GLOBal short wave solar radiation monitoring system (GLOB)

System overview

ABSTRACT

Solar radiation and wind, coupled with the use of power banks, hold promise as viable energy sources for the Arctic. However, it is crucial to comprehensively understand and describe the unique properties of these energy sources in this challenging environment.To address this, researchers from the Arctic Technology Department at the University Centre in Svalbard (UNIS) have developed and deployed a system to measure incoming and reflected short-wave radiation across 25 array planes. This innovative system enables high-frequency data logging at 5-second intervals, with on-line transmission to a central server for analysis.

Sebastian Sikora

GLOBal short wave solar radiation monitoring system (GLOB)

System overview v. 1.0

Authors: Sebastian Sikora[1], Aleks Klein-Paste[2], Aleksey Shestov[3]

[1]guliksen@gmail.com, [2]alex.klein-paste@ntnu.no, [3]AlekseyS@unis.no

Longyearbyen, 14.10.2023

INTRODUCTION

The Arctic region presents numerous challenges for human presence, ranging from constructing on thawing permafrost to ensuring adequate food supplies. However, one of the most pressing challenges revolves around energy sources, encompassing both heat and electricity supply. Arctic settlements are typically considered off-grid systems, necessitating the development of systems of locally generated energy that are both reliable and sustainable.

Traditionally, these energy systems heavily rely on fossil fuels, resulting in significant per capita CO_2 emissions. To address this issue, there is a growing need to explore the utilization of renewable energy sources as a pathway towards establishing clean and sustainable power stations for these remote locations. Policy makers highlight the importance of transitioning towards renewable energy sources in the Arctic. By embracing clean energy technologies, such as wind and solar power, it becomes possible to reduce the environmental impact and carbon footprint associated with energy generation in these regions. Additionally, sustainable energy solutions contribute to Arctic settlements' overall resilience and self-sufficiency.

Solar radiation and wind, coupled with the use of power banks, hold promise as viable energy sources for the Arctic. However, it is crucial to comprehensively understand and describe the unique properties of these energy sources in this challenging environment.

To address this, researchers from the Arctic Technology Department at the University Centre in Svalbard (UNIS) have developed and deployed a sophisticated system to measure incoming and reflected short-wave radiation across 25 array planes. This innovative system enables high-frequency data logging at 5-second intervals, with on-line transmission to a central server for analysis.

This research initiative aims to identify the untapped energy potential and optimize energy capture by accurately assessing solar radiation and wind patterns specific to the Arctic region. Moreover, it facilitates the development of efficient and sustainable energy systems tailored to Arctic applications.

SYSTEM DESCRIPTION

The goal of GLOB is to measure solar irradiation on as many as possible array planes to verify recorded values with modeled. Due to technical and costs limitations a sphere with 26 faces called rhombicuboctahedron (Rhombicuboctahedron - Wikipedia) has been pointed as most suitable for this project.

DESIGNING OF RHOMBICUBOCTAHEDRON

The rhombicuboctahedron is a polyhedron comprised of 8 triangles and 18 squares. For this project, elements with edges measuring 10 cm for both squares and triangles have been constructed. The bottom-facing square element isn't utilized for irradiation data collection. This is because a stainless steel pipe has been employed as a support leg, passing through the bottom square and reaching the top face, which is horizontally oriented. All components required for building the rhombicuboctahedron have been meticulously designed using CAD software, and the corresponding files compatible with most 3D printers have been attached for your convenience.

