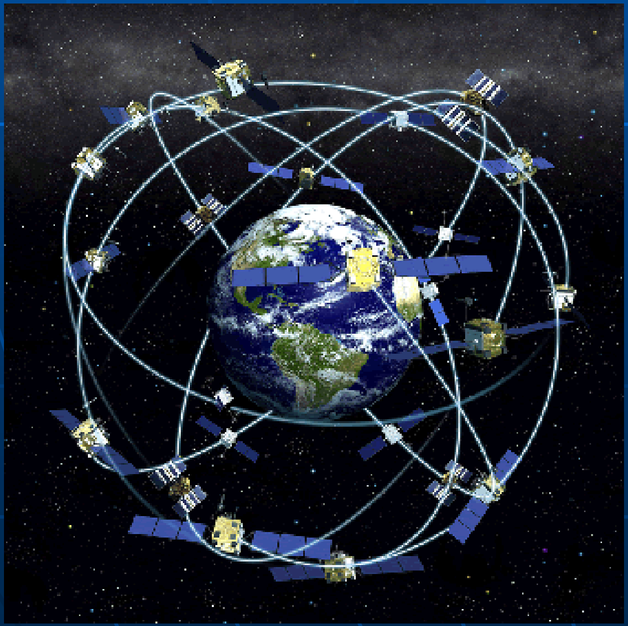


Space segment

Downlink data
- Coded ranging signals
- Position information
- Atmospheric data
- Almanac

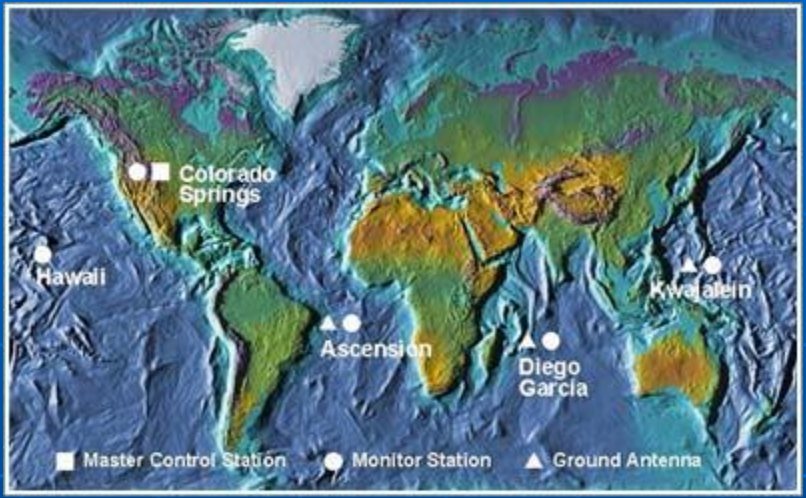
Uplink data
- Satellite ephemeris position constants
- Clock-correction factors
- Atmospheric data
- Almanac

Space segment



- 31 satellites
    - 24 at 6 orbital planes
    - 7 are orbited as spares
- Orbital radius is 26 600 km (20 200 km altitude)
- 2 complete orbits each 24 hours(the same ground track each day)
- from 5 to 8 satellites above horizon from any point on the Earth

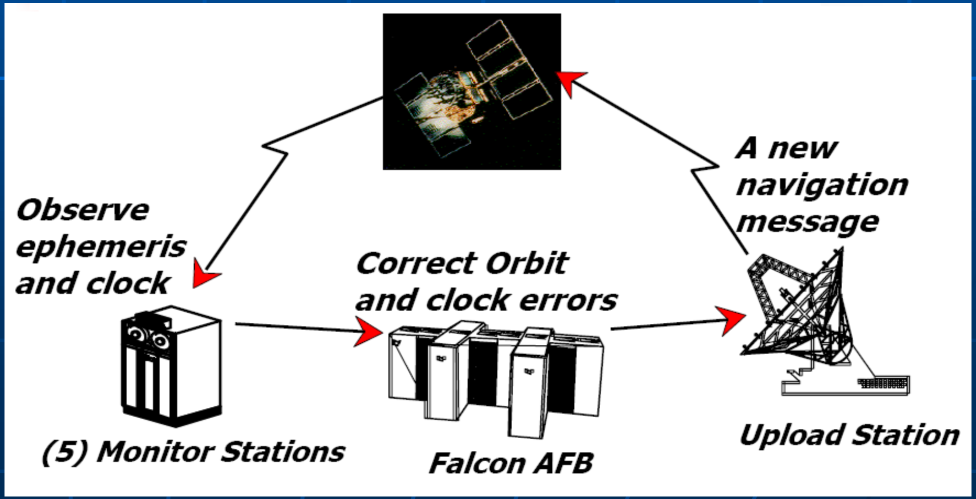
Control segment



consists of:

