Cat. No. 101-446

実験の手引き

グリーンハウス

ST-2997

本書をよくお読みのうえ,安全事項・警告内容を 充分ご理解いただき,正しくご使用ください。 いつでも使用できるように大切に保管ください。

株式会社 島津理化

ご使用に際しての安全上の注意事項

	ご使用に際しての安全上の注意事項					
本書では安全にご	使用いただくためのいろいろな絵表示をしています。					
その内容を理解することなく誤った取り扱いをすることによって生じる						
内容を次のように	規定しています。					
記載されている内	容をよく理解してからご使用ください。					
▲注意	誤った取り扱いをすると、人が傷害を負ったり、物 的損害の発生が想定される内容を示します。					
注記	装置を正しく使用していただくための情報を示し ています					

絵表示の意味

	絵表示の意味
	この絵表示は, 「警告」「注意」を促す事項を示しています。 この絵表示の近くに, 具体的な警告・注意内容を表記しています。
\bigcirc	この絵表示は, 禁止事項を示しています。 この絵表示の近くに, 具体的な禁止内容を表記しています。
	この絵表示は, 必ず実行して欲しい事項を示しています。 この絵表示の近くに, 具体的な指示内容を表記しています。

安全上の注意

	▲ 注意
\bigcirc	グローライトは非常に明るい LED を使用しています。操作中に光を直接 見ないでください。
\bigcirc	グローライトは熱くなります。 操作中または長時間の実験後は冷えるまで 触らないでください。
\bigcirc	センサボックス,電源プラグ,露出した電子基板に水を近づけないでくだ さい。

もくじ

はじめに
実験1:晴れた日のプログラム6
1. 1. 目標
1.2. 使用機器
1.3.安全上の注意6
1.4. ブロックの役割
1.5. 事前練習
1.6. 実験
1.7. 追加実験
実験 2 : USB ファンを使った風のプログラム14
2.1. 目標
2.2.使用機器
2.3.安全上の注意14
2.4. ブロックの役割15
2.5. 事前実験
2.5.1. セットアップ
2.5.2. センサベースおよびコード生成のデータ表示
2.5.3. 複合条件
2.6. 実験
2.7.追加実験
実験 3:雨のプログラム
3.1. 目標
3.2. 使用機器
3.3.安全上の注意
3.4. 取扱いのヒント
3.5. ブロックの役割
3.6.事前実験
3.6.1. グリーンハウスセンサのセットアップ 31
3.6.2. ソフトウェアのセットアップ 31
3.6.3. リザーバーのセットアップ 32
3.6.4. 給水の準備
3.7. 実験
3.8.追加実験
実験 4 : グリーンハウス内の水の移動を最適化する
4.1.目標
4.2. 使用機器

4.3.安全上の注意	 43
4.4.ブロックの役割	 44
4.5. 事前実験	 45
4. 6. 実験	 55
4. 7. 追加実験	 56
付録 A:10 の小型屋内植物の特性	 57
実験5:グリーンハウスの状態感知および制御システム	 58
5. 1. 目標	 58
5.2. 使用機器	 58
5.3.安全上の注意	 59
5.4. 比例制御	 59
5.5.事前実験	 61
5.6. 実験	 66
5. 7. 追加実験	 72

はじめに

この度は「グリーンハウス ST-2997」をお買い上げいただきまことにありがとうございます。

グリーンハウスは,温度・湿度・光・土壌水分センサから得たデータを用いて光源の色や USB フ アン,ウォーターポンプをプログラミングで制御でき,学生が独自の温室を設計することができ ます。

コントロールノードはグリーンハウスのコントロールセンターとして機能し, USB ファンなどの アクセサリへの電力供給やプログラムの保存が可能です。コントロールノードを接続すると,ア クセサリをコントロールすることが可能になり,学生は部品の組み合わせ,状態の監視,散水の 自動化,グリーンハウス内の様々な条件の影響の研究を行うことができます。

プログラムは専用ソフトウェア SPARKvue と PASCO Capstone のプログラムツールでブロックコードを使用します。ブロックコードを使用すると、学生は構文を気にすることなく、プログラミングの基本を簡単に習熟できます。

基本をマスターした後は、学生は条件をカスタムし、例えばデータ表示等を備えた独自のプログ ラムを作成できます。

本書では SPARKvue の使用を想定して記載しておりますが, PASCO Capstone でも手順は変わりません。

グリーンハウスを用いて以下のような学習をすることができます。

- 晴れた日のプログラム
- ・ USB ファンを使った風のプログラム
- 雨のプログラム
- ・ グリーンハウスの水の移動を最適化する
- ・ グリーンハウスの状態感知および制御システム

実験1:晴れた日のプログラム

x やyを使って数学の問題を解いたり,理科の実験をしたりしたことがある人は,「変数」を使っ たことがあるでしょう。変数とは,値を変更できるあらゆる情報です。数学の授業において x は変 数で,理科の実験で植物の成長速度を勉強している場合においては植物の高さと時間が変数です。 コンピュータプログラムは,プログラム実行中に参照できる値を格納するために変数を使用しま す。

ループについても経験があるでしょう。プログラマーは何かを複数回実行したいときにループを使 用します。例えば、スマートフォンで間違ったパスワードを入力した場合、ループを使用して、正 しいパスワードの入力を促します。

この調査の目標は、変数とループを使用して、グローライトを太陽のように動作させるコードを作 成することです。グローライトが通常の電球のような白色光を生成しないことに驚くかもしれませ ん。グローライトには、赤と青のLEDライトしかありません。植物の観点からみると、グローラ イトは太陽光に似ています。太陽光は虹のすべての色が利用できるにもかかわらず、植物は主に赤 と青の光を利用します。植物の葉の緑色のクロロフィルは、赤色光と青色光から光合成のためのエ ネルギーを最も効率的に吸収します。

1.1. 目標

- 変数とループを使用して、太陽の1日のサイクルをモデル化した光の出力を制御するプログラム を作成します。
- ・ 様々な種類のループがどのように機能するかを調べます。

1.2. 使用機器

・専用ソフトウェア	・コントロールノード
SPARKvue または PASCO Capstone	
・グローライト	・Eco チャンバーの蓋
(パワーアウトケーブルと USB 電源アダプタ付き)	
・フラット1穴ストッパー(#6)	

1.3. 安全上の注意

グローライトは非常に明るい LED を使用しています。 操作中に光を直接見ない でください。
クローライトは熱くなります。操作中または長時間の実験後は冷えるまで触ら ないでください。

1.4. ブロックの役割

【繰り返しブロック】

[繰り返し]ブロックを使用して、プログラムがコードを実行する回数を制御します。



上記の[繰り返し]ブロックは, [続ける条件](条件が有効な間)または[終わる条件](条件が満たされるまで)タスクを繰り返します。パズルのようにブロックを追加することで繰り返し条件を設定できます。

💽 を 🚺 から 🚺 まで 🚺 ずつカウントする

上記の[繰り返し]ブロックは,カウントブロックです。この例では,1から10までの整数を一度 に1ずつカウントアップするように設定しています。ブロック数(i)を減らすか,別の範囲(1~10) を使用するか,別の整数(1)でカウントするかを選択できます。

【変数ブロック】

[変数]ブロックはプログラム実行中に参照する値を格納するために使用されます。

[変数]ブロックカテゴリで変数を作成できます。変数を作成すると、その変数名を持つ新しいブ ロックが[変数]カテゴリに表示されます。

【時間ブロック】

[時間]ブロックを使用してコードを一時停止したり、プログラムの開始時からのタイムスタンプ を取得したりします。[時間]カテゴリには指定した時間コードを一時停止する[sleep]ブロックが含 まれています。スリープ中のプログラムは、他のタスクを実行しません。

以下の[sleep]ブロックは、プログラムを 20 ミリ秒(0.02 秒)停止させます。必要に応じて時 間単位をミリ秒(ms)から秒(s)に変更できます。

sleep for 20 ms

【グローライトのオン/オフスイッチ】

グローライトが接続されたコントロールノードが SPARKvue に接続されている場合,次のプロ グラムを実行するまでグローライトはオフのままです。各色の光の強さを 0(オフ)から 10(最大照 度)の範囲で変化させるコードを記述します。

【グローライトブロック】

[グローライト]ブロックを使用してグローライトのオン/オフを切り替え、赤と青のカラーミックスを制御します。ブロックは[機器]ブロックカテゴリにあります。

0 set grow light for **//control.Node** port A to brightness R

-7 -

●port: グローライトが接続されているポートをAまたはBから選択します。

❷brightness:赤[R]と青[B]の明るさを0から10の間で設定します。0に設定するとライトがオフになり、1から10の間に設定するとライトがオンになります。10が最大の明るさです。この明るさは整数単位で設定できます。

1.5. 事前練習

パート1:グローライトのセットアップ

- 図1①に示すように、グローライトのポスト(支柱) を1穴ストッパーにはめて Eco チャンバーの蓋に セットします。
- 2) ②グローライトのフラットパワーアウトプットケ ーブルをコントロールノードの Power Out ポート Aに挿し込みます。
- コントロールノードを接続します。[デジタル]テン プレートを選択します。



- 4) プログラムツール
 な
 を
 開きます。ツールがすでに開いている場合は、既存のブロックを ドラッグして右下にある薄いグレーのゴミ箱マークまで持って行き、削除します。
- 5) グローライトの USB ケーブル ③を電源アダプタに接続し、電源アダプタをコンセントに差し込みます。
- パート2: ライトの明るさと色をプログラムする
 - 1) 赤と青の両方のライトの明るさを最大値に設定します。[時間]ブロックを追加して, 10 秒から 30 秒の間でライトをオンにし, コードをテストします。

✓ 時間単位をミリ秒(ms)から秒(s)に変更します。

【プログラム例】															
set grow light for //control	.Nod	e 🔻) por	t A	•	to b	righ	tne	ss I	R (10		в	10	
sleep for 20 S	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•

2) コードを修正して、赤いライトだけを数秒間点灯させてから、青いライトだけを数秒間点灯 させます。どのようなプログラムを作成すれば、赤色のライトだけをオンにして青色をオフ にすることができますか?



- パート3:繰り返すライトサイクルを設定する
 - 1) ループを使用して[停止]を押すまで赤いライトを半分の明るさで7秒点灯させ、3秒間消灯 させるシーケンスを繰り返します。繰り返されるコードはループ内に入れます。

繰り返す:続ける条件 V true V 実行	
【プログラム例】	
「繰り返す:続ける条件・」」 true・	
実行 set grow light for (//control.Node * port A * to brightness R 15 B 0	
sleep for (7 ms)	
set grow light for <i>[//control.Node</i> > port A > to brightness R (0 B 0	
sleep for 1 3 ms v	

2) [カウントループ]ブロックをコード領域にドラッグし, [変数]を開きます。

追加した[カウント]ブロックには[i]という変数が含まれているため、ブロックリーによって[i]という変数が自動的に作成されます。必要に応じて、変数名を変更できます。 以下のコードを作成します。



指定されたコードがどのように動作するかを観察し、コード値、変数、外側のループの有無を変 更して、次の a~d について調べます。

a. 外側の[繰り返し条件]ループを[カウント]ループと組み合わせて使用するのはなぜですか。

【回答例】

[カウント]ループ自体は 1 回だけ実行される。[繰り返し条件]ループはプログラムを停止するまで一連の色の変更を続ける。

b. 変数[i]は[カウント]ループでどのように機能しますか。

【回答例】

[カウント]ループは変数[i]のステップの範囲およびステップのサイズや範囲内の各値の変化 量を定義できる。上記のコードでは、開始は青色の光が 0(オフ)に設定され、10 になるまで 1 ずつ増加する。変数[i]を青色光から赤色光に移動し R[i] B[10]に設定すると赤色光で同じ ことが起こる。

c. [繰り返し条件]ループを[(10)回繰り返す]ループに切り替え,値を 10 より小さい値に変更しま す。このループと[繰り返し条件]ループの違いは何ですか。

【回答例】

[(10)回繰り返す]ループを使用すると、手動で停止するまでコードを繰り返すのではなく、コードを繰り返す回数を定義することができる。

d. [数学]ブロックカテゴリを開き、加算ブロックを探します。

[(10)回繰り返す]ループの数字を、一桁の答えになる単純な計算に置き換えて、結果を観察しま す。[数式]ブロックのドロップダウンメニューを使用して "+"以外の様々な数式関数を試しま す。この特定の[数式]ブロックを使用して学んだ最も興味深いと思ったことを説明してください。

【回答例】

数学の負の答えが正の数と同じように機能したのは興味深かった。これはすべての状況で機 能するわけではないが、このコードではそのようになっている。

3) グローライトを1日24時間点灯したままにするのではなく,植物の成長のために光エネル ギーを変えることが重要なのはなぜですか。 【回答例】

生物は地球の太陽周期に対応して睡眠周期などの周期をたどるように進化してきた。人間が 睡眠を必要とするように,植物は健康を維持するためのメンテナンス作業を完了するために 夜のサイクルを必要とする。

1.6. 実験

24 時間の連続ループで太陽の動きをシミュレートするために、学んだことをすべて組み合わせてください。

1日24時間のサイクルは毎日真夜中に始まると仮定し、1日を通して起こる光の強度と色の変化を考慮します。単純化するために、あなたが住んでいる場所の日の出と日没の間に、一方の地 平線から他方の地平線まで180度の完璧なルートを動いていると仮定することもできます。