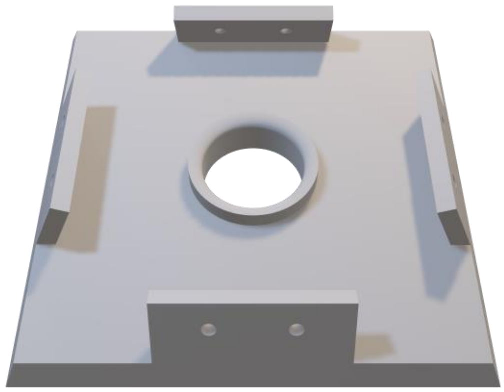

Fig. 1 The square element (1A) created in CAD software (https://doi.org/10.5281/zenodo.8411341)

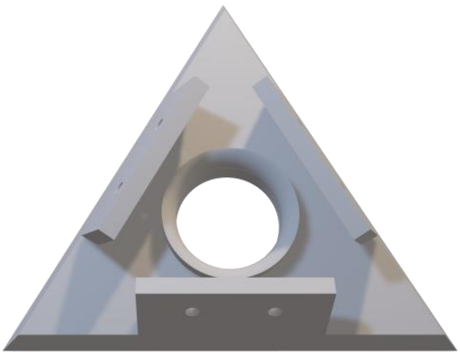

Fig. 2 The triangle (1B) element created in CAD software (https://doi.org/10.5281/zenodo.8411341).

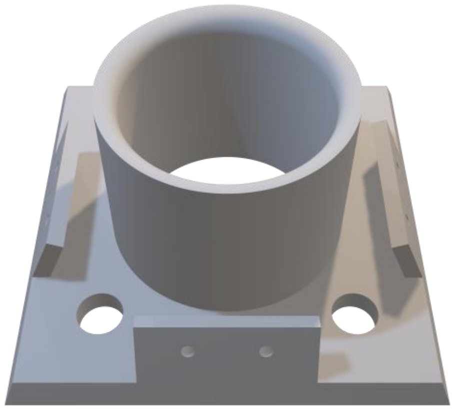

Fig. 3 The square (2A) element created in CAD software (https://doi.org/10.5281/zenodo.8411341).

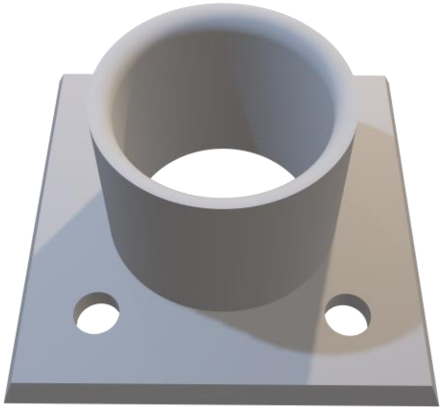

Fig. 4 The square (3A) element created in CAD software (https://doi.org/10.5281/zenodo.8411341).

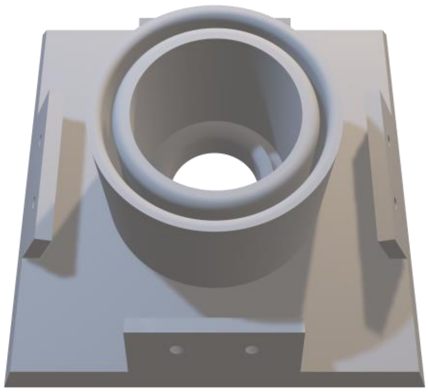

Fig. 5 The square (4A) element created in CAD software (https://doi.org/10.5281/zenodo.8411341).

To build complete sphere following number of elements are needed:

- eight 1B,
- sixteen 1A,
- one of 2A, 3A and 4A.

The choice of black ASA filament (acrylonitrile styrene acrylate) was deliberate, as it offers superior mechanical properties and UV resistance of 3D prints. Depending of filling percentage during printing out process elements are getting different mechanical properties. The sphere was mounted on a stainless steel pipe with an external diameter of 48 mm, which is a part of the CM110 Campbell Sci. tripod. In the center of each element (except the downward-facing one), a pyranometer was installed. Cables from these sensors were discreetly routed inside the metal pipe. To ensure structural integrity, all elements of the sphere were securely fastened using plastic strips and bonded with black silicon glue.