- Master control station (Colorado Spring, USA)
- Alternate master control station
- Four dedicated ground antennas
- Six dedicated monitor stations

The main role is observing the ephemeris and clock of each satellite and correction of the orbit and clock errors



User segment

- U.S. military users of the secure GPS Precise Positioning Service
- Civil, commercial and scientific users of the Standard Positioning Service

Principle of work

The GPS receiver receives the following information from each satellite:

- The almanac data about the approximate position of satellite
- The ephemeris data about the exact position of satellite
- Message containing the time of transmission

The receiver measures the time of arrival of the satellite signals

$$\Delta t = t_{Arrival} - t_{Transmission}$$

*Distance=Propagation Rate (speed of light) * Time*

User position can be computed using satellite positions and the distance to them

GPS satellite clocks

GPS time is accurate to about 14ns.
To achieve such accuracy each satellite is equipped with 2 cesium and 2 rubidium clocks ($100,000 - $500,000 each).
Moreover these clocks are continually adjusted with atomic clock by Master Control Station

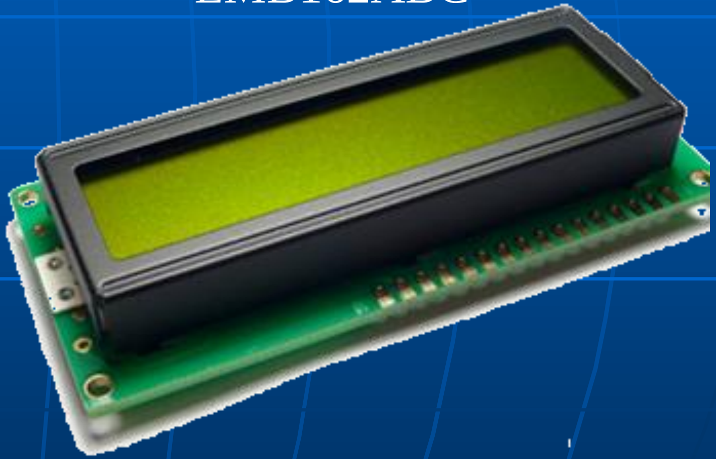
Hardware

- Microcontroller as the main processing and controlling unit
- GPS module as data provider
- An alphanumeric LCD as display
- RS232 standard peripherals for communication purpose