別の用紙に、プログラムのスクリーンショットと印刷またはスケッチを作成し、コーディング 方法を説明します。後で参照できるように、作業内容は SPARKvue に保存しましょう。

 ✓ 時間ブロックは秒(s)で設定しますが、テストの準備ができたら時間単位をミリ秒 (ms)に変更します。これにより、24時間分のコードを約1.5分(86,400 ms または 86.4 s)でテストできます。コードに自信がある場合は時間単位を秒に戻します。



グローライトが非常に熱くなるため,実験が終了したら,グローライトの電源を 必ず取り外してください。



このコードは真夜中に開始する設計なので,スタートから6時間(21,600秒)は両方の ライトがオフになります。日の出と日の入りの時間帯は,青色光の散乱により光が暖色 になるため,その時間帯は青色光を含まないで弱い赤色光のみを表示するようにしま す。

次に午前7時から午前11時まで、一定の割合で(1時間または3600秒ごとに2単位の明るさ)で赤色光と青色光の両方の明るさを維持します。

2番目の[カウント]ループでは、午後1時から午後6時(日の入り)まで一定の割合 (3600秒ごとに2単位)で明るさを減らす計算を使用しています。日の出前と日の入 り後は両方とも完全に消灯します。

1.7. 追加実験

プログラムを次のように変更します。

- ・ 真夜中から真夜中ではなく,授業開始から次回の授業開始までの24時間サイクルを実行します。
- ・ 毎日のライトサイクルに自動補正を追加して,太陽光の強度と持続時間の季節変化をシミュレー トします。
- ・ 1時間の間に赤から紫,青への光の色をシームレスに移行させることで,24時間のサイクルを通じて光の変化をスムーズにします。
- ・ 24時間ではなく、数秒の連続サイクルで、赤、紫、青、紫から赤へと移行します。





グローライトが非常に熱くなるため,実験が終了したら,グローライトの電源を 必ず取り外してください。

実験 2: USB ファンを使った風のプログラム

この実験では、グリーンハウス内を理想的な温度に保つため、グリーンハウス内の温度測定値を設定 条件として USB ファンをオンにするタイミングと駆動時間を決定するプログラムを作成します。 風は温度調節を行うことに加えて、葉の近くの空気を動かして光合成に関係する二酸化炭素・酸素・ 水蒸気が周囲の空気と混ざるのを助けます。植物を強く成長させるためには頻繁な空気の移動を必要 とします。

2.1. 目標

- ・ リアルタイムのセンサ入力値を使用して,条件ステートメントを評価するコードを作成します。
- ・ ファンを動作させてグリーンハウス内の温度を調整します。

2.2. 使用機器

・専用ソフトウェア	・コントロールノード
SPARKvue または PASCO Capstone	
・グリーンハウスセンサ	・グリーンハウスセンサモジュール
	(センサモジュールケーブル付き)
・グローライト	・パワーアウトプットモジュール
(パワーアウトケーブルと USB 電源アダプタ付き)	(パワーアウトケーブル付き)
・USB ファン	・Eco チャンバー
・浅めの器	・ジップ付袋
・氷	

2.3. 安全上の注意

- センサボックス,電源プラグ,露出した電子基板に水を近づけないでください。
」 露出した電子基板が金属または導電性の表面に接触しないようにしてくださ い。
グローライトは非常に明るい LED を使用しています。 操作中に光を直接見ない でください。
グローライトは熱くなります。操作中または長時間の実験後は冷えるまで触ら ないでください。

2.4. ブロックの役割

コンピュータプログラムは,条件が満たされているかどうかをチェックするために[true]と[false] という単語を使用します。

【If ブロック】※本ブロックの説明は英語ブロックが分かりやすいため、日本語と英語を併記します



If ブロック①は,条件が満たされて[true]のときにアクションを実行します。このブロックは [論理]カテゴリ内にあります。[実行/do]は[もし/if]が満たされた場合に,アクションを実行します。 ブロックの設定アイコン(歯車)を押すと,ブロックを修正することができます。

Ifブロックがどのように変更できるかの例を示します。

2は[そうでなければ/else]を追加したブロックです。

◆ [もし/if]寒ければ[実行/do]厚着をして、[そうでなければ/else]そうでなければ好きなもの を着ます。

③は[そうでなくもし/else if]は最初の if 文の条件が満たされない場合に、アクションを実行します。

◆ [もし/if]寒ければ[実行/do]厚着をして、[そうでなくもし/else if] 暑ければ[実行/do]薄着に します

ステートメントは上から下に順番にチェックされます。複数の[そうでなくもし/else if]を組み合わせて一連の条件を定義し,[true]ごとに異なるアクションを設定できます。

❹は歯車マークでブロックを修正し[そうでなければ/else]を追加したブロックです。

- ◆ [もし/If]寒ければ[実行/do]厚着をして、[そうでなくもし/else if]暑ければ[実行/do]薄着を、
 [そうでなければ/else]どちらでもなければ好きなものを着ます。
- ▶ If ブロックに条件を追加する
 - 1. [比較]ブロック 「 「 = 三 ト

:2つの値を比較するために使用します。比較記号は次の中から選択します。

2. [論理演算]ブロック 「「 <u>かつつ</u>」 / 「 and 」 「

:[かつ/and]は前後のブロックが両方とも[true] (真) であるかどうかを評価します。[かつ/and] を[または/or]に変更して,前後のブロックの内,少なくともどちらか1つが真であるかどうか を評価することも可能です。

 $= \neq < \leq > \geq$

複数の条件を評価する必要がある場合は、前後のブロックの中にさらに条件ブロックを追加して、条件文を入れ子にします。

入力と出力

グリーンハウスセンサはコードの条件設定に使用するための環境情報を取得します。センサの 測定値を条件文に入れてそれが真かどうかを評価するのに使用するため、センサのデータは 「入力」になります。

グローライトと USB ファンは, If 文が[true]または[false]のいずれかであると判断された場合 に動かす「出力」アクションになります。

[機器]カテゴリでは、グリーンハウスセンサなどの接続されたすべての PASCO センサによる [入力]と、グローライトや USB ファンなどの[出力]にアクセスできます。

[プログラム出力]カテゴリには、データ表示に表示するテキストまたは数値出力を作成するブ ロックが含まれています。

【センサ測定値ブロック】

グリーンハウスセンサの測定値は、センサ測定値ブロックで使用できます。センサがコントロー ルノードに接続されている場合、デフォルトでは温度測定が表示されます。

value of Temperature • °C •

ドロップダウンメニューを使用して、別のセンサ測定値を選択したり、℃から K など単位を変更 したりできます。

プログラムで測定が1回だけ呼び出される場合は、センサ測定値ブロックをコードに直接追加し ます。センサの呼び出しの数が多いほどプログラムの実行速度は遅くなってしまいます。プログ ラムの実行速度の低下を回避するには、センサの測定値を使用して変数を定義し、センサの測定 値を直接使用するのではなく、コード内で変数を呼び出します。

▶ 明るさ:入力と出力

明るさは入力と出力の両方として使用されます。

グリーンハウスセンサモジュールの光センサは「入力」として,0%~100%の強度の範囲で明 るさを報告します。

グローライトは「出力」として、LEDの明るさを0(オフ)~10(最大輝度)の範囲で設定します。

【パワーアウトプットブロック】

パワーアウトプットブロックで USB ファンを制御します。このブロックでは、パワーアウトプ ットモジュールが接続されたコントロールノードをペアリングしている場合に、[機器]カテゴリ で使用できます。ここでは次の4つの設定を使用できます。



①コントロールノードの選択:複数のコントロールノードをペアリングしている場合、使用する コントロールノードを選択できます。 パワーアウトプットとチャンネル選択:このドロップダウンを使用してコントロールノードのパワーアウトプットポート(AまたはB)およびパワーアウトプットモジュールチャンネル(CH1またはCH2)を設定します。コントロールノードのパワーアウトプットモジュールチャンネル(CH1ワーアウトプットモジュールのチャンネル1と2に何が接続されているか確認してください。
 電源セレクタ:各パワーアウトプットモジュールチャンネルには、電力を受け取ることができる3つの異なるプラグがあります。USBプラグと2種類のターミナルプラグ(ピンとブロック)です。この実験ではUSBプラグのみを使用しますが、同じチャンネルのすべてのプラグの電源が同時に入ることに注意してください。

④true/false 値:ファンなどの USB デバイスの場合, on の後が[true]のときは、選択したチャンネル内の USB プラグが電力を受け取ります。on の後が[false]のときは、USB プラグに電力が供給されません。ターミナル出力は、0%(オフ)~100%(最大)まで調整できます。

2.5. 事前実験

- 2.5.1. セットアップ
 - グリーンハウスセンサをコントロールノードのセンサポートに接続します(図2①)。
 グローライトはポート A, パワーアウトプットモジュールをポート B に接続します(図2②
 ③)。



図2. コントロールノードのセットアップ

- 2) センサモジュールから2穴のストッパーを取り外します。外したナット, ワッシャー, モジ ュールをわきに置きます。
- 3) 図3①に示すように、クラフトスティックを使用して2穴のストッパーの切り込みを広げ、 センサモジュールケーブルの一部を広げた穴の端に押し込みます。
- 4) スティックを取り外し,残りのカット部分にケーブルを押し込み,ケーブルがストッパーの 穴を自由に通過できるようにします(図32)。



- 5) ストッパーの細くなっている端にあるプラグの矢印記号を見つけます。図3**3**のようにプラ グをグリーンハウスセンサモジュールに接続します。
- 6) もう一方のプラグをグリーンハウスセンサの LIGHT HUMIDITY TEMP ポートに接続し ます。図34のようにプラグの矢印をグリーンハウスセンサの背面の矢印に合わせます。
- 7) 次の作業を行うときは図4を参照してください。
 - a. グリーンハウスセンサモジュールと数センチ余裕を持たせたケーブルを Eco チャンバ ーの蓋の上の穴に通します。2 穴ストッパーで穴を塞いでください。
 - b. グリーンハウスセンサモジュールのネジをストッパー穴の底側から通して(図4 A), ストッパーとモジュールが接触するようにします。ストッパーの上部に出ているネジの 上にワッシャーをはめ、ワッシャーの上からナットを手で絞めます。これでモジュール がカバーの下に固定されます。
 - c. ストッパーから余分なケーブルをゆっくりと引き出します。



図4. 光センサを蓋の中央に配置する

- 8) 蓋の開いている穴に平らな1穴のストッパーを押し込みます。グローライトのネジをストッパー穴に挿し、図4のようにライトを蓋にセットします。光センサがグローライトの中心に来るまで、グローライトとセンサモジュールのストッパーを回します。図4 Bの矢印はセンサモジュールの光センサを指しています。
- 9) USB ファンをパワーアウトプットモジュールのチャンネル1(CH1)に接続します。
- 10) Eco チャンバーを閉じるためにチャンバーと蓋の間の2つの側面の穴を合わせます。残った穴は穴の開いていないストッパーで塞いでください。
- 11) コントロールノードの電源を入れ、ソフトウェアに接続します。[センサデータ設定]で次の手順を実行します。
 - a. グリーンハウスセンサで[温度]と[照度]の測定値のみを選択↓します。

- b. コントロールノードのオンボードセンサ測定のスライダをオン ●からオフ に変え て無効にします。
- c. [デジタル表示とグラフ表示]のテンプレートを選びます。
- 12) データ収集を開始します。室温と明るさを記録し、データ収集を終了します。

【回答例】

室温 21.6℃, 照度 9.45%

13) グローライトの USB ケーブルを電源アダプタに挿し込み,コンセントに接続します。

2.5.2. センサベースおよびコード生成のデータ表示

- プログラムツールアイコンをクリックしてプログラム画面を開きます。
 両方の色を最大強度に設定してグローライトをオンにするコードを記述します。
- 2) プログラムツールアイコンをもう一度押すと、データ表示画面に戻ります。
- 3) X 軸の下の測定値[時間(s)]を選択します。時間の単位を秒から分に変更するには[s]をクリックして[min]を選択します。
- 4) データ収集を開始します。温度が 30 秒以上同じ状態になるまでデータ収集をし、その後デ ータ収集を停止します。待っている間に手順7)を確認し、コードの作成を開始します。 温度が安定したら、最終的なチャンバーの温度と照度レベルを記録します。

【回答例】

赤色光と青色光の強度が最大の場合、温度は 24.2℃、照度は 47.38%

5) チャンバーの側面 (蓋との境) のストッパーを取り外します。グローライトまたはグリーン ハウスセンサモジュールの位置を動かさないように注意して蓋を取り外します。 グローライ トが冷めるようにテーブルの上に置いてください。

 ✓ 光センサとグローライトは、実験全体を通じて同じ位置を維持する必要があります (図4参照)。
 いずれかのコンポーネントが移動した場合は、最大照度の明るさの読取値が手順4)
 で記録した明るさと等しくなるまで慎重に位置を調整します。

6)図5に示すように、実験に使用する冷却システムを構築します。 Aのように浅めの器に氷 を入れます。





図5. 冷却システム

- 7) ジップ付袋に器を入れ、袋の空気を抜いて密封します。
- 8) センサモジュールの温度センサの真下に図50の冷却システムを配置して、チャンバーの 蓋をテーブルにセットします。このとき、図50のように冷却システムがセンサモジュー ルのどの部分にも接触しないようにします。
- 9) プログラムツールを開きます。グローライトの色と強度の組み合わせの違いが、閉じた Eco チャンバー内の温度にどのように影響するかをテストするプログラミングを作成します。 テストとテストの間は冷却システムを使用してモジュールを室温に戻し、各テストは同じ温 度で開始します。
 - a. チャンバーの温度は、赤色光と青色光のどちらの影響を強く受けるか、または両方の 色が同じ影響を与えるか、説明してください。ただし両方のライトが同じ強度に設定 されていると仮定します。

【回答例】

青色光は赤色光にくらべて,チャンバーに多くの熱を与えます。例えば青色光の みが強度3で点灯している場合,同じ強度で赤色光のみが点灯している場合と比 べて,温度はより早く上昇し,より高いレベルに到達します。

b. 光の強さと色を使ってチャンバー内の温度をコントロールできますか?理由は何ですか?