The sphere with sensors is installed on a steel pipe and the top pyranometer (GHI) is at 148 cm height above ground layer. Entire sphere is about 25 cm height. The steel pipe is placed on metal frame (to anchor system against strong wind). At a distance of 2 m (West from the sphere) another steel pole is installed on the same frame which holds photovoltaic module (facing S) and logger enclosure (facing N; Fig. 9.).

SENSORS AND OTHER ELECTRONIC COMPONENTS

Following components has been used to build complete system:

- 26 self-powered silicon-cell pyranometers type SP110 Apogee (https://www.apogeeinstruments.com/sp-110-ss-self-powered-pyranometer/),
- 1 temperature probe 107 manufactured by Campbell Sci. (https://www.campbellsci.com/107) for inside-sphere temperature monitoring,
- CR1000x logger (https://www.campbellsci.com/cr1000x),
- AM16/32B Multiplexer (https://www.campbellsci.com/am16-32b),
- CELL215 LTE modem (https://www.campbellsci.com/cell215),
- 7 Ah, 12V battery,
- Solar charging regulator,
- 20W photovoltaic module.

SYSTEM LOCATION

GLOB is an integral component of the Arctic Solar Energy Test Sites (ASETS, http://158.39.149.181/Adventdalen_PV/index.html) operated by The University Centre in Svalbard. It is situated approximately 4.5 km southeast of Longyearbyen, in close proximity to The Old Aurora Lights Research Station.

For precise reference, the geographical coordinates are:

- N 78.200318,
- E 15.840308

Altitude is about 7 m a.s.l.

Fig. 6. The GLOB system view in spring (3.03.2023, https://doi.org/10.5281/zenodo.8410795)

Fig. 7. The GLOB system view in summer (30.06.2023, https://doi.org/10.5281/zenodo.8410805)

Fig. 8. The GLOB system view in autumn (5.10.2023, https://doi.org/10.5281/zenodo.8412962)