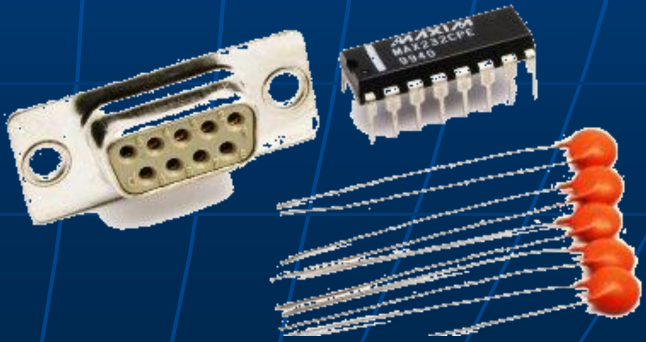
PIC18f4431

PMB-648

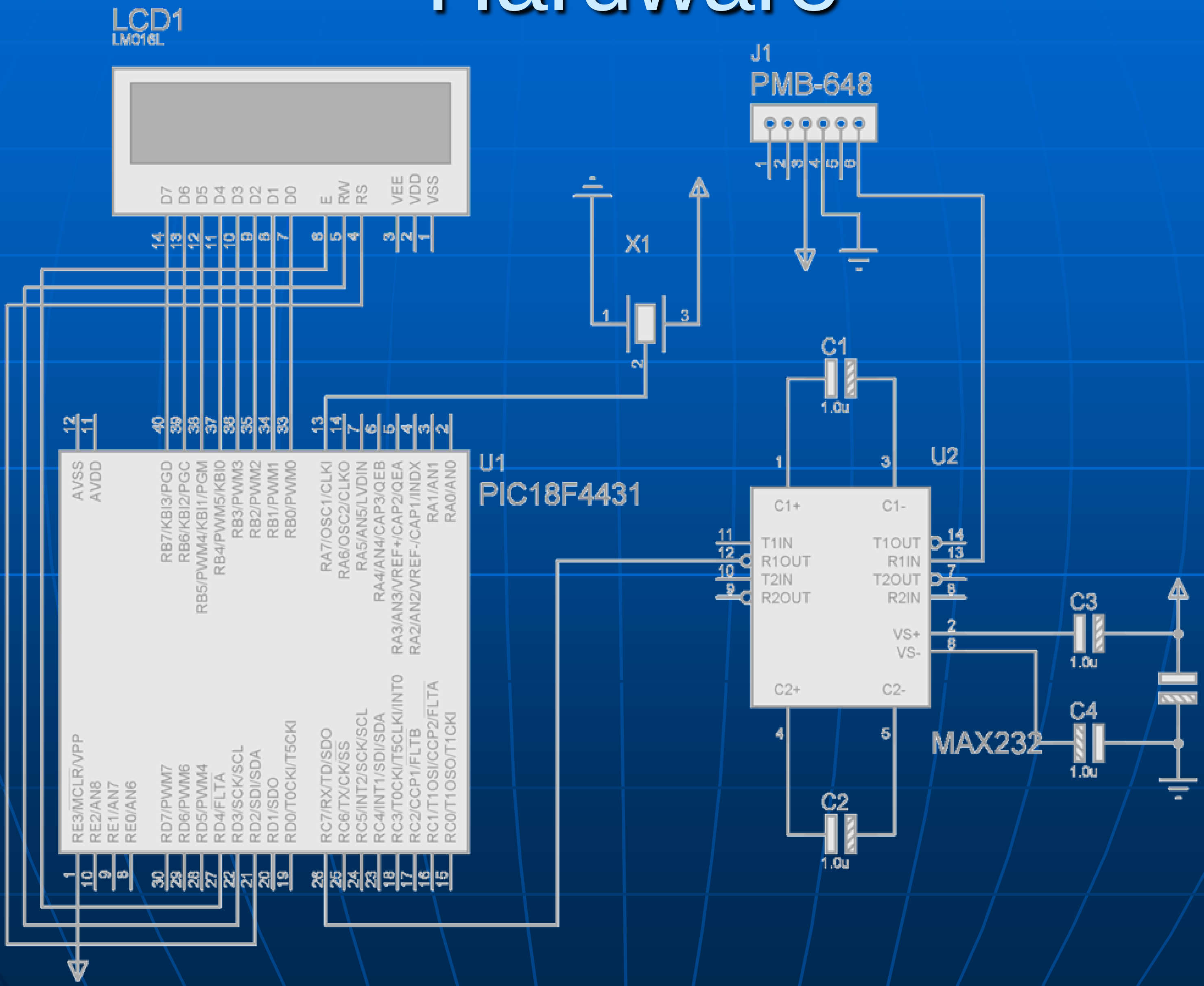
LMB162ABC



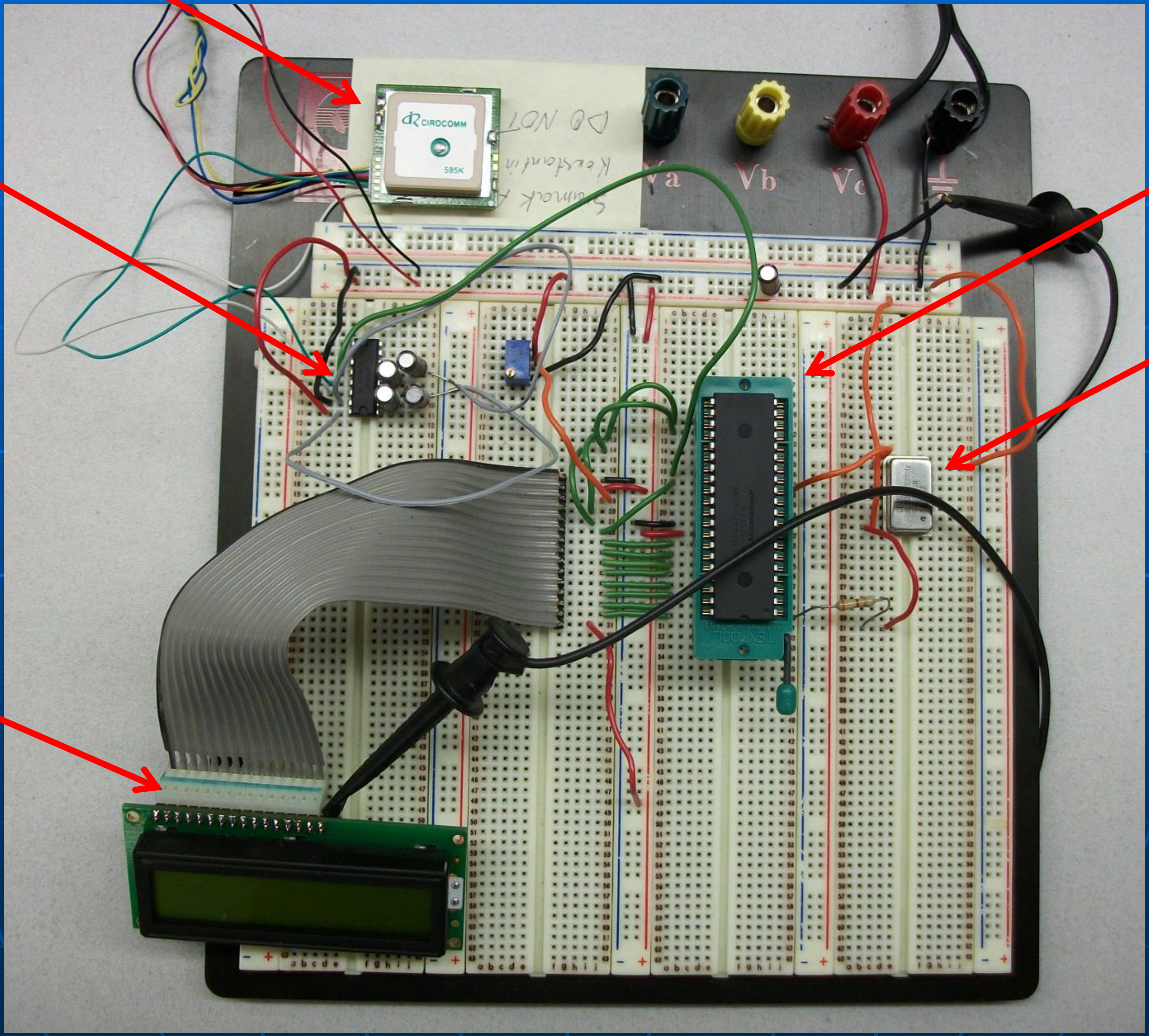
MAX232



Hardware


LCD1
LM016L
D7 D6 D5 D4 D3 D2 D1 D0 E RW RS VEE VDD VSS
14 13 12 11 10 9 8 7 6 5 4 3 2 1
J1
PMB-648
1 2 3 4 5 6
X1
1 2 3
C1
1.0u
C2
1.0u
C3
1.0u
C4
1.0u
C5
1.0u
U2
MAX232
C1+ C1- T1IN R1OUT T2IN R2OUT T1OUT R1IN T2OUT R2IN VS+ VS- C2+ C2-
1 3 11 12 10 9 14 13 7 8 2 6 4 5
U1
PIC18F4431
AVSS AVDD
RB7/KBI3/PGD
RB6/KBI2/PGC
RB5/PWM4/KBI1/PGM
RB4/PWM5/KBI0
RB3/PWM3
RB2/PWM2
RB1/PWM1
RB0/PWM0
RA7/OSC1/CLKI
RA6/OSC2/CLKO
RA5/AN5/LVDIN
RA4/AN4/CAP3/QEB
RA3/AN3/VREF+/CAP2/QEA
RA2/AN2/VREF-/CAP1/INDX
RA1/AN1
RA0/AN0
RE3/MCLR/VPP
RE2/AN8
RE1/AN7
RE0/AN6
RD7/PWM7
RD6/PWM6
RD5/PWM4
RD4/FLTA
RD3/SCK/SCL
RD2/SDI/SDA
RD1/SDO
RD0/T0CKI/T5CKI
RC7/RX/TD/SDO
RC6/TX/CK/SS
RC5/INT2/SCK/SCL
RC4/INT1/SDI/SDA
RC3/T0CKI/T5CKI/INT0
RC2/CCP1/FLTB
RC1/T1OSI/CCP2/FLTA
RC0/T1OSO/T1CKI
12 11 40 39 38 37 36 35 34 33 13 14 7 6 5 4 3 2
1 10 9 8 30 29 28 27 22 21 20 19 26 25 24 23 18 17 16 15