【回答例】

どのような強さの光の色でも、チャンバー内の温度をいくらか上昇させるので、必要に応じてチャンバーを温めるために光を利用することができますが、チャンバー を冷やすことはできません。光が必要だが温度をあまり上昇させたくない場合は、 妥協案として光の強度を下げることになります。 c. すべての実験を表示するには、グラフの凡例のチェックボックスにチェックを入れます。 各実験の0秒における明るさについてわかることは何ですか?

【回答例】

センサはプログラムの実行開始前にデータの収集を開始するため,時間=0秒の最初のデータポイントではグローライトはオフになります。最初のデータポイントは,ライトが点灯する前のバックグラウンドの光の強度を測定します。2番目のデ ータポイントまでにグローライトがオンになり,センサはバックグランドの光と合わせた明るさを報告します。

- 10) センサモジュールを室温に戻し, チャンバーの蓋をします。 チャンバーの側面の穴に USB ファンをセットします。 チャンバー側面のもう一つの穴はストッパーで塞ぎます。
- 1 1) 次のコードを作成し, [室温]の値を[123]から P.19 2.5.1 の 12)で記録した室温に置き換え ます。また,時間の単位を[s]に変更します。時間または温度の値を自由に変更して,テスト を高速化できます。



- 12) プログラムツールアイコンをクリックして計測画面を開きます。上部のデジタル数値表示 は[温度]のままで、下部のデジタル数値表示画面を変更します。[照度]をクリックして開い たリストの[センサ]タブを[ユーザ設定]タブに切り替え、[チャンバーの状態]テキスト出力に 設定します。
- 13)データ収集を開始します。データ表示の測定値とテキスト出力に基づいて、コードがどの ように動作するかを観察します。ファンが自動的にオンになり、温度が下がり始めるまで、 データを収集し続けます。データ収集が終わったら次の質問に答えてください。
 - a. 最初のいくつかのコードブロックがループ内に含まれていないのはなぜですか?

【回答例】

最初の[変数セット]ブロックは[繰り返し]ブロックの参照であるためループの外側にあ ります。室温と光の強度はどちらも固定された変わらない値なので、コードは新しい値 をチェックし続ける必要はありません。 b. コードを翻訳して指示内容を説明してください。

【回答例】

現在のチャンバー温度が室温より 2.5°C 以上高い場合は、"暑い"を[チャンバーの状態]テキスト出力で表示し、ファンを 30 秒間オンにします。ただしチャンバー内の 温度が室温以下の場合は[室温と同じくらいか下回っている]と表示し、ファンをオ フのままにします。プログラムが停止するまで、これらの手順を繰り返します。

14) P.19 2.5.1 の 12)で記録した周囲の明るさを測定値から差し引いて補正したグローライト の明るさを表示する数値出力を作成します。温度の表示を数値出力に置き換えます。必要に 応じて数値出力が目的どおりに動作するようにコードを変更します。コードに追加された新 しいブロックをメモし、加えた変更をまとめます。



2.5.3. 複合条件

1) [tempReading]という変数を作成し、下のプログラムに従って[論理]ブロック内を変更しま す。プログラムをテストし、少なくとも2種類のテキスト出力メッセージが表示されるまで データを監視します。



 センサの測定値を直接使用するのではなく変数を使用する理由を説明し、1つまたは複数の [そうでなくもし/else if]文を追加すると、プログラムによる情報の評価方法がどのように変 わるのかを説明します。

【回答例】

この場合プログラムは1つの温度値を複数回評価します。変数を使用して温度値を呼び 出すことで、プログラムをできるだけ高速に実行できます。If ブロックを変更する前は コードが評価できる条件は1つだけでした。[そうでなくもし/else if]文を使用すると複 数の条件を評価できます。今回のプログラムは USB ファンがチャンバー内の温度を下 げることができる条件を満たすのに十分な温度を維持できない場合に、グローライトの 強度を変更するためのテキスト出力アドバイスを提供できるようになりました。

2.6. 実験

次のすべての条件下で,可能な限り最高の明るさになるように光の強度を調整するプログラムを作成 してください。

- ・ ファンを常時作動させずにグリーンハウスの温度を室温よりも2.5℃以上高くしてはならない。
- チャンバーの温度が最高温度に達すると、ファンが温度を維持できるようになるまで、グローラ イトの光の強度を徐々に下げる(赤色光と青色光の強度は必要に応じて異なる値にすることがで きます)。
- ・ グリーンハウス観察者に有用な情報を提供する 1 つ以上の新しいテキストまたは数値出力を組

み込む(2.5.3のテキストまたは数値出力を使用する必要はありません)。

新しいテキストまたは数値出力を観察者に示すデータ表示を少なくとも1つのセンサの測定表示を含めて作成する。

p.18 図4に示しているように,蓋に取り付けたグローライトとセンサモジュールは所定の位置に保持するようしたうえで,穴の開いていないストッパーを追加したり取り外したりして通気を試してみてください。

別の用紙にコードをスクリーンショットして印刷またはスケッチし、その動作方法とデータ表示の選択肢を説明します。後で参照できるように、作業内容を SPARKvue に保存します。



2.7. 追加実験

グリーンハウスの外の状況は今と同じとは限りません。プログラムを次のように変更します。

- ・ 日中の室温と自然光レベルの変化に対応させます。
- ・ チャンバーが温度制限を超えた場合は、一時的にグローライトをオフにするコードを追加します。
- ・ チャンバーの温度が室温よりもかなり低くなった場合はグローライトの明るさを高い値にジャ ンプさせます。





実験3:雨のプログラム

水は少なすぎても多すぎても植物に害を及ぼします。温室(グリーンハウス)は空気の流れが限られ ているので,湿度が高すぎると葉っぱにカビが生えたり,水をやりすぎると根が腐ったり,微生物が 過剰に繁殖したりして植物が枯れてしまうことがあります。

この実験では、土壌水分状態を自動的に監視するプログラムを作成し、土壌水分データから土壌が必要とするときに適切な量の水を供給し、システムの水供給を補充するときに警告を発するウォーター ポンプシステムを設計します。これらの自動化されたタスクはそれぞれ関数、つまり一連のステップ に分けられた再利用可能なタスクで対処します。

3.1. 目標

- ・ データを使用して、自動給水システムの設計に必要な情報を得ます。
- ・ プログラム内のタスクを整理する関数を作成します。

3.2. 使用機器

・SPARKvue または PASCO Capstone	・コントロールノード
・グリーンハウスセンサ	・土壌水分プローブ
・USB ポンプ	・パワーアウトプットモジュール
	(パワーアウトケーブル付き)
・グリーンハウスアクセサリキット	・Eco チャンバー
・小さなカップまたはビーカー	・植木鉢
	(直径約 10cm,高さ約 10cm 以下)
・鉢植え用の土	・メスシリンダー10mL
・リザーバー	・ゴムバンドまたはジップタイ
・バインダークリップ	・はさみ

3.3. 安全上の注意

	センサボックス, 電源プラグ, 露出した電子基板に水を近づけないでく ださい。
注記	露出した電子基板が金属または導電性の表面に接触しないようにして ください。

3.4. 取扱いのヒント

- この実験では, [変数][ループ][測定値へのアクセス][パワーアウトプットモジュールの操作][数値 およびテキストの表示作成]に関する背景知識が必要です。
- 事前に生徒用の土を用意しておきます。大き目のジップ袋に,湿った状態になるまで鉢植えの土

に徐々に水を加えます。1カップの水と5カップの乾いた土から始めます。湿っている土を使用 する場合は、水の量を減らします。

複数の土壌サンプルを必要とする場合は、植木鉢3個分の十分な土を用意してください。

- 植木鉢は本キットには含まれていません。直径と高さが約10cm以下の鉢が必要です。植木鉢がない場合は、底にいくつかの排水用の穴を開けた同じサイズのプラスチックカップを使用します。
- リザーバーは本キットには含まれていません。水のオーバーフローを防ぐために、理想的なリザ ーバーの水量は、Ecoチャンバーの下半分の空きスペースよりも少なくする必要はあります。チ ューブと USB ケーブルを収納できる蓋つきの容器を探します。生徒はリザーバーのおおよその 容積を知る必要がありますので、リザーバーには油性マジックで容量(mL)を記入します。
- プログラミング時間を最大限確保するために、「3.6.1 グリーンハウスセンサのセットアップ 1)
 ~6)」と「3.6.3 リザーバーのセットアップの 1)~5)」を事前に完了しておく必要があります。
- チューブを接続するには、かなりの握力が必要です。若い学生には援助が必要かもしれません。
 チューブからの挿入と取り外しを容易にするために、コネクターウェッジに少量のグリセリンを 加えるとよいでしょう。
- チューブは必要に応じて短くカットできます。交換用チューブアクセサリキット(チューブ、コネクタ、ドロップヘッド)は型番 PS-3348 で購入できます。また、交換用チューブとドロップヘッドを個別に購入することもできます。交換用のポリウレタンチューブはホームセンターなどでも購入できます。チューブのサイズは次のとおりです。

USB ポンプに接続された透明チューブ:外径 = 3/8 インチ,内径 = 1/4 インチ

青色チューブ:外径 = 1/4 インチ,内径 = 3/16 インチ

付属のドロップヘッドの流量は2 GPH です。必要に応じて流量の異なるヘッドを使用してくだ さい

3.5. ブロックの役割

【関数ブロック】

関数ブロックは[関数]カテゴリにあります。例えば下図のようなプログラムを作成します。

② ? 関数 makeNoise	
set //control.Node Speaker, 651-416 (0-20000) * frequency to 100	Hz
sleep for 200 ms	• •
set //control.Node Speaker, 651-416 (0-20000) * frequency to 500	Hz
sleep for 200 ms	• •
set //control.Node Speaker, 651-416 (0-20000) * frequency to 1000	Hz
sleep for 200 ms	• •
tum (//control.Node Speaker, 651-416 •) on: (false •	• •

① [関数]カテゴリから関数ブロックを出し、[何かする]の部分をクリック(タップ)して関数名を

-27 -

入力します。今回は[makeNoise]とします。

② 関数何かする
 ② [機器]カテゴリと[時間]カテゴリから各ブロックを出して組み合わせます。
 Set //control.Node Speaker, 651-416 (0-20000) 「frequency to 1 Hz

[関数]はプログラムに挿入するためのもので作成後にプログラムを実行しても何も起こりません。 [関数]カテゴリを開くと, [makeNoise]関数が作成されています。



作成した[makeNoise]関数を繰り返しブロック内に入れると、関数の内容が繰り返され音を演奏 します。

繰り返	豆す:続ける条	件	tr	ue	2
実行	makeNoise				

【入力を持つ関数】

[関数]ブロックの歯車アイコンを選択して機能設定を開き、次の操作を行います。



1) [入力名]ブロックを3つ入力領域にドラッグして、それぞれに周波数1、周波数2、周波数3 という名前を付けます。



2) [関数]ブロックの「枠」を右クリックもしくは長押しすると開くテキストボックスから['周波数1を取得'を作成します。]を選択すると個別のブロックが作成されます。

	② 및数 makeNoise 引数:	周波数1,周波数2,周波数3
国油粉1-	set (//control.Node スピーカー	(0-20000) frequency to 100 Hz
	sleep for 200 ms	
周波数2 ▼	複製	(0-20000) T frequency to 500 Hz
周波数3 🔹	コメントを削除	
	ブロックを折りたたむ	(0-20000) T frequency to 1000 Hz
	ブロックを無効にする	
	15個のブロックを削除	on: false
	'makeNoise' を作成	
	'周波数1 を取得' を作成します。	
	'周波数2 を取得' を作成します。	
	'周波数3 を取得' を作成します。	

3) 作成した[周波数 1][周波数 2][周波数 3]のブロックを各周波数の数値の部分にはめます。

😋 ? 関数 [makeNoise] 引数:周波数1,周波数2,周波数3		
set [//control.Node スピーカー (0-20000) *) frequency to () 周	波数 1	• Hz
sleep for 200 ms		
set //control.Node スピーカー (0-20000) 🔹 frequency to 🔰 盾	波数 2	• Hz
sleep for 200 ms	• •	
set //control.Node スピーカー (0-20000) 🔹 frequency to し 盾	波数 3	• Hz
sleep for 200 ms		
turn (//control.Node スピーカー 、 on: (false 、		