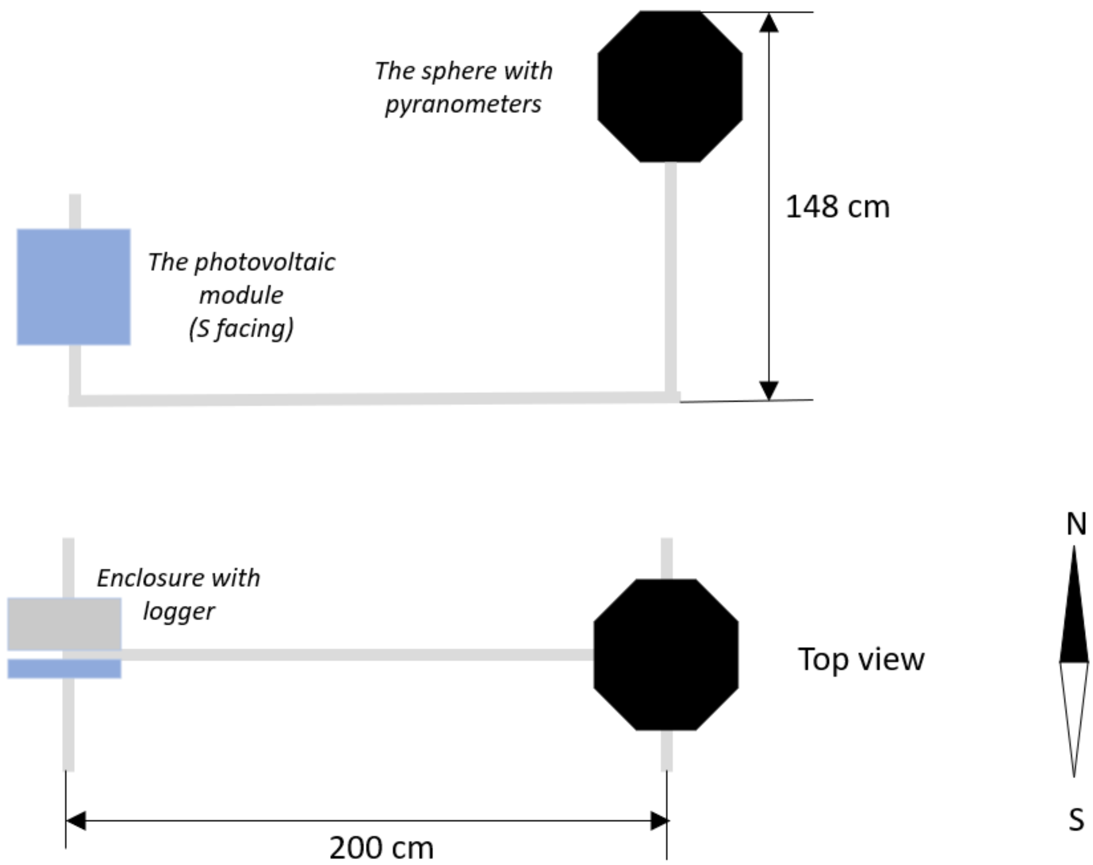

Fig. 9. Sketch of the GLOB system (https://doi.org/10.5281/zenodo.8411391)

DATA

A total of 26 pyranometers and one thermometer probe have been installed to collect data. The table below outlines the orientation of the array planes for 25 of the pyranometers.

The 26th pyranometer, referred to as "Kglob," is an additional unit situated on the ground (on levelled base with aluminium pole with centimetre scale markings to detect snow depth; sensor might be useful to evaluate solar irradiation under snow cover). It can potentially be utilized as a sensor for measuring reflected solar irradiation.

Table 1. Pyranometers orientation

Name	Data Table	azimuth	altitude	comments
GHI	SIR_Avg(1)	0	90	horizontal plane GHI
SIR135S	SIR_Avg(2)	180	45	
SIR90S	SIR_Avg(3)	180	0	
SIR45S	SIR_Avg(4)	180	-45	
SIR135SW	SIR_Avg(5)	225	45	
SIR90SW	SIR_Avg(6)	225	0	
SIR45SW	SIR_Avg(7)	225	-45	
SIR135W	SIR_Avg(8)	270	45	
SIR90W	SIR_Avg(9)	270	0	
SIR45W	SIR_Avg(10)	270	-45	
SIR135NW	SIR_Avg(11)	315	45	
SIR90NW	SIR_Avg(12)	315	0	
SIR45NW	SIR_Avg(13)	315	-45	
SIR135N	SIR_Avg(14)	0	45	
SIR90N	SIR_Avg(15)	0	0	
SIR45N	SIR_Avg(16)	0	-45	
SIR135NE	SIR_Avg(17)	45	45	
SIR90NE	SIR_Avg(18)	45	0	
SIR45NE	SIR_Avg(19)	45	-45	
SIR135E	SIR_Avg(20)	90	45	
SIR90E	SIR_Avg(21)	90	0	
SIR45E	SIR_Avg(22)	90	-45	
SIR135SE	SIR_Avg(23)	135	45	
SIR90SE	SIR_Avg(24)	135	0	
SIR45SE	SIR25_Avg	135	-45	

The logger generates 3 tables:

- Sec (5 s interval – RAW data),
- Hour (1 h average),
- Daily (daily average).

COSTS

The entire system cost approximately 170,000 NOK as of the beginning of 2023. In this estimation working hours to print out, designing 3D elements and programming of the logger are not included.

