

## **Plan de Test pour le Projet HelioHeat**

### **Tests Électriques Passifs (avant alimentation)**

Ces tests permettent de s'assurer que le circuit imprimé est correctement réalisé et que les composants sont correctement soudés, sans alimenter le circuit. Cela évite d'endommager les composants si une erreur matérielle est présente.

#### **Test de continuité**

##### **Objectif**

Le but de ce test de continuité est de vérifier que toutes les pistes critiques sont bien connectées selon le schéma électrique. Ce schéma électrique peut être retrouvé dans les différents documents générés par *Kicad*.

##### **Méthode**

Pour les différents point, il nous faudra utiliser un multimètre en mode continuité. On pourra alors tester :

- La connexion entre le pin Vcc de l'ATTiny et l'entrée Vcc du PCB.
- La masse commune (GND) sur tous les composants (ATTiny, capteur thermique, afficheurs, etc.).
- Les signaux de données (par ex. entre ATTiny et 74HC595).

#### **Test d'isolement (absence de court-circuit)**

##### **Objectif**

Dans le test d'isolement, il nous suffit de vérifier qu'il n'y a pas de court-circuit involontaire, en particulier entre Vcc et GND.

##### **Méthode**

On pourra alors mesurer les différentes résistances entre Vcc et GND avec un multimètre.

Si une valeur très basse ( $<10\ \Omega$ ), elle sera alors suspecte.

#### **Vérification de polarité des composants**

##### **Objectif**

Pour vérifier la polarité des composants, il nous faut vérifier l'orientation correcte des composants polarisés.

### **Méthode**

Visuellement, on peut vérifier les différents composants avec leurs repères, notamment sur l'ATTiny et les afficheurs 7 segments.

### **Vérification des soudures**

#### **Objectif**

Nous pouvons également s'assurer que toutes les soudures sont propres et fonctionnelles.

#### **Méthode**

À la loupe, nous pouvons vérifier qu'il n'y a pas de ponts de soudure, ni de broches non soudées.

### **Vérification du brochage des connecteurs**

#### **Objectif**

Le but de ces tests est de confirmer que les connecteurs sont câblés conformément au schéma.

#### **Méthode**

En prenant le schéma PDF, généré par Kicad, on peut repérer chaque ligne de signal. On peut alors vérifier que chaque broche correspond bien à la bonne ligne (ex. : GND, VCC, DATA, CLK, etc.).

### **Tests Fonctionnels (composant par composant)**

Ces tests visent à vérifier que chaque composant du circuit fonctionne indépendamment.

### **ATTiny**

#### **Objectif**

Afin de tester l'ATTiny, on peut tester la programmation et le bon fonctionnement du microcontrôleur.

#### **Méthode**

On peut flasher un code qui permettrait de faire clignoter une LED connectée à une sortie.

### **Capteur thermique MCP9700AT**

### **Objectif**

Nous pouvons vérifier que la tension de sortie du capteur varie correctement avec la température.

### **Méthode**

En mesurant la sortie au multimètre du capteur de température, on peut soit chauffer, soit refroidir légèrement le capteur et vérifier les valeurs.

### **Registre à décalage**

#### **Objectif**

Pour vérifier le registre à décalage, nous pouvons vérifier que les bits de données sont correctement transférés.

#### **Méthode**

On peut utiliser une carte Arduino pour envoyer un signal série. On peut alors observer les sorties avec un oscilloscope.

### **Décodage BCD vers 7 segments**

#### **Objectif**

Cette partie a pour but de vérifier la conversion correcte de l'entrée BCD en segments.

#### **Méthode**

En appliquant des valeurs de 0 à 9 sur les entrées, on peut vérifier l'affichage sur les différents 7 segments présents sur le PCB.

### **Afficheur 7 segments**

#### **Objectif**

Vérifier la lisibilité et la fonctionnalité des segments nous permettrait de tester les différents afficheurs 7 segments.

#### **Méthode**

En alimentant directement chaque segment individuellement, on peut vérifier que les segments s'allument clairement.

### **Tests Système avec Firmware**

#### **Chargement du firmware**

### **Objectif**

On peut tester le code final (`main.ino`) pour tester le chargement du firmware.

### **Méthode**

On peut utiliser *Arduino IDE*, on peut compiler et téléverser le firmware.

### **Affichage de température**

#### **Objectif**

On peut alors vérifier que la température lue est affichée correctement.

#### **Méthode**

On peut alors comparer la température affichée avec un thermomètre réel.

### **Réactivité à la variation de température**

#### **Objectif**

L'objectif est de s'assurer que le système réagit aux changements.

#### **Méthode**

Pour cela, on peut appliquer une variation thermique rapide, que ce soit en chauffant ou refroidissant la carte.