4) [繰り返し]ブロック内に入れた[makeNoise]関数ブロックの形状が変わり, 作成した 周波数 1~3 が追加されます。



5) 引数に[数学]カテゴリから[123]数値ブロックを引き出し取り付けます。

繰り	返す:続ける条件・)(1	rue 🔹											
実行	makeNoise 引数:												
	周波数1	100											
	周波数2	500											
	周波数3	1000											
								-					
	関数 makeNoise 引	数: 周波	数1,	周波	数2,	周波	数3	3					
s	et //control.Node スピース	カー (0-20	0000)	• fre	eque	ncy	to ()	周	波数	<u>x</u> 1	•	Hz	2
s	leep for 200 ms 🔹				•	•	-			-			
s	et //control.Node スピー:	カー (0-20	0000)	• fro	eque	ncy	to ()	周	波数	2	۲.	Hz	2
s	leep for 200 ms				•	•			•	•			
s	et //control.Node スピー:	カー (0-20	0000)	• fro	eque	ncy	to ()	周	波数	X 3	¥.	Hz	2
s	leep for 200 ms												
tı	urn [//control.Node スピー	カー・	on: 🌔	false	e v)								

関数入力により[makeNoise]関数の周波数の値を[繰り返し]ブロック内で直接変更できるようになります。

【[関数]に対するコメント】

[?]アイコンを押すとコメントバブルが開きます。

į	この関数の説明		-	•			· ·	•	•	-	-			
E	2 🕜 関数 makeNois	se) 51	数:	周波	数1	,周波	皮数 2	, 周)	皮数	3	-			
	set //control.Node 2	スピー	カー	(0-20	0000) •	freque	ency		盾]波数	(1)	2	Hz
	sleep for 200	ms 🔹				-				•				
	set (//control.Node 2	スピー	カー	(0-20	0000) •	freque	ency		盾]波数	(2)		Hz

作成した[makeNoise]関数について簡単な説明を記入できます。コメントは、他のユーザと関数を 共有する場合特に便利です。

3.6. 事前実験

- 3.6.1. グリーンハウスセンサのセットアップ
 - 1) グリーンハウスセンサをコントロールノードのセンサポートに接続します(図5①)。
 - 2) パワーアウトプットモジュールをコントロールノードのポートAに接続します(図52)。
 - 3) 土壌水分プローブをグリーンハウスセンサの[MOISTURE PROBE]ポートに接続します。グ リーンハウスセンサの裏面に描いてある2重矢印さとプローブのプラグの2重矢印が一致し ていることを確認してください(図53)。



図5. コントロールノードとグリーンハウスセンサのセットアップ

- 4) プローブのケーブルをフラット1 穴ストッパーに挿入し Eco チャンバーの蓋の穴にプローブ を通して図6に示すように穴にストッパーを取り付けます。
- 5) 蓋とチャンバーの横の穴を合わせて Eco チャンバーを閉じます。
- 6) プローブケーブルの長さを調整し、プローブがチャンバー内で直立し、図6のように底面に ちょうど触れるようにします。



図6. プローブのセットアップ

3.6.2. ソフトウェアのセットアップ

- 1) SPARKvue で新しい実験を開始します。コントロールノードをデバイスに接続します。
- 2) [センサデータ]を選択し, [センサデータ設定]を開きます。デフォルトではグリーンハウスセ ンサの[温度]にチェックが入っていますが, [温度]のチェックを外して[%VWC(ローム)]に チェックを入れます。

✓「VWC」は土壌水分率の尺度である体積含水率の略で「ローム」は土壌の一種です。使用 する土が粘土か砂かわかっている場合は、「ローム」ではなく「粘土」または「砂」を選択 します。

3) コントロールノードのオンボードセンサを無効にします。[//control.Node オンボードセン サ]の右にあるスライダをオンからオフに変更します。

//control.Node オンボードセンサ

- 4) テンプレートから[デジタル表示とグラフ表示]を選択し、データ表示画面を開きます。
- 3.6.3. リザーバーのセットアップ
 - 1) USB ポンプをパワーアウトプットモジュールの CH1 に接続し ます。
 - 2) 太めの透明チューブを USB ポンプのノズルに取り付けます。
 - 3) リザーバーに水道水を半分入れます。
 - 4) 吸盤を利用して USB ポンプをリザーバーの底に固定し,図7の ようにリザーバーを水で満たします。
 - 5) USB ポンプのケーブルとチューブを容器の外に出します。容器 に蓋がない場合は、厚紙やアルミホイルで蓋を作ります。



図7. リザーバー

✓ ポンプは常に完全に水面の下にあるように十分な水を入れておいてください。
 藻類の成長を防ぐために容器をアルミホイルで包む場合も、水位を監視するために一部
 は包まないようにしましょう。

- 6) 給水システムを計画するときは、図8を参照してください。 ドロップヘッドは1つで良いですか?それとも2つにしますか?計画が完了したら次に進みます(図8では2つ使用しています)。
- 7) 青色のチューブを少なくとも5cm以上の長さで切ります(図8の①から2に取り付ける分)。 図8のコネクタを使用して、一端をUSBポンプに繋がった太めの透明チューブと接続し、 もう一端を穴付ゴム栓大の幅広の方に接続します(図802)。
- 8) 切った残りの青色チューブにコネクタを取り付けて、穴付ゴム栓大の幅が狭い方に接続しま す(図83)。
- 9) 青色チューブを切断し、ドロップヘッドを追加します。ドロップヘッドはT型分岐コネクタ →青色チューブ→ドロップヘッドと接続しますが、このとき、間の青色チューブは 5cm 以 上の長さにします(図8④)。
- 10) チューブはドロップヘッドを除き密封されていることを確認します。1つのドロップヘッドを開いたままにし、追加のドロップヘッドがあれば、ゴムバンドまたはジップタイで一時的に閉じます(図8**5**)。



図8. チューブ、コネクタ、およびドロップヘッドのセットアップ

3.6.4. 給水の準備

- パート1:水量データの収集
 - 1) 開いたドロップヘッドを空のカップまたはビーカーにセットします。
 - プログラムツールに既存のプログラムがある場合は削除または無効にしてから、【プログラムを停止するまでポンプを実行するプログラム】を作成します。
 [繰り返し]カテゴリ、[論理]カテゴリ、[機器]カテゴリから以下のブロックを引き出し組み立てます。



- すべての気泡がチューブから除去されるまでポンプを作動させます。
 気泡が無くなったらドロップヘッドを指で閉じ、プログラムを停止します。
 ※指示があるまでドロップヘッドから指を離さないでください。
- ✓ この時点でチューブに空気を入れないでください。ドロップヘッドは気泡を避けるため、 チューブの他の部分よりも高い位置に置いてください。ポンプを連続運転して閉じ込め られた空気を取り除きます。
- 4) ポンプを正確に10秒間実行するようにコードを修正します。

繰り返	፯す:続ける条件・】	true	e •)										÷				÷
実行	set power output for	//co	ntrol	Node	•) port	Α,	CH	1 🔻	using	U	SB 🔹	on	: 6	tru	e v	
	sleep for 10 s	V	• •	•	•	•	•	•	•	• •	•	•	•	•	•	•	•

- 5) メスシリンダーに滴下するようにドロップヘッドを配置します。指を離してプログラムを実行し、プログラムが停止したら再びドロップヘッドを指で閉じます。
- 6) 目の高さでメスシリンダーの体積を読み取り, 10 秒間で1つのドロップヘッドから流出した水量(mL)を表1に記録します。

実験#	10 秒間の水量 (mL)
1	
2	
3	
平均	
ドロップヘッドの数の分乗算した 総平均水量	
給水速度(mL/s)	

表1.水供給データ

表2. サンプルデータ

実験#	10 秒間の水量 (mL)
1	5.2
2	5.2
3	5.3
平均	5.23
総平均水量(ドロップヘッド2個分)	10.46
給水速度(mL/s)	1.05

- 7) メスシリンダーの水をリザーバーに戻し、メスシリンダーから余分な水を軽く振ります。
- 8) 手順5) ~7) をさらに2回繰り返します。
- ひとつのドロップヘッドによって 10 秒間に供給される水の平均体積を計算し、結果を表 1 の[平均]に記録します。
- 10)この平均値に、取り付けたドロップヘッドの数を掛けると、10秒間に供給される水の総平 均水量が得られます。結果を表1の[総平均水量]に記録します。
- 11) 平均水量を10秒で割り,給水速度(mL/s)を求めます。結果を表1の[給水速度]に記録します。
- パート2:土壌水分データによる給水管理
 - 1) 植木鉢に土を入れて固めます。土は植木鉢の縁から 1cm 下まで 入れます。
 - ドロップヘッドを留めているゴムバンドまたはジップタイを外します。
 - バインダークリップとゴムバンドを使用して、図9①のように ドロップヘッドをチャンバー内の植木鉢に固定します。このと きドロップヘッドは少し上に向けてチューブを図9②のように 植木鉢の横にカーブさせてセットし、気泡や余計な給水を避け ます。



図9. ドリップのセットアップ

- 4) すべての空気・気泡がチューブから除去されるまでポンプを起動させ,除去後は直ちにポン プを停止します。
- 5) ドロップヘッドから1cm以内のできるだけ深い土壌中に土壌水分プローブを直立させます。 土壌水分プローブの両側の土を圧縮して、土との接触を最大にします。
- 6) チャンバーの蓋を閉じ、必要に応じてチャンバー内の土壌水分プローブのコード長さを調整 します。チューブストッパーはチャンバーの側面の穴に配置します。
- 7) ソフトウェアに戻ります。既存のコードを削除または無効にし、プログラムアイコンを切り替えてデータ表示画面に戻ります。
- 8)新しいページを作成します。レイアウトは画面 1/3 領域で上下 2 段, 2/3 領域で 1 段 します。
- 9) 1/3 領域の上の段を[数字表示] 1.23, 2/3 領域を[折れ線グラフ表示] 区にします。



1 0) [測定を選択]を押して[数字表示]を[%VWC(ローム)] (または植木鉢に入れた土壌タイプ) に設定し、グラフ表示の Y 軸も[%VWC(ローム)]にします。

ここでは、下の段のディスプレイは割り当てしません。



11) データ収集を開始します。土壌水分レベル(%VWC(ローム))が安定したら、データ収集を 停止し、安定した値を初期土壌水分値として記録します。

【回答例】 %VWC(ローム) = 14%

- プログラミングに戻ります。土壌水分量を上回るまでポンプを1秒稼働→5秒停止を繰り 返し、土壌水分量を上回ったときにループを抜け出してポンプを停止させるプログラムを組 みます。
- [関数]ブロックは例えば[ウォータープラント]と名付け,引数として[土壤水分量]を設定します。

◎ ? 関数 ウォータープラント 引数:土壌水分量

② [関数]カテゴリから

ウォータープラント 引数: 土壌水分量 ●

ブロックを引き出し, [土壌水分量]の値を設定します。

[土壌水分量]の初期入力値は手順11)でメモした初期土壌水分値より1%高く設定します。

③ [関数]ブロックの縁を右クリックまたはタップ長押しをして, ['土壌水分量 を取得']を作成 します。

[繰り返し], [論理], [数学], [機器], [時間]カテゴリから必要なブロックを引き出して組み合わせます。

%VWC(ローム)の小数点を丸めた値が土壌水分量を上回るまでポンプを1秒稼働→5秒停止 を繰り返し,土壌水分量を上回ったときにループを抜け出してポンプを停止させるプログラ ムです。 ✓ 土壌水分プローブは小数点を含む値を取得するため、[四捨五入]ブロックを使用して、読み取り値をデータ表示と直接比較できる整数にします。

27	関数 <mark>ウォーターブラント</mark> 引数:土壌水分量
繰り逃	をす:終わる条件 ▼
実行	set power output for //control.Node > port A, CH1 > using USB > on: true >
	sleep for (1 S
	set power output for //control.Node > port A, CH1 > using USB > on: false >
	sleep for 15 S
set po	ower output for <i>[//control.Node</i>) port A, CH1) using USB on: false
	· · · · · · · · · · · · · · · · · · ·

④ さらに、[プログラム出力]カテゴリから[テキスト出力を作成...]を押し、テキスト出力の名前を[アラート]と入力します。

☆ 論理	テキスト出力を作成	新しいテキスト出力の名前
♀ 繰り返し	数値出力を作成	新しいテキスト出力の名前 アラート
☑ 数学		
₩ テキスト		
■ リスト		
\$ 変数		
Σ 関数		
6.4 ## 32		
ペノロシラム西川		キャンセルOK

⑤ [テキスト出力]ブロックが作成されるので,["abc"]に適当なテキストを入力します。
 in text output アラート・ enter (abc ")

ここでは土壌に水が足りている状態かどうかを表示したいので,給水中と給水停止後にそれ ぞれ追加し以下のプログラムを作成します。

• •	関数 ウォータープラント 引数:土壌水分量
繰	D返す:終わる条件▼ (四捨五入▼) value of %VWC(ローム)▼ (%▼) ≥▼ ↓ 土壌水分量▼
実行	set power output for <i>//control.Node</i> port A, CH1 v using USB v on:
	in text output アラート v enter 0 44 給水中 ジ
	sleep for 1 SI
	set power output for //control.Node > port A, CH1 > using USB > on: false >
	sleep for 15 ST
set	power output for //control.Node port (A, CH1) using USB) on: (false)
in t	ext output フラート 🔹 enter 肯 🧐 十分な水分量 🥬 🔹 👘 👘 👘 👘 👘 👘 👘 👘

13) データ表示に戻ります。

まだ設定していないレイアウト部分を[数字表示]^{1.23}にします。[測定を選択]を押して[ユー ザ入力]タブの[アラート]にしてテキスト出力します。



14) ループが期待通りに動作するまで、必要に応じてプログラムをテスト・修正します。

- ✓ 土が完全に水分飽和し、余分な水がチャンバー内にたまることがあります。水がたまり始めたら、チャンバーから余分な水を捨て、必要に応じて新しい土を植木鉢に補充してください。
- 15) 所定のスペースに作業ループをスケッチします。(ファンクション自体の内容を再現する 必要はありません。)



16) 関数内の各時間遅延の目的は何でしょうか。これらの時間ブロックが取り除かれたら、何 が起こるのでしょうか?