LOGGER PROGRAMM

To get the program as .CR1X file follow the link:

https://doi.org/10.5281/zenodo.8411428

```
'CR1000X Series

'Created by Seb guliksen@gmail.com

'Program running to measure solar irradiation in different asimuth and PoA. Totaly 25 pyranometers Apogee SP-110 installed on 3D printed 26 walls sphere (Rhombicuboctahedron).


'Declare Variables and Units

Public BattV

Public PTemp_C

Public LCount

Public SIR(25)

Public Kglob

Public KglobMJ

Public SIR25

Public SIRMJ25

Public SIRMJ(25)

Public Mult(25)={5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5}

Public Offs(25)={0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0}

Public ModuleState As Boolean

Public Temp


Units BattV=Volts

Units PTemp_C=Deg C

Units Temp=Deg C
```

```
Units SIR=W/m^2

Units SIRMJ=MJ/m^2

Units SIR25=W/m^2

Units SIRMJ25=MJ/m^2

Units Kglob=W/m^2

Units KglobMJ=MJ/m^2


'Define Data Tables

DataTable(Sec,True,-1)

	DataInterval(0,5,Sec,10)

	CardOut(0,-1)

	Sample(1,Temp,FP2)

	Sample(1,SIR(1),FP2)

	Sample(1,SIR(2),FP2)

	Sample(1,SIR(3),FP2)

	Sample(1,SIR(4),FP2)

	Sample(1,SIR(5),FP2)

	Sample(1,SIR(6),FP2)

	Sample(1,SIR(7),FP2)

	Sample(1,SIR(8),FP2)

	Sample(1,SIR(9),FP2)

	Sample(1,SIR(10),FP2)

	Sample(1,SIR(11),FP2)

	Sample(1,SIR(12),FP2)

	Sample(1,SIR(13),FP2)

	Sample(1,SIR(14),FP2)

	Sample(1,SIR(15),FP2)

	Sample(1,SIR(16),FP2)

	Sample(1,SIR(17),FP2)
```

```
        Sample(1,SIR(18),FP2)
        Sample(1,SIR(19),FP2)
        Sample(1,SIR(20),FP2)
        Sample(1,SIR(21),FP2)
        Sample(1,SIR(22),FP2)
        Sample(1,SIR(23),FP2)
        Sample(1,SIR(24),FP2)
        Sample(1,SIR25,FP2)
        Sample(1,Kglob,FP2)
EndTable


DataTable(Hour,True,-1)
        DataInterval(0,60,Min,10)
                CardOut(0,-1)
        Sample (1,Temp,FP2)
        Minimum (1,Temp,FP2,False,False)
        Maximum (1,Temp,FP2,False,False)
        Sample (1,BattV,FP2)
        Minimum (1,BattV,FP2,False,False)
        Maximum (1,BattV,FP2,False,False)
        Average (1,SIR(1),FP2,False)
        Average (1,SIR(2),FP2,False)
        Average (1,SIR(3),FP2,False)
        Average (1,SIR(4),FP2,False)
        Average (1,SIR(5),FP2,False)
        Average (1,SIR(6),FP2,False)
        Average (1,SIR(7),FP2,False)
        Average (1,SIR(8),FP2,False)
        Average (1,SIR(9),FP2,False)
```

Average (1,SIR(10),FP2,False)

Average (1,SIR(11),FP2,False)

Average (1,SIR(12),FP2,False)

Average (1,SIR(13),FP2,False)

Average (1,SIR(14),FP2,False)

Average (1,SIR(15),FP2,False)

Average (1,SIR(16),FP2,False)

Average (1,SIR(17),FP2,False)

Average (1,SIR(18),FP2,False)

Average (1,SIR(19),FP2,False)

Average (1,SIR(20),FP2,False)

Average (1,SIR(21),FP2,False)

Average (1,SIR(22),FP2,False)

Average (1,SIR(23),FP2,False)

Average (1,SIR(24),FP2,False)

Average (1,SIR25,FP2,False)

Average (1,Kglob,FP2,False)

Totalize (1,SIRMJ(1),FP2,False)

Totalize (1,SIRMJ(2),FP2,False)

Totalize (1,SIRMJ(3),FP2,False)

Totalize (1,SIRMJ(4),FP2,False)