Hardware



PMB-648 GPS Module

MAX232

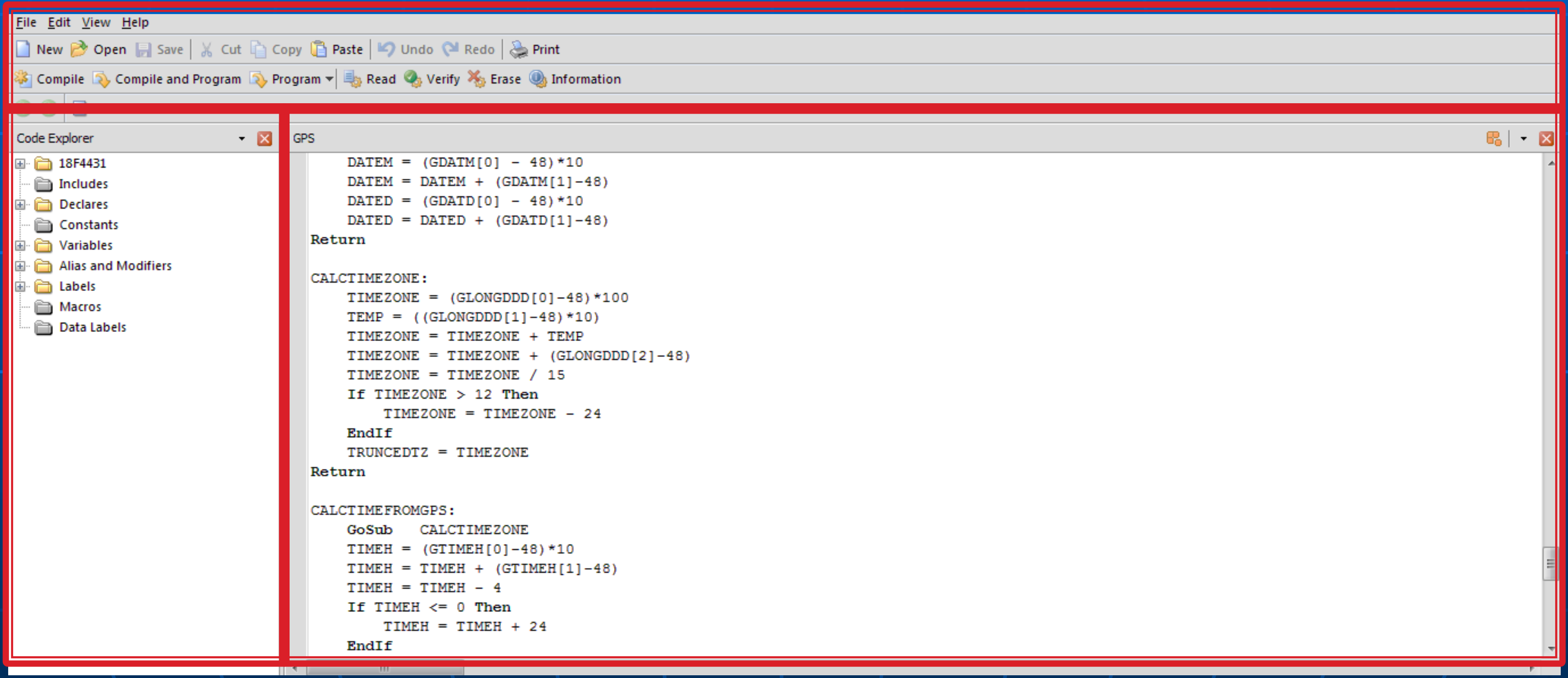
PIC18F4431

Oscillator
8MHz

LCD Module
LMB162ABC

Software

- High level programming language, Proton Basic employed
- Proton IDE used to develop and compile code



```
    DATEM = (GDATM[0] - 48)*10
    DATEM = DATEM + (GDATM[1]-48)
    DATED = (GDATD[0] - 48)*10
    DATED = DATED + (GDATD[1]-48)
Return

CALCTIMEZONE:
    TIMEZONE = (GLONGDDD[0]-48)*100
    TEMP = ((GLONGDDD[1]-48)*10)
    TIMEZONE = TIMEZONE + TEMP
    TIMEZONE = TIMEZONE + (GLONGDDD[2]-48)
    TIMEZONE = TIMEZONE / 15
    If TIMEZONE > 12 Then
        TIMEZONE = TIMEZONE - 24
    EndIf
    TRUNCEDTZ = TIMEZONE
Return

CALCTIMEFROMGPS:
    GoSub   CALCTIMEZONE
    TIMEH = (GTIMEH[0]-48)*10
    TIMEH = TIMEH + (GTIMEH[1]-48)
    TIMEH = TIMEH - 4
    If TIMEH <= 0 Then
        TIMEH = TIMEH + 24
    EndIf
```

Software

- One-shot HSerIn instruction of Proton Basic used with necessary arguments to place each data in its own associated variable