【回答例】

時間を遅らせることで水が土壌にゆっくりと滴り落ちるので、土壌水分プローブが水を追加 する前に正確に値を読み取ることができます。時間遅延が無ければ、すぐに水を与えすぎてし まう可能性があります。これは、センサが追加された水をすぐに検知できるほど水は土の中を 素早く移動できないからです。 パート3:給水状況の把握

1) 1秒の時間遅延とポンプをオフにする電源出力ブロックの間に,次のブロックを関数に追加 します。

関数ブロックの引数に[平均水量(mL)]を設定します。表1の給水速度を関数ブロックの[平 均水量(mL)]の値として使用します。(下記例では表2のサンプルデータを使用しています。) [変数]カテゴリから[変数の作成...]を押し, [カウント], [累積水量(mL)]を作成します。

Blockly Lua		
r 論理	変数の作成	繰り返す:続ける条件 ·) true ·
◎ 繰り返し	【累積水量(mL) ▼ に 🐂 をセット	実行 ウォータープラント 引数: 土壌水分量 (15)
☑ 数学	【累積水量(mL) ▼ を 1 増やす	平均水量(mL) 1.05
゙ テキスト		
= リスト	カウント・	繰り返す:終わる条件・ (四捨五入・)
\$ 変数	● 土壤水分量 ▼	美行 set power output for //control.Node p
Σ 関数	平均水量(mL) ▼	sleep for (1 S
秋 横架	■ 【累積水量(mL) * 】	set power output for (//control.Node) p

2) プログラムの最初のブロックとして,繰り返しループの外側に次のコードを追加し,[カウント]を0から開始します。

カウン	・ト・ に (🚺 をセット
繰り返	zす:続ける条件 🔹 true 🔹
実行	ウォータープラント 引数:
	土壤水分量 15
	平均水量(mL) 1.05

3) [数値表示]を変更して、プログラムのテスト時に[繰り返し数]および[給水量]の数値出力の値 を監視します。

[プログラム出力]カテゴリから[数値出力を作成...]を押し,[繰り返し数]と[給水量]を作成し ます。作成すると[プログラム出力]カテゴリにブロックが作成されるので,

in number output 保り返し数 🔹 enter 📋 🚺

[繰り返し数]の enter の値の部分に[変数]カテゴリ内の[カウント]を入れます。

同様に, [給水量]の enter に[累積水量(mL)]を入力します。[給水量]は[繰り返し数]のプルダ ウンで選択できます。

[数学]カテゴリから足し算のブロックを引き出し,



始めの数値部分に[変数]カテゴリ内の[カウント],間の記号をプルダウンから[x],2つ目の 数値部分に[平均水量(mL)]を入れます。

作成した数式ブロックを[累積水量(mL)]にセットします。



- 4) 必要に応じて値を変更し、土壌が水分を必要とする場合と必要としない場合のプログラムの 動作を確認してから、次の質問に答えます。
 - a. 変数[カウント]はどんな役割ですか?

【回答例】

この変数はポンプがオンになった回数を記録します。プログラムが開始すると[カ ウント]は0から始まり、ポンプがオンになるたびに1ずつ増加します。

b. [累積水量]計算ではどのような情報が提供されますか?

【回答例】 この計算により,プログラムの実行が開始されてから供給されたリアルタイムの 累積水量が得られます。ポンプがオンになった回数を取得し,ポンプが1秒間オ ンになるたびに供給される平均水量を乗算します。

【プログラム例】
カウント・ に (0) をセット
繰り返す:続ける条件・ Ctrue・ Ct
美打 土壌水分読取値 τ に $($ value of %VWC (ローム) τ % τ をセット
リオーダーフラント 51級: 土壌水分量 ↓ 15 平均水量(mL) ↓ 1.05
😢 😨 関数 ウォータープラント 引数:土壤水分量,平均水量(mL)
繰り返す:終わる条件 ▼
寒行 set power output for //control.Node v port A, CH1 v using USB v on: U true v
sleep for 1 s
in number output (採り返し数 enter) フリント・ 累積水量(mL) 、 に (カウント × ・) 平均水量(mL) ・ をセット
in number output (給水量 venter (累積水量(mL) v
set power output for //control.Node v port A, CH1 v using USB v on: false v
set power output for <i>[//control.Node •</i> port A, CH1 • using USB • on: [false •]
in text output アラート enter 。 《 充分な水分量 "

-40-

3.7. 実験

次のことを実現するために,繰り返しループ内で入力を持つ関数を使用した関連データを表示するプ ログラムを作成します。

- 目標土壌水分レベルに達するまで、土にゆっくりと水を加えます。
- コントロールノードのスピーカーを使用して、リザーバーの水が残り約 1/3 となったときに音を 鳴らし、アラートテキスト出力にメッセージを表示します。
- ・ 土壌水分が目標レベルより大幅に高い場合に新しいアラート(視覚的または聴覚的, あるいはそ の両方)を追加します。
- 可能な場合は、センサの直接測定値の代わりに変数を使用して、プログラムをできるだけ高速で 実行します。

後で参照できるように,作業内容を SPARKvue に保存します。別の用紙に関数の内容を含むプログ ラムをスクリーンショットして印刷またはスケッチし,その動作方法とデータ表示の選択肢を説明し ます。



があり,内1つ[リザーバーアラート]関数は,メインの[ウォータープラント]関数に入れ子に なっています。

[ウォータープラント]関数を変更して、リザーバーの水量の下限しきい値を[最大水量(mL)]と し、[累積水量]が[最大水量(mL)]に達したときに[リザーバーアラート]関数で警報とテキスト の表示を行います。リザーバーの容量は約 1.5 L なので、約 1,000 mL の給水が完了すると警 報が鳴ります。

[水分過剰アラート]関数は、土壌水分が高すぎるとユーザに警告します。この関数には、アラ ートをトリガーする[目標水分量%]をユーザが決めて数値入力する必要があります。この例で は[目標水分量%]を17%としています。

センサから直接測定値を呼び出す代わりに,変数[土壌水分読取値]を作成します。

このプログラムはディスプレイには給水量とテキスト警告,土壌水分のグラフを表示しま す。データを素早く確認でき,アラームが鳴った場合に何が起こったのかを把握できるよう に,画面にできるだけ多くの情報を表示させることができます。

3.8. 追加実験

リザーバーに水が 1/4 しか満たされていないときにポンプを無効にするようにプログラムを変更して ください。



-42 -

実験4:グリーンハウス内の水の移動を最適化する

ー度に一つや二つのことを覚えるのは簡単ですが、15以上のことを覚えやすくするにはどうすればいいのでしょうか?例えば、人気の高い観葉植物の中には、土壌の水分、光、湿度、温度の条件がよくわからないものもあります。その情報をリストに書き留めておくと、記憶に頼るのではなく、その情報を調べることができます。

コンピュータプログラムは、メモリに余分な情報を格納する代わりに、必要に応じて格納された情報 に簡単にアクセスするために「リスト」を使用します。人間と同じように、コンピュータプログラム も使用可能なメモリが多いほど効率的に動作します。

この実験では、水を使用してグリーンハウスの条件を変更する方法を学習し、リストを使用してグリ ーンハウスの条件に適した植物を決定するプログラムを作成します。

4.1. 目標

・温度と湿度のデータを使用して、プログラムの設計を最適化します。

・リストまたは配列を使用して、プログラム内の計算タスクを簡略化します。

4.2. 使用機器

・SPARKvue または PASCO Capstone	・コントロールノード
・グローライト	・グリーンハウスセンサ
(パワーアウトプットケーブルと USB 電源アダプタ付き)	
・グリーンハウスセンサモジュール	・土壌水分プローブ
(センサモジュールケーブル付き)	
・グリーンハウスアクセサリキット	・USB ファン
・パワーアウトプットモジュール	・Eco チャンバー
(パワーアウトケーブル付き)	
・水道水が半分程度入った小さなカップまたは	・土が入った植木鉢
ビーカー	
・5×5cm のスポンジ	・ペットボトルのキャップ3つ
・ジップ付袋(M サイズ)	

4.3. 安全上の注意

センサボックス,電源プラグ,露出した電子基板に水を近づけないでください。
グローライトは非常に明るい LED を使用しています。操作中に光を直接見ないで ください。



4.4. ブロックの役割

【リストブロック】

[リスト]ブロックは、複数の関連情報を処理する場合に便利です。リストの例を表示するには、次の手順を実行します。

- 1) コントロールノードを接続します。任意のテンプレートを選択してデータ表示を作成し、プロ グラムツールを開きます。
- 2) 右上隅にあるコードライブラリを開きます。
- 3) カテゴリを[PASCObot]から[温室]に変更し、[太陽光モデル]を開きます。
- 4) 折りたたまれた

ブロックを右クリックまたはタップ長押してブロックを展開します。

1	関数	mo	delS	Sunl	iaht	// " F	
	12382	ine		John	gin		複製
						•	ブロックを展開する
•			•			•	ブロックを無効にする
	٠	*	•	*	*	•	86個のブロックを削除
	٠	٠	•	٠	*	•	'modelSunlight' を作成

この[modelSunlight] 関数の目的については、コメントの吹き出しと注記を参照してください。 プログラムがどのように構成されているかを調べ、どのように動作するかを予測してから読み進 めましょう。

[growLight]変数は、リストで構成されたリストです。

リスト内のリストを「サブリスト」、個々のサブリスト値は「項目」と呼ばれます。

[modelSunlight] 関数では、サブリストアイテムは、秒単位の継続時間、赤色 LED 強度、およ び青色 LED 強度を表す編集可能な入力です。必要に応じてカラーミックスやグローライトの持 続時間を変更できます。



その下のループは、コントロールノードがグローライトに送信するための命令です。

וגע	「growLight 可 の各項目 sublistitem 」について
実行	timeInSec ・ に リスト (sublistitem ・) 取得 ・ (1) をセット
	red · C リスト (sublistitem · 取得 · # · (2) をセット
	blue・に リスト (sublistitem ・ 取得 ・ # ・ (3) をセット

[リストループ]ブロックの仕組みを詳しく見てみましょう。

リストループは、一度に1つのサブリストを上から下へ順番に読み取ります。ループが最終リストに到達すると、[リストループ]ブロックが[繰り返し]ブロック内に保管されていない限り、プロ グラムは終了します。

[timeInSec][red][blue]変数はループ内に含まれ,項目1[継続時間],項目2[赤色 LED 強度],項 目3[青色 LED 強度]の値から[sublistItem]というリストを作成します。

ー般的なプログラマーは、リスト項目の位置を識別するために「インデックス」という用語を使用します。ブロックコードでは、[継続時間]のインデックスは 1, [赤色 LED 強度]のインデックスは 2, [青色 LED 強度]のインデックスは 3 です。そのため、リストブロックの最後に 1, 2, 3 という数値を与えます。

4.5. 事前実験

- パート1:センサのセットアップ(1日目)
 - 1) コントロールノードの電源がオフになっていることを確認しグリーンハウスセンサのプラ

グをコントロールノードの図10の Sensor ポートに接続します。

- 2) パワーアウトプットモジュールをポートAに接続します(図10**2**)。
- 3) USB ファンをパワーアウトプットモジュールの CH1 に接続します(図108)。
- 4) センサモジュールを Eco チャンバーの蓋に図104のようにセットします。
- 5) 土壌水分プローブをグリーンハウスセンサに接続します。プラグの二重矢印≉とセンサの背 面の二重矢印≉を合わせます(図10⑤)。今はプローブを脇に置いておきましょう。
- 6) センサモジュールをグリーンハウスセンサに接続します。プラグの矢印は、センサの背面に 合わせます(図106)。
- 7)3つのペットボトルの蓋を Eco チャンバーの底の中央に置きます。3つのキャップの上に実験3で作成したドロップヘッド+チューブ付きの植木鉢をセットします。気泡対策として植木鉢の側面を通しチューブがU字に曲がるように調整してください。
- 8) Eco チャンバーの側面の穴に蓋を合わせ,図10中央のように USB ファンを側面の穴に固 定して Eco チャンバーを閉じます。