Totalize (1,SIRMJ(5),FP2,False)

Totalize (1,SIRMJ(6),FP2,False)

Totalize (1,SIRMJ(7),FP2,False)

Totalize (1,SIRMJ(8),FP2,False)

Totalize (1,SIRMJ(9),FP2,False)

Totalize (1,SIRMJ(10),FP2,False)

Totalize (1,SIRMJ(11),FP2,False)

Totalize (1,SIRMJ(12),FP2,False)

```
		Totalize (1,SIRMJ(13),FP2,False)
		Totalize (1,SIRMJ(14),FP2,False)
		Totalize (1,SIRMJ(15),FP2,False)
		Totalize (1,SIRMJ(16),FP2,False)
		Totalize (1,SIRMJ(17),FP2,False)
		Totalize (1,SIRMJ(18),FP2,False)
		Totalize (1,SIRMJ(19),FP2,False)
		Totalize (1,SIRMJ(20),FP2,False)
		Totalize (1,SIRMJ(21),FP2,False)
		Totalize (1,SIRMJ(22),FP2,False)
		Totalize (1,SIRMJ(23),FP2,False)
		Totalize (1,SIRMJ(24),FP2,False)
		Totalize (1,SIRMJ25,FP2,False)
		Totalize (1,KglobMJ,FP2,False)
	EndTable


	DataTable(Daily,True,-1)
		DataInterval(0,1440,Min,10)
		CardOut(0,-1)
		Average (1,SIR(1),FP2,False)
		Average (1,SIR(2),FP2,False)
		Average (1,SIR(3),FP2,False)
		Average (1,SIR(4),FP2,False)
		Average (1,SIR(5),FP2,False)
		Average (1,SIR(6),FP2,False)
		Average (1,SIR(7),FP2,False)
		Average (1,SIR(8),FP2,False)
		Average (1,SIR(9),FP2,False)
```

Average (1,SIR(10),FP2,False)

Average (1,SIR(11),FP2,False)

Average (1,SIR(12),FP2,False)

Average (1,SIR(13),FP2,False)

Average (1,SIR(14),FP2,False)

Average (1,SIR(15),FP2,False)

Average (1,SIR(16),FP2,False)

Average (1,SIR(17),FP2,False)

Average (1,SIR(18),FP2,False)

Average (1,SIR(19),FP2,False)

Average (1,SIR(20),FP2,False)

Average (1,SIR(21),FP2,False)

Average (1,SIR(22),FP2,False)

Average (1,SIR(23),FP2,False)

Average (1,SIR(24),FP2,False)

Average (1,SIR25,FP2,False)

Average (1,Kglob,FP2,False)

Totalize (1,SIRMJ(1),FP2,False)

Totalize (1,SIRMJ(2),FP2,False)

Totalize (1,SIRMJ(3),FP2,False)

Totalize (1,SIRMJ(4),FP2,False)

Totalize (1,SIRMJ(5),FP2,False)

Totalize (1,SIRMJ(6),FP2,False)

Totalize (1,SIRMJ(7),FP2,False)

Totalize (1,SIRMJ(8),FP2,False)

Totalize (1,SIRMJ(9),FP2,False)

Totalize (1,SIRMJ(10),FP2,False)

Totalize (1,SIRMJ(11),FP2,False)