RMC standard

```
$GPGSV,1,1,00*79
$GPRMC,003304.749,V,0000.0000,N,00000.0000,E,,,300311,,,N*78
$GPGGA,003305.758,0000.0000,N,00000.0000,E,0,00,,0.0,M,0.0,M,,0000*4C
```

RMC standard

```
$GPGSA,A,1,,,,,,,,,,,,,,,*1E
$GPRMC,003305.758,V,0000.0000,N,00000.0000,E,,,300311,,,N*79
$GPGGA,003306.749,0000.0000,N,00000.0000,E,0,00,,0.0,M,0.0,M,,0000*4F
```

RMC standard

```
$GPGSA,A,1,,,,,,,,,,,,,,,*1E
$GPRMC,003306.749,V,0000.0000,N,00000.0000,E,,,300311,,,N*7A
$GPGGA,003307.751,0000.0000,N,00000.0000,E,0,00,,0.0,M,0.0,M,,0000*47
```

RMC standard

```
$GPGSA,A,1,,,,,,,,,,,,,,,*1E
$GPRMC,003307.751,V,0000.0000,N,00000.0000,E,,,300311,,,N*72
$GPGGA,003308.760,0000.0000,N,00000.0000,E,0,00,,0.0,M,0.0,M,,0000*4A
```

RMC standard

```
$GPGSA,A,1,,,,,,,,,,,,,,,*1E
$GPRMC,003308.760,V,0000.0000,N,00000.0000,E,,,300311,,,N*7F
$GPGGA,003309.750,0000.0000,N,00000.0000,E,0,00,,0.0,M,0.0,M,,0000*48
$GPGSA,A,1,,,,,,,,,,,,,,,*1E
$GPGSV,1,1,00*79
```

GPS Module
Output
NMEA Standard

- Valid example of RMC data:

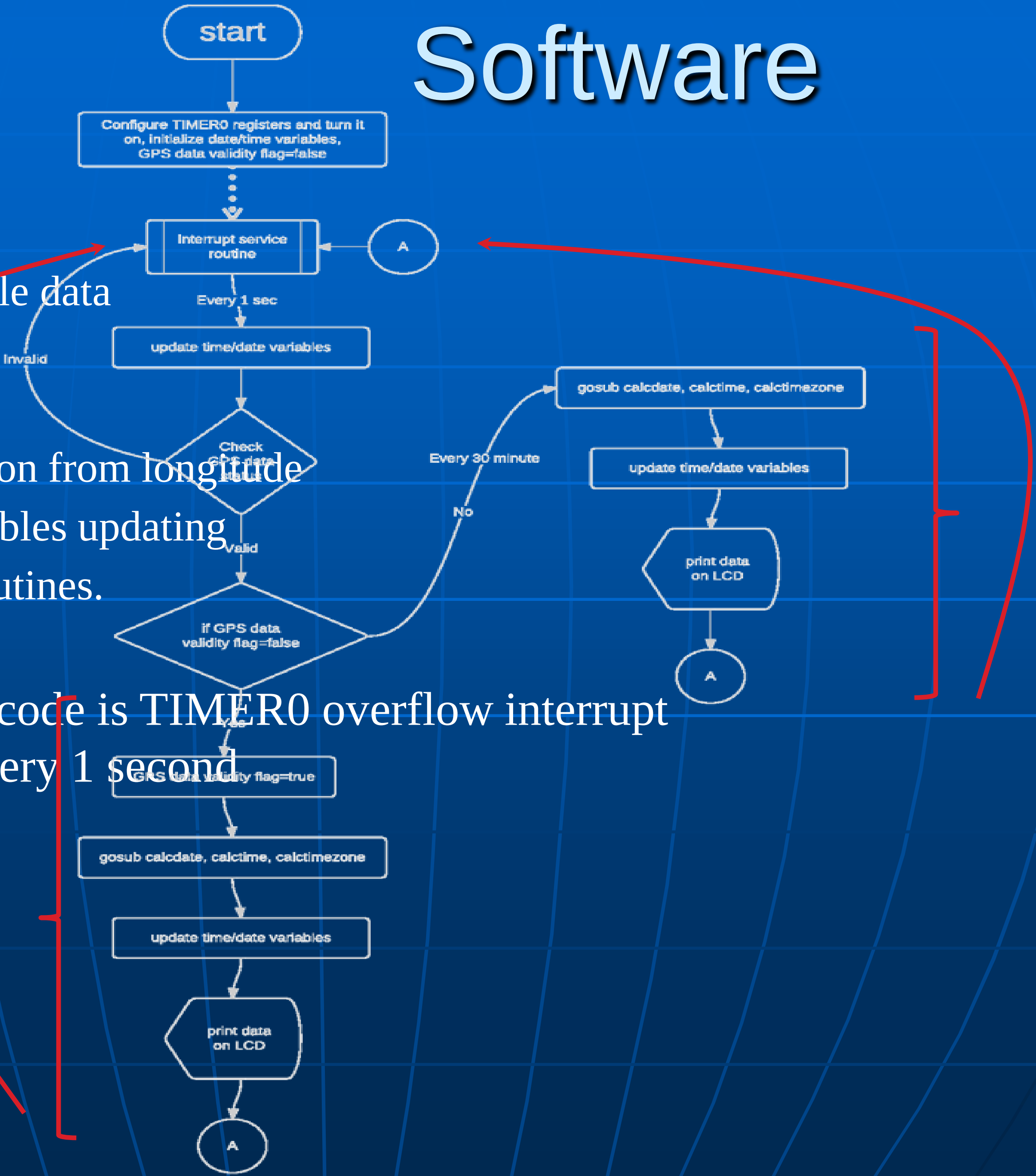
$GPRMC,161250.487,A,3723.2475,N,15030.3416,W,0.13,309.62,120511

Software

Code Sections:

- Initialization
- Reading GPS module data
- Date calculation
- Time calculation
- Time zone calculation from longitude
- Time and date variables updating
- Data printing subroutines.

- Main core of this code is TIMER0 overflow interrupt
- Occurs exactly every 1 second



Modeling & Simulation

- Proteus ISIS Professional advanced simulation software employed for design verification of this system

Model Components:

- PIC18F4431 microcontroller
- Standard LM016L 2×16 LCD model
- Virtual terminal to illustrate data flow in serial data line
- HDL (Hardware description language) code to define a virtual model for the GPS module.

Practical Testing

Indoor

Practical Testing

Outdoor

Conclusion and practical application

Thank you for your attention