図 10. コントロールノード, グリーンハウスセンサ, USB ファンのセットアップ

- 9) SPARKvue で新しい実験を開始します。
- 10) コントロールノードの電源を入れ、表示デバイスと接続します。[センサデータ設定] 画面 で次の手順を実行します。
 - i) [温室監視センサ]で, [温度]と[相対湿度]の測定値のみ選択↓します。
 - ii) [//control.Node オンボードセンサ]のスライダをオン ●からオフ ●に変更して無効 にします。
 - iii) [デジタル表示とグラフ表示] テンプレートを選択します。



- 11) データ表示ページが開いたら、次の手順で1分後にデータの収集を自動的に停止するよう に SPARKvue を設定します。
 - i) サンプリングオプションメニューの

 を開き, [停止条件]を開きます。
 - ii) [条件タイプ:]を[時間ベース]にして[記録時間:]を1に設定します。単位を[min]に設定 して, OKを選択します。
- パート2:湿度と温度の調査
 - チャンバー内外の状態が安定した時点(コントロールされた状態)から、1分間の温度と相対湿度の測定を開始します。停止条件を設定しているためデータ収集は1分後に自動的に停止します。
 - データ収集が停止したら、グラフの凡例を確認します。温度の実験マークをクリックして、 赤いボックスを温度データに移動します。



- 3) グラフツールメニュービを開きます。グラフに座標ボックスを追加するには,座標ツール を選択します。
- 4) 座標ボックスを最初の温度データ点に移動し、開始時の温度を表3に記録します。
- 5) 座標ボックスを最後の温度データ点に移動し、終了時の温度を表3に記録します。
- 6) 温度変化を計算し表3に記録します。
- *4)~6)は表3の「コントロール」の欄に記録します。

条件	開始温度	終了温度	温度変化	開始相対湿度	終了相対湿度	相対湿度変化
	(°C)	(°C)	(°C)	(%)	(%)	(%)
コントロール						
ファン						
ファン +						
スポンジ						
ファン + 水						

表3. 異なる条件下での1分間の温度および相対湿度性能

- 7)凡例の相対湿度の実験マークをクリックして、赤いボックスを相対湿度データに移動します。 温度と同様に座標ツールを使用して、表3の相対湿度の開始値、終了値、変化値を記入します。
- プログラムツールアイコンをクリックして、USB ファンを継続的に実行する簡単なプログ ラムを作成します。
- 9)作成後、再びプログラムツールアイコンを押すと、データ表示に戻ります。次の1分間のデ ータを収集して、ファンに対する温度の変化を観察し、その結果を表3の「ファン」の欄に 記入します。データ収集の間、ファンは継続的に動作する必要があります。必要に応じてコ ードを修正します。

10) スポンジを水道水に浸し、余分な水をしぼって、垂れないように湿らせます。スポンジを ファンの1cm後ろに持ちます(図11を参照)。



図11. ファンから1 cm 後ろで持ったスポンジ

- 11) データ収集を開始します。ファンを作動させ、スポンジはデータ収集の1分間ずっと同じ 位置に置かなければなりません。データ収集が終了したら、スポンジを脇に置き、結果を表 3の「ファン+スポンジ」の欄に記録します。
- 12) グリーンハウスを開きます。チャンバーに底から 1cm くらいまで水道水を入れます。
- 13) データ収集を開始します。データ収集の間,ファンは作動しているはずです。データ収集 が停止したら,結果を表3の「ファン+水」の欄に記録します。
- 14) オプション:安全なファン/湿度シナリオを考えられる限り追加でテストし、表3の続きを 別の用紙に記入します。
- 1 5) SPARKvue のサンプリングオプションメニュー[☉]に戻ります。自動停止条件で,条件タ イプを[なし]に変更し, OK を選択します。
- 16)表3の結果を確認します。各条件についての結論を共有します。この結論は、グリーンハウスに植物を入れる準備ができたときに、土壌水分の管理とグリーンハウスの最適な気温のバランスをとるのに役立ちます。
 - a. グリーンハウスの内外で何も変化させなかった場合(対照条件),温度と相対湿度はどの ようになりましたか?

【回答例】

温度と相対湿度は安定している。システムにもその周囲にも変更は加えられてい ないため、どちらの測定値も変化はない。

b. ファンがグリーンハウス内に空気を吹き込むと,温度と相対湿度はどのようになりましたか?

【回答例】

ファンをオンにすると相対湿度は低下したが,温度は安定したままだった。ファ ンからの空気循環により,乾燥した空気がチャンバー内に移動した。 c. ファンを通してグリーンハウス内に水分を多く含んだ空気を吹き込むと,温度や相対湿 度はどのようになりましたか?

【回答例】

チャンバー内への水蒸気の移動により相対湿度は上昇したが、温度は安定していた。

d. ファンを回しながらチャンバーに滞留水を加えると,温度と相対湿度はどのようになり ましたか?

【回答例】

相対湿度は上昇し,温度はわずかに低下した。水はチャンバーの外側ではなく内側 で蒸発しているので,水がエネルギーを吸収して液体から気体に変化するときにチ ャンバー内の温度が低下する。蒸発した水は、チャンバーを出るときに熱を排出す る。

パート3:プログラムでリストを使用する(2日目)

- 1) 蓋の空いている穴にフラット1穴ストッパーをセットします。
- 2) グローライトのポスト(支柱)をストッパーに挿して固定します。図12に示すように、光 センサが中央に来るようにセンサモジュールとグローライトを調整します。



図12. 光センサとグローライトの位置

- 3) データ表示画面から Bluetooth アイコンを押して機器接続メニューを開きます。
- 4) ×を押して、コントロールノードとの通信を切断します。メニューを開いたままにします。
- 5) コントロールノードの電源をオフにし、グローライトをパワーアウトプットポート B に接 続します。
- 6) グローライトに電源用の USB ケーブルを接続し、電源アダプタに差し込みます。
- 7) コントロールノードの電源をオンに、再び表示デバイスと接続します。
- 8) 次のことを実行するコードを記述します。
- ・ グローライトをオンにして、実際の植物で長期的に使用する赤と青の強度に設定します。
- 進行中のプログラムなどのステータスを示すために[デジタル表示]で表示する[ステータス]
 テキスト出力を作成します。
- グローライトで発生する熱を考慮するため、プログラムを進める前にライトのウォームアッ

プタイム用の時間ブロックを追加します。赤色 LED と青色 LED の強度が低い場合は最低 1分間,高い場合は最高5分間にします。

9) 温度,湿度,照度の[変数]ブロックを追加作成します。

注 記 センサの測定値名と同じ名前は使用しないでください。

10) [論理]ブロックを使用して、温度、湿度、および照度の変数を、表4に示すセンサの測定 範囲と一致させます。開始時に役立つように、不完全なロジックブロックを示します。例で は温度の変数を[temp]としています。



表4. 室内温度, 湿度および光条件の基準

測定項目	高い	中くらい	低い
温度			
湿度			
照度			
測定値:			

表5. 室内温度, 湿度および光条件の基準サンプル

測定項目	高い	中くらい	低い	
温度	23℃より大きい	17 ~ 23℃	17℃未満	
湿度	55%より大きい	$42 \sim 55\%$	42%未満	
照度	42%より大きい	28~42%	28%未満	
測定値:				

- 11)温度,湿度,および照度の変数のテキスト出力を作成して,測定値に応じて[高い],[中くらい],または[低い]の結果を表示します。これで,4つの固有のテキスト出力ができます。
- 12) SPARKvue のデータ表示画面に戻ります。新しいページ を追加し、4 分割されたレイ アウト を選択します。
- 13) すべてのパネルを[デジタル表示] 1.23に設定します。
- 14) [測定を選択]を押します。開いたパネルで, [センサ]タブから[ユーザ入力]タブに切り替え ます。



- 15) ユーザ入力データ測定リストから,作成した4つのテキスト出力のいずれかを選択します。
- 16)各表示に異なるテキスト出力を割り当て、コードをテストします。コードが想定どおりに 動作するまで、次の手順に進まないでください。
- 17)表4の最終行に温度,湿度および明るさのレベル(高,中,低)を記録します。
- プログラムツールに戻り、コードに追加します。
 [プラントリスト]という変数に練習用の全テキスト配列を作成します。
 次の点を考慮してください。
 - A. サブリストの最初の列は,表の見出しのように機能し,情報の整理に役立ちます。 以降の列のサブリストは,温度,湿度,および照度の測定結果の可能な組み合わせです。
 - B. 最後のサブリストを完了します。表4の結果が指定された組み合わせのいずれかに一致 する場合は、最後のサブリストを[中くらい]で開始して新しい組み合わせを取得します。 それ以外の場合は、結果を使用して最後のサブリストを完成させます。



【プログラム 作成途中の例】	
フラントリスト・ に 🔋 以下を使ってリストを作成:	以下を使ってリストを作成: (" 温度 " 「 " 温度 " 「 " 明るさ " 「 " ステータス " をセット
	◎ 以下を使ってリストを作成: ("高い") ("高い") ("高い") ("高い") ("高い") ("ステータスA")
	◎ 以下を使ってリストを作成: ("高い") ("高い") ("中") ("ステータスB"
	◎ 以下を使ってリストを作成: ("高い") ("高い") ("低い") ("ステータスC")
	◎ 以下を使ってリストを作成: 4 4 4 4 (ステータスD)

19) [繰り返し]カテゴリの[リストループ]ブロック



に作成した[プラントリスト]を追加し、以下のブロックを完成させます。

① [論理]カテゴリから ゆうつう と ゆうつう を使用して [リスト]カテゴリの

リスト 🛌 💵 🏭 🗰 🖡

ブロックでリストの各項目番号と変数名を一致させます。

- ② [temp][humi][light]を対応させたら、残りの[ステータス]も設定します。リストの最後の項目を指定する場合は、[リスト取得]ブロックの[#]をプルダウンして[最後]を選んで指定します。
- ③ [プログラムの出力]カテゴリのテキスト表示ブロックを追加して[ステータス]変数を表示します。

リスト (IT 取得す # 2) = 1 (humi * がつ *) リスト (IT 取得す # * 3) = 1 (humi * 新 ステータス・ に リスト (・・・) 取得 ・ 最後 ・ をセット utput ステータス・ enter

20) データ表示に切り替えて、コードをテストします。作業コードは、ステータス数字表示に ステータス A, B, C, または D を表示します。必要に応じてコードを修正します。この特 定のコードで[リストループ]がどのように機能するかを説明し、コードの[リスト取得]ブロ ックの[最後]部分を設定する別の方法を提案します。



- 21) 温度,湿度,および明るさの組み合わせの総数を決定します。別の用紙に, P.57の付録 A を参照して,考えられる温度,湿度,明るさの組み合わせごとに1つの植物を特定する表を 作成します。
- 22) 植物の水やりの設定の列を追加します。10 種類の植物をすべて使用する必要があり、多く の植物は複数回使用できます。

【プログラム例】	
set grow light for //control Node + port B + to brightness R 5 B 5	
in text output ステータス enter II 44 進行中 ??	
sleep for (60 S・) の もし (value of 温度・ *C・ >・ 23	
実行 temp に (" 扈い " をセット	
そうでなくもし value of 温度 * 'C * >** 17	
そう C/aty/1 (は temp · に 0 " (低い) " をセット	
© もし value of 相対温度 · % · > · 55 実行 burging に (● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●	
そうでなくもし (value of 相対温度 % * > 42)	
実行 [humi]に ("中くらい"をセット	
そうでなければ [humi]に ("(まい)" をセット	
○ もし (value of 照度・ %・ ▶・ 42)	
実行 に「「「「「」」」 をセット	
マンパンマン value of 脱皮・%・ ・ 28 実行 Light・に "中くらい" をセット	
そうでなければ Tont に "原の" をセット	
in text output [temp] enter temp -	
in text output (humi +) enter (humi +)	
フラントリスト・に 0 以下を使ってリストを作成: 0 以下を使ってリストを作成:	(4 (温度) 7 4 (温度) 7 4 (明るさ) 7 4 (植物名) 7 4 (水分要求量) 7
	((高))) (高)) (高)) (シェフレラ) ((セクト14乾いたら水をあげる)) ((高)) ((市)) ((市))) ((市)) ((市)) ((市)) ((市))) ((市)) ((市)) ((市))) ((市)) ((市)) ((市))) ((市)) ((市)) ((n)) ((n)
 ※ 人工を送ってリストを作成: 	((高い)) ((高い)) ((低い)) ((ZZブラント)) ((上力完全に認いたら水をあげる))
	(" (鳥い) " (" (中くらい) " (" (鳥い) " (" シェフレラ) " (" 主の上14乾いたら水をあげる) "
	(" 高い " (中くらい " (中くらい " (カランコエ " (丘の上半分が乾いたら水をあげる "
 回 以下を使ってリストを作成: 回 以下を使ってリストを作成: 	((高い) ((低い) ((高い) (アロエペラ) ((エカ元至に起いたら水をあける))
④ 以下を使ってリストを作成:	(" (高い) " (" (低い) " (" (中くらい) " (" (サンセベリア)" (" 土がほぼ乾いたら水をあげる "
◎ 以下を使ってリストを作成:	((底の)) ((底の)) ((低の)) (リンセベリア)) ((土がほぼむいたら水をあげる))
 ・ 以下を使ってリストを作成: ・ ・	(中くらい)) (鳥い)) (鳥い)) (ホトス)) (正か完全に起いたら水をあける)) ((中くらい)) (市くらい)) (ホトス)) (正か完全に起いたら水をあける))
 回 以下を使ってリストを作成: 	((中くらい)) ((高い)) (低い)) (ボトス)) (甘が完全に認いたら水をあげる))
◎ 以下を使ってリストを作成:	(" 中くらい】 " 「 " 中くらい " 、 " 高い " 、 " 八一ブ " 、 " 水分量を維持する "
	((中くらい)) ((中くらい)) ((中くらい)) ((ボトス)) ((土力完全に載いたら水をあげる)) ((中くらい)) ((中くらい)) ((中へらい)) ((市トス)) ((市トス-2)(時いたら水をあげる))
	("中くらい"、"信い"、"高い"、"ジェイト"、"上の主告にもいたられたものがる" ("中くらい"、"信い"、"高い"、"ジェイト"、"土の上5cmが思いたられたものがる"、
・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	(" 中くらい " 『 " 低い " 『 " 中くらい " 『 " サンセベリア " 『 " 土がままむいたら水をあげる "
	() 中くらい) () (低い) () () () () () () () () ()
	((低い) (「高い
・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	(* (低い) * (* (高い) * (* (低い) * (* 22ブラント * (* (主が完全に載いたら水をあける) *
 図 以下を使ってリストを作成: 	(低い) (中くらい) (高い) (アイビー) (主の上5cmが乾いたら水をあげる)
	** (EVU) ** (中くらVI) ** (中くらVI) ** (アイビー) ** (土の上5cmが乾いたら水をあげる) ** ** (低VI) ** (中くらVI) ** (低VI) ** (アイビー) ** (土の上5cmが乾いたら水をあげる) **
回 以下を使ってリストを作成:	() (近い) () () () () () () () () () (
・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・	" (低い) " " (低い) " " (主くらい) " " (サンセベリア " " (土がほぼむいたら水をあげる "
◎ 以下を使ってリストを作成: をセット	・ " 良い " 」 " 便い " 」 " 便い " 」 " サンセベリア " 」 " 土が良難いたら水をあげる "
リスト (フラントリスト・) の各項目 [1] について	
第行 ステータス に リスト 11 取得 最後 をセット	
intext output (ステータス・) enter () ステータス・)	