Totalize (1,SIRMJ(12),FP2,False)

```
	Totalize (1,SIRMJ(13),FP2,False)
	Totalize (1,SIRMJ(14),FP2,False)
	Totalize (1,SIRMJ(15),FP2,False)
	Totalize (1,SIRMJ(16),FP2,False)
	Totalize (1,SIRMJ(17),FP2,False)
	Totalize (1,SIRMJ(18),FP2,False)
	Totalize (1,SIRMJ(19),FP2,False)
	Totalize (1,SIRMJ(20),FP2,False)
	Totalize (1,SIRMJ(21),FP2,False)
	Totalize (1,SIRMJ(22),FP2,False)
	Totalize (1,SIRMJ(23),FP2,False)
	Totalize (1,SIRMJ(24),FP2,False)
	Totalize (1,SIRMJ25,FP2,False)
		Totalize (1,KglobMJ,FP2,False)
EndTable


'Main Program
BeginProg
	'Main Scan
	Scan(5,Sec,1,0)
		'Default CR1000X Datalogger Battery Voltage measurement 'BattV'
		Battery(BattV)
		'Default CR1000X Datalogger Wiring Panel Temperature measurement 'PTemp_C'
		PanelTemp(PTemp_C,50)
		'107 Thermistor sensor inside GLOB
  Therm107(Temp,1,4,VX1,50,50,1,0)
		'SP110 laying on ground (radiation under snow cover)
		VoltSe(Kglob,1,mV1000,5,True,50,60,5,0)
		If Kglob<0 Then Kglob=0
```

```
KglobMJ=Kglob*0.000005
VoltSe(SIR25,1,mV1000,6,True,50,60,5,0)
If SIR25<0 Then SIR25=0
SIRMJ25=SIR25*0.000005

'Turn AM16/32 Multiplexer On
PortSet(C2,1)
Delay(0,150,mSec)
LCount=1
SubScan(0,uSec,8)
    'Switch to next AM16/32 Multiplexer channel
    PulsePort(C1,10000)
    'Generic Single Ended Voltage measurements 'SIR()' on the AM16/32 Multiplexer
    VoltSe(SIR(LCount),3,mV1000,1,True,0,60,Mult(LCount),Offs(LCount))
    LCount=LCount+3

NextSubScan
'Switch to next AM16/32 Multiplexer channel
PulsePort(C1,10000)
'Generic Single Ended Voltage measurements 'SIR()' on the AM16/32 Multiplexer
VoltSe(SIR(25),1,mV1000,1,True,0,60,Mult(25)(),Offs(25)())
    If SIR<0 Then SIR=0

SIRMJ(1)=SIR(1)*0.000005
SIRMJ(2)=SIR(2)*0.000005
SIRMJ(3)=SIR(3)*0.000005
SIRMJ(4)=SIR(4)*0.000005
SIRMJ(5)=SIR(5)*0.000005
SIRMJ(6)=SIR(6)*0.000005
```

```
SIRMJ(7)=SIR(7)*0.000005

SIRMJ(8)=SIR(8)*0.000005

SIRMJ(9)=SIR(9)*0.000005

SIRMJ(10)=SIR(10)*0.000005

SIRMJ(11)=SIR(11)*0.000005

SIRMJ(12)=SIR(12)*0.000005

SIRMJ(13)=SIR(13)*0.000005

SIRMJ(14)=SIR(14)*0.000005

SIRMJ(15)=SIR(15)*0.000005

SIRMJ(16)=SIR(16)*0.000005

SIRMJ(17)=SIR(17)*0.000005

SIRMJ(18)=SIR(18)*0.000005

SIRMJ(19)=SIR(19)*0.000005

SIRMJ(20)=SIR(20)*0.000005

SIRMJ(21)=SIR(21)*0.000005

SIRMJ(22)=SIR(22)*0.000005

SIRMJ(23)=SIR(23)*0.000005

SIRMJ(24)=SIR(24)*0.000005

SIRMJ(25)=SIR(25)*0.000005


            'Turn AM16/32 Multiplexer Off

            PortSet(C2,0)

            Delay(0,150,mSec)

            'Call Data Tables and Store Data


If TimeIsBetween(8,10,24,Hr) Then

ModuleState=True

IPNETPower(5,1)

Else
```

```
ModuleState=False

IPNetPower(5,0)

EndIf

If BattV<11.5 Then

IPNetPower(5,0)

EndIf


                    CallTable Sec

                    CallTable Hour

                    CallTable Daily

          NextScan

EndProg
```