4.6. 実験

作成した植物情報表のすべての情報を格納するように配列を再設計します。 6分割されたレイアウトを使用してすべての結果を表示します。



次の点を考慮してください。

- ・ テキスト出力を追加し,要素の名前を変更します。
- ・ プログラムが成功すると、グローライトが点灯しているときのグリーンハウス内の温度、湿度、 明るさの測定値に基づいて、理想的な植物とその水やりの必要性が表示されます。

プログラムが正常に終了したら,次の手順を実行します。

- 1) 植木鉢に乾いた土を入れ、上から1cmのところまで固めます。
- 2) いずれかのデジタル表示を[%VWC (ローム)] に変更します。
- 3) 植木鉢の底に届くまで土壌水分プローブを土の中に静かに押し込み, プローブの周りの土を 圧縮します。
- 4) 乾燥した土の土壌水分レベルを測定し、その結果を記録します。
- 5) ジップ付袋に植木鉢の1/4杯分の土を入れます。
- 6)実験で使用する植物に適した土壌水分レベルに達するのに必要と思われる水よりも、少し少ない水を加えます。
- 7)袋を密封し、土と水をよく混ぜます。必要に応じて、土が目的の土壌水分レベルに達するまで手順2)~3)を繰り返します。
- 8) 袋の土を植木鉢に入れ、土の高さが植木鉢の縁から1cm下になるまで固めます。
- 9) 4)と同様に土壌水分を測定します。
 - i) 植物名,土壌水分要求量,理想的に湿った土壌の%VWC(ローム),乾燥した土壌の%VWC(ローム)を記録します。
 - ii) コードをスクリーンショットして印刷またはスケッチします。
- iii) 実験で使用する植物にとって理想的なグリーンハウスの条件を維持するために、どの温度または湿度の制御が最も効果的であるかを説明します(ファンのみ、ファンの後ろにスポンジを設置する、水を溜めた状態でファンを作動する、または自分で考えた方法)。
- 10)後で参照できるように、作業内容を SPARKvue に保存します。次回の使用のために土と 一緒に植木鉢をそのままにすべきかどうか、先生に確認してください。

4.7. 追加実験

- [数字]表示の代わりに[表]を使用して,植物の識別と給水要求量を表で表示します。
- 温度,湿度,明るさ,土壌水分の状態が特定されたプラントの理想的な範囲を超えたときに通知 するテキストおよび音声アラートを作成します。



付録A:10の小型屋内植物の特性

暖かい室内に耐える植物

- シェフレラ:中~高湿度,高照度が必要。土の上半分が乾いたら水をあげる。
- カランコエ:開花させるには多湿・多光を好む。土の上半分が乾いたら水をあげる。
- ZZ プラント (ザミオクルカス):様々な条件下で生育する丈夫な室内植物。水やりと水やりの間 は土を完全に乾燥させる。
- アロエベラ:低湿度,高照度を好む。土壌がほぼ完全に乾燥しているときに水を与える。
- サンセベリア:様々な条件下で生育する丈夫な室内植物。土壌がほぼ完全に乾燥しているときに 水を与える。

中温室を好む植物

- ポトス:中~高湿度,さまざまな光量を好みます。水やりの間は土を完全に乾燥させる。
- ハーブ:適度な湿度と明るい光を好む。土を湿らせておく。
- ZZ プラント (ザミオクルカス):様々な条件下で生育する丈夫な室内植物。水やりと水やりの間 は土を完全に乾燥させる。
- ジェイド:低湿度,中~高照度を好む。土の上1インチが乾いたら水をやる。
- サンセベリア:様々な条件下で生育する丈夫な室内植物。土壌がほぼ完全に乾燥しているときに 水を与える。

涼しい室内に耐える植物

- ゼラニウム:開花させる場合,多湿と光を好む。土の上1インチ半が乾いたら水をやる。
- ZZ プラント (ザミオクルカス):様々な条件下で生育する丈夫な室内植物。水やりと水やりの間 は土を完全に乾燥させる。
- アイビー:中~高湿度と様々な光のレベルを好む。土を湿らせておく。
- ジェイド:低湿度,中~高照度を好む。土の上1インチが乾いたら水をやる。
- サンセベリア:様々な条件下で生育する丈夫な室内植物。土壌がほぼ完全に乾燥しているときに 水を与える。

実験5:グリーンハウスの状態感知および制御システム

ー般家庭のプログラム可能なサーモスタット(設定された温度に保つ機能を持つ装置)と自動車の クルーズコントロールシステム(アクセル操作なしで車の速度を一定に保つシステム)の共通点は何 でしょうか。どちらも、プログラムの入力としてユーザが理想的な値を設定し、システムを目的値付 近で維持します。ただし、クルーズコントロールシステムは車の速度が理想値に非常に近い状態を維 持することが非常に重要であるためサーモスタットプログラムよりも複雑です。

標準的なサーモスタットプログラムは ON-OFF 制御に基づいています。気温が理想より数度低い 値まで下がると ON となりヒーターが点きます。ヒーターは温度が理想より数度高くなるまで作動 し、その後 OFF となりヒーターが切れます。実験1~4は、1 つまたは複数のセンサ入力値が出力 機器をオン/オフする必要があるかどうかを決定する ON-OFF 制御システムをプログラムしました。 比例制御 (P制御) はクルーズコントロールシステムが車速を一定に保つために使用するプログラミ ング戦略の一部に過ぎません。比例制御もセンサの入力を必要としますが、出力は目標値に近づくよ うに常に調整されます。車速の入力は綿密にモニターされ、システムはスロットルポジションなどの 出力設定を常に微調整し、理想的な車速にできるだけ近づけます。

この実験では完全に自動化されたグリーンハウス内で植物を健康な状態に保つために,新しいコード,変更されたコード,既存のコードの組み合わせと,少なくともひとつの比例制御を統合する 24 時間のプログラムを作成します。

5.1. 目標

- ・ 既存のコードを新しいプログラムに組み込みます。
- グリーンハウスを自律的に維持するために、さまざまな制御構造を組み合わせたプログラムを設 計します。

5.2. 使用機器

・SPARKvue または PASCO Capstone	・コントロールノード
・グローライト	・グリーンハウスセンサ
(パワーアウトプットケーブルと USB 電源アダプタ)	
・グリーンハウスセンサモジュール	・土壌水分プローブ
(センサモジュールケーブル付き)	
・グリーンハウスアクセサリキット	・USB ポンプ
・USB ファン	パワーアウトプットモジュール
	(パワーアウトプットケーブル付き)
・Eco チャンバー	・植木鉢
・浅めの器	・ジップ付袋
・ 氷	・ゴムバンド
・バインダークリップ	

5.3. 安全上の注意

	センサボックス, 電源プラグ, 露出した電子基板に水を近づけないでく ださい。
	グローライトは非常に明るい LED を使用しています。 操作中に光を直 接見ないでください。
	グローライトは熱くなります。操作中または長時間の実験後は冷える まで触らないでください。
注記	露出した電子基板が金属または導電性の表面に接触しないようにして ください。

5.4. 比例制御

このセクションでは,比例制御プログラミング手法の設定と使用方法について説明します。 次の手順を実行します。

- 1) コントロールノードを接続します。任意のテンプレートを選択してデータ表示を作成し、プ ログラムツールを開きます。
- 2) 右上隅にあるコードライブラリを開きます。
- 3) カテゴリを[PASCObot]から[温室]に変更し, [温度調節]を開きます。
- 4) 折りたたまれた

関数 regulateTemperature 引数: ... 4

ブロックを右クリックまたはタップ長押してブロックを展開します。

関数	(reg	julat	teTe	mpe	erati	ure 弓	数: >
							複製
							ブロックを展開する
							ブロックを無効にする
		*	*		*	•	39個のブロックを削除
	*		•	·	•	•	'regulateTemperature' を作成
*	*	*	*	*	*	•	

- 5) この[regulate Temperature]関数の目的については、 マコメントの吹き出しを参照してく ださい。
- 6) コードがどのように構成されているかを調べます。25,10,20がプリセットされた最初の
 関数ブロックの入力値と、展開された関数ブロック内のプログラムの関係を読み解き、比
 例制御システムがどのように機能するかを予測してから読み進めてください。
 (ヒント:ポート B, CH1の項目は USB ファンです。)



USB ファンは ON-OFF 制御を使用して, グローライトがオンの間グリーンハウスの温度を [idealTemperature](目標温度)近くに維持します。ファンだけでは理想的な温度を維持できない 場合, [lightIntensity](光の強度)は温度が理想的な値に近く安定するまで, 光の強度を減少さ せます。温度が安定している場合は, 理想的な温度である限り, 光の強度を徐々に増加させるこ とができます。では, [error]変数と[proportionalityConstant]変数は何をするのでしょうか。

まず編集可能な3つの関数入力を見てみましょう。

- ・[idealTemperature] 目標温度を 25°C に設定します。
- ・[lightIntensity] グローライトの赤と青の強度はレベル 10 に設定します。
- ・[proportionalityConstant] 比例定数は20です。

具体的にはどういうことなのでしょうか?

次に示すコードからわかるように、比例制御プログラムには3つの要素が必要です。

A : [セットポイント]と呼ばれる変数の理想値。

B:[プロセス変数]と呼ばれ,設定値に対してリアルタイムで測定値をチェックします。

▲-B:[error]は測定値と実際の温度との差です。



実際の温度が設定値から離れるほど、誤差は大きくなります。実際の温度と設定値の間に差がなければ誤差は0であり、システムは安定です。測定温度が設定値よりも高い場合、誤差は0未満となるので(○)ファンの電源がオンになり、そうでなければオフになります。

USB ファンは異なる電力レベルに設定できないため、ON-OFF 制御しかできません。グローラ

イトは強度を調整できるので、比例制御を使用できます。

グローライトの出力は誤差に比例します。これは、グローライトの強度と誤差の比率が比例定数 に等しい固定値であることを意味します。この比例定数[proportionalityConstant]は誤差を乗算 し、それを関数入力で指定されたグローライトの初期強度[lightIntensity]に加算して[intensity] 変数を計算します(●)。 [intensity]変数は必要に応じてグローライトを新しい強度に更新する ために使用されます(●)。

誤差の大きさと、その正負の状態には違いがあります。負の誤差は光の強度を初期強度より低く 設定し、正の誤差は光の強度を初期強度より高く設定します。比例定数が大きいほど、グリーン ハウスの温度が理想的な温度から離れたときの光の強度の変化が大きくなります。



上記のグラフは, [regulateTemperature]関数を[繰り返し]ループで10分間実行した場合の温度 データ(左)と明るさデータ(右)を示しています。

温度データでは,設定値は22.6℃です。実線は比例定数を20に設定した場合で,点線は定数を40に設定した場合の結果です。温度データはどちらの比例定数でも設定値の周りで温度が振動します(上下を繰り返します)。比例定数を40に設定すると,振動間の温度がより安定しています。

次に明るさデータを見ます。比例定数を高く設定すると、明るさは大きく低下します。そのため チャンバーに与える強い光と熱が減ります。これが、比例定数を高く設定すると温度がより安定 する理由です。

5.5. 事前実験

パート1:センサ,グリーンハウスのセットアップ



図13. グリーンハウスのセットアップ

1) コントロールノードの電源をオフにし、SPARKvueを閉じます。

- 2) 必要に応じて植物に水を与えます。
- 3) 図13のようにグリーンハウス(Eco チャンバー)の蓋にグリーンハウスセンサモジュール
 ①, 土壌水分プローブ②, グローライト③をセットします。
- 4) 蓋を通して約 30cm の余分な土壌水分プローブケーブルをゆっくり引っ張ります。
- 5) 図13④に示すように、バインダークリップと輪ゴムを使用してチューブとドロップヘッド を植木鉢に固定します。
- 6) 土壌水分プローブを図13のようにドロップヘッドの近くでセットします。土壌水分プローブの周りの土はプローブとの間に隙間ができないよう軽く圧縮します。
- 7)ドロップヘッドを少し上に向け、チューブを図135のようにフック状にセットして気泡や 過剰な排水を避けるようにセットします。
- 8) オプション:湿度管理として水を溜めておく場合は、必要に応じてボトルキャップ、ドロップヘッド、水を配置します。濡れたスポンジで管理する場合は、ファンを取り付けた後、スポンジを適切な高さとファンからの距離を保つシステムを考案します。
- 9) グリーンハウスの蓋をして,水供給チューブが通っているストッパーと USB ファンを側面 の穴に固定します。
- 10) 蓋を通して土壌水分プローブの余分なケーブルをゆっくり引き戻し, グローライトの中央が光センサの真上にくるように調整します。
- 11)図14 A に示すように冷却システムを構築します。浅めの器に氷を入れます。
- 12) ジップ付袋に手順11)の器を入れ、袋の中の空気を抜いて密封します。



図14. 冷却システム

13)テスト中に急激に温度を下げる必要がある場合は、図14の冷却システムを使用します。
 使用するには、チャンバーの蓋を外して冷却システムをセンサモジュールの温度センサの真下に配置します。※図14 Bのように冷却システムがセンサモジュールのどの部分にも接触しないようにしてください。

【ハードウェアのセットアップ】

 土壌水分プローブ(①),光・湿度・温度センサのセンサモジュールのケーブル(②)をグリーン ハウスセンサに接続し、グリーンハウスセンサはコントロールノードのセンサポートに接続 します(図153)。



図15. ハードウェアのセットアップ

- グローライトのコネクタをコントロールノードのポートAにつなぎ、グローライトの電源 アダプタをコンセントに接続します(図154)。
- パワーアウトプットモジュールをコントロールノードのポートBに接続します(図15
 ⑤)。
- 4) USB ファンをパワーアウトプットモジュールの CH1 に, USB ポンプを CH2 に接続しま す (図15**6⑦**)。

パート2:コントロールノードにプログラムをアップロードする

コントロールノードにはプログラムをデバイス自体に直接保存できる機能があるため、プログラム を実行するために毎回 SPARKvue に接続する必要がありません。この機能は数日、数週間、または 数ヶ月にわたって単独で実行する必要があるグリーンハウスの実験では大変便利な機能です。

- 1) SPARKvue を開き、デバイスにコントロールノードを接続します。
- 2) [温度]と、その他グリーンハウスセンサで使用する測定項目にチェック / を入れます。
- 3) コントロールノードのオンボードセンサは使用しないので、スライダをオン●からオフ● に変更します。
- 4)任意のデータ表示テンプレートを選択します。測定画面が開いたら、データ収集を開始して 現在の温度を記録し、データ収集を停止します。
- 5) プログラムツールアイコン 2を押してプログラム作成画面を開きます。
- 右上隅にあるコードライブラリを開きます。 カテゴリを[PASCObot]から[温室]に変更し、 [温度調節]を開きます。
- [idealTemperature]変数をデフォルト値の 25℃から、手順4) で記録した現在の温度より
 0.2℃高い値に変更します。
- 8) 関数ブロックを[繰り返し]ブロック内に配置し、コードを数秒間テストして動作することを 確認します。必要に応じて修正してください。

- 9) 右上の[コードアップロード]アイコン図を押して、コントロールノードにプログラムを送信します。アイコンを押すとアップロードが成功したことを示す音が鳴ります。
- 10) 右上の緑色の[コード実行]ボタン▶を押して、コントロールノードに保存されているプロ グラムを実行します。アップロードされたプログラムが実行されるとコントロールノードの Bluetooth 表示灯が青色に点滅します。
- 11) SPARKvue を閉じます。グリーンハウスで出力が動作し続け、コントロールノードの Bluetooth 表示灯が青色に点滅し続けていることに注目してください。
- 12) SPARKvueを開き、コントロールノードを接続します。手順2)と3)を行います。
- ✓ アップロードされたプログラムはセンサ測定値のみを返すことができます。[プログラム出力]カテゴリの[テキスト出力]および[数値出力]はアップロードされたプログラムからは取得できません。どちらも、SPARKvue内の[開始ボタン]●でプログラム実行されたときのみ、データ表示で取得できます。
- 13) データ取得を開始します。数秒実行し、停止します。データ収集が実行中かどうかにかか わらず、アップロードされたプログラムが実行され続けていることに注意してください。
- 14) コントロールノードのプログラムを停止するには、プログラムツールを開き、右上にある
 [コード実行の停止] アイコン■を押すか、コントロールノードの電源をオフにします。
- パート3:既存コードのレビュー
 - SPARKvue のメインメニュー
 に移動し、[開く...]を選択します。実験1~4で保存した ファイルの内、プログラムを含んだファイルを開きます。または、実験1~4で作成したコ ードのスクリーンショットやスケッチを参照します。
 - 2) コードを確認します。プログラムが 24 時間サイクル内で機能するために必要な変更箇所を 決めます。
 - すべてのグリーンハウス実験について、手順1)と2)を繰り返します。必要に応じてコー ドライブラリの関数を使用します。
 - 4) テキスト出力を持つ以前に作成したプログラムを再設計するか、開いて実行します。データ
 収集中にデータ表示に移動して[新規ページ] ■を作成します。
 - 5) フルページレイアウト を選択し, [表] 間を追加します。
 - 6) [測定項目を選択]を押して、グリーンハウスセンサの測定項目を選択し、コードからテキスト出力を1つ以上追加します。テキスト出力を追加するには、[測定項目を追加]を選択し、右側に表示されるメニューで[センサ]タブから[ユーザ入力]タブに切り替えて選択します。
 - 7) 表の下の[表ツール]メニューを開いて、表の列を追加または削除します。
 - 8) 表に追加されるデータを確認します。このデータ確認は、プログラムのトラブルシューティングを行うときに使用します。新しい情報を明らかにするのに役立つことがあります。

パート4:24時間サイクルプログラムを計画する

別の用紙に,既存のコードを変更および結合して次のような一連のプログラムを作成する方法を示す フローチャートを描きます。

- 1) 定義した開始時刻で24時間サイクルを実行します。
- 2) 適切な[繰り返し]ブロックと[時間]ブロックを使用して、入出力の論理順序を決定します。
- 3) 関数を使用して、プログラムを分離および整理します。
- 4)各測定の適切な時間間隔で状態を監視するため、定期的にセンサ入力値(温度・相対湿度・ 明るさ・土壌水分)を入手します。
- 5) 測定値を入力として使用して, USB ポンプ・USB ファンの動作およびグローライトの光の 強度をトリガーし, 植物に適した温度, 相対湿度, 明るさ, 土壌水分の条件を1日中維持し ます(理想温度は現在の室温よりも少し高く設定してください)。
- 6) 視覚的な警告(テキスト出力)とコントロールノードのスピーカーを使用した音による警告 を結合します。
- 7) ON-OFF 制御と比例制御を採用します。

フローチャートには,各イベントが発生するおおよその時間と,使用するブロックの種類を含める必要がありますが,すべてのコードを表示する必要はありません。

フローチャートはプログラムを成功させるために必要な正しい種類の変数,ループ,時間遅延などを 使用して,論理的な順序で発生する必要があることを計画し考えることに役立ちます。



5.6. 実験

SPARKvue で[新しい実験を開始]し,フローチャートからプログラムのコードへの変換を開始します。 トラブルシューティングが容易になるように,作業中に各コード機能をテストし,作業内容を保存す ることを忘れないでください。テスト時には時間単位を秒からミリ秒に変更して,1時間を3.6秒に 圧縮すると24時間が約1.5分でテストできるようになります。トラブルシューティング中に行き詰 った場合は,[表]を作成してプログラムの結果を直接確認してみてください。

プログラムは前のセクションのパート4で説明したすべての条件を満たす必要があります。

長期間にわたってグリーンハウスを自律的に実行する場合は,コントロールノードを電源に接続した ままにし,プログラムをコントロールノードにアップロードして,SPARKvueを終了する前にプログ ラムを実行します。

完了したらプログラムのコードをスクリーンショットで撮影し,別の用紙に印刷またはスケッチし, 問題 a~d について考えます。

a. 24 時間サイクルプログラムで発生するイベントを要約し,24 時間にわたって正しいタイミングを確保するためにプログラムをどのように記述するのか説明してください。

【回答例】

主なコードは24時間の繰り返しで、コードの中では1日は日の出前の午前5時に始まり、 土壌水分をチェックし、必要に応じて植物に水を与えます。給水はこの時間にのみ行うこと ができ、貯水量が少なくなりすぎるとアラートが鳴り警告が表示されます。水が少なすぎる 場合はポンプを停止することもできます。日の出は午前6時に始まります。グローライトは 赤色のみからスタートし、正午のピークまで青色とともに徐々に明るくなります。正午を過 ぎてから午後6時の日没までの間は徐々に暗くなります。ライトが点灯している間、温度設 定値を維持するために必要があれば比例制御を使用して明るさを減らします。リザーバーの 水位、相対湿度、温度、土壌水分を1時間に1回チェックし必要に応じてアラートを鳴らし たり警告を表示したりします。日没後は24時間サイクルが終わるまでライトは消灯し、翌 日の同じ時間にサイクルが再び開始されます。 b. 複数の入力イベントと出力イベントを1つのプログラムに結合したときに直面した課題を 上げてください。

【回答例】

ほとんどの実験では[繰り返し]ブロックの[繰り返す:続ける条件]が動作しましたが, 24 時間サイクルプログラムでは,プログラムが永遠に繰り返されるループで止まり次 の関数に到達しないため,個々の関数でループを使用することはできませんでした。ル ープがある場合はすべての関数に使用できるループの種類を考えるか,前日のアラート が次の日に確実につながるように条件ステートメントを追加する必要がありました。 また,1つのイベントを次のイベントにつなげる方法と,各イベントの間の[sleep]時間 を把握することも課題でした。24 時間サイクルがいつ停止し,次のサイクルがいつ開 始するかがわからないため,[日数カウント]変数を作成して数字表示に追加することを 考えるまでが困難でした。

c. 複数の入力イベントと出力イベントを1つのプログラムに結合して成功する方法について, みんなにどのようなアドバイスをしますか?

【回答例】

プログラムのすべての部分を, [sleep]ブロックを使用して時間の経過とともに分けられ る個別の機能に分解する必要があります。物事がどのような順序で行われなければなら ないかをよく考えてください。変数と測定値の表を使用して, プログラムした挙動と実 際の挙動が同じであることを確認するなどのトラブルシューティングを行い, 少しコー ドを記述してからテストします。テストは長時間になってしまうため, 表を使用したト ラブルシューティングで時間を節約しましょう。また, 必要なコードを既に作成してい る場合は, 不必要にコードを始めから作成するのではなく, そのコードの一部またはす べてを使用できるかどうかを確認し, 修正しましょう。

d. 24 時間プログラムで比例制御を使用する方法を説明してください。このタイプの制御が ON-OFF 制御より優れている点を少なくとも1つ説明してください。

【回答例】

温度の調整関数のコードをリスト配列と[リストアイテム繰り返し/ for each item]ブロ ックに結合して、1日の時間帯に適した量に明るさを調整しました。ライトの明るさは 最大値を超えて増加することはありませんが、ファンだけでグリーンハウス内の温度を 目標温度に維持できない場合は、ライトの明るさ最大値を徐々に減らします。比例制御 の利点の1つは、ON-OFF 制御と比較して、理想的な温度と明るさに近い状態でより 多くの時間を維持できることです。



-68-



-69-



-70-


5.7. 追加実験

- 植物に推奨される施肥方法を調べます。テキストとオーディオのリマインダーを追加して, 推奨 されるスケジュールに遅れないようにします。
- ビープ音の代わりに,音楽を演奏するプログラミングします。例えば USB ポンプで植物に水や りをしているときに音楽を流します。
- 色が変化したり、点滅したりする特徴的で注意を惹くグローライトのパターンを使用して、テキ ストベースのアラートからライトベースのアラートに変更します。



輸入販売元

株式会社 島津理化

〒101-0051 東京都千代田区神田神保町1丁目 32番出版クラブビル TEL (03) 6854-0274 URL: https://www.shimadzu-rika.co.jp

本製品の技術的お問い合わせはこちらへ

株式会社島津理化 ホームページ「お問い合わせ」フォーム

(URL : https://www.shimadzu-rika.co.jp)

e-mail : soudan@shimadzu-rika.co.jp Fax : (03) 6854-0275

101446D